

## Systems and Methods for Presenting Information on Mobile Devices

### Summary

Certain embodiments of the present invention are directed to a "thin client" that addresses the evolving requirements for distributing content across the mobile device landscape, and specifically one or more of the problems with prior art systems.

The "thin client" begins with the definition of an application model for running on a mobile device, where the model is implemented inside a language. In one embodiment, a method and apparatus for authoring content with the language is provided. Embodiments may range from quasi-programming to a full-featured authoring environment. By this definition the browser itself is a "thin client" but will be treated below as a separate category. Other prominent examples of "thin clients" on the desktop are Flash and PDF readers. Browser-based implementations include, but are not limited to, WAP, HTML, AJAX, and/or possible scripting Languages. Since the browser initially defined the desktop internet experience it is reasonable to expect that, as the incumbents, the browser-based solutions could play a significant role in bringing content to mobile devices.

In contrast to a "thin client," a "fat client" is generally an application that is installed and remains on the device. The "fat client" utilizes the APIs available natively on the device, or relies on the APIs that are exposed by the virtual machine. Apple's iPhone is an example of a "fat client," as are Yahoo Go Live 2.0 and implementations developed by various vendors such as Surf Kitchen.

### Detailed Description

The following includes a description of specific embodiments of the present invention. These embodiments are illustrative, and are not meant to limit the scope of the present invention.

## **A: Comprehensive Application Model**

All languages, whether they are programming languages such as C, C#, C++, Java or BREW, scripting languages such as JavaScript, or mark-up languages such as WAP and HTML, implement an application model, which is the abstraction of what capabilities the language was designed for.

Productivity tools, such as word processors and spread sheets, also have application models, although they usually have very limited extensibility as compared to programming languages. Typically, they sacrifice the robustness of a programming language for ease-of-use.

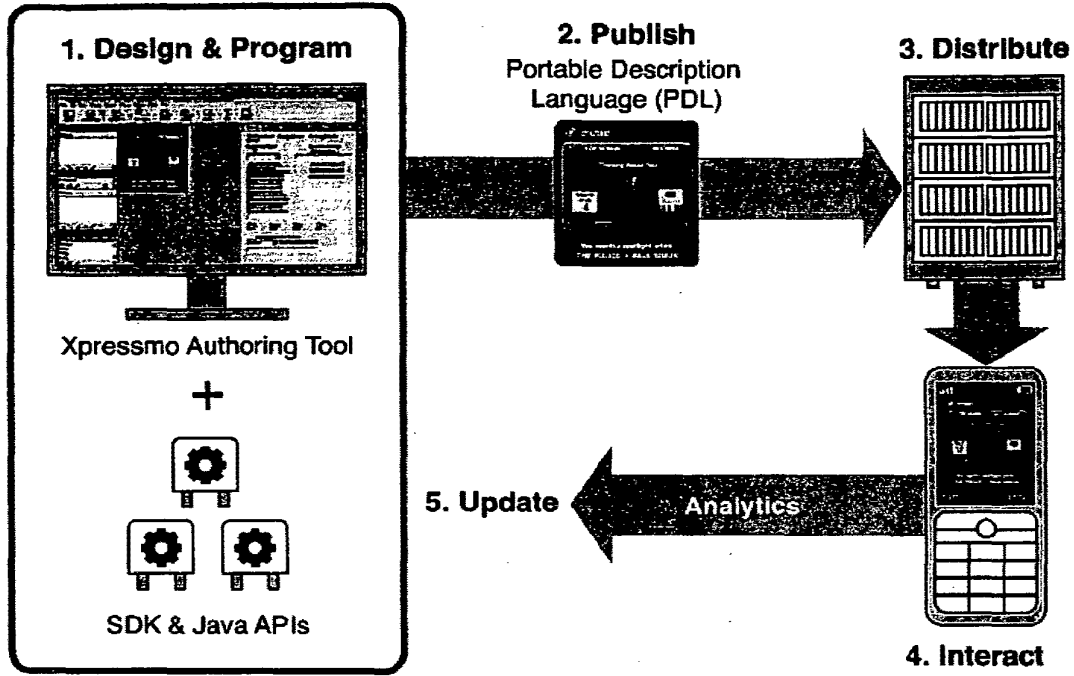
There must be an authoring platform that supports a comprehensive application model that combines the ease-of-use of well designed productivity tools with the robustness of a programming language.

### **Application Model Life Cycle**

The implementation of a robust application model requires a Portable Description Language (PDL). As *Graphic 1* depicts, the PDL is the common language for both the Authoring Tool and the SDK.

Systems and Methods for Presenting Information on Mobile Devices

Graphic 1: Application Model Life Cycle



## Systems and Methods for Presenting Information on Mobile Devices

### **Portable Description Language**

The PDL can be conceptually viewed as a device, operating system and virtual machine agnostic representation of serialized objects. While either designing the application with the Authoring Tool, or programming with the SDK, the internal representation of the programming logic can be, for example and without limitation, Java. {STEVE: ARE THERE OTHER EXAMPLES THAT YOU MIGHT ADD HERE?}

At "publish" time this Java code, for example, gets translated into the PDL. This translation would also occur in real-time during the execution of any Web Services or backend business logic that interacts with the user.

### **Some of the advantages of the PDL as opposed to other languages are:**

#### **1: Code and Data Compaction**

Code, as well as vector, integer and Boolean data are compacted and then compressed resulting in a size reduction of 40 to 80 times that of the original Java serialized objects. This is important not only for performance over the network but for utilizing the virtual memory manager of the player more efficiently. See Graphic 2.

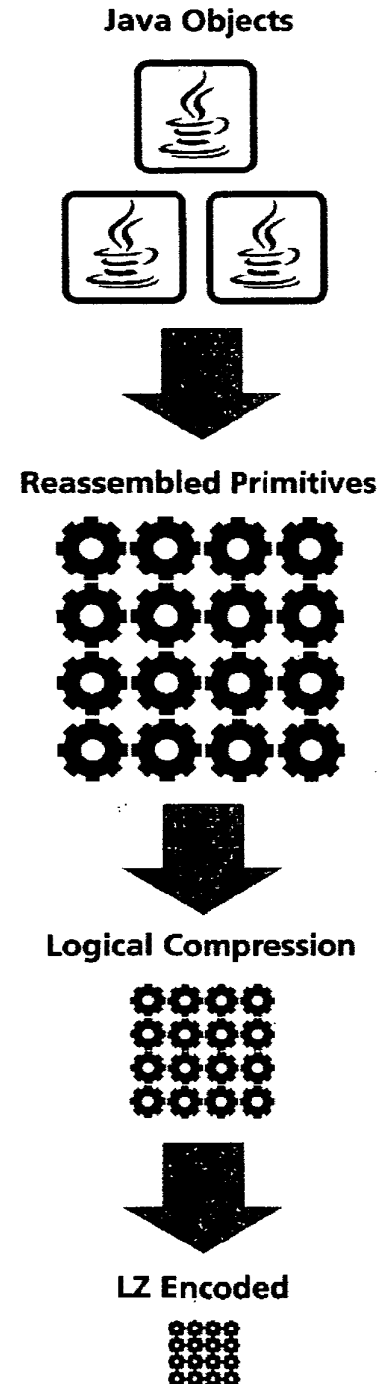
#### **2: Virtual Machine and Operating System Independence**

The reassembled primitives of the application no longer have any dependencies from the original programming language (Java) that they were defined in. The player architecture takes full advantage of this by abstracting all the virtual machine and/or operating system interfaces from the code that processes the PDL.

#### **3: Extensibility and Compatibility**

Because the PDL is defined by the means of nested arrays of primitives, there are no constraints in terms of extending the player seamlessly as market demands and device capabilities continue to grow. Compatibility with other languages is inherent based on the various player abstraction implementations.

Graphic 2: PDL Code and Data Compaction



## **B: Authoring Platform and API Extensions**

### **1. WYSIWYG Full-Featured Authoring Tool**

The delivery platform authoring tool provides developers with a what-you-see-is-what-you-get (WYSIWYG) PC- and Mac-based full featured editor. Most of the complicated Java programming functions is presented with easy-to-use pull-down menus, dialog boxes, and buttons.

**Graphic 3: Authoring Tool**



#### **a: Intelligent Interactive Object Libraries for Sophisticated Development Environment**

Designers of the delivery platform must adopt and support the most sophisticated and powerful features of advanced web and web services applications, in addition to location-based services (LBS) and photo services, and integrate them within the platform's authoring environment. Now, solution providers can use the intelligent interactive object libraries to offer to their customers and mobile users:

- ▶ Slide shows, which incorporate images, video, audio, animated transitions, multiple chats, and mouse interaction.
- ▶ Full 2-D vector graphics.
- ▶ GIS (advanced LBS), including:
  - Multiple raster and vector layers.
  - Feature sensitive interactions.
  - Location awareness.
- ▶ Streaming and embedded audio/video.
- ▶ Virtual tours.
- ▶ Image processing and enhancement.
- ▶ Widgets

Child objects, which can be activated by user interaction and/or time. Uses include:

- ▶ Mouse over (object selection), hover and fire events.
- ▶ Launching of object-specific, rich-media experiences.

## Systems and Methods for Presenting Information on Mobile Devices

### **b: Advanced Interactive Event Model for Rich, Compelling Applications**

The distinguishing characteristic of the 21st Century Internet is that it is a highly interactive and personalized platform for all consumer, business, and public sectors. Mobile devices must be as interactive and personal as PCs and gaming consoles. This requires timed events, smooth movements, and interesting animations. The vendor delivery platform should have the following advanced features built into its authoring environment:

- ▶ User-, time- and/or location-initiated events, which allow content developers to base interactivity on specific user interactions and/or instances in time and space.
- ▶ Timelines, which are critical for timing of multiple events and for animations when entering, on, or exiting pages of the application.
- ▶ Waypoints, which act similar to key frames, to allow smooth movement of objects within pages of the application. Waypoints define positions on a page object's animation trajectory. When an object reaches a specific waypoint:
  - Other object timelines can be initiated, creating location-sensitive multiple object interaction.
  - Audio can be defined to play until the object reaches the next waypoint.

### **c: Full Style Inheritance**

Both Master Page inheritance (for structural layout inheritance and repeating objects) and Object Styles (for both look and feel attribute inheritance) are supported.

After a style has been defined for an object, the object will inherit the style. Style attributes include both the look and the feel of an object, including mouse interaction, animations, and timelines.

For example, if the content developer creates various text objects using a style that sets the font to red, the fonts of these objects will be red. Suppose the developer changes the font color of a specific button to green. If later, the developer changes the style to blue; all other text objects that were created with that style will become blue except for the button that had been specifically set to green.

### **d: Page View, Style, Object, Widget and Application Template Libraries**

Templates created in private libraries can be used to provide consistency throughout an application. There are also public libraries that can be made available to all developers.

Templates can be used to define the look and feel of the entire application, specific pages, or specific slide shows and virtual tours.

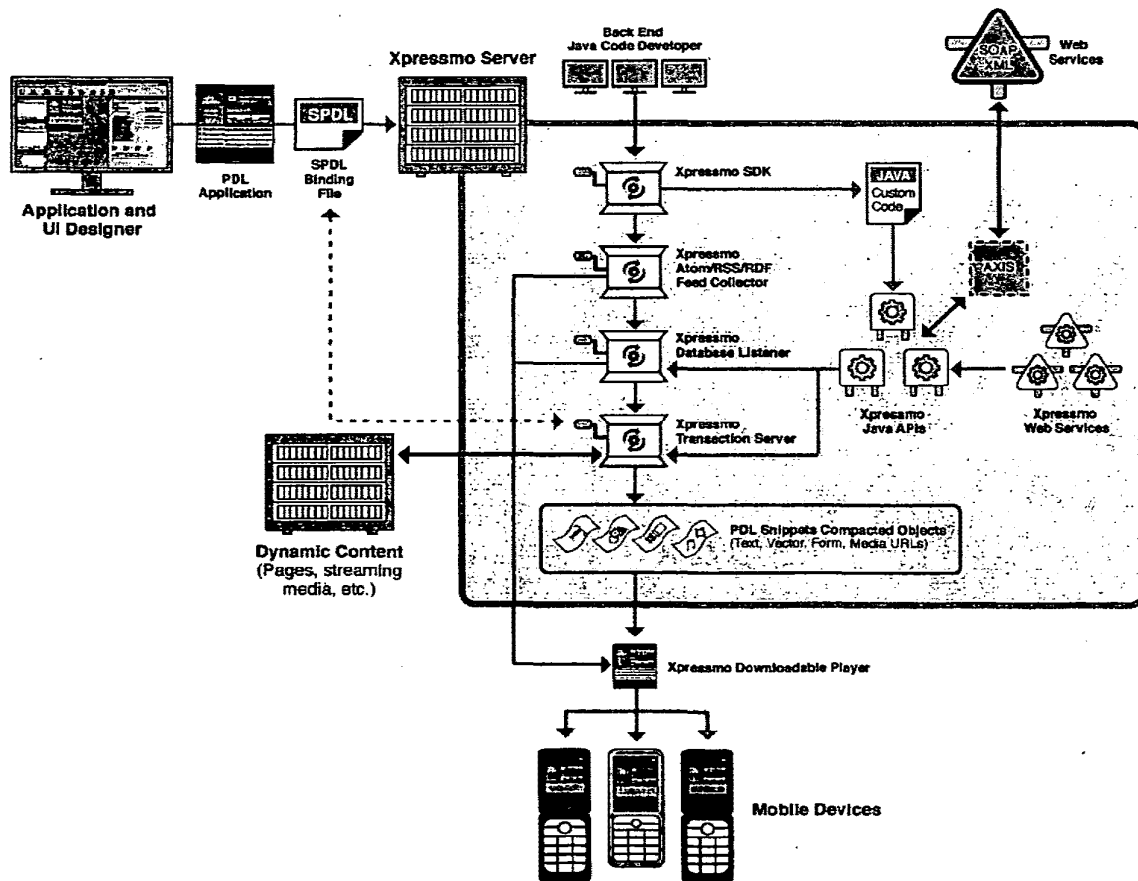
## Systems and Methods for Presenting Information on Mobile Devices

### 2. Java APIs, Web Services and the SDK

The authoring platform should also include the following components:

- ▶ Vendor supplied SDK for custom code development.
- ▶ A full set of Java APIs should be included in the vendor's System Development Kit (SDK) to make it easy to add extensions and functionality to mobile applications, and tie applications to back-end databases through the vendor's Content Server.
- ▶ A growing set of vendor supplied Web Services. The vendor should offer an expanding set of Web Services, available through both the authoring tool and the SDK. They include:
  - Dynamic Binding of Real-time Content
    - ▶ This includes a RSS/Atom/RDF Feed Collector with XML Rules based APIs
  - Broadcasting
  - Rich Messaging
  - Server Pages
  - Widgets
  - Persistent Storage Variables
  - Access to Phone Resources
- ▶ A Web Services interface to SOAP/XML enabled Web Services

Graphic 4: Authoring Platform



## Systems and Methods for Presenting Information on Mobile Devices

### **a: The SDK and Java APIs**

The vendor's SDK should work with various popular IDEs including Eclipse. Available through the SDK should be a large and powerful set of APIs and interfaces that permit the seamless extension of any application to back end business logic, Web Services, etc. These interfaces and APIs support:

#### ▶ **Listeners**

There should be a large set of listeners that expose both player-side events and dynamically linked server-side data base events. Some examples are:

- Player-side forms-based content.
- Player-side user interactions
- Player-side object status
- Server-side driven dynamic content events.

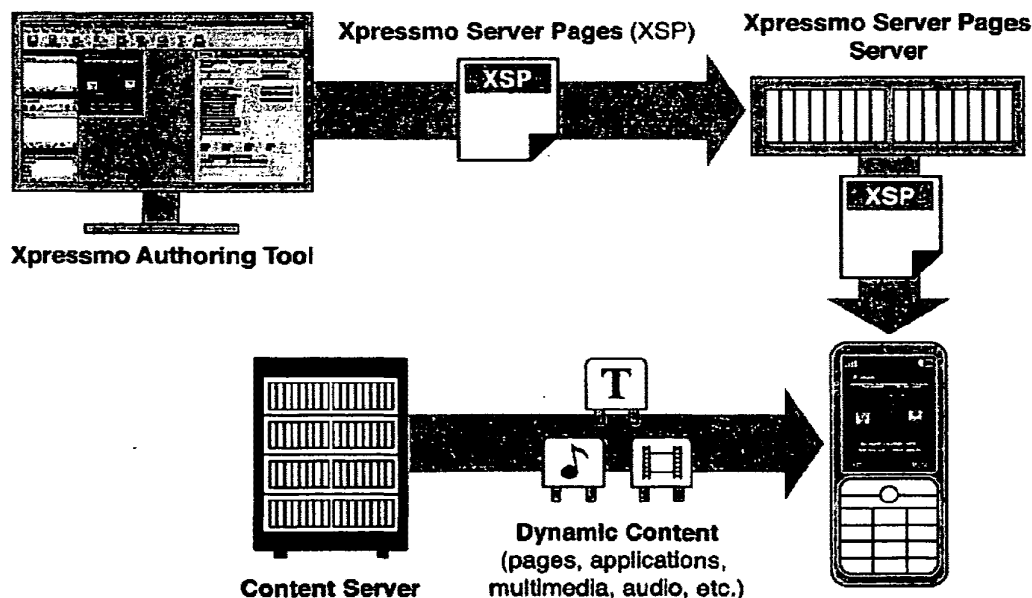
#### ▶ **Player-side Object Operations**

There should be a superset of all authoring tool functionality that is available through APIs. They include:

- Page View Level APIs for inserting, replacing, and or modifying any page object.
- Object Level APIs for modifying any attribute of existing objects, adding definitions to attributes, and adding, hiding or replacing any object.

### **b: Web Services for Real-Time, Shared, and Broadcast Content for Timely Distribution**

Magazines, newspapers, websites, and other forms of published material are based on templates and styles that offer a consistent, familiar look and feel to those viewing the material. These forms of publications offer content that can be accessed (i.e., read) in near real-time (depending upon the frequency of the publication), shared with others (i.e., showing an article to a friend), and broadcast to groups of people (i.e., subscribers). It should be no different for the mobile market. That's why the suggested platform offers Server Pages (XSPs), which, like Java Server Pages, serve objects and dynamic content to individual pages (that have been set with styles and templates) without the need to open other pages.



## Systems and Methods for Presenting Information on Mobile Devices

### **Graphic 5: Server Pages**

However, unlike JSPs and ASPs, which are restricted to the functionality supported by the browser, the functionality of XSPs can be remarkably extended through the vendor's Java APIs. Also, unlike the slow performance inherent in browser operations, XSPs take full advantage of the compaction technology of the proposed platform, resulting in response times usually less than 2 seconds.

With the dynamic binding functionality of an XSP, a page can be saved as a page object in an author's "pages" library, and then can be dynamically populated with real-time content simultaneously as the page is downloaded to a given handset player based on a newly expanded API. Server Pages could also be produced programmatically, but in most cases the delivery platform authoring tool will be a much more efficient way to generate and maintain libraries of dynamically changing XSPs.

With XSPs, pages that have dynamic content built into them can be sent directly to the handsets. Without XSPs, content authors would have to define each page in the application. With the proposed implementation, no pages need be defined. For example, in a World Cup application, one page could represent real-time scores that change continuously on demand. With polling (for example, a prompt to the users asking who they predict will win a game), a back-end database would tabulate the information and then send the results dynamically to the handsets. With a bar chart, the application could use dynamic Portable Description Language (PDL) with scaling on the fly. For example, the server could recalibrate the bar chart for every ten numbers.

### **c: Web Services for Widgets**

An extension of an XSP is a Widget object. Widgets are supplied with the following services:

- ▶ Widgets can be developed from numerous sources:
  - The Authoring Tool
  - A Consumer Publishing Tool
  - An XML to Widget Conversion Tool
- ▶ The SDK Widget Libraries are automatically populated and managed.
- ▶ Widget Selection Lists are available and can be populated with author defined Icons.

### **d: Web Services for Sharing Through Chats and Forums**

User-generated content and chats have become the trend in 2006. For example, YouTube.com delivers more than 40 million video views every day with 35,000 new videos uploaded daily.

A USA Today article revealed that MySpace has reached well over 47.3 million members. The Blog Herald also reported that MySpace "is growing by an amazing 160,000 new users a day, according to MediaPost" (comScore Media Metrix, a division of comScore Networks, Inc.).

The delivery platform should have an integrated social networking message board technology within its solution. With the mobile chat objects, mobile handset users can communicate within groups of communities. Entertainers and artists can create personalized Chat Rooms, and sport teams can create team chats. Other benefits target MVNOs and media companies.

- ▶ **MVNOs**
  - Increase average revenue per user (ARPU).
  - Increase upgrades and premium service plans.
  - Expand subscriber base with incremental users sharing photos and chats.
  - Leverage photos and chats with seasonal marketing events.
- ▶ **Media companies**
  - Build community and capture new users with chatting.
  - Increase consumer interaction with brands/properties.



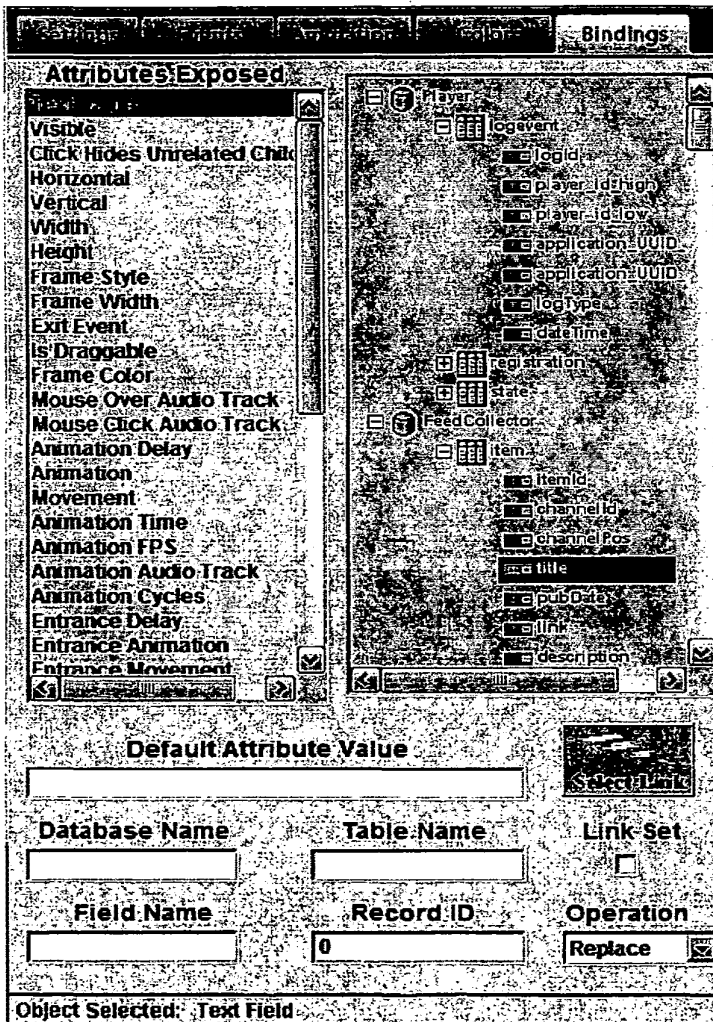
## Systems and Methods for Presenting Information on Mobile Devices

### e: Web Services for Dynamic Data Binding (RSS feed and Chats)

Inserting RSS feeds and other forms of dynamic content into mobile applications must be easy with the delivery platform. The authoring tool should contain an easy-to-use dialog box that dynamically links objects to data and feeds determined by RSS and chat databases. Any relevant attribute for a page view and/or object can be dynamically bound to a value in a server-side database. This includes elements within complex objects such as:

- ▶ Any Icon or Text Element within a Graphical List
- ▶ Any Icon within a Launch Strip
- ▶ Any Feature within any Geographical View of a GIS Service Object
- ▶ Any Virtual Room within a Virtual Tour.

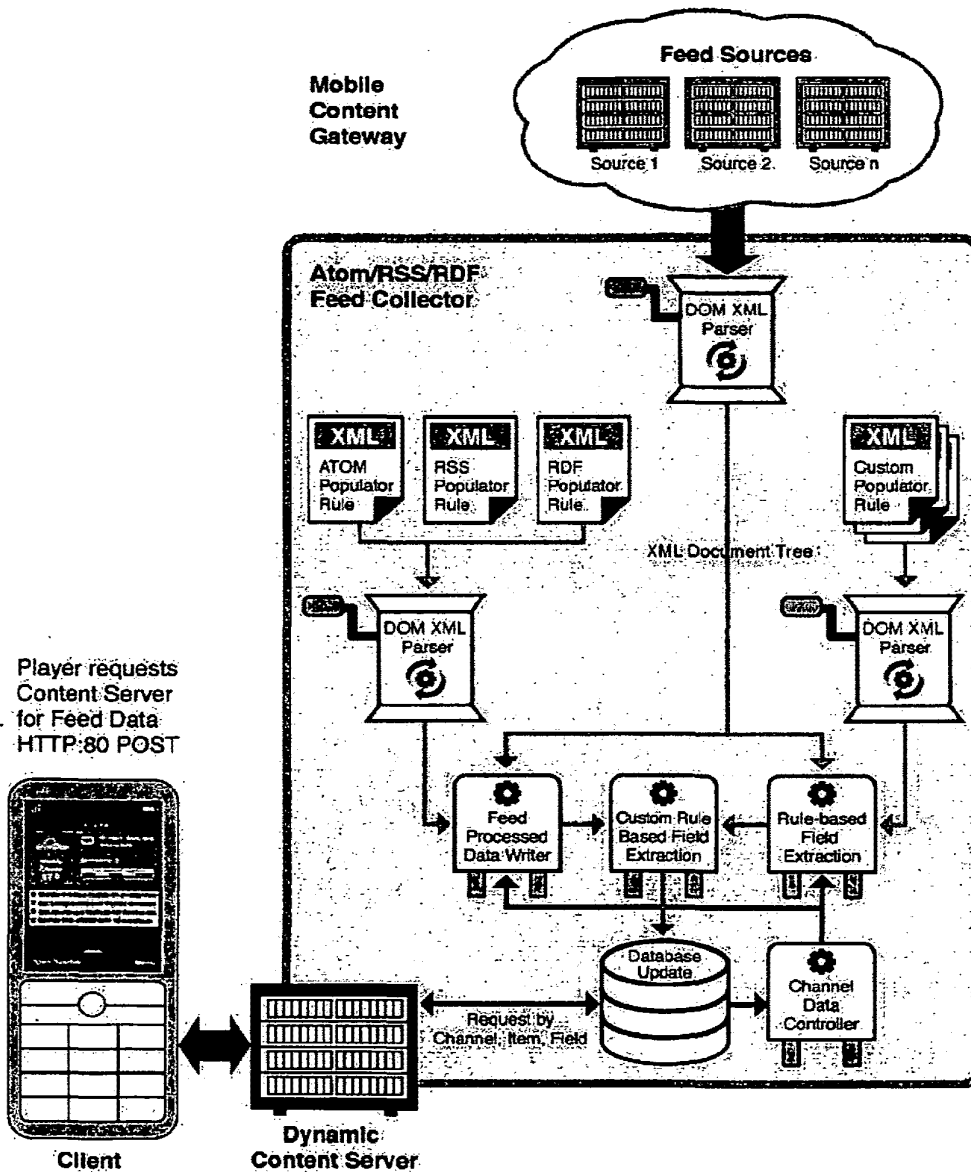
Graphic 6: Dynamic Data Binding Tool



## Systems and Methods for Presenting Information on Mobile Devices

### f: Web Services for Converting RSS/Atom and RDF Feeds

Graphic 7 shows a proposed implementation for a Mobile Content Gateway. It starts with the identification of an ATOM, RDF or RSS feed source. Through a Vendor supplied rules-based populator, any of these feeds can be logically parsed, with any type of data extraction methodology, either by using vendor supplied rules, or by the author defining their own custom extraction rule.



Graphic 7: RSS/Atom and RDF Feed Conversion Web Service

## Systems and Methods for Presenting Information on Mobile Devices

### C: Intuitive Navigation

Efficient Platform User Interface with Touch Minimization for Optimizing Form Factors and Navigation

The delivery platform should offer a very powerful and broad set of extensible navigation objects, as well as object- and pointer-navigation options to make it easy to populate small mobile device displays with content and to navigate easily:

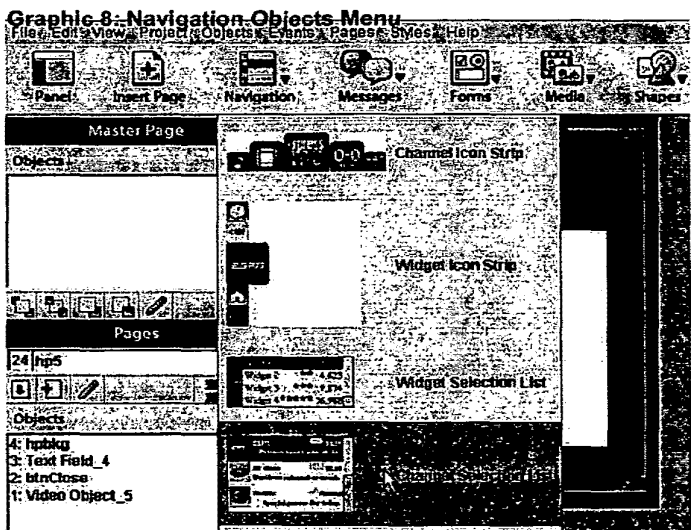
- ▶ among page views
- ▶ between applications
- ▶ within objects in a single page view of an application

#### a: Navigation Objects

Navigation objects include various types of launch strips, various intelligent and user-friendly text fields and scrolling text boxes, as well as desktop level business forms. In fact, every type of object can be used for navigation by assigning a navigation event to it.

The authoring tool offers a list of navigation Object Templates, which then can be modified in numerous ways.

(Graphic 8)



## Systems and Methods for Presenting Information on Mobile Devices

### **b: Launch Strips and Graphical List Templates**

#### **Launch Strips** (Graphic 9)

Launch Strips can be designed by the author with almost no restrictions. They can be stationary or appear on command from any edge of the device, their size, style, audio feedback, and animations can be freely defined to create highly compelling experiences.

In the example to the left a portal type Launch Strip becomes visible from the left edge when requested, while a channel type Launch Strip, which may be application specific, becomes visible from the bottom when requested. The channel type Launch Strip could have an appropriate sound effect for each channel when being selected, as well as popup bubble help.

#### **Graphical Lists** (Graphic 10)

As seen in the examples below, graphical lists can contain items with many possible text and image elements. Each element can be defined at authoring time and/or populated dynamically through either vendor supplied Web Services or APIs.

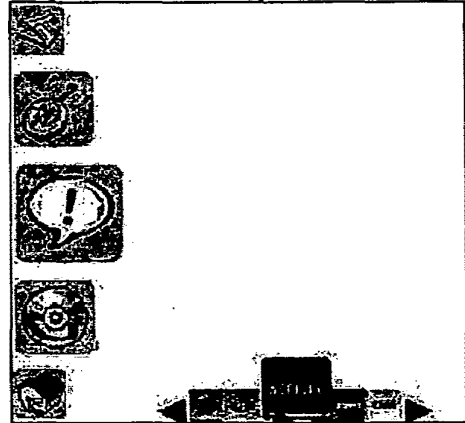
### **c: Assignable Navigation Events**

All objects, and/or all elements within an object, can be assigned navigation events that can be extended to the vendor's Web Services or APIs. For example, a Rolodex type of navigation event can dynamically set the starting slide of the targeted page view (or the starting view of a targeted application).

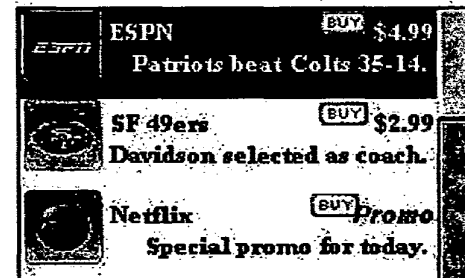
#### **d: Intuitive Object and Pointer Navigation**

The proposed platform has created options to choose between object and Mouse navigation for other devices. Object Navigation permits selection of objects with a Joy Stick and/or Cursor Keys in all 4 directions. When within a complex object the navigation system automatically adopts to the navigation needs for that object. For coordinate-sensitive objects, such as geographical information services (GIS), location-based services (LBS), or virtual tours, a soft cursor appears. For lists, scrolling text areas and chats, launch strips, and slide shows the navigation process permits intuitive selection of elements within the object. Scroll bars and elevators are optionally available for feedback. If the device has a pointing mechanism then scroll bars are active and simulate the desktop experience.

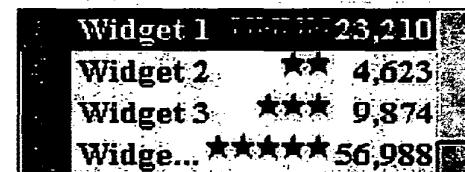
**Graphic 9: Launch Strips**



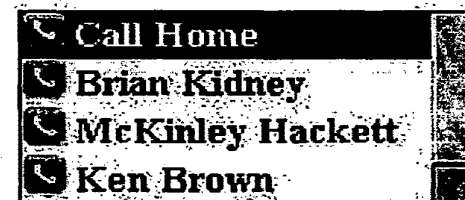
**Graphic 10: Graphical Lists**  
Channel Selection List



Widget Selection List



Phone list



## D: Advanced Feature Support

### 1: Messaging and Social Networking

A seamless integration of all phone communication functions with any robust mobile application would be an essential prerequisite for market adoption. This means that communication functions should be:

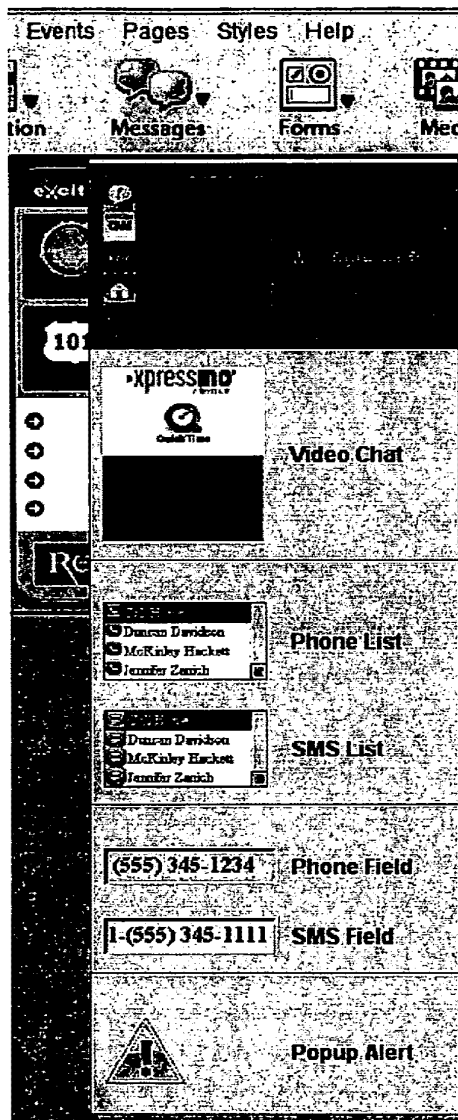
- ▶ **Easier**  
The number of touches should be less than (certainly no worse than equal to) that of using the phone communications UI directly.
- ▶ **More Robust**  
The communication choices should be an appropriate superset of what the phone already offers.
- ▶ **More Entertaining**  
The communication choices should offer appropriate imagery, video, text, and audio to make the user experience more compelling.
- ▶ **Personalization**  
Users should be able to adjust the communication metaphor to whatever they feel works best for them.

In addition, the content that is available for messaging and social networking should only be limited by the imagination of the user. This means that any content capturing functions on the device must be seamlessly supported. Such content should be customized as the user wishes. Messages could be easily assembled based on any content that is available to the user. Messages could easily be shared with any individual and/or group.

Communication metaphors must be state-of-the-art and easily available to both the authors and the users of this suggested platform. Two examples of this are shown in Graphic 11 above.

- ▶ **Video Chat**  
A chat session can be conducted simultaneously, inside an integrated page view, with a video or television station.
- ▶ **Multiple Chat**  
Multiple chat sessions, each with a designated individual and/or group, can be conducted simultaneously, with each of the chat threads visible inside an integrated page view.

Graphic 11: Messaging Objects



## Systems and Methods for Presenting Information on Mobile Devices

### 2: Personalization and Temporal Adoption

Personalization has become a prerequisite for any comprehensive consumer-facing mobile solution. It consists of two customization methodologies.

▶ **Adoption**

The choices, navigation options, etc. are based on user usage patterns.

▶ **Customization**

The user can select which skins, choices, layouts, dynamic content, widgets, etc. are available either through a customization on the phone or one that is on the desktop but dynamically linked to the user's other internet connected devices.

Personalization also must be extended to include temporal adoption. That means that the skins, choices, layouts, content, widgets, etc. are further influenced by:

▶ **Location**

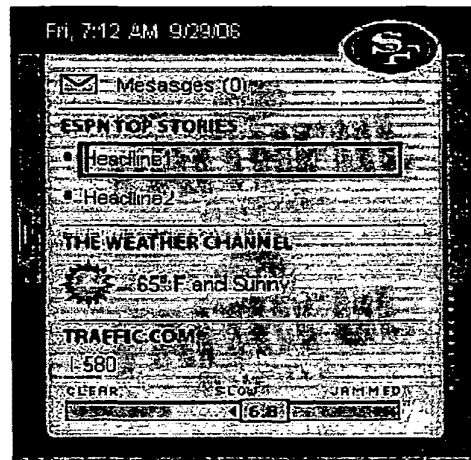
Information requirements will significantly change based on whether the user is at home, at the office or when travelling.

▶ **Time of Day**

Information requirements could also significantly change by the day of the week, which season of the year, and the time of day.

For example, a possible phone top layout for a workday when a user wakes up might look like Graphic 12 to the left. Any messages received overnight would be flagged, the user's favorite RSS sports feeds would be visible, today's weather forecast would be available, and the current traffic conditions between the user's home and office would be graphically depicted.

Graphic 12: Temporal Example



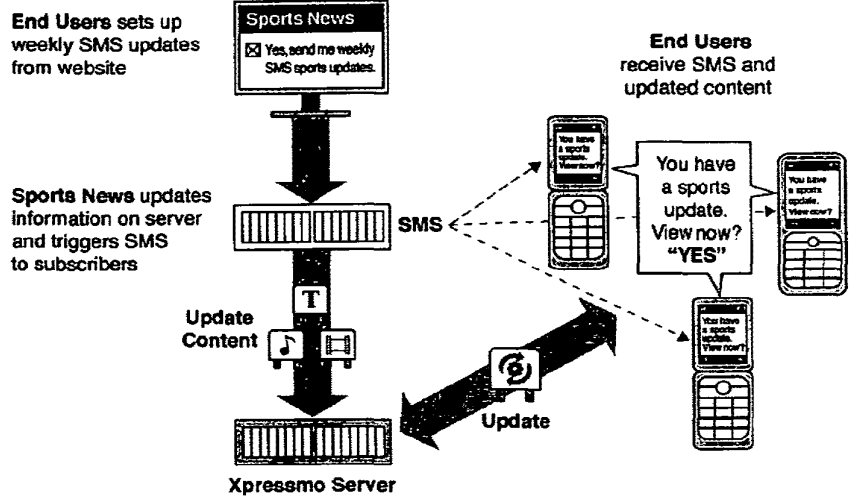
Systems and Methods for Presenting Information on Mobile Devices

**3: Push Capable**

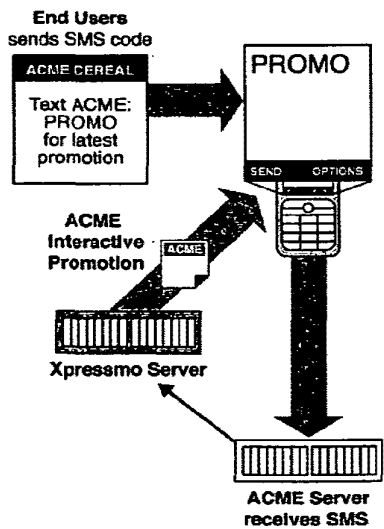
Most agencies and brands want more than just rich, interactive content. They want to reach, acquire, and retain customers through the mobile channel. This requires a sophisticated back-end database, as well as push and pull technology. The suggested platform must work with both.

For example, short codes can be applied to cereal boxes and beverage containers, and SMS text fields can be applied to promotional websites. In either case, a consumer can text to a vendor SMS server, which then serves the appropriate application link back to the consumer. (Graphics 13A and 13B)

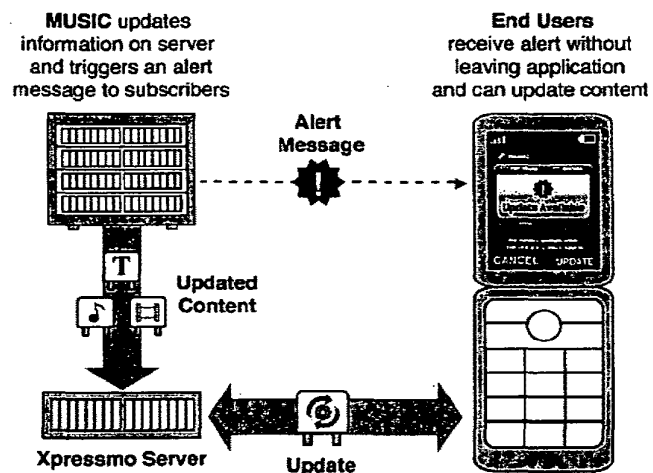
**Graphic 13A: Content Pushing using SMS Example 1**



**Graphic 13B: Content Pushing using SMS Example 2**



**Graphic 14: Content Pushing using Rich Media Multi-layer Messaging**



(Graphic 14) A more powerful example is utilizing the Rich Media Multi-layer Messaging Architecture so that powerful and entertaining messages can be broadcast to any logical group of devices. The consumer is notified in real-time of the pending alert, and can view and interact with the message without interrupting the current application and view.

## Systems and Methods for Presenting Information on Mobile Devices

### 4: Tight Integration with Phone Functions

As described above as a prerequisite for messaging and social networking, there are a large and expanding set of phone based resources that must be tightly integrated with any consumer facing application to be a viable robust solution. This list will continue to grow, but it already includes:

- ▶ Placing and receiving phone calls
- ▶ Access to the phone's PIM data
- ▶ Access to the phone's camera

### 5: Mashups

The efficient and robust integration of different web services to create a value-added seamless user experience should be an inherent attribute of the Authoring Platform. Some of the facilities that support mashups are:

- ▶ **The Feed Collector**

Utilizing a vendor supplied rules based populator, dynamic content from different feeds and/or web services can be logically consolidated.

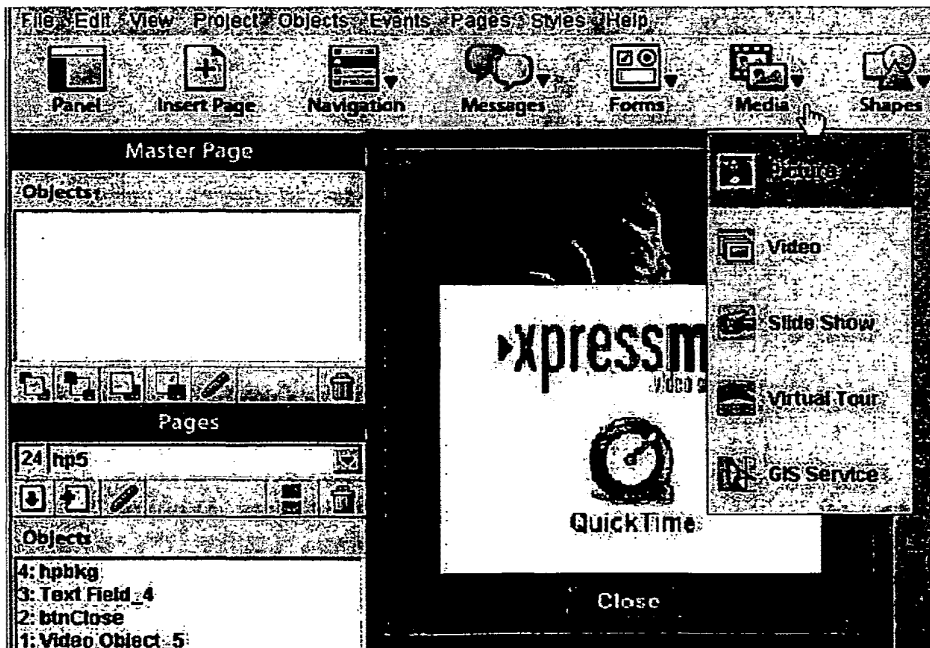
- ▶ **Extensible Attributes for Complex Objects**

Slide shows, virtual tours, GIS Service objects, and child objects, can consolidate content in a logical way, whether it be text, video, audio, pictures and/or vectors.

- ▶ **Vendor Supplied Web Services**

Services such as dynamic binding and Server Pages further help facilitate the mashup of logically related dynamic content.

Graphic 15: Web Services Objects










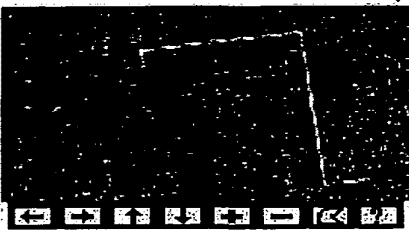
## Systems and Methods for Presenting Information on Mobile Devices

### **Example: Location Awareness Integration with all Relevant Services**

More and more social networking applications require the use of geographical information services (GIS) and location-based services (LBS) as integral components. For example, a social networking application may detect the location of "friends" within a certain radius of the mobile device using GIS and phone triangulation (location based on the three nearest mobile cell towers near the respective phone) or through the GPS received on the phone. GIS images may be used for mapping to theaters or restaurants, detect optimal travelling routes based on traveling conditions, etc. A broad array of mobile GIS web services have been built into the proposed delivery platform for location-aware applications.





Graphic 16A and Graphic 16B are two examples of possible implementations.

#### **Graphic 16A: LBS Optimal Travel Route**

<p><b>Step 1</b></p> <p>Select the  icon or fire after moving a soft cursor to zoom in to the desired geography.</p>	
<p><b>Step 2</b></p> <p>Request directions from two User-Defined geospatial points. Select the  icon and move cursor to a desired start-point 'A'. Move your cursor to a desired end-point 'B'.</p>	
<p><b>Step 3</b></p> <p>To get the directions, select  icon again</p>	
<p><b>Result</b></p> <p>You will start off with a raster image of a map with two points selected. The optimal route will be displayed as a green vector graphic on top of the raster image of the map. You can zoom in closer to the optimized directions for a closer look. You can also see points of Interest, by viewing a 360-degree virtual tour. See Graphic 17 LBS Enhanced Services</p>	

Systems and Methods for Presenting Information on Mobile Devices

**Graphic 16B: LBS Enhanced Services Demo**

<p><b>Step 1</b></p> <p>The user could, as a short cut, select a point of interest through a button or any other navigation object.</p>	
<p><b>Step 2</b></p> <p>Alternatively, the user could select the  icon or fire after moving a soft cursor to zoom in to the desired geography so that the point of interest becomes visible and is selected. Or the user could enter in or select an address. The user could then fire on the button labeled 'Take Tour'</p>	
<p><b>Step 3</b></p> <p>Immediately a virtual tour of the point of interest becomes available in which the user can now pan and zoom throughout the virtual representation of the point of interest.</p> <p>Alternatively, if a video cam is operational, a live video feed could be displayed.</p>	

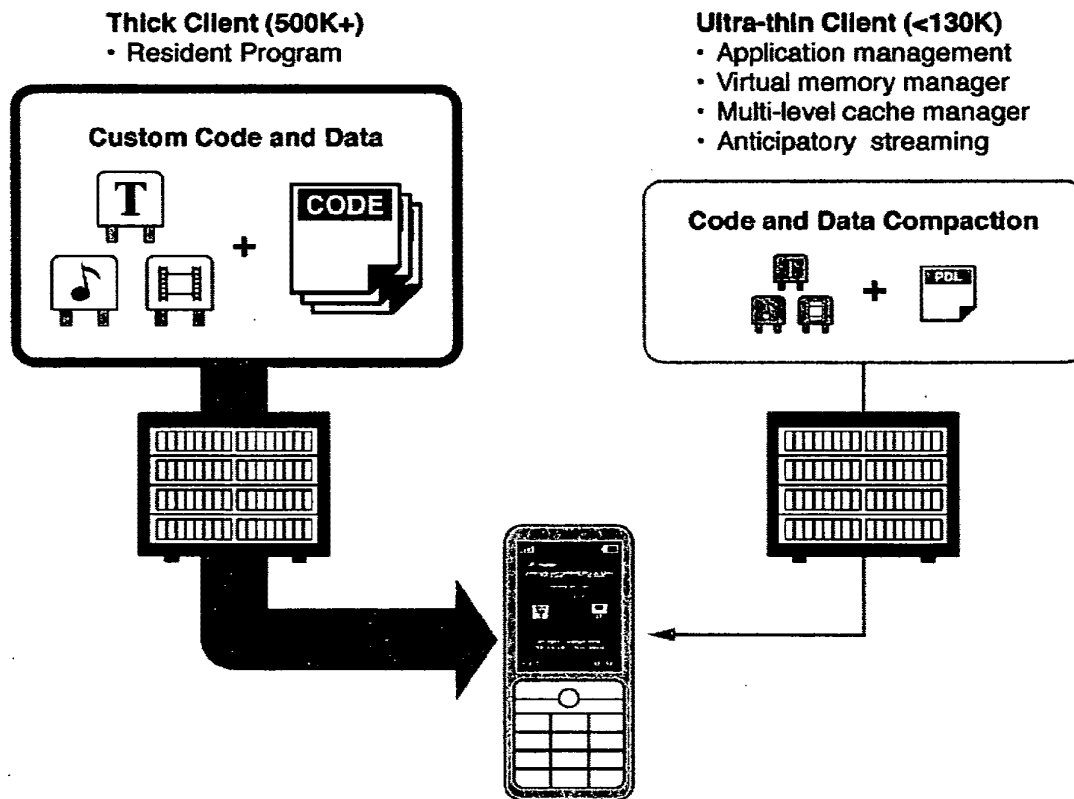
## E: Maximum Distribution to Internet Connected Devices

### 1: Removal of Most Device & Network Constraints

#### Thick vs. Thin Client

Most current implementations that support rich and highly interactive user experiences are computer programs that are written either to the Operating System (OS) or Virtual Machine (VM) of the device. From a development point-of-view this is a far quicker path to take. However, it means that all significant changes to the application require reprogramming the application and then complete the required Q/A cycle, a process that can take several months. Also, for feature phones, the entire application must be downloaded and placed into the limited heap of the device. For any robust and full featured application this design will, at best, quickly restrict its use to a very limited number of devices.

Graphic 17: Thick vs. Thin Client



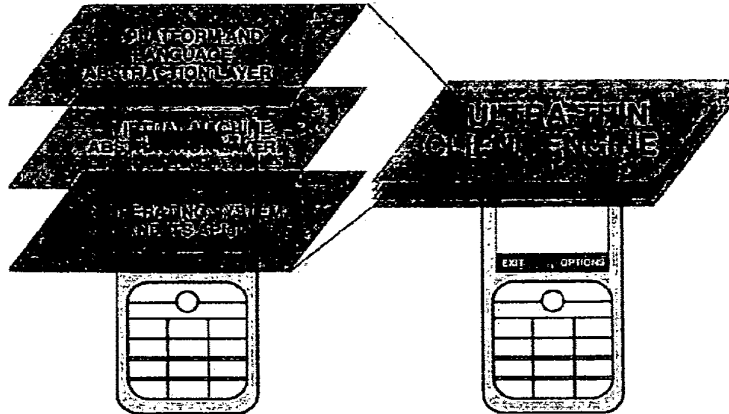
## Systems and Methods for Presenting Information on Mobile Devices

As mentioned before, the Proposed Platform is based on an ultra-thin client architecture. The Client is light weight and extends the OS and/or VM of the device to:

- ▶ Extend the capability of the device to that of a desktop computer
- ▶ Manage all applications and application upgrades
- ▶ Resolve device, OS, VM and language fragmentation

A thin client architecture requires that a language be adopted that manages resources efficiently, is extensible, supports a robust application model, and has no device specific dependencies.

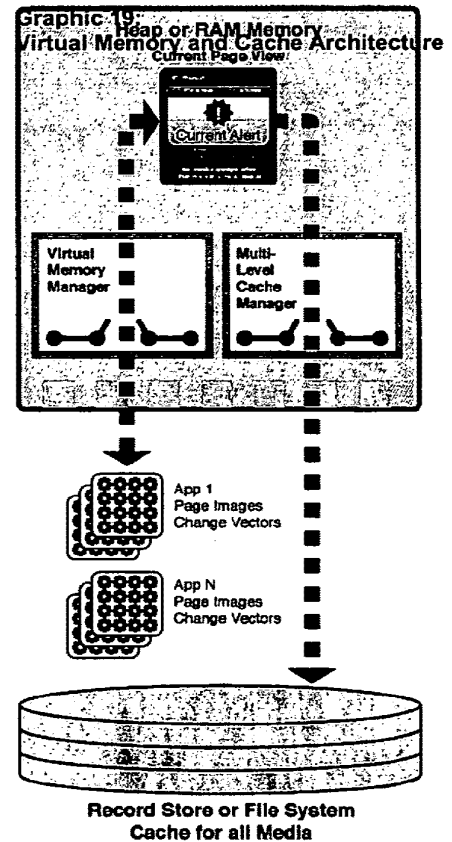
Graphic 18: Ultra-thin Client Engine and its extensions



## Virtual Memory and Multi-Level Cache Architecture

### Virtual Memory

Desktop systems all support virtual memory by using a physical page model with supporting hardware. The proposed platform should implement a logical page virtual memory manager. This architecture requires no supporting hardware and works efficiently with constrained devices. All page view images, which could span multiple applications, are placed in a table as highly compacted and compressed code. A typical page view will range from 500 bytes up to about 1,500 bytes. When rolled into the heap and instantiated this code increases to the more typical 50,000 up to 250,000 bytes. Additional alert pages may also be rolled into the heap and superimposed on the current page view. Any changes to any page currently downloaded are placed in a highly compact change vector for each page, and rolled out when the page is discarded. Note that whenever an application is visited that had previously been placed in virtual memory the Server is interrogated to see if a more current version is available, and, if so, downloads it. This means that application logic can be changed in real-time and the results immediately available to mobile devices.



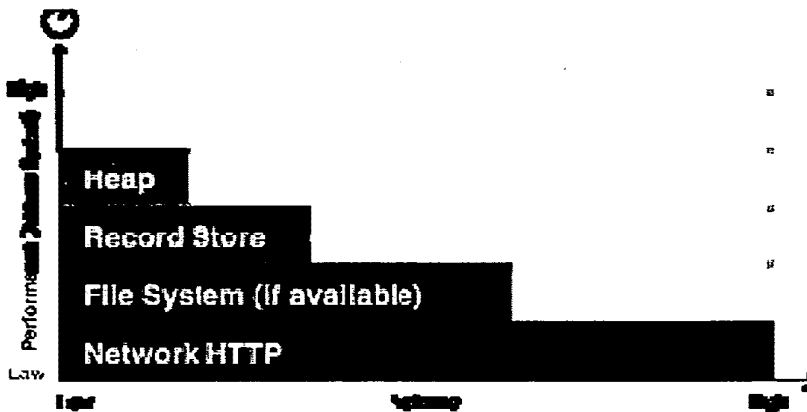
## Systems and Methods for Presenting Information on Mobile Devices

### Anticipatory Streaming and Multi-Level Caching

To operate efficiently with the bandwidth constraints of mobile devices, this delivery platform also features anticipatory streaming™ and multi-level caching. Anticipatory streaming looks ahead in the current application to see if there is content that is likely to be visited in yet untouched page views.

With the multi-level caching system, mobile applications are bounded so that the handsets don't run out of memory. Multi-level caching is a memory management system with results similar to embedding, without the overhead of instantiating the content. In other words, with multi-level caching, handset users get an "embedded" performance without the embedded download. Multi-level caching determines the handset's heap through an API, and also looks at the record store to see how much memory is resident. This content is placed in record store and/or the file system, and may, if there is available heap, also place the content there as well. Note that when content is flagged as cacheable and is placed in persistent storage, a digital rights management (DRM) solution will be used.

Graphic 20: Memory Speed vs. Volume



### Code and Data Compaction

This delivery platform uses compaction to transform the code and data in an intelligent way while preserving all of the original classes, methods and attributes. This requires both an intelligent server engine and client (handset) player, both of which fully understand what the data means and how it will be used.

This compaction technology should include transformation algorithms that deconstruct the logic and data into their most primitive representations, and then reassembles them in a way that can be optimally digested by further compression processing. This reassembled set of primitive representations define the proposed PDL.

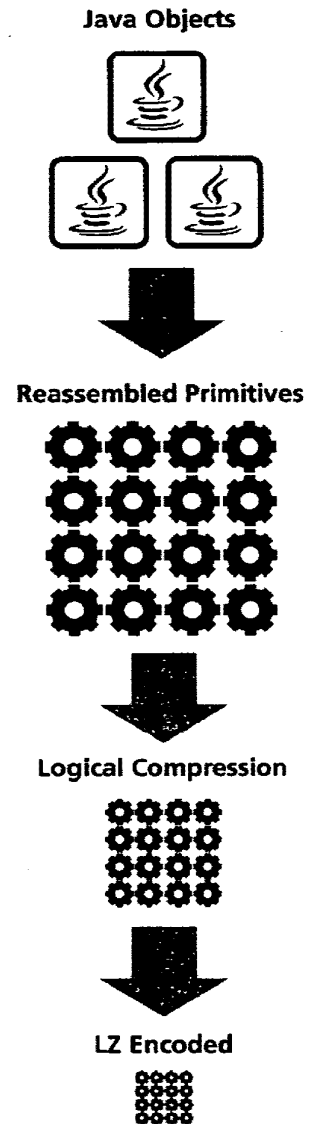
Prior to compression the code has already been transformed so that:

- ▶ There are no dependencies on the original programming language (Java)
- ▶ The code and data have been reduced by 4 to 10 times

Compression has two distinct phases. The first takes advantage of how the primitive representations had been assembled, while the second utilizes standard LZ encoding.

Docket: XPR002PR

Graphic 21:  
Code and Data Compaction



## Systems and Methods for Presenting Information on Mobile Devices

The final result is an overall reduction of 40 to 100 times the original size as represented by Java serialized objects.

The player, when preparing a page view for execution, will decompress and then regenerate the original objects, but this time in compliance with the programming APIs exposed by that device.

## 2: Solution for Device Fragmentation

### Response Director for wide coverage of the wireless device market

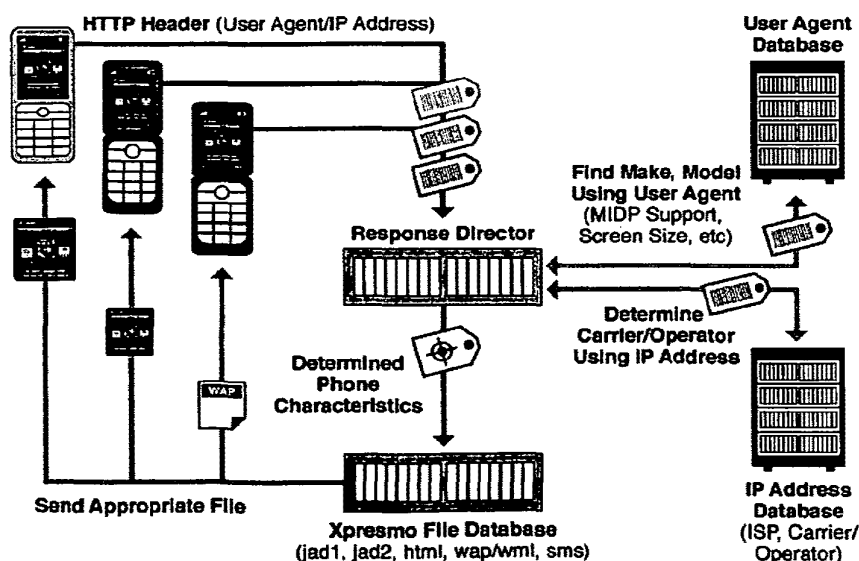
The heart of this proposed deployment platform resides in its Response Director, which determines a user's handset, fetches the correct application from different databases, and delivers a respective highly compressed application in a Postscript-like Portable Description Language (PDL) format over the air (OTA).

With the Response Director, any mobile device can be serviced, and the most appropriate application for the device will be delivered to the device, based on the characteristics of the device.

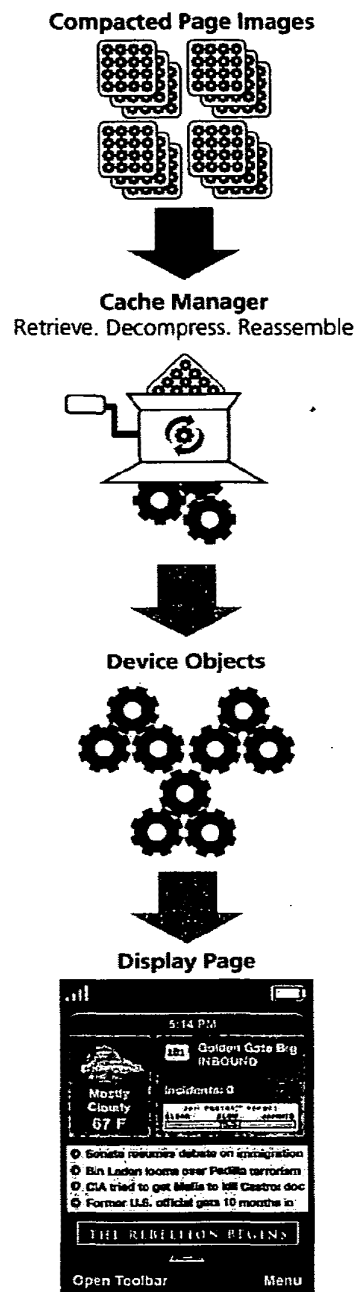
### How the Response Director Works

Let's take a look at how it works. First, an SMS message is generated to the mobile device, which then automatically sends an http:// stream that includes handset information and its phone number to the Response Director. The Response Director then looks at a field in the http header (which includes the user agent and IP address) that identifies the browser (i.e., the User Agent). The User Agent prompts a database lookup table that brings back a slew of data (for example, make, model, attributes, MIDP 1.0 MIDP 2.0, WAP). The database also distinguishes the same models from different countries. There is another IP address database, which identifies the carrier/operator. A decision tree determines which application to fetch and send to the handset.

Graphic 23: Response Director



Graphic 22: Decompression



## Systems and Methods for Presenting Information on Mobile Devices

Another Vendor supplied database should contain the data types such as jad1, jad2, html, wml/wap2, and so on. A list of available applications are returned to the decision tree, which then returns, to the handset, the application that is appropriate for the respective handset. For each file type, there is an attributes list (e.g., streaming video, embedded video, streaming audio, etc.) so that there is enough information to determine what to send to the handset.

If there is an application that has a data type that the handset cannot support, for example, video, the Response Director sends an alternative application to the handset, for example one that has a slide show instead. If the device cannot support a slide show, an application might have text and images and display a message that indicates it does not support video.

Another proposed feature of the Response Director is its exposed API from the decision tree that permits the Vendor solution providers to override the default output of the decision tree. The Vendor solution providers will be given a choice of applications and then can decide to use the defaults or force other applications.

### **Device Scaling for Reduced Fragmentation**

One of the most visible forms of fragmentation resides in the various form factors of wireless, and particularly mobile, devices, which range from 128x128, 176x208, 240x260, 320x320, and many other customized sizes in between. Often, developers must create hundreds of builds for one mobile application.

This proposed platform should automatically scale applications at publishing time to various form factors to reduce the amount of fragmentation among devices, and the Response Director would then serve the appropriately scaled version to the device. For example, a QVGA application will automatically scale to the QCIF form factor.

### **Player Abstraction and Device Adaptation for Eliminating Fragmentation**

The proposed player architecture includes an abstraction interface that separates all device, operating system and virtual machine dependencies from the player's application model business logic.

The advantages of this abstraction interface are:

- ▶ Porting to other operating systems and virtual machines is far more efficient.
- ▶ Adding extensions to the application model and PDL can be implemented once and then seamlessly propagated across all platform implementations.
- ▶ Less robust platforms can be augmented by extending higher end capabilities inside that platform's abstraction interface implementation.

The players also extend the power of the Response Director by adapting the application to the resources and limitations of any particular device.

Some of these areas of adaptation include:

- ▶ The speed of the device's microprocessor.
- ▶ The presence of device resources such as cameras and touch screens.
- ▶ The Heap, Record Store and File System memory constraints.

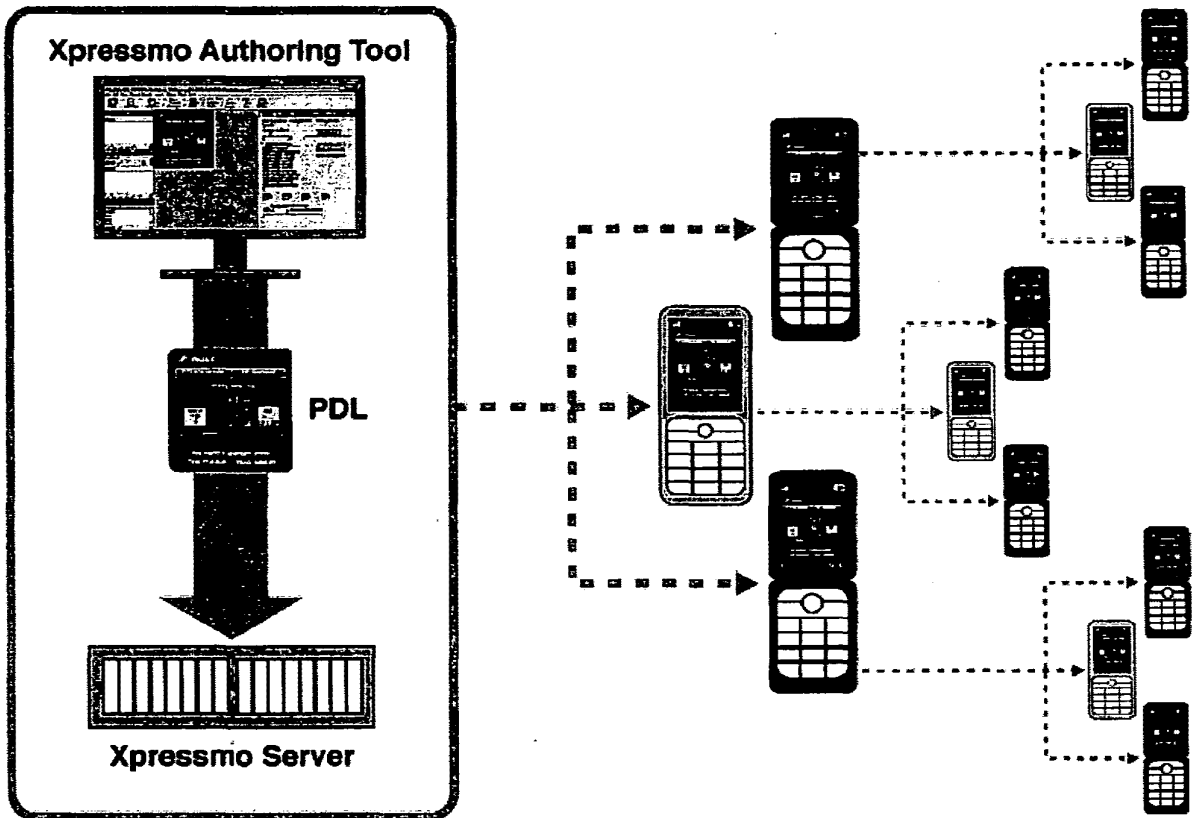
For example, the player will automatically throttle down an animation to the frame rate that the device can handle so that the best possible user experience is preserved.

Systems and Methods for Presenting Information on Mobile Devices

**Automatic Deployment around the Globe**

Finally, the Response Director can automate the deployment of applications across national boundaries, supporting both traditional and double byte Asiatic languages.

**Graphic 24: Reduce Fragmentation**





Systems and Methods for Presenting Information on Mobile Devices

**3: Application Model Support for Operator Extensions and Advanced JSRs**

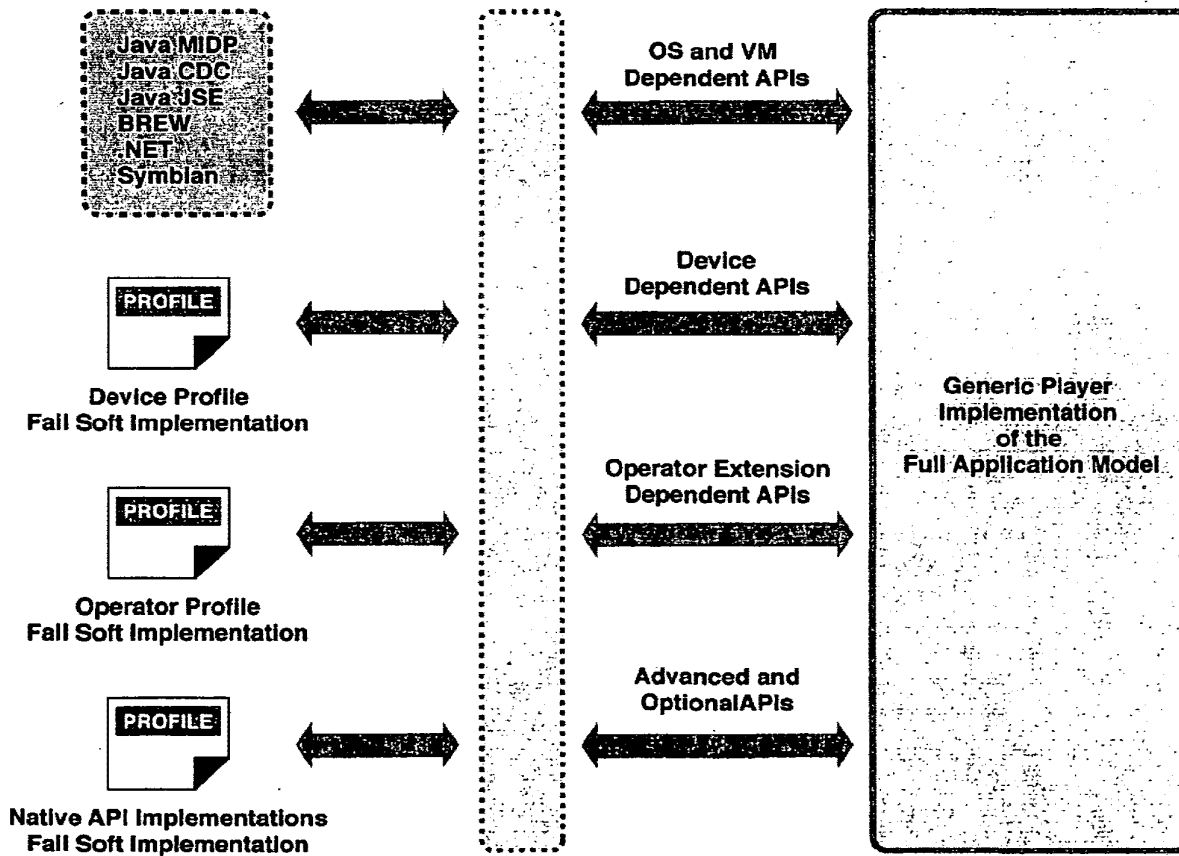
Over time, the operators, device manufacturers, operating system and virtual machine vendors, responding to market demand, will increase the robustness of the IP Stack. This is already being seen as numerous optional JSRs have been released that extend the power of the Java MIDP virtual machine, new versions of operating systems with ever increasing capability are now available on mobile devices, and operators such as DoCoMo, Vodafone and Sprint have released numerous extensions that are specific to their networks.

The proposed platform should have a facility, when it makes business sense, to extend its application model to support these new capabilities, without fragmenting application deployment over legacy devices that do not support these capabilities. Generally, this would be accomplished by using the following criteria to determine which new capabilities to add to the application model.

- ▶ Is this capability useful to a broad segment of the market?
- ▶ Is there a reasonable comparable or fail-soft implementation that can be added to the players for devices that do not support this capability?

The processes below are largely automated in order to efficiently generate and support an expanding data base of Player Profiles.

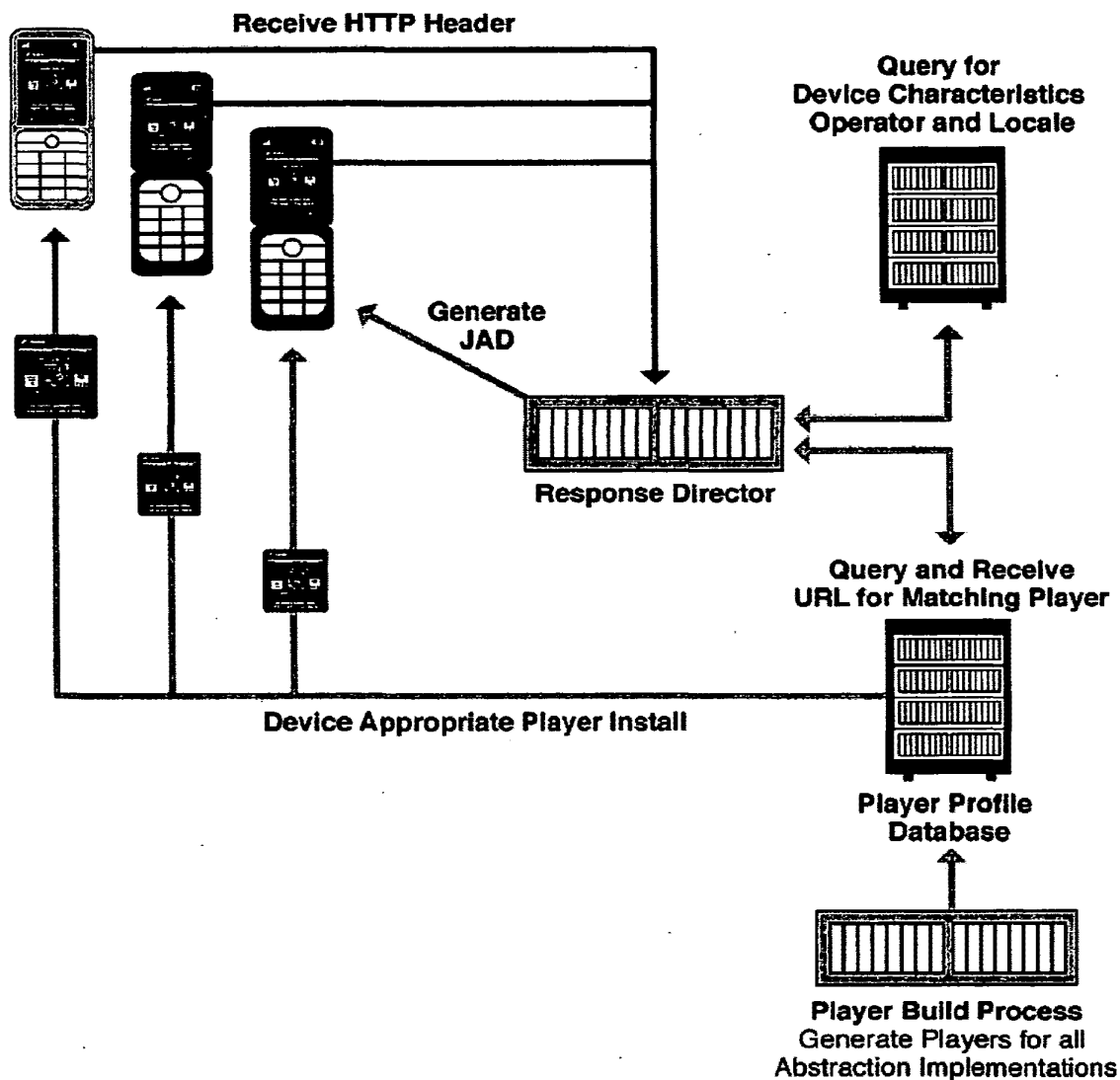
Graphic 25: Player Profiles



## Systems and Methods for Presenting Information on Mobile Devices

As discussed previously, the Response Director manages the deployment of all players.

Graphic 26: Player Deployment



This does not preclude the support for advanced capabilities when there is no reasonable way to implement some reasonable representation on legacy devices. It does, however, infer that:

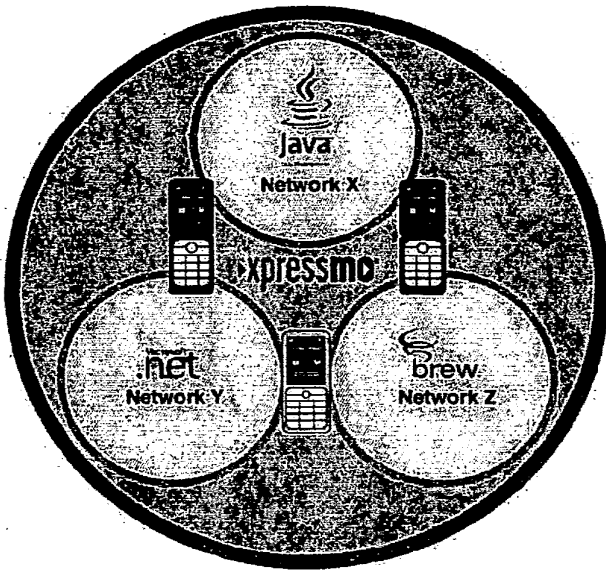
- ▶ Legacy devices will, when receiving PDL instructions that it cannot support, ignore those instructions.
- ▶ The vendor, or its customers, may release advanced versions of this proposed Platform targeted for these more capable environments.

Systems and Methods for Presenting Information on Mobile Devices

**4: Robust and Extensible Cross-Platform Application Model**

- It would not be possible, with modern technology, to have a solution for deploying rich intelligent content to the wide array of 21st Century Internet applications that span set-top boxes, game consoles, PCs, and mobile devices without the implementation of a well conceived Application Model as the conceptual framework for all applications and services.

**Graphic 27: Cross Platform Application Model**



## F: Complete Analytic Reporting

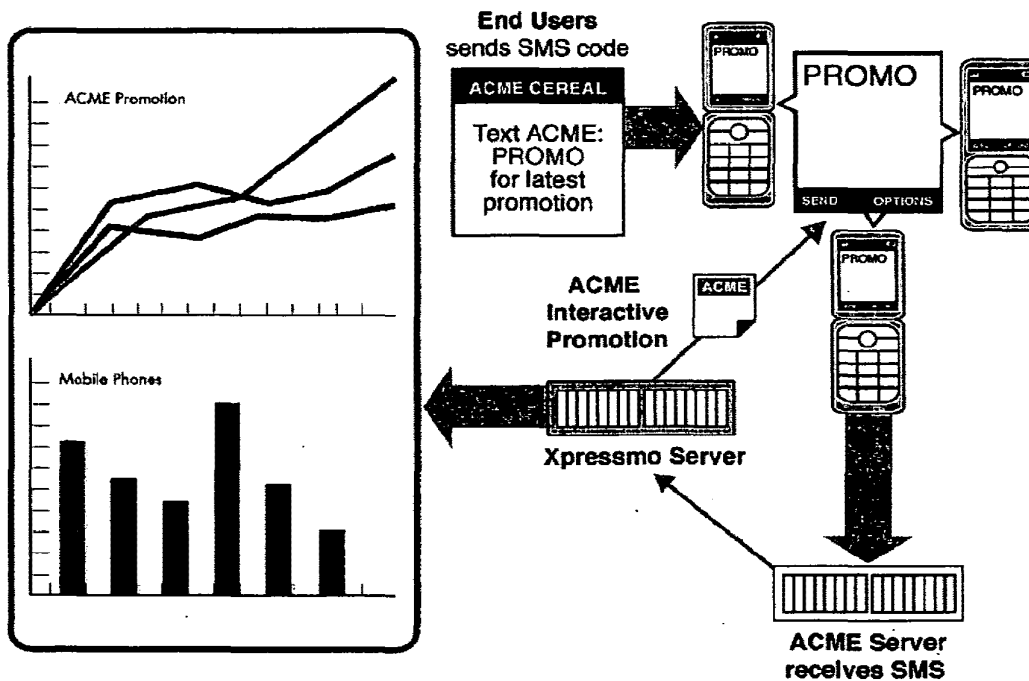
Analytics can be tracked and reported by:

- ▶ Application
- ▶ User Demographic
- ▶ Time of Day
- ▶ Location

In addition, the type of analytic content is really only limited by which listeners have been activated for which objects and for which pages. Some of the choices are:

- ▶ Player-side forms-based content
- ▶ Player-side user interactions
- ▶ Player-side object status
- ▶ Server-side driven dynamic content events

Graphic 26: Reporting and Management



## Systems and Methods for Presenting Information on Mobile Devices

### **G: Enhanced User Experience**

The quality of the user experience will be determined by a number of factors. All of the sections above address various components that will affect the user experience. In summary, they are:

▶ **Minimized Handset Response Time Delay**

This will have been addressed by the suggested Cache and Virtual Memory Manager, anticipatory streaming, compaction of code and data, and the design of the proposed player.

▶ **Navigation**

This includes the various launch strips, graphical lists, intelligent test objects, and the overall power the proposed Event Manager Model.

▶ **Entertaining Presentation Layer**

By empowering the designer and UI Engineer with the Publishing Tool, by offering, through massive analytic reporting the behaviors of the consumer, and with the rapid iteration capability to continue to adapt the presentation layer to consumer needs, the likelihood of success goes way up while the risk of failure is greatly minimized.

▶ **Personalization**

With both extensive customization choices available to the consumer on both the phone and desktop, and with the adaptive technologies based on actual usage, the consumer will find the experience familiar, pleasing and intimate.

▶ **Coverage**

Although this discussion has been primary focused on the feature phone, the consumer already is connected to the internet with office and home PCs, through the television and its STB, and possibly through other mobile devices such as laptops, PDAs, Blackberries, etc. Since this proposed platform can operate on all these devices, and can create an integrated and adaptive experience that will follow the consumer anywhere in this interconnected world, the internet world becomes human centric, not device centric.

## Systems and Methods for Presenting Information on Mobile Devices

### **Appendices**

The following appendices are illustrative of one embodiment of the present invention.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR
		Application Number	
Title of Invention	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES		
The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76. This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.			

**Secrecy Order 37 CFR 5.2**


<input type="checkbox"/>	Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)
--------------------------	---

**Applicant Information:**



<b>Applicant 1</b>					
<b>Applicant Authority</b>		<input checked="" type="radio"/> Inventor		<input type="radio"/> Legal Representative under 35 U.S.C. 117	
				<input type="radio"/> Party of Interest under 35 U.S.C. 118	
<b>Prefix</b>	<b>Given Name</b>	<b>Middle Name</b>	<b>Family Name</b>	<b>Suffix</b>	
	Steven	H.	Rempell		
<b>Residence Information (Select One)</b> <input checked="" type="radio"/> US Residency <input type="radio"/> Non US Residency <input type="radio"/> Active US Military Service					
<b>City</b>	Novato	<b>State/Province</b>	CA	<b>Country of Residence i</b>	US
<b>Citizenship under 37 CFR 1.41(b) i</b>		US			
<b>Mailing Address of Applicant:</b>					
<b>Address 1</b>	38 Washington Street				
<b>Address 2</b>					
<b>City</b>	Novato	<b>State/Province</b>	CA		
<b>Postal Code</b>	94947	<b>Countryi</b>	US		
<b>Applicant 2</b>					
<b>Applicant Authority</b>		<input checked="" type="radio"/> Inventor		<input type="radio"/> Legal Representative under 35 U.S.C. 117	
				<input type="radio"/> Party of Interest under 35 U.S.C. 118	
<b>Prefix</b>	<b>Given Name</b>	<b>Middle Name</b>	<b>Family Name</b>	<b>Suffix</b>	
	David		Chrobak		
<b>Residence Information (Select One)</b> <input checked="" type="radio"/> US Residency <input type="radio"/> Non US Residency <input type="radio"/> Active US Military Service					
<b>City</b>	Pleasant Hill	<b>State/Province</b>	CA	<b>Country of Residence i</b>	US
<b>Citizenship under 37 CFR 1.41(b) i</b>		US			
<b>Mailing Address of Applicant:</b>					
<b>Address 1</b>	132 Shadowood Drive				
<b>Address 2</b>					
<b>City</b>	Pleasant Hill	<b>State/Province</b>	CA		
<b>Postal Code</b>	94523	<b>Countryi</b>	US		
<b>Applicant 3</b>					
<b>Applicant Authority</b>		<input checked="" type="radio"/> Inventor		<input type="radio"/> Legal Representative under 35 U.S.C. 117	
				<input type="radio"/> Party of Interest under 35 U.S.C. 118	
<b>Prefix</b>	<b>Given Name</b>	<b>Middle Name</b>	<b>Family Name</b>	<b>Suffix</b>	
	Ken		Brown		
<b>Residence Information (Select One)</b> <input checked="" type="radio"/> US Residency <input type="radio"/> Non US Residency <input type="radio"/> Active US Military Service					
<b>City</b>	San Martin	<b>State/Province</b>	CA	<b>Country of Residence i</b>	US

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>	Attorney Docket Number	XPR.002PR
	Application Number	
Title of Invention	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES	

Citizenship under 37 CFR 1.41(b) i	US		
<b>Mailing Address of Applicant:</b>			
Address 1	2485 Church Avenue		
Address 2			
City	San Martin	State/Province	CA
Postal Code	95046	Countryi	US
All Inventors Must Be Listed - Additional inventor information blocks may be generated within this form by selecting the <b>Add</b> button.			

**Correspondence Information:**

Enter either Customer Number or complete the Correspondence Information section below. For further information see 37 CFR 1.33(a).			
<input type="checkbox"/> An Address is being provided for the correspondence information of this application.			
Customer Number	40280		
Email Address	svosen@PhDpatents.com		

**Application Information:**

Title of the Invention	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES		
Attorney Docket Number	XPR.002PR	Small Entity Status Claimed	<input checked="" type="checkbox"/>
Application Type	Provisional		
Subject Matter	Utility		
Suggested Class (if any)		Sub Class (if any)	
Suggested Technology Center (if any)			
Total Number of Drawing Sheets (if any)		Suggested Figure for Publication (if any)	

**Publication Information:**

<input type="checkbox"/> Request Early Publication (Fee required at time of Request 37 CFR 1.219)
<input type="checkbox"/> <b>Request Not to Publish.</b> I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application <b>has not and will not</b> be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

**Representative Information:**

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32). Enter either Customer Number or complete the Representative Name section below. If both sections are completed the Customer Number will be used for the Representative Information during processing.			
Please Select One:	<input checked="" type="radio"/> Customer Number	<input type="radio"/> US Patent Practitioner	<input type="radio"/> Limited Recognition (37 CFR 11.9)



Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR	
		Application Number		
Title of Invention	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES			
Customer Number	40280			

### Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, or 365(c) or indicate National Stage entry from a PCT application. Providing this information in the application data sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78(a)(2) or CFR 1.78(a)(4), and need not otherwise be made part of the specification.

Prior Application Status				<a href="#">Remove</a>
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)	

Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the **Add** button.

### Foreign Priority Information:

This section allows for the applicant to claim benefit of foreign priority and to identify any prior foreign application for which priority is not claimed. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55(a).

				<a href="#">Remove</a>
Application Number	Country <sup>i</sup>	Parent Filing Date (YYYY-MM-DD)	Priority Claimed	
			<input type="radio"/> Yes <input checked="" type="radio"/> No	

Additional Foreign Priority Data may be generated within this form by selecting the **Add** button.

### Assignee Information:

Providing this information in the application data sheet does not substitute for compliance with any requirement of part 3 of Title 37 of the CFR to have an assignment recorded in the Office.

**Assignee 1**

If the Assignee is an Organization check here.

Prefix	Given Name	Middle Name	Family Name	Suffix

**Mailing Address Information:**

Address 1			
Address 2			
City		State/Province	
Country <sup>i</sup>		Postal Code	
Phone Number		Fax Number	
Email Address			

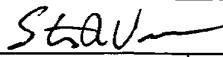
Additional Assignee Data may be generated within this form by selecting the **Add** button.

### Signature:

A signature of the applicant or representative is required in accordance with 37 CFR 1.33 and 10.18. Please see 37 CFR 1.41(a) for the form of the signature.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR	
		Application Number		
Title of Invention	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES			

Signature				Date (YYYY-MM-DD)	2008-04-07
First Name	Steven	Last Name	Vosen	Registration Number	45186

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

040708  
14023 U.S. PTO

PTO/SB/16 (10-07)

Approved for use through 06/30/2010. OMB 0651-0032  
U.S. Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

**PROVISIONAL APPLICATION FOR PATENT COVER SHEET - Page 1 of 2**

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53(c).

Express Mail Label No. ED 707635166 US

**U.S. PTO**  
**61/123438**  
**04/07/2008**

INVENTOR(S)		
Given Name (first and middle [if any])	Family Name or Surname	Residence (City and either State or Foreign Country)
Steven H.	Rempell	Novato, CA
David	Chrobak	Pleasant Hill, CA
Ken	Brown	San Martin, CA

Additional inventors are being named on the \_\_\_\_\_ separately numbered sheets attached hereto

**TITLE OF THE INVENTION (500 characters max):**

**SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES**

Direct all correspondence to: **CORRESPONDENCE ADDRESS**

The address corresponding to Customer Number: 40280

**OR**

Firm or Individual Name

Address

City	State	Zip
Country	Telephone	Email

**ENCLOSED APPLICATION PARTS (check all that apply)**

Application Data Sheet. See 37 CFR 1.76  CD(s), Number of CDs \_\_\_\_\_

Drawing(s) Number of Sheets \_\_\_\_\_  Other (specify) \_\_\_\_\_

Specification (e.g. description of the invention) Number of Pages 251

**Fees Due:** Filing Fee of \$210 (\$105 for small entity). If the specification and drawings exceed 100 sheets of paper, an application size fee is also due, which is \$260 (\$130 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

**METHOD OF PAYMENT OF THE FILING FEE AND APPLICATION SIZE FEE FOR THIS PROVISIONAL APPLICATION FOR PATENT**

Applicant claims small entity status. See 37 CFR 1.27. \$625.00

A check or money order is enclosed to cover the filing fee and application size fee (if applicable).

Payment by credit card. Form PTO-2038 is attached **TOTAL FEE AMOUNT (\$)**

The Director is hereby authorized to charge the filing fee and application size fee (if applicable) or credit any overpayment to Deposit Account Number: \_\_\_\_\_ A duplicative copy of this form is enclosed for fee processing.

**USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT**

This collection of information is required by 37 CFR 1.51. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 8 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

**PROVISIONAL APPLICATION COVER SHEET**  
**Page 2 of 2**

PTO/SB/16 (10-07)

Approved for use through 06/30/2010. OMB 0651-0032

U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.


No.

Yes, the name of the U.S. Government agency and the Government contract number are: \_\_\_\_\_

**WARNING:**

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

SIGNATURE



Date April 7, 2008

TYPED or PRINTED NAME Steven R. Vosen

REGISTRATION NO. 45,186

(if appropriate)

TELEPHONE (510) 841-4711

Docket Number: XPR.002PR

## Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Under the Paperwork Reduction Act of 1995 no persons are required to respond to a collection of information unless it displays a valid OMB control number

Effective on 12/08/2004.

Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818).

**FEE TRANSMITTAL  
For FY 2008** Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT (\$) 625.00

**Complete if Known**

Application Number	to be assigned
Filing Date	to be assigned
First Named Inventor	Rempell, Steven H.
Examiner Name	
Art Unit	
Attorney Docket No.	XPR.002PR

**METHOD OF PAYMENT (check all that apply)** Check  Credit Card  Money Order  None  Other (please identify): \_\_\_\_\_ Deposit Account Deposit Account Number: \_\_\_\_\_ Deposit Account Name: \_\_\_\_\_

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

 Charge fee(s) indicated below  Charge fee(s) indicated below, except for the filing fee Charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17  Credit any overpayments

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

**FEE CALCULATION****1. BASIC FILING, SEARCH, AND EXAMINATION FEES**

Application Type	FILING FEES		SEARCH FEES		EXAMINATION FEES		Fees Paid (\$)
	Fee (\$)	Small Entity Fee (\$)	Fee (\$)	Small Entity Fee (\$)	Fee (\$)	Small Entity Fee (\$)	
Utility	310	155	510	255	210	105	
Design	210	105	100	50	130	65	
Plant	210	105	310	155	160	80	
Reissue	310	155	510	255	620	310	
Provisional	210	105	0	0	0	0	\$105.00

**2. EXCESS CLAIM FEES****Fee Description**

	Fee (\$)	Small Entity Fee (\$)
Each claim over 20 (including Reissues)	50	25
Each independent claim over 3 (including Reissues)	210	105
Multiple dependent claims	370	185

Total Claims	Extra Claims	Fee (\$)	Fee Paid (\$)	Multiple Dependent Claims	Fee (\$)	Fee Paid (\$)
_____ - 20 or HP = _____	x _____	= _____	_____	_____	_____	_____

HP = highest number of total claims paid for, if greater than 20.

Indep. Claims	Extra Claims	Fee (\$)	Fee Paid (\$)
_____ - 3 or HP = _____	x _____	= _____	_____

HP = highest number of independent claims paid for, if greater than 3.

**3. APPLICATION SIZE FEE**

If the specification and drawings exceed 100 sheets of paper (excluding electronically filed sequence or computer listings under 37 CFR 1.52(e)), the application size fee due is \$260 (\$130 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets	Extra Sheets	Number of each additional 50 or fraction thereof	Fee (\$)	Fee Paid (\$)
251 - 100 = 151	/ 50 = 4	(round up to a whole number) x 130	= \$625.00	

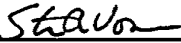
**4. OTHER FEE(S)**

Non-English Specification, \$130 fee (no small entity discount)

Other (e.g., late filing surcharge): \_\_\_\_\_

Fees Paid (\$)

**SUBMITTED BY**

Signature		Registration No. (Attorney/Agent) 45186	Telephone (510) 841-4711
Name (Print/Type)	Steven R. Vosen		Date April 7, 2008

This collection of information is required by 37 CFR 1.136. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 30 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

## Appendix B

# Icon Strip Requirements Specification

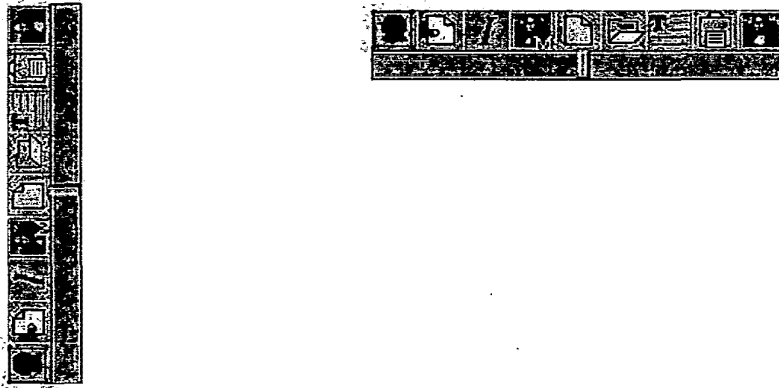
### Introduction

- This document describes the technical requirements for the implementation of an icon strip for enhanced portal navigation.
- For brevity, all references to LEFT and RIGHT would translate to TOP and BOTTOM if the icon strip has a vertical orientation.

### 1: UI

#### *Appearance*

The following illustration shows how the Icon Strip may appear in an application.



### 2: *Page "Icon Strip Aware" Behavior Specifications*

1. If a page is "icon strip aware" then up to 2 icon strips will be available, one for the horizontal edges of the page (top or bottom) and one for the vertical edges (left or right)
2. If an icon strip has a "scrolls" setting (as opposed to a static setting) then a tab will draw in the center of the appropriate page edge to indicate that an Icon strip is available but hidden.
3. A navigation operation (joystick or cursor), which moves the selection in the direction of the icon strip, will now select the icon strip only (the focus is on the icon strip and the centermost icon will have a selection rectangle drawn around it. Any navigation operations for other objects must use the unassigned navigation operations.

## Systems and Methods for Presenting Information on Mobile Devices

4. If the icon strip has a “scrolls” setting and the activation navigation event is detected, the icon strip will slide onto the page as per the settings in “5” below prior to the focus placed on it.

### **3: Icon Strip Behavior Specifications (MIDP Player)**

1. There will always be an Icon Selected if the Icon Strip is selected. That Icon will be in the center of the visible icons.
2. The Selected Icon may have the following attributes:
  - a. A pan transition animation
  - b. A transition audio effect for the entering icon.
  - c. A background audio track. Starts when the Icon is selected, plays or loops until the focus is off that icon
  - d. Bubble help (Will draw below/top, left or right of selected icon.
  - e. A background color for icon separation and end-of-icon strip feedback.
  - f. All exit events that can be assigned to a slide.
3. When the Icon strip is Selected a 3D red rectangle will be drawn around the implicitly selected icon..



4. LEFT and RIGHT actions of the Joy Stick will select the next icon as follows:
  - a. If LEFT, the icon strip will appear to slide right (if the animation is enabled) with a transition audio playing if assigned. If available, the next icon on the icon strip to the right will appear. When there are less icons available to the left than required to fully populate the icon strip then only the color assigned to the icon will draw in those slots. However, the Icon strip will wrap if the selected icon is first or last and the user performs another navigation event..
  - b. If RIGHT, the icon strip will appear to slide left (if the animation is enabled) with a transition audio playing if assigned. If available, the next icon on the icon strip to the left will appear. When there are fewer icons available to the right than required to fully populate the icon strip, then only the color assigned to the icon will draw in those slots. However, the Icon strip will wrap.
  - c. If bubble help is assigned it will appear just to the right of the selected icon, and remain for 1.5 seconds.
5. FIRE will execute the exit event assigned to the icon. If there is no exit event then the Icon will act as a label, and cannot be selected.



## Systems and Methods for Presenting Information on Mobile Devices

6. If a stylus is available then the additional navigation choices for the generic player will be available.

### **4: *Icon Strip Behavior Specifications (Generic Player)***

1. All MIDP navigation behaviors will be available, with the left and right cursor keys operating in a manner similar to the LEFT and RIGHT toggle stick behaviors.
2. Clicking on the Elevator Strip to the left of the elevator will be equivalent to a LEFT operation. Even if an elevator has been defined for this icon strip, it will only draw when the number of available icons is greater than the number of visible icon slots.
3. Clicking on the Elevator Strip to the right of the elevator will be equivalent to a RIGHT operation.
4. Dragging the elevator will change the selected icon appropriately.
5. Clicking on any visible icon will execute its exit event, if any.
6. If the cursor is placed over any visible icon then the bubble help will appear for 1.5 seconds.
7. Cursor keys and the Enter will be supported. There will be a selection rectangle (see MIDP implementation). Pressing the Enter Key will be analogous to FIRE. The cursor keys will be analogous to the navigation events with the joy stick.

### **5: *Authoring Pages for Icon Strip Awareness***

Pages may or may not be defined as Icon Strip aware pages. If not aware, the icon strip will function as described in this document, but there will be no automatic positional, tab feedback, or entrance and exit animations based on the navigation heuristics defined below. In the Edit Pages dialog box there will be the following additional choices.

Horizontal Icon Strip Mode. It will have the following choices.

- None
- Top Static
- Top Scrolls
- Bottom Static
- Bottom Scrolls

Selected Horizontal Icon Strip Object

If any choice other than “none” is selected, then a drop down will be enabled that will display the object names of all the Icon Strip objects defined for that page. Which ever is selected will now be repositioned on the page at its visible location. (where it will appear

## Systems and Methods for Presenting Information on Mobile Devices

after its entrance (static means that it will be visible all the time.)

Vertical Icon Strip Mode. It will have the following choices.

None  
Left Static  
Left Scrolls  
Right Static  
Right Scrolls

Selected Vertical Icon Strip Object

If any choice other than “none” is selected, then a drop down will be enabled that will display the object names of all the Icon Strip objects defined for that page. Which ever is selected will now be repositioned on the page at its visible location. (where it will appear after its entrance (static means that it will be visible all the time.)

Number of Frames for the Animation. (a drop down with a range from 1-20)

Time for the Animation (from 0.1 seconds to 2 seconds0, in tenth of a second increments.

### **6: *Authoring Icon Strips***

Icon strips are a variant of the Slide Show Object. To change a slide show into an Icon strip take the following steps.

- Right click with the Slide Show selected.
- Choose “Edit” (not “Edit Slides”)
- Set the maximum number of visible slides to greater than 1.
- You can also set the number of pixels that separate adjacent icons.
- You can set whether to draw the Icon Strip horizontally or vertically. Note if this icon strip is selected in a “Icon Strip Aware” page then the orientation will be consistent with that selection, possibly overwriting this setting.

Once set, then editing a slide show will have a few different choices.

- Both “Slide Settings” and “Generic Settings” will only have 2 transition animation choices, “None” or “Slide Icon Strip”. The settings for frames per Second and Transition Time will control its animation behavior.
- “Slide Settings” will have a text field for entering in bubble help. There is currently no restriction as to the number of characters. A single pixel black frame will draw around the help, with the background color being the background color assigned to the currently selected icon.

## Systems and Methods for Presenting Information on Mobile Devices

- “Phlog Settings” already permits the definition of a font. The font has a default setting of Times Roman Bold 9 point (this would map to the Small size on a MIDP phone. This can be changed and will be applied to the bubble help.
- “Generic Settings” already gives the author the choice on whether to draw a slide bar, and 6 different positions relative to the Icon strip.

### **7: New APIs**

In order to facilitate the adaptive and customizable intelligence for the icon strip 2 new APIs will be made available to the backend code. One API permits the PDL to be extracted from an object created by the authoring tool, and the other permits the same function but with the entire definition generated by backend code.

**@param** Function defines the operation. Possible values are APPEND, INSERT or REPLACE.

**@param** objectName is the name for this object.

**@param** pageSource is the location indicator for the page.

**@param** pageName is the page name that the player will be directed to visit.

**@param** insertion position number is the object number for this object. This value is ignored if operation is APPEND.

**@param** SlideShow Library Object PathName is the .avd pathname that defines this SlideShow for extraction of a PDL specification of a particular slide.

**@param** SlideShow Library Object Slide Number is the number of the slide to be extracted.

InsertSlide (Function, objectName, pageName, objectName, insertion position number, SlideShow Library Object PathName, SlideShow Library Object Slide Number)

**@param** Function defines the operation. Possible values are APPEND, INSERT or REPLACE.

**@param** objectName is the name for this object.

**@param** pageSource is the location indicator for the page.

**@param** pageName is the page name that the player will be directed to visit.

**@param** insertion position number is the object number for this object. This value is ignored if operation is APPEND.

**@param** SlideBooleanAttributes is an array of boolean attributes for this slide.

**@param** SlideIntegerAttributes is an array of int attributes for this slide

**@param** SlideStringAttributes is an array of String attributes for this slide.

**@param** SlideColorAttribute is an int value containing the background color for this slide.

InsertSlide (Function, objectName, pageName, objectName, insertion position number, SlideBooleanAttributes[], SlideIntegerAttributes[], SlideStringAttributes, SlideColorAttribute)

## Systems and Methods for Presenting Information on Mobile Devices

# Appendix A PDL SPECIFICATION

## 1 AKI Header

Boolean ScaleToWindow  
Boolean VirtualPageSizeSet  
Int VirtualWidth  
Int VirtualHeight

## 2 PDL Header

Short CurrentFileFormatNumber  
Long ApplicationUUID1  
Long ApplicationUUID2  
String UserName  
**Boolean** ScaleToWindow  
Int VirtualWidth  
Int VirtualHeight  
Boolean EnableAPIs  
Boolean DataBindingEnabled  
Boolean ServerListener  
Boolean LoginRequired  
Boolean RunTimeEventsInterfaceEnabled  
Boolean PhlogConnectionRerquired  
Short StartingPageNumer

## 3 Engine Variables

EngineBoolean[0]=EnableAPIs  
EngineBoolean[1]=HasServerListener  
EngineBoolean[2]=HasDynamicDataConnection,DataBindingEnabled  
EngineBoolean[3]= LoginRequired  
EngineBoolean[4]=IsWebSite  
EngineBoolean[5]=Terminate  
EngineBoolean[6]=initialization  
EngineBoolean[7]=PlayDummyAudio  
EngineBoolean[8]=IsDummyStreamThreadPlaying  
EngineBoolean[9]=PlayAudio  
EngineBoolean[10]=appStarted  
EngineBoolean[11]=appClosed  
EngineBoolean[12]=RunTimeEventsInterfaceEnabled  
EngineBoolean[13]=PhlogConnectionRerquired  
EngineBoolean[14]=SwitchingApplications Armor  
EngineBoolean[15]=Draw Popup Alert  
EngineBoolean[16] [NOT USED]  
EngineBoolean[17]=Do full screen paint.  
EngineBoolean[18]=Two dynamic icon strips.

## 4 Generic Icon Strip Definitions

IconStrip GenericHorizontalIconStrip:  
    SlideShow Object with ObjectIntegerAttribute[17]>0  
IconStrip GenericVerticalIconStrip:  
    SlideShow Object with ObjectIntegerAttribute[17]>0  
GenericHorizontalIconStripName = "HorizontalIconStrip":  
GenericVerticalIconStripName = "VerticalIconStrip":  
boolean VerticalIconStripVisible,HorizontalIconStripVisible:  
    true when visible  
SelectedVerticalIcon, SelectedHorizontalIcon:  
    Currently selected Icon

## 5 WebPage Definitions

### 5.1 BooleanAttribute[] ;

0= Page Audio Exists[0]  
1= Page Transition Audio Exists[1]  
2= Page Icon Exists  
3= has a defined ExternalURL or Exit PageName  
4= Report to Server Listener  
5= Select by Object Navigation  
6= MasterPageOnly Dim Page  
7= Restore page defaults  
    MasterPageOnly (Size by Pixels)  
8= Runtime: Random transition  
9= Runtime: Do exit event  
10= Page transition Animation (transitionlock)  
11= Pause/Resume Page Timeout Toggle  
12= Toggle for going to a specific page on a mouse event  
13= Runtime: mouse listeners need to be added toggle  
14= Is a complex page  
15= MIDP wait state while page transition image is being constructed  
16= IsDynamic (An XpressMo Server Page)  
17= If false draw full page for MIDP Player.  
18= Draw Page Background  
19= Force full screen refresh during an animation  
20= Runtime: load object events toggle  
21= Suppress Horizontal Icon Strip  
22= Suppress Vertical Icon Strip  
23= Suppress BackViewCommand  
24= Draw Horizontal Scroll Bar  
25= Draw Vertical Scroll Bar

### 5.2 IntegerAttribute[] :

0=TransitionType[0]  
    Runtime: MasterPage Only: Light Selection Color top 2 bytes  
1=TransitionTime[1]  
    Runtime: MasterPage Only: Light Selection Color bottom 2 bytes  
2=TransitionFrame[2]  
    Runtime: MasterPage Only Dark Selection Color top 2 bytes  
3=BoardTime[3] (30000=stay)  
    Runtime: MasterPage Only: Dark Selection Color bottom 2 bytes

## Systems and Methods for Presenting Information on Mobile Devices

- 4=Page Icon Width
- 5=Page Icon Height
- 6=PageType
  - 1=MasterPage
  - 0=Normal Application Page (resides in the Apps PDL)
  - 1=Alert Page Non-Modal
  - 2=Alert Page Non-Modal with Timeout
  - 3= Alert Page Modal
  - 4= Alert Page Modal, Fire Enabled
  - 5=Rich Media IM
- 7=ExitMode
  - 0=Next Web Page
  - 1=External Page
  - 2=External Page in Window
  - 3=Specific Web Page
  - 4=External Page in Top Frame
  - 5=Call a JavaScript Method
  - 6=Execute an XpressMo Application
  - 7=Goto Start Page
- 8=PageAudioMode[8]
  - 0=Inactive
  - 1=Play while Active
  - 2=Loop while Active
  - 3=Stream while Active
- 9=PageTransitionAudioMode[9]
  - 0=Inactive
  - 1=Play while Active
  - 2=Loop while Active
  - 3=Stream while Active
- Runtime: MasterPage Only: Dark Selection Color
- 10=Page Width
- 11=Page Height
- 12=Runtime: Number of objects on page that require Event Manager Thread
- 13= Horizontal IconStrip Mode
  - 0=Convert to SlideShow
  - 1=Top Static
  - 2=Bottom Static
  - 3=Top Scrolls
  - 4=Bottom Scrolls
- 14 = Vertical IconStrip Mode
  - 0= Convert to SlideShow
  - 1=Left Static
  - 2=Right Static
  - 3=Left Scrolls
  - 4=Right Scrolls
- 15= Frame delay time (in milliseconds)
- 16= Number of Animation Frames
- 17=Normal Application Page Number of Suppressed MasterPage Objects that are suppressed.  
Master Page Only: Used to Generate the light and dark Selection Rectangle Colors
- 18= Normal Page: Current Scroll Horizontal Origin  
MasterPage: Number of Icons on Horizontal IconStrip that are initially being set to visible (all or up to 20) 0=all;
- 19= Normal Page: Current Scroll Vertical Origin  
MasterPage: Number of Icons on Vertical IconStrip that are initially being set to visible (all or up to 20) 0=all;
- 20=Number of Objects on Page (used by interface)

## Systems and Methods for Presenting Information on Mobile Devices

21=build: page Number  
runtime: Ticker  
22= runtime: pagenumber  
23= for horizontal translation of graphics context with MIDP centering  
24= for Vertical translation of graphics context with MIDP centering  
25= Current Selection x origin  
26= Current Selection y origin  
27= Current Selection width  
28= Current Selection height  
29=Previous Selected object Child Count  
30= Current canvas x origin  
31= Current canvas y origin

### **5.3 StringAttribute []**

0=Page Audio Object[0]  
1=Page Transition Audio Object[1]  
2=BackgroundImageURL  
3=ExternalURL, Exit PageName  
4=PageName  
5=Horizontal IconStrip Object Name  
6=Vertical IconStrip Object Name  
7=Runtime (Pathname for all Media on Page) (from 6)

### **5.4 PageBackgroundColor**

#### Soft Key Command Attributes

short NumberOfCommands;

Vector SoftKeyCommand()

Vector of Shorts where:

0=none.  
1=goto page  
2=back  
3=goto page w/ Listener  
4=back w/ Listener  
5=goto logical first page  
6=goto logical first page w/ Listener

Vector SoftKeyCommandString()= Vector of Strings for the soft key menu

Vector SoftKeyParameter()=Vector of String Parameters

for executing the command. For "goto page" it is the pagename.

boolean[] hasNumericKeyBindings= new boolean[10];

State of whether the numeric key, as defined by the array index, is bound.

byte[] NumericKeyBindings= new byte[10];

A byte pointer into the SoftKeyCommand() Vector to bind rthat key top a particular SoftKey Command.

ScrollBarDefinition=new int[8];

0=Horizontal Scroll Bar Height  
1=Vertical Scroll Bar Width  
2=ScrollBarBackground  
3=ElevatorBackground  
4=ScrollBarFrame1  
5=ScrollBarFrame2  
6=ElevatorFrame1  
7=ElevatorFrame2

DrawFullPage

## 6 Object Definitions

### 6.1 ObjectBooleanAttribute [] :

- 0=IsDynamicObject
- 1=DrawobjectBackground
  - GIS: Raster Layer Exists
  - SlideShow: Is Selectable (MIDP Player only)
- 2=Dampen Animation
  - GIS: Draw Coordinates
  - SideShow: Phlog Object
  - Image, TextField, Text Area, Choice, List; Restore default upon leaving page.
  - Text Button, Submit and Clear Button: Don't Scale
- 3=Image: isAudio Mute Image
  - TextArea/TextField: true=is selectable
  - List: ~~Don't Scale~~
  - SlideShow: If true Phlog is not editable, fire on Phlog will execute slide exit event
- 4=Image IsScaled
  - SlideShow: Is Vertically Drawn
  - Video: Is Streamed
  - GIS: Is a Mr. Side file
  - VR: Preload all VR Rooms/Rictures/Walls
  - Paragraph/Button: Underlined
  - Polygon/Polyline true=IsPolyline
  - XpressMo Check Box State
  - TextArea/TextField: true=IsEditable
- 5=IsParent
- 6=MousePressTimelineEvent
  - TextArea/TextField: true=scroll begins on select (mouseover)
- 7=EntryAnimationSuppressed
  - SearchResponseList: Title Marquee enabled.
- 8=IsDraggable
- 9= Build: isDeleted (for undo)
  - Runtime: Initial Object Selected
- 10=AvoidCursor
  - Text Field/Text Area/ListObject/Submit/Clear: Draw Frame
- 11=IsHidden (runtime only)
  - SlideShow: Is Animated iconstrip
- 12=Child Click Enabled
- 13=DragState
  - runtime initialization: Parent Object: Child object has a timeline
- 14=Parent: Disable Nephew Visibility
  - Child: ClickState Visibility
- 15=: ObjectListener is Enabled
- 16=SlideShow: Has Placeholder Slides
  - Runtime: Submit Form Listener is enabled.
- 17=build: LineWidthSet
  - runtime: isMouseOver\_
- 18=build: FillColorSet
  - runtime: AnimationClick
- 19=LineColorSet
  - Runtime: MouseOverAudioToggle
- 20=Build: InitialObjectSelected (moved to [9] in player)
  - Runtime: MousePressAudioToggle
- 21=Requires ObjectName for Players



## Systems and Methods for Presenting Information on Mobile Devices

22=Object is locked  
23=Build: Object is a template Object  
Runtime: Activate Timeline from the Server  
24=gis: Show Navigation Icons  
image: DrawingRotatingImage\_  
TextArea: This phlog is selected  
Runtime SlideShow: used for navigation between chat and SS  
false=PhlogStyle==1 (To right or bottom)  
true=PhlogStyle==2 (To left or top)  
DropDown Select From List (Selection list in open state, don't switch to form)  
25=gis: Show Popup menus  
XpressMoChoiceObject is Open  
SlideShow: PhlogSelected  
TextField/TextArea: Draw Insertion Point  
26: Initial State of Checkbox for Clear  
TextArea: True means it's a phlog trail  
SlideShow: True means that the chat Text Entry Field is visible and active  
TextField: True means that the Text Entry Field should be painted  
ChoiceObj: DrawIcon  
27: IsObjectSelected Used for drawing insetion points in form objects  
28: IsObjectTextEntryEnabled  
29=build: CanPaste  
Runtime: Object Requires Drawing  
30: InAnimation  
31: Generic Icon Strip  
32: build: SetInvisible  
49=If True a Dynamic Link has been set for this object  
99=if true cam from style page.

### 6.2 ObjectIntegerAttribute [ ] :

0=ObjectType  
where ObjectType value means:  
0=Paragraph  
1=Button  
2=Polygon  
Type of Polygon  
2= Polygon Library  
6=Free-form Polyline  
7=Free-form Polygon  
8= Polyline Library  
3=MultiShape  
4=GIS  
5=Arc  
Type of Arc  
4=Free-form Arc  
5=Library  
6=Video  
7=Slide Show  
8=VR  
9=Image  
10=game object  
21=Text Edit Field  
22=Polygon Extended  
23=MultiShape Extended  
27=Slide Show Extended

## Systems and Methods for Presenting Information on Mobile Devices

- 32=XpressMo Submit Buttons
  - 33=XpressMo Clear Buttons
  - 34=XpressMo Check Boxes
  - 36=XpressMo Text Field
  - 37=XpressMo Text Area
  - 38=XpressMo Choice
  - 39=XpressMo List
  - 42=XpressMo Submit Buttons Extended
  - 43=XpressMo Clear Buttons Extended
  - 44=XpressMo Check Boxes Extended
  - 46=XpressMo Text Field Extended
  - 47=XpressMo Text Area Extended
  - 48=XpressMo Choice Extended
  - 49=XpressMo List Extended
- 
- 1=Current Drawing Order
  - 2= Text Objects: Vertical Offset  
Polygon & Arc: line width  
GIS: Pan Range  
SlideShow: Maximum number of Visible Slides  
Game: Number of Game Objects  
Text Objects Pixel Mode: Line Height in Pixels
  - 3= Build: ID number (for undo)  
Runtime: Number of child Objects
  - 4=XValue
  - 5=YValue (For Text Objects it represents the baseline of the top line.  
Text Objects Pixel Mode: Top of the first line's bounding box.
  - 6=ObjectWidth
  - 7=ObjectHeight
  - 8=FrameWidth
  - 9=Frame Style
    - 0=none
    - 1=raised
    - 2=pressed
    - 4=solid
  - 10= Button: Origin position  
Arc:=Start Angle  
Polygon:=Rotate Angle  
Paragraph:=# of lines  
XpressMo ChoiceObject=# of items  
Video: Audio Mute
    - 0=play with audio
    - 1=video onlyGIS:=Number of Vector Layers  
SlideShow: Draw Icons and location
    - 0=None
    - 1=Above Slide
    - 2=Straddle Top
    - 3=Top of Slide
    - 4=Below Slide
    - 5=Straddle Bottom
    - 6=Bottom of SlideXpressMo Text Field: Current Position Origin  
XpressMo Text Area: # of lines
  - 11= Button: Format  
format: 1=text

## Systems and Methods for Presenting Information on Mobile Devices

2=HH:MM:SS  
3=DD:MM:YY  
4=DD:MM:YY HH:MM:SS  
5=MM:DD:YY  
6=MM:DD:YY HH:MM:SS  
7=Text Entry

Arc:=Arc Angle

Polygon:=Gradient Fill Type

0=Single Color  
1=Linear Gradient Fill  
2=Radial Fill

Paragraph:=max width as % of screen width

GIS:=Digital Zoom Range

Game:=Communication Mode

0=On Demand  
1=Socket Always Open

XpressMo Choice= Currently Selected Item (-1 if null)

XpressMo Text Field: Current End Visible Position

XpressMo Text Area: First Line Index

SlideShow: Number of pixels separating slides (default is 1)

Video : Video Mode

0= Automatic Play  
1= Automatic Loop  
2= Play on Fire/SoftKey  
3= Loop on Fire/Softkey

12=ExitEvent

0=None

1=Goto External Web Page replacing Current Frame

2=Goto External Web Page Launched in a New Window

3=Goto a specific Internal Web Page

5=Goto the next Internal Web Page

4=Goto External Web Page replacing the Top Frame

6=Execute JavaScript Method

7=Pause/Resume Page Timeout

8=Execute an XpressMo Application

9= Goto a specific Internal Web Page with setting starting slide

10=Exit Application

11=Exit Player

12=Place PhoneCall from linked Text Field

13= Text Field/Area: Send String on FIRE

ChoiceObject : Add Icon to Launch Strip

14= Text Field/Area: Send String on FIRE or Numeric Keys

ChoiceObject : Remove Icon from Launch Strip

15=Goto a specific Internal Web Page with Alert. "Backend Synchronization"

16=Goto Widget Object

17=Generate Alert. "With a Fire Event"

18=Send SMS Message from Linked Text Field

19=Toggle Alert. "Display OnFocus, Hide OffFocus"

20= Execute an XMO Application with Alert. "With a Fire Event"

21=Goto Logical First Page

22= Generate Alert with Backend Synchronization

23=Send SMS Message with Share (Player Download)

24= Place PhoneCall from linked Text Field with Share (Player Download)

25= N/A

26=N/A

27=Send IM Alert from linked Text Field or Text Area

## Systems and Methods for Presenting Information on Mobile Devices

- 28=Set Starting Page
- GIS: Port Number
- 13= InternalPageNumber
- SS: Starting Slide Number
- GIS: Compression Percentage
- 14= mouseOver Cursor Shape
  - 0=Default Cursor
  - 1=Cross Hair Cursor
  - 2=Move Cursor
  - 3=Hand Cursor
  - 4=Text Cursor
  - 5=Wait Cursor
- VR: Crop
- GIS:=Navigation Mode
  - 0: Automatic
  - 1: Modal
- List: ListItemHeight
- 15= mousePressed Cursor Shape
  - 0=Default Cursor
  - 1=Cross Hair Cursor
  - 2=Move Cursor
  - 3=Hand Cursor
  - 4=Text Cursor
  - 5=Wait Cursor
- Paragraph
  - 11-20 Number of Lines/Seconds per scroll event
- GIS & VR: PanSpeed
- SlideShow Width of Slidebar
- ChoiceObject
  - 0=default
  - 1= Lists add visible icons/channels from generic icon strip
  - 2= Lists remove invisible icons/channels from generic icon strip
  - 3= Selection List
  - 4=WidgetSelection List from User's Widget Library
  - 5=Phone Selection List
  - 6=SMS Selection List
  - 7=WidgetSelectionList from Runtime Library
  - 8=Phone Selection List from Pim
  - 9=SMS Selection List from Pim
  - 10=SearchResponseList
- ListObject
  - 0=default
  - 1= Multiple Select
  - 2= Category
  - 3=Lists all icons/channels from generic icon strip
  - 4=WidgetSelection List from User's Widget Library
  - 5=Phone Selection List
  - 6=SMS Selection List
  - 7=WidgetSelectionList from Runtime Library
  - 8=Phone Selection List from Pim
  - 9=SMS Selection List from Pim
  - 10=SearchResponseList
- 16= mouseOver Audio Mode
  - 0=Inactive
  - 1= Play Once while Selected
  - 2= Loop while Selected

## Systems and Methods for Presenting Information on Mobile Devices

3= Play Once until Fire  
4= Loop until Fire  
GIS & VR: PanFrame  
17= mousePressed Audio Mode  
0=Inactive  
1= Play Once while Selected  
2= Loop while Selected  
3= Play Once until Fire  
4= Loop until Fire  
VR: Zoom Speed (time for a 2X zoom)  
GIS: Zoom Speed (time for the full digital zoom)  
SlideShow Type \  
0=Standard  
1=Single exploding icon Icon Strip  
2=Mac-like large and intermediate sized icons Icon Strip  
18= mouseDragged Cursor Shape  
0=Default Cursor  
1=Cross Hair Cursor  
2=Move Cursor  
3=Hand Cursor  
4=Text Cursor  
5=Wait Cursor  
Paragraph  
11=Manual Scroll  
12=Auto Scroll Down  
13=Auto Scroll Up  
XpressMo Choice= MaximumNumberVisibleItems  
XpressMo Text Area= Number of Visible lines  
SlideShow: Percent Explode Slide is greater than unselected slide.  
VR: ZoomFrame (number of frames for a 2X zoom)  
GIS: ZoomFrame FPS  
Text Field: Type  
0: default  
1: Password  
2: Numeric  
3: North America formatted Phone Number  
4: North America formatted SMS Number  
5: Unformatted Phone Number  
6: Unformatted SMS Number  
19= Build Object Style Number  
Runtime Paragraph Line Height  
Choice: Pixel Separator if ObjectIntegerAttribute[15]>0  
GIS: Layer Preload Methodology  
0=No Preloading for pan or zoom  
1=Preload Vector Pan  
2=Preload Vector Pan/Zoom  
3=Preload Vector and Raster Zoom  
4=Preload Vector and Raster Pan  
5=Preload Vector Pan/Zoom and Raster Zoom  
6= Preload Vector Pan/Zoom and Raster Pan  
7= Preload Vector Pan/Zoom and Raster Pan/Zoom  
if (IsDynamicObject{0) {  
20= ObjectAnimation (see Animation Specs)  
21= ObjectMovement (see Animation Specs)  
22= AnimationSoundEffectMode  
VR pan sound track mode

## Systems and Methods for Presenting Information on Mobile Devices

- 23=ObjectAppearDelay
- 24=ObjectAppearType (see Animation Specs)
- 25=ObjectAppearMovement (see Animation Specs)
- 26=ObjectAppearTime
- 27=ObjectAppearFrame
- 28=ObjectAnimationTime
- 29=ObjectAnimationFrame
- 30=     AppearSoundEffectMode  
        VR zoom in sound track mode
- 31=ObjectDepartDelay
- 32=ObjectDepartType (see Animation Specs)
- 33=ObjectDepartMovement (see Animation Specs)
- 34=ObjectDepartTime
- 35=ObjectDepartFrame
- 36=ObjectAnimationCycles
- 37=RelatedIDNumber for BuildObj
- 38=     DepartSoundEffectMode  
        VR zoom out sound track mode
- 39=ObjectAnimationDelay
- 40=ObjectAnimationCustomZoom
- 41=     Build: Part of Selected Group  
        Runtime:         xOffset\_
- 42=     Build: Next Child Group Number  
        Choice: Pixel Separator if ObjectIntegerAttribute[15]>0  
        Runtime:         yOffset\_
- 43=     Build: Alert Selection  
        GIS X  
        Runtime: FrameWidth
- 44=     Build: Share Selection  
        GIS Y  
        Runtime: OriginalSize
- 45=     Build: GIS Width  
        Runtime: timelineEvent\_
- 46=     Build: GIS Height  
        SS: Starting Slide Number (proposed 2/13/06) Set by calling object  
        Runtime: currentTopParagraphLine  
        XpressMoChoiceObject: currentTopItem
- 47=     Build: GIS Resolution\*100  
        Runtime: RSS Paragraph Number of Lines
- 48=     Build: GIS Scenario  
        XpressMoChoiceObject: Expanded Height when Open  
        SlideShow IconStrip: Xpos
- 49=     SlideShow IconStrip: Ypos  
        Build: GIS Server Listener Port Number  
        Runtime: Initial Selected Item for ChoiceObject for Clear  
        All text objects: length of string in char. Used for drawing RSS feeds.
- 50=     SlideShow IconStrip: Width  
        Runtime textField: Marquee offset. Left offset while in animation loop
- 51=     SlideShow IconStrip: Height  
        All text objects:: Caret position when returning from text box

## Systems and Methods for Presenting Information on Mobile Devices

### 6.3 ObjectColorAttribute [] :

0=ObjectColor  
VR/GIS=Icon Color  
1=BackgroundColor  
VR/GIS=Icon Fill Color  
2=FrameColor  
3=VR/GIS: SelectIconFillColor  
Polygon: Gradient Color  
4=VR/GIS: IconFrameColor

### 6.4 ObjectFont [] :

0=Text/Button Font

### 6.5 ObjectString [] :

0= Button: Value  
GIS GIS Directory Name  
XpressMo Choice: Currently selected String  
Image: Saved URL for reset()  
1= Image: URL  
GIS: Raster Layer URL  
SlideShow: String for Empty Chat Area  
Search Response List: Linked SlideShow OBJ for Media Display  
2=  
exitUrl (1,2,4)  
Goto PageName (3,9,15)  
Goto Widget(16)  
XMO Application Name (8,20)  
JavaScript method Name (6)  
Starting PageName (28)  
Search Response List: Linked Button or Text Field for Title Display  
Linked Text Object name with text message for:  
sending SMS Message (18,23)  
sending IM Alert (27)  
3= StyleSelected  
Runtime:  
Selected SlideShow Object Name for Exit Event to a Specific Slide (9)  
Search Response List: Linked Para or Text Area for Message Display  
Linked Text Field name with phone number for:  
placing a phone call (12,24)  
sending SMS Message (18,23)  
sending IM Alert (27)  
Or Starting Page for ExitMode=8 and 20 (Execute an XpressMo Application)  
Or Alert POPUp Page for:  
ExitMode=15 (Goto Specific Internal Page & Display Alert)  
ExitMode=17 (Display Alert w/Fire)  
ExitMode=19 (Toggle Alert. "Display OnFocus, Hide OffFocus")  
ExitMode=22 (Generate Alert with Backend Synchronization)  
4= Entry Audio URL  
VR Zoom In Audio URL  
Alert Avatar URL for ExitMode= 27  
5= Main Audio URL  
VR Pan Audio URL  
6= Exit Audio URL

## Systems and Methods for Presenting Information on Mobile Devices

- VR Zoom Out Audio URL
- Alert POPUp Page for ExitMode=20 and 27
- 7= Object Name  
For Style Object it is the List Layout Style Name
- 8= MouseOver Audio URL
- 9= MousePressed Audio URL
- 10= Runtime: Initial State of Text Field and Text Area for Clear
- 11= build:
  - Search Response List: Linked Para or Text Area for Message Display
  - Selected SlideShow Object Name for Exit Event to a Specific Slide (9)
  - Linked Text Field name with phone number for:
    - placing a phone call (12,24)
    - sending SMS Message (18,23)
    - sending IM Alert (27)
  - Or Starting Page for ExitMode=8 and 20 (Execute an XpressMo Application)
  - Or Alert POPUp Page for:
    - ExitMode=15 (Goto Specific Internal Page & Display Alert)
    - ExitMode=17 (Display Alert w/Fire)
    - ExitMode=19 (Toggle Alert. "Display OnFocus, Hide OffFocus")
- 98= GIS: GIS Server URL
- 99= GIS: Thumbnail Raster URL

### **6.6 Audio[] :**

- 0= Mouse Over AudioClip
- 1= Mouse Click AudioClip
- 2= animation AudioClip



## 7 Animation Specifications

1= Fade In  
2= Fade Out  
3= Zoom In  
4= Zoom Out  
5= Grow NW  
6= Grow NE  
7= Grow SE  
8= Grow SW  
9= Shrink SE  
10= Shrink SW  
11= Shrink NW  
12= Shrink NE  
13= Enter N  
14= Enter NE  
15= Enter E  
16= Enter SE  
17= Enter S  
18= Enter SW  
19= Enter W  
20= Enter NW  
21= Exit N  
22= Exit NE  
23= Exit E  
24= Exit SE  
25= Exit S  
26= Exit SW  
27= Exit W  
28= Exit NW  
30= Custom  
29= Multi-Point  
31= Fade  
32= Hover  
33= Zoom  
34= Zoom SE  
35= Zoom SW  
36= Zoom NW  
37= Zoom NE  
38= Carom N  
39= Carom NE  
40= Carom E  
41= Carom SE  
42= Carom S  
43= Carom SW  
44= Carom W  
45= Carom NW  
50= Seek Cursor  
52= Attach  
53= Deposit  
54= Send Home  
62= Rotate W-E  
61= Swing W-E  
60= Swivel W-E  
63= Rotate N-S

## Systems and Methods for Presenting Information on Mobile Devices

64= Swing N-S  
65= Swivel N-S  
70= Spin Left  
71= Spin Right  
72= Flip Left  
73= Flip Right  
80= Scroll Left (Tickertape)  
81= Scroll Up  
82= Scroll Right (Marquee)

## 8 GIS Object Definitions

### 8.1 ObjectString [] :

Vector Layer File Definitions have an offset of 200. ie:

200= GIS Build: Vector Layer 0 Name  
201= GIS Build: Vector Layer 1 Name  
202= GIS Build: Vector Layer 2 Name  
203= GIS Build: Vector Layer 3 Name  
204= GIS Build: Vector Layer 4 Name  
205= GIS Build: Vector Layer 5 Name

### 8.2 ObjectIntegerAttribute [] :

Vector Layer Minimum Scale have an offset of 100  
Vector Layer Foreground Color have an offset of 200  
Vector Layer Background Color have an offset of 300  
Vector Layer Line Width have an offset of 400

## 9 Paragraph and XpressMo Choice Object

Paragraph:                    TextArea Paragraph;  
Paragraph/Choice:           String paragraphstr[i]  
Choice:                      int LineStart[i]  
Paragraph/Choice:           int LineEnd[i]  
Paragraph:                   int LineBreak[i]  
Paragraph:                   int LineWidth[i]  
where i=line number.

## 10 Polygon Definitions

```
int PolySides
PolySides=in.readShort();
PolyX            = new int[3][PolySides];
PolyY            = new int[3][PolySides];
for (int j=0;j<PolySides;j++) {
    if (j==0) {
        PolyX[0][j]= in.readShort();
        PolyY[0][j]= in.readShort();
    }
    else {
        PolyX[0][j]= in.readByte();
    }
}
```

## Systems and Methods for Presenting Information on Mobile Devices

```
    if (PolyX[0][j]==-128) PolyX[0][j]= in.readShort();
    PolyX[0][j]+=PolyX[0][j]-1];
    PolyY[0][j]= in.readByte();
    if (PolyY[0][j]==-128) PolyY[0][j]= in.readShort();
    PolyY[0][j]+=PolyY[0][j]-1];
  }
}
```

Gradient Definition:  
Int GradientMidPoint, Noise;  
Radial only: int PointX,PointY  
Linear only: int Angle

## 11 MultiShape Definitions

```
byte MetaDataEntries=0; // to be defined
int NumberOfShapes_, Xorigin,Yorigin, ShapeWidth, ShapeHeight;
int Xoffset; // the offset, in pixels, of the shape to the left of the logical x origin.
int Yoffset; // the offset, in pixels, of the shape past the top of the logical y origin.
XOrigin=(int)in.readShort();
YOrigin=(int)in.readShort();
ShapeWidth=(int)in.readShort();
ShapeHeight=(int)in.readShort();
XOffset=(int)in.readShort();
YOffset=(int)in.readShort();
NumberOfShapes_=(int)in.readShort();
FillColor= new Color [NumberOfShapes_];
LineColor= new Color [NumberOfShapes_];
x_ = new int [NumberOfShapes_];
y_ = new int [NumberOfShapes_];
width_ = new int [NumberOfShapes_];
height_ = new int [NumberOfShapes_];
Initx_ = new int [NumberOfShapes_];
Inity_ = new int [NumberOfShapes_];
Initwidth_ = new int [NumberOfShapes_];
Initheight_ = new int [NumberOfShapes_];
LineWidth= new int [NumberOfShapes_];
for (int i=0;i<NumberOfShapes_;i++) {
    LineWidth[i]=1;
    FillColor[i]= new Color(255,255,255);
    LineColor[i]= new Color(0,0,0);
}
ShapeType= new int [NumberOfShapes_];
NumberOfElements= new int [NumberOfShapes_];
DrawFill= new boolean [NumberOfShapes_];
ElementX= new int [NumberOfShapes_][];
ElementY= new int [NumberOfShapes_][];
InitElementX= new int [NumberOfShapes_][];
InitElementY= new int [NumberOfShapes_][];
Gradient Definition:
    GradientType= new int [NumberOfShapes_];
    Noise= new int [NumberOfShapes_];
    GradientMidPoint= new int [NumberOfShapes_];
```

## Systems and Methods for Presenting Information on Mobile Devices

```
Gradient Color= new Color [NumberOfShapes_];
GradientAngle= new int [NumberOfShapes_];      (linear)
GradientX= new int [NumberOfShapes_];          (radial)
GradientY= new int [NumberOfShapes_];          (radial)

BooleanAttribute= new boolean [NumberOfShapes_][[]];
IntegerAttribute= new int [NumberOfShapes_][[]];
StringAttribute= new String [NumberOfShapes_][[]];
for (int i=0;i<NumberOfShapes_ ;i++) {
    ShapeType[i]=(int)in.readShort();
    x_[i]=(int)in.readShort();
    y_[i]=(int)in.readShort();
    width_[i]=(int)in.readShort();
    height_[i]=(int)in.readShort();
    MetaDataEntries=(int)in.readByte();
    BooleanAttribute[i]= new boolean [MetaDataEntries];
    IntegerAttribute[i]= new int [ MetaDataEntries];
    StringAttribute[i]= new String [ MetaDataEntries];
    for (int b=0;b<MetaDataEntries ;b++) {
        BooleanAttribute [i][b]=(int)in.readBoolean();
        if (BooleanAttribute[i][b]) {
            /*
            This is where IntegerAttribute and StringAttribute data is read, based on the definition of each meta data
            element.
            */
        }
        if (ShapeType[i]==2) {
            NumberOfElements[i]=(int)in.readShort();
            ElementX[i]= new int [NumberOfElements[i]];
            ElementY[i]= new int [NumberOfElements[i]];
            InitElementX[i]= new int [NumberOfElements[i]];
            InitElementY[i]= new int [NumberOfElements[i]];
            for (int j=0;j<NumberOfElements[i];j++) {
                if (j==0) {
                    ElementX[i][j]= (int)in.readShort();
                    ElementY[i][j]= (int)in.readShort();
                }
                else {
                    ElementX[i][j]= (int)in.readByte();
                    if (ElementX[i][j]==-128) ElementX[i][j]= (int)in.readShort();
                    ElementX[i][j]+=ElementX[i][j-1];
                    ElementY[i][j]= (int)in.readByte();
                    if (ElementY[i][j]==-128) ElementY[i][j]= (int)in.readShort();
                    ElementY[i][j]+=ElementY[i][j-1];
                }
            }
        }
        if (ShapeType[i]==5) {
            NumberOfElements[i]=1;
            ElementX[i]= new int [NumberOfElements[i]];
            ElementY[i]= new int [NumberOfElements[i]];
            InitElementX[i]= new int [NumberOfElements[i]];
            InitElementY[i]= new int [NumberOfElements[i]];
            ElementX[i][0]=(int)in.readShort();
            ElementY[i][0]=(int)in.readShort();
        }
    }
}
```

## Systems and Methods for Presenting Information on Mobile Devices

```
LineWidth[i]=(int)in.readShort();  
DrawFill[i]=in.readBoolean();  
if (DrawFill[i]) FillColor[i]=new Color(in.readInt());  
LineColor[i]=new Color(in.readInt());
```

## 12 Event Definitions

### 12.1 Definition of TimelineEvents

0= Initial Entry Delay  
1= Entry Animation  
2= Main Animation Delay  
3= Main Animation  
4= Exit Animation Delay  
5= Exit Animation

### 12.2 Definition of Mouse Events

6= Mouse Press  
7= Mouse Over  
8= Mouse Click  
9= Mouse Up  
10= Mouse Drag  
11= Mouse Off  
12= Mouse Double Click

## 13 Virtual World Definitions

### 13.1 Virtual World:

Int cropW  
Int cropE  
Int cropN  
Int cropS  
Int VrcropPercent  
Int panSpeed\_ // Number of frames per second during the VR pan animation  
Int panFrame // The number of pixels that each frame will be offset.  
Int zoomSpeed // Time for a 2X VR Zoom animation  
Int zoomFrame // The number of frames for an 2X Zoom closer loop  
Color Fill Color of Unselected Icon Frame  
Color Color of Icon Frame  
Color Color of Icon  
Color Fill Color of Selected Icon Frame  
Hashtable virtualRooms\_ (RoomName will be the KEY)  
PanLocation[3]; // used for vr image positioning  
ZoomState; // used for controlling vr zoom  
PanState; // used for controlling vr pan  
VRActive; // used for continuing vr pan on drag over vr window  
Stack backupRoom\_

### 13.2 Virtual Room:

Boolean DynamicLink  
RoomType  
0=Virtual Room  
1=Virtual Picture  
2=Virtual Wall  
3=Virtual Window  
4=Video

## Systems and Methods for Presenting Information on Mobile Devices

Int RoomX  
Int RoomY  
Int RoomWidth  
Int RoomHeight  
URL VRRoomImageHref  
Image RoomImage  
URL VRRoomExitURL  
URL VRPictureSoundTrackURL  
Hashtable virtualDoors\_ (DoorName will be the KEY)  
Hashtable drawDoor\_ ( Boolean, DoorName will be the KEY)

### **13.3 Virtual Door:**

String DoorName  
Boolean DoorVisible  
Int DoorInitX  
Int DoorInitY  
Int DoorInitWidth  
Int DoorInitHeight  
Int DoorX  
Int DoorY  
Int DoorWidth  
Int DoorHeight  
Int DoorExit

## 14 Slide Show Definitions

### 14.1 StyleIntegerAttribute [] :

0=TransitionAnimationType  
1=TransitionTime  
2=TransitionFrame  
3=SlideDelay  
4=Scroll Bar Definition  
0=Hidden  
1=Above Slide Show  
2=Bracketed Above  
3=Superimposed Above  
4=Below Slide Show  
5=Superimposed Below  
6=Bracketed Below  
Colors:

~~0=Hidden~~  
ElevatorBackground  
ElevatorFrame1  
ElevatorFrame2

SlideWidth  
SlideHeight  
MouseEvent;  
AudioMode;  
Define Slides

### 14.2 SlideBooleanAttribute [] ;

0= True=video  
False=image  
1= Slide Audio Exists  
2= Slide Transition Audio Exists  
3=Goto External Page  
4=Mouse Enabled  
5=BubbleHelp Enabled  
6=DynamicLink Set (build only)  
Runtime: Icon is active in icon strip  
7=Icon came from UberApplication

### 14.3 SlideIntegerAttribute [] :

0=TransitionAnimationType  
0=None  
1=Expand Over  
2=Shrink Out  
3=Zoom to Upper Left  
4=Zoom to Lower Right  
5=Rotate to the Left  
6=Rotate to the Right  
7=Rotate from Bottom to Top  
8=Rotate from top to Bottom  
9=Slide to the Left  
10=Slide to the Right  
11=Slide from Bottom to Top



## Systems and Methods for Presenting Information on Mobile Devices

12=Slide from top to Bottom  
13=Pan to the Left  
14=Pan to the Right  
15=Pan from Bottom to Top  
16=Pan from top to Bottom  
17=Fade  
18=Zoom In  
19=Zoom Out  
20=Random  
1=TransitionTime  
2=TransitionFrame  
3=SlideDelay  
4=Slide audio mode  
5=Transition audio mode  
6=Exit Slide Mode (red is current implementation as of 6/19/07)  
0=Next Slide  
1= Go to a Specific Slide  
2=Goto External Web Page Launched in a New Window  
3=Goto External Web Page  
4=Goto External Web Page replacing the Top Frame  
5= Execute an XpressMo Application  
6= Goto a specific Internal Web Page  
7= Goto a specific Internal Web Page with setting starting slide  
8= Goto a specific Internal Web Page with Alert  
9= Goto a widget  
10= Goto Logical First Page  
11=N/A  
12=N/A  
13=N/A  
14=N/A  
15=Goto a specific Internal Web Page with Alert. "w/ Backend Synchronization"  
16=Goto Widget Object  
17=Generate Alert. "With a Fire Event"  
18=N/A  
19=Toggle Alert. "Display OnFocus, Hide OffFocus"  
20= Execute an XMO Application with Alert. "With a Fire Event"  
21=Goto Logical First Page  
22= Generate Alert with Backend Synchronization  
25=N/A  
26=N/A  
27=Send IM Alert to Chat Members  
28=N/A  
7=Goto Slide number  
8=primary Key  
9= video:width  
10=video:height

### **14.4 SlideString[];**

0=Image/Video String (used for exit event==27 for Avatar URL)  
1=Slide Audio String  
2=Slide Transition Audio String  
3=  
ExitEvent=1,2,4: External URL  
ExitEvent=16: Widget URL  
ExitEvent=8,20: ApplicationName

## Systems and Methods for Presenting Information on Mobile Devices

ExitEvent=3,9,15: GoToPageName  
ExitEvent=27: Symbolic Name for Chat Group Members  
4=  
Selected SlideShow Object Name for Exit Event to a Specific Slide  
Starting Page for ExitMode=8 and 20 (Execute an XpressMo Application)  
ExitEvent=27: Linked TextObj Name for IM Alert Message  
Alert POPUp Page for:  
ExitMode=15 (Goto Specific Internal Page & Display Alert)  
ExitMode=17 (Display Alert w/Fire)  
ExitMode=19 (Toggle Alert)  
ExitMode=20 (Execute an XMO Application)  
ExitMode=22 (Generate Alert with Backend Synchronization)  
5=BubbleHelp  
ExitEvent=27: Alert POPUp Page name  
6=Database Name (build only)  
Runtime:extracted image/video name for managing icons  
7=Table Name (build only)  
8=Field Name (build only)

### **14.5 SlideAudioObject[];**

0=Slide Audio  
1=Transition Audio

### **14.6 SlideImage**

### **14.7 SlideColor**

### **14.8 Phlog Attributes:**

FlogFont  
PhlogWidth:  
0=Slideshow Width  
1=Slideshow Width + Slide Width  
2=Slideshow Width + Slide Width x 2  
3=Maximum Width  
NumberOfVisibleLines (Horizontal SS only);  
Vertical SS is calculated by the slide height.  
PhlogStyle;  
0=Hidden (Not a Phlog)  
1=Below or Right of Slide Show  
2=Above or Left of Slide Show  
DateStamp;  
0=None  
1=DD:MM:YYYY  
2=MM:DD:YYYY  
TimeStamp;  
0=None  
1=HH:MM AM/PM  
2=HH:MM  
Color TextColor;  
Color BackColor;  
Color TextEditColor;

## Systems and Methods for Presenting Information on Mobile Devices

Color BackEditColor;

ScrollWindow (runtime) Maximum String width before scroll starts.

### **15 Game Definitions**

Vector            AudioEffectName  
Vector            AudioEffectURL  
Hashtable        childObjects\_;

### **16 Form Object Definitions**

- 32: Submit Button**  
Bounds(x,y,width,height)  
Foreground Color  
Background Color  
Font(Name,Style,Size)  
Label  
Form Number
- 33: Clear Button**  
Bounds(x,y,width,height)  
Foreground Color  
Background Color  
Font(Name,Style,Size)  
Label
- 34: Check Box**  
Bounds(x,y,width,height)  
State
- 35: Radio Button (N/A)**  
Bounds(x,y,width,height)  
State
- 36: Text Field**  
Bounds(x,y,width,height)  
Foreground Color  
Background Color  
Font(Name,Style,Size)  
Text String  
IsHidden  
isEditable

## Systems and Methods for Presenting Information on Mobile Devices

### **37: Text Area**

Bounds(x,y,width,height)  
Foreground Color  
Background Color  
Font(Name,Style,Size)  
Text String  
IsHidden  
isEditable

### **38: Choice**

Bounds(x,y,width,height)  
Foreground Color  
Background Color  
Font(Name,Style,Size)  
NumberOfItems  
ItemLabel[NumberOfItems]  
ItemExitEvent[NumberOfItems]  
0=None  
1=Goto External Web Page replacing Current Frame  
2=Goto External Web Page Launched in a New Window  
3=Goto a specific Internal Web Page  
5=Goto the next Internal Web Page  
4=Goto External Web Page replacing the Top Frame  
6=Execute JavaScript Method  
7=Pause/Resume Page Timeout  
8=Execute an XpressMo Application  
9= Goto a specific Internal Web Page with setting starting slide  
10=Set by IconStrip. Display Icon  
11=N/A  
12=Place Phone Call  
13=Add Icon  
14=Remove Icon  
15=Goto a specific Internal Web Page with Alert. "with Backend Synchronization"  
16=Goto Widget Object  
17=Generate Alert. "With a Fire Event"  
18=Send SMS Message  
19=Toggle Alert. "Display OnFocus, Hide OffFocus"  
20= Goto an XMO Application with Alert. "With a Fire Event"  
21=Goto Logical First Page  
22= Generate Alert with Backend Synchronization  
23=Send SMS Message with Share (Player Download)  
24= Place PhoneCall with Share (Player Download)  
25=Send SMS Message with Share All (Player Download)  
26= Place PhoneCall with Share All (Player Download)  
27=Send IM Alert  
28=Set Starting Page  
30= Public Widget URL // and list is set and populated by back end automatically when a List of  
// ObjectIntegetAttribute[15]=7 (WidgetSelectionList from Runtime Library type )  
ListItemParameter [NumberOfItems]  
ExitEvent=0: Parameter  
ExitEvent=18, 23,25: Linked TextObj Name for SMS Message  
ExitEvent=27: Linked TextObj Name for IM Alert Message

## Systems and Methods for Presenting Information on Mobile Devices

ExitEvent=20:	AlertPageName
ListItemPageName [NumberOfItems]	
ExitEvent=1,2,4:	External URL
ExitEvent=12,18, 23-27:	PhoneNumber
ExitEvent=13,14:	Widget IconName
ExitEvent=16:	Author Widget URL
ExitEvent=30:	Public Widget URL
ExitEvent=8,20:	ApplicationName
ExitEvent=28:	Starting Page Name
ExitEvent=3,9,15,17:	GoToPageName
ItemSlideShowName[NumberOfItems]	
ExitEvent=8,20:	StartingPageName
ExitEvent=9:	SlideShowObjectName
ExitEvent=15,17,19,22	AlertPageName
ItemGotoSlideNumber[NumberOfItems]	

- 39: List**  
Bounds(x,y,width,height)  
Foreground Color  
Background Color  
Font(Name,Style,Size)  
IsHidden  
MultipleSelect  
NumberOfItems

### **16.1 Complex List Generic Attributes**

ItemString[0]=ItemLabel	
ListItemParameter [NumberOfItems]	
ExitEvent=0:	Parameter
ExitEvent=18, 23,25:	Linked TextObj Name for SMS Message
ExitEvent=27:	Linked TextObj Name for IM Alert Message
ExitEvent=20:	AlertPageName
SearchResponseList	Message Object Visible String(ObjectString[0])
ListItemPageName [NumberOfItems]	
ExitEvent=1,2,4:	External URL
ExitEvent=12,18, 23-27:	PhoneNumber
ExitEvent=13,14:	Widget IconName
ExitEvent=16:	Widget URL
ExitEvent=8,20:	ApplicationName
ExitEvent=28:	Starting Page Name
ExitEvent=3,9,15,17,21:	GoToPageName
SearchResponseList	Title Object Visible String (ObjectString[0])
ItemSlideShowName[NumberOfItems]	
ExitEvent=8,20:	StartingPageName
ExitEvent=9:	SlideShowObjectName
ExitEvent=15,17,19,22	AlertPageName
SearchResponseList	SlideShow Slide URL(SlideString[0])
ListItemSlideNo	
SearchResponseList	SlideShow Slide Index Number
ExitEvent=9:	SlideShow Slide Index
Color ItemColor	
Font ItemFont	
IconType[0]:	For defining the selected icon where:

## Systems and Methods for Presenting Information on Mobile Devices

0=none.  
1=custom image icon  
2=Checkmark  
3=X  
4=Checkmarked Box  
5=X in Box  
6=Filled Circle

IconType[1]: For defining the unselected icon where:

0=none.  
1=custom image icon  
2=empty box  
3=empty circle

If IconType[0]=1:

isIconDefined[0]=true  
IconURL[0]=the Selected Icon File Name

If IconType[1]=1:

isIconDefined[1]=true  
IconURL[1]=the Unselected Icon File Name

boolean hasSecondLine: Enables and draws second line attributes

### **16.2 Complex List Layout Attributes**

short secondLineIndent: If 2<sup>nd</sup> line # of pixel indent  
short DefaultItemNumber: List item # for setting default values  
boolean[4] isItemStringDefined:  
boolean[4] isItemImageDefined:

Each Complex List item may have up to 2 lines, and each line may have up to 2 elements. If set to true the actual items will have these attributes available for definition. 1<sup>st</sup> Line Body is always available. The elements are:

1<sup>st</sup> Line Title  
    Optional Image Icon  
    Mandatory String  
1<sup>st</sup> Line Body  
    Optional Image Icon  
    Optional String  
2<sup>nd</sup> Line Title  
    Optional Image Icon  
    Optional String  
2<sup>nd</sup> Line Body  
    Optional Image Icon  
    Optional String

boolean[2] RightJustify:

Each line may have the body element right justified if its related RightJustify[line number]=true.

### **16.3 Complex List Item Attributes**

For each item in a complex list, there are definitions for all of the possible elements that were enabled by the respective Layout Booleans.

**For Strings:**

## Systems and Methods for Presenting Information on Mobile Devices

boolean[4] isItemStringDefined:

If set to true the actual item will have these attributes defined and displayed. 1<sup>st</sup> Line Body is always defined.

The String elements are:

String[4] ItemString: The string for the element:

boolean[4] UseItemColor: True means use ItemColor

boolean[4] UseItemFont: True means use ItemFont

where:

0=1<sup>st</sup> Line Title

1=1<sup>st</sup> Line Body

2=2<sup>nd</sup> Line Title

3=2<sup>nd</sup> Line Body

### **For Image Icons:**

boolean[4] isItemImageDefined:

If set to true the actual item will have these attributes defined and displayed.

The Icon Element is defined by:

String[4] ImageURL: Image Icon File Name

short[4] ItemImageWidth: Image Icon width

where:

0=1<sup>st</sup> Line Title

1=1<sup>st</sup> Line Body

2=2<sup>nd</sup> Line Title

3=2<sup>nd</sup> Line Body

## **16.4 Runtime Attributes for dynamic content**

String IconPathName: transmitted to the List Object from the populateComplexChoiceObject API.

Boolean Append: transmitted to DynamicDataConnection from the populateComplexChoiceObject API to either append or replace the existing items in the List object.

For each List Item:

boolean[4] DownloadIcon

If true then the IconPathName: will be used for an http request. If false, then the ImageURL[index] must be prepended by either "Icons/" "Images/" or "Slides/", and the image that is to be used must already have been defined in the PDL.

## 17 Dynamic Linkage Definitions

A dynamic link record consists of the following fields

- 0: FileFormatNumber
- 1: Link Type
  - 0= Web Site
  - 1= Web Page
  - 2= Parent Object
  - 3= Child Object
  - 4= Not Used
  - 5= Slide
  - 6= Virtual Room
  - 7= List
- 2: page Number
- 3: page name
- 4: Attribute Number
- 5: Parent object number
- 6: Parent object name
- 7: Child object number, slide number or List item number
- 8: Object Type (equivalent to ObjectIntegerAttribute[0])
- 9: DatabaseName
- 10: UserName
- 11: Password
- 12: DatabaseFormat
- 13: TableName
- 14: Field Name
- 15: Field Format
- 16: Primary Key
- 17: Primary Key Field Name
- 18: Primary Key Field Format

### Pages :

For LinkType =1

Attribute Definitions

- 0= Page Width
- 1= Page Height
- 2= Background Image
- 3= Background Color
- 4= Page Delay Time
- 5= Transition Animation
- 6= Transition Time
- 7= Frames per Second
- 8= Page Audio Track
- 9= Transition Audio Track



## Systems and Methods for Presenting Information on Mobile Devices

10= Page Exit

### **17.1 BooleanAttribute [] :**

9=If True a Dynamic Link has been set for this object

True= Object Attribute is set to a dynamic link with an offset of 10

i.e. ObjectBooleanAttribute[10]=attribute 0

### **17.2 StringAttribute [] :**

If a dynamic link then the Database name has an offset of 10

If a dynamic link then the Table name has an offset of 30

If a dynamic link then the Field name has an offset of 50

### **17.3 IntegerAttribute [] :**

If a dynamic link then the Record Number has an offset of 30

## **18 Page Objects:**

### **18.1 ObjectBooleanAttribute [] :**

48=Permission is granted for serverside timeline activation

49=If True a Dynamic Link has been set for this object

True= Object Attribute is set to a dynamic link with an offset of 50

i.e. ObjectBooleanAttribute[50]=attribute 0

For Link Types =2 & 3

Attribute Definitions

0=

Object Types 0/1	Text Value
Object Types 2/5	Shape Object .AVG URL
Object Types 6/9	Video/Image URL
Object Type 7	Slide Show Object .AVD File URL
Object Type 8	Virtual Object .AVR File URL

For Object Types 0,1,2,5,6,7 ,9 , 31-39

1= IsVisible  
2= Click Hides Unrelated Children  
3= Horizontal  
4= Vertical  
5= Width  
6= Height  
7= Frame Style  
8= Frame Width  
9= Exit Event  
10= Is Draggable  
11= Frame Color  
12= mouseOver Audio Track  
13= mouseClicked Audio Track

## Systems and Methods for Presenting Information on Mobile Devices

- 14= Animation Delay
- 15= Animation
- 16= Movement
- 17= Animation Time
- 18= Animation FPS
- 19= Animation Audio Track
- 20= Animation Cycles
- 21= Entrance Delay
- 22= Entrance Animation
- 23= Entrance Movement
- 24= Entrance Time
- 25= Entrance FPS
- 26= Entrance Audio Track
- 27= Departure Delay
- 28= Departure Animation
- 29= Departure Movement
- 30= Departure Time
- 31= Departure FPS
- 32= Departure Audio Track
- 33= Activate Timeline
- 34= MouseEvent Activate Timeline
- 35= EntryAnimationSuppressed
- 36= DrawobjectBackground
- 37=
  - Object Types 0/1 Typeface
  - Object Types 2/5 Line Width
  - Object Type 7 Goto a specific Slide Number
- 38= Object Types 0/1/2/5 Foreground Color
- 39= Object Types 0/1/2/5 Background Color
- 40= Object Types 0,1,2,3,5,6,and 9 : Exit URL

For LinkType =5  
SlidePathname

- For LinkType =7
- 0= List Item Title
  - 1= List Item Body
  - 2= 2nd Line Title
  - 3= 2nd Line Body
  - 4= Title Icon Filename
  - 5= Body Icon Filename
  - 6= Line2 Title Icon Filename
  - 7= Line2 Body Icon Filename

For LinkType =6  
1= IsVisible

## Systems and Methods for Presenting Information on Mobile Devices

- 3= Horizontal
- 4= Vertical
- 5= Width
- 6= Height
- 7= Frame Style
- 8= Frame Width
- 9= Exit Event
- 10= Is Draggable
- 11= Frame Color
- 12= Icon Color
- 13= Icon Fill Color
- 14= Icon Frame Color
- 15= Icon Select Color
- 16= Crop Percent
- 17= Pan Speed
- 18= Pan Frames/Second
- 19= Zoom Speed
- 20= Zoom Frames/Second
- 21= Pan Audio Track
- 22= Zoom Out Audio Track
- 23= Zoom In Audio Track
- 24= Typeface
- 25= Goto a specific VR Room

### **18.2 ObjectString[] :**

If a dynamic link then the Database name has an offset of 50

If a dynamic link then the Table name has an offset of 100

If a dynamic link then the Field name has an offset of 150

### **18.3 ObjectIntegerAttribute[] :**

If a dynamic link then the Record Number has an offset of 50

## 19 GIS Geography View (.gis)

```
gisOut.writeUTF(WebSiteName);      Application Name
gisObj.getBooleanAttribute(1));     Has Raster Layer
gisObj.getIntAttribute(11));        digital zoom
gisObj.getIntAttribute(13));        compression percentage
gisObj.getIntAttribute(19));        bandwidth load technique
gisObj.getIntAttribute(20));        allowable pan range (except for bottom layer)
gisObj.getIntAttribute(6));         GIS Window Width
gisObj.getIntAttribute(7));         GIS Window Height
gisObj.getIntAttribute(97));        GIS Object Width
gisObj.getIntAttribute(98));        GIS Object Height
gisOut.writeInt(gisObj.getIntAttribute(43)); GISVirtualX
gisOut.writeInt(gisObj.getIntAttribute(44)); GISVirtualY
gisOut.writeInt(gisObj.getIntAttribute(45)); GISVirtualWidth
gisOut.writeInt(gisObj.getIntAttribute(46)); GISVirtualHeight
gisOut.writeInt(gisObj.getIntAttribute(47)); Resolution
gisOut.writeInt(gisObj.getIntAttribute(48)); Scenario
gisOut.writeInt(gisObj.getIntAttribute(49)); ServerListenerPort_
if (gisObj.getBooleanAttribute(1)) {
    gisOut.writeUTF(GISRasterURL0);  root for JPEGs generated by MrSid
    gisOut.writeUTF(GISRasterURL1);  URL pointing to the MrSid File
}
gisOut.writeInt(gisObj.getIntAttribute(10)); Number of Vector Layers
for (int j=0;j<gisObj.getIntAttribute(10);j++) {
    gisOut.writeInt(gisObj.getIntAttribute(j+100));    MinimumVectorVisibleScale
    gisOut.writeUTF(gisObj.getStringAttribute(j+200)); Vector FileName(ShapeFile)
}
```

## **20 Client to Server File Management**

### **20.1 File Source**

- 0=upload (selected off client)
- 1=server private Audio library
- 2=server private Photo library
- 3=server private Video library
- 4=local private Audio library
- 5=local private Photo library
- 6=local private Video library
- 7=server XpressMo Audio library
- 8=server XpressMo Photo library
- 9=server XpressMo Video library
- 10=Server Private Slide Lib
- 11=Local Private Slide Lib
- 12=Server XpressMo Slide Lib
- 13=server private VR lib
- 14=local Private VR lib
- 15=server XpressMo VR lib
- 16=server Private Page lib
- 17=local Private Page lib
- 18=server XpressMo Page lib
- 19=server Private Website lib
- 20=local Private Website lib
- 21=server XpressMo Website lib
- 22=server Private Website Template lib
- 23=local Private Website Template lib
- 24=server XpressMo Website Template lib
- 25=server Private Style Template lib
- 26=local Private Style Template lib
- 27=server XpressMo Style Template lib
- 28=server Private Game lib
- 29=local Private Game lib
- 30=server XpressMo Game lib
- 31=server Private Icon lib
- 32=local Private Icon lib
- 33=server XpressMo Icon lib
- 34=server Private PageIcon lib
- 35=local Private PageIcon lib
- 36=server XpressMo PageIcon lib
- 37=server Private Widget lib
- 38=local Private Widget lib
- 39=server XpressMo Widget lib
- 40=server Private Alert lib
- 41=local Private Alert lib

## Systems and Methods for Presenting Information on Mobile Devices

42=server XpressMo Alert lib  
43=server Private WidgetReferenceIcons lib  
44=server Private List lib  
45=local Private List lib  
46=server List lib

### **20.2 LibraryName**

Required for:

Slide Lib  
VR Lib  
Page Lib  
Widget Lib  
Website Lib  
Website Template lib  
Style Template lib  
Game lib

## 21 Client Database Schema

```
String DataBaseName
Int FileFormat
String UserName2
String Password
int Number of Tables (TableNum)
for (int j = 0; j < TableNum; j++) {
    String TableName
    int Number of Fields (FieldNum)
    for (int k = 0; k < FieldNum; k++) {
        String FieldName
        int FieldFormat: where:
            0=Long Integer
            1=Integer
            2=Short
            3=Byte
            4=Double
            5=Float
            6=String
            7=Boolean
            8=Timestamp
            9=Decimal
            10=Date
            11=Time
            12=National Char
            13=Tiny Blob
            14=Blob
            15=Medium Blob
            16=Long Blob
        boolean isPrimaryKey
```

# Appendix H

## RSS/ATOM/RDF Feed Collector

### Specifications

#### 1 Introduction

FeedCollector, formerly known as RSSFeeder, is a server side component of Publisher system which collects RSS, ATOM and RDF format feeds from various sources and aggregates them into a database for use by the applications built using Publisher. FeedCollector is primarily driven by two sets of parameters: one is the Database Schema (written as SQL DDL) which defines the tables in the database, as well as parameters for each of the feeds to be examined. The other is the feed parse rules, written in xml, which can be used to customize the information that is extracted from the feeds. Each of the feeds is collected at intervals specified by the feed parameter set in the SQL DDL.

##### **1.1 Advantages of FeedCollector over RSSFeeder**

1. Foremost is the ability to custom pick the fields to place in the feed database by providing text based parsing rule for each channel. In fact, even the standard fields for RSS, ATOM and RDF feeds are extracted using the same methodology. A custom rule will override any definitions in the standard.
2. More database fields can be added or modified without changes to the program. All database definitions and parsing rules are in externalized text files.
3. Has the ability to scan not only RSS but RDF and ATOM format feeds. Other non-conforming formats could be parsed using custom rules, as long as the feed is xml 1.0 compliant.
4. The scan interval for each channel can be set individually, and at a granularity of 1 second. If the scan interval is not important, the default scan interval, set in the properties file, can be used by setting the channel's scanInterval field to 0.
5. Channel specific data from the feed is updated and placed in the channel table. RSSFeeder lacked this functionality.
6. A channel status field is available, which reports whether the channel is OK or has an error. This field can be used in a simple monitoring program to check on the health of each channel.
7. Because of the expanded capabilities, more useful data can be extracted from the feeds. Information such as item specific image URL's, channel image URL's, and even URL's to enclosures such as audio or video files can be retrieved from the feed and stored in the database. RSSFeeder supports a very limited set of fields, and only for item fields.
8. Allows up to 999 feed items per channel (can be modified in properties file); the number of items in a channel can change without upsetting the itemId's of other channels, by adopting a new scheme for assigning itemId's. (See prepopulated channel field chart below).
9. The new channel schema defines channel dependencies. A channel can declare its parent channel in the field parentId, whose value is the Channel ID of the

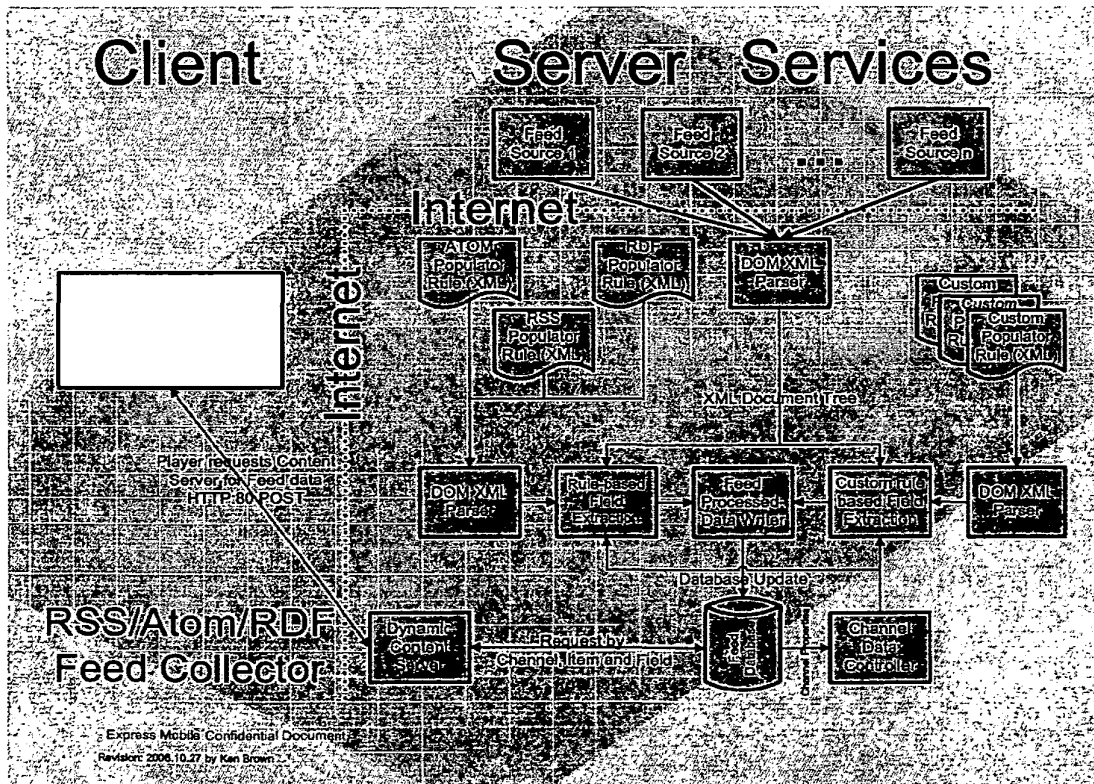


## Systems and Methods for Presenting Information on Mobile Devices

parent channel, or 0 if this channel has no dependencies. This data allows a program to recreate a dependency tree of the channels.

10. The new channel schema stores the number of items read in "numItems". This may be equal to or less than "maxItems", if maxItems is greater than 0, or is the number of items found in the channel, if maxItems is 0. This is a change in semantics for this field "numItems" from the RSSFeeder schema. In the previous schema "numItems" took the place of "maxItems" in the new schema. This change allows the item count to reflect whatever is available in the feed, rather than the number of items predetermined at database initialization time.
11. The channel schema contains an additional optional field, "ruleName", which, if populated, defines the custom parse rule to use. This allows the customization of field extraction. This field can be unspecified, in which case it takes the value in channelName. FeedCollector looks in the rules folder to see if a custom rule by the same name as ruleName exists, and if so, uses this to parse the fields.

## 2 Components Diagram



## Systems and Methods for Presenting Information on Mobile Devices

### 3 Input Data

#### 3.1 MySQL DDL (Schema)

The database engine used by FeedCollector is MySQL 5.0. The database name is "FeedCollector". The schema for database FeedCollector defines two tables: channel and item. Channel holds the feed channel information. It loosely corresponds to the <channel> element in RSS. Item holds the individual feed items records, and loosely corresponds to the <item> elements in RSS.

##### Definition of channel:

```
# Field descriptions for table "channel"
# Static fields:
# channelId          primary key
# parentid           Channel ID of the parent channel, or 0 if no parent channel exists.
#                   Used for building channel dependency tree. If parent channel's
#                   lastBuildDate changes then all child channels are rescanned.
# url                URL for accessing this feed channel
# channelName        unique descriptive name for the channel; this is also the default
#                   value for field "ruleName", the name of xml file
#                   for custom extraction of fields into the database.
#                   Alphanumeric only.
# ruleName           Name of a parsing rule file to apply to this channel. The file
#                   name is the value in this field with ".xml" appended.
#                   If this field is NULL the channelName value is used.
# scanInterval       Time between scans of this feed, in seconds, '0' = use default
#                   interval.
#                   A default interval is configurable via TIME_INTERVAL property.
#                   If the channel supports <ttl>,<skiphours>, or <skippedays>, these
#                   are used to compute the next scan time instead of using refreshRate.
# maxItems           Maximum number of feeds to collect from this channel, must be at
#                   least 1.
#                   If there are more items than maxItems, then the first maxItems
#                   items are
#                   read into the database. If this value is set to 0, then all items
#                   in the feed are read in, subject to FeedCollector's maximum which
#                   is in property MAX_ITEMS_PER_CHANNEL.
# Dynamic fields:
# Mandatory:
# numItems           Set by FeedCollector: Number of items that were read in from the
#                   feed
# status             Set by FeedCollector: OK for no errors, or error message of last
#                   scan
# -----
# title              Set from feed. Title provided by the channel
# link               Set from feed. The link to website corresponding to this feed.
# description        Set from feed. Description field provided by the feed channel
#
# Optional (all set from feed):
#
# language           is the language the channel is written in. E.g. "en-US"
# copyright          copyright notice
# managingEditor     E-mail address for person responsible for editorial content
# webMaster          E-mail address for person responsible for technical issues.
# pubDate            Publication date, in RFC 822 format, e.g.,
#                   "Sat, 07 Sep 2002 00:00:01 GMT"
# lastBuildDate      Last time content changed
# category           One or more categories this channel belongs to
#                   Special category value _FeedList_ indicates master list of feeds:
#                   Each item link element is url to another feed.
# generator          Program used in generating this channel
# docs               Documentation for this RSS feed format
# cloud              callback mechanism
# ttl                Time to live, in minutes
# imageUrl           It is the URL of the image file representing the channel
```

## Systems and Methods for Presenting Information on Mobile Devices

```

# imageUrl          title of the image representing the channel
# imageLink         link to website related to this feed
# imageWidth        Width of image in pixels max=144, default=88
# imageHeight       Height if image in pixels max=400, default=31
# imageDesc         Description from the TITLE attribute of the link formed around
#                   the image in HTML
# rating            PICS rating for channel
# textInputTitle    +
# textInputDesc     \ Not known what
# textInputName     / these fields are for
# textInputLink     +
# skipHours         Hours of day (0-23) to skip reading the feed, comma delimited
# skipDays          Days of week on which to skip reading the feed, comma delimited
# field01-10       Extra channel fields that can be populated by a custom feed parser.
CREATE TABLE FeedCollector.channel (
    channelId        INT          NOT NULL AUTO_INCREMENT,
    parentId         INT          NOT NULL DEFAULT 0,
    url              VARCHAR(512) NOT NULL,
    channelName      VARCHAR(64)  NOT NULL,
    ruleName         VARCHAR(64)  DEFAULT NULL,
    scanInterval     INT          NOT NULL DEFAULT 0,
    maxItems         INT          NOT NULL DEFAULT 0,
#-----
    numItems         INT          NOT NULL DEFAULT 0,
    status           blob,
    title            VARCHAR(512),
    link             VARCHAR(512),
    description      blob,
    language         VARCHAR(64),
    copyright        VARCHAR(512),
    managingEditor   VARCHAR(255),
    webMaster        VARCHAR(255),
    pubDate          VARCHAR(512),
    lastBuildDate    VARCHAR(255),
    category         VARCHAR(512),
    categoryDomain   VARCHAR(512),
    generator        VARCHAR(255),
    docs             VARCHAR(512),
    cloudDomain      VARCHAR(512),
    cloudPort        INT,
    cloudPath        VARCHAR(512),
    cloudRegProc     VARCHAR(255),
    cloudProtocol    VARCHAR(255),
    ttl              INT,
    imageUrl         VARCHAR(512),
    imageUrl         VARCHAR(512),
    imageLink        VARCHAR(512),
    imageWidth       INT          DEFAULT 88,
    imageHeight      INT          DEFAULT 31,
    imageDesc        VARCHAR(512),
    rating           VARCHAR(512),
    textInputTitle   VARCHAR(512),
    textInputDesc    VARCHAR(512),
    textInputName    VARCHAR(512),
    textInputLink    VARCHAR(512),
    skipHours        VARCHAR(255),
    skipDays         VARCHAR(255),
    field01          VARCHAR(255) DEFAULT NULL,
    field02          VARCHAR(255) DEFAULT NULL,
    field03          VARCHAR(255) DEFAULT NULL,

```

## Systems and Methods for Presenting Information on Mobile Devices

```
field04          VARCHAR(255) DEFAULT NULL,
field05          VARCHAR(255) DEFAULT NULL,
field06          VARCHAR(255) DEFAULT NULL,
field07          VARCHAR(255) DEFAULT NULL,
field08          VARCHAR(255) DEFAULT NULL,
field09          VARCHAR(255) DEFAULT NULL,
field10          VARCHAR(255) DEFAULT NULL,
CONSTRAINT UNIQUE KEY (channelName),
PRIMARY KEY (channelId)
);
```

### This is the definition of the item table:

```
# Field Descriptions for table "item"
# Static fields:
# itemId          is the primary key
# channelId       is the ID of the channel for this feed item
# channelPos      is the ordinal within the channel, starting with 1
#-----
# Dynamic Fields
# title           is the title field of the rss feed item
# pubDate         is the publication date
# link            is the website corresponding to the news or article.
# description     is the text of description field; all html tags are filtered out.
#
# Predefined Optional Fields:
# author          e-mail address of the author
# category        categorization. More than one can occur in item
# categoryDomain  URL that identifies categorization taxonomy.
# comments        Comments page URL to allow readers to comment on article
# enclosureURL    URL of "attachment"
# enclosureLength Size of "attachment" in bytes
# enclosureType   Type of attachment as a MIME type, e.g. "audio/mpeg".
# guid            Globally Unique Identifier
# guidIsPermaLink If "true" then guid is a URL, default is "true"
# source          The RSS channel that the item came from.
#
# field01-10     are subfields extracted from the item. The content
# of the subfields are dependent on the RSS channel. An XML driven parser
# extracts the data from the item and places them in the fieldxx fields.
#
```

```
CREATE TABLE FeedCollector.item (
  itemId          INT NOT NULL,
  channelId       INT NOT NULL,
  channelPos      INT NOT NULL,
#-----
  title           VARCHAR(512),
  pubDate         VARCHAR(512),
  link            VARCHAR(512),
  description     blob,
  author          VARCHAR(255),
  category        VARCHAR(512),
  categoryDomain  VARCHAR(512),
  comments        VARCHAR(512),
  enclosureUrl    VARCHAR(512),
  enclosureLength INT,
  enclosureType   VARCHAR(255),
  guid            VARCHAR(512),
  guidIsPermaLink VARCHAR(255),
  source          VARCHAR(512),
  field01         VARCHAR(255) DEFAULT NULL,
  field02         VARCHAR(255) DEFAULT NULL,
```

## Systems and Methods for Presenting Information on Mobile Devices

```
field03          VARCHAR(255) DEFAULT NULL,  
field04          VARCHAR(255) DEFAULT NULL,  
field05          VARCHAR(255) DEFAULT NULL,  
field06          VARCHAR(255) DEFAULT NULL,  
field07          VARCHAR(255) DEFAULT NULL,  
field08          VARCHAR(255) DEFAULT NULL,  
field09          VARCHAR(255) DEFAULT NULL,  
field10          VARCHAR(255) DEFAULT NULL,  
PRIMARY KEY (itemId),  
CONSTRAINT UNIQUE INDEX (channelId, channelPos)  
);
```

The channel table is prepopulated with fixed channel data. These are:

1. **channelId**: The primary key for this table, which is also used in calculating the itemId for the item table.
2. **parentId**: Channel ID of parent channel, or 0 if there is no parent channel.
3. **scanInterval**: The time in seconds between scans for this channel. If set to 0, FeedCollector uses the default interval specified in the properties file.
4. **numItems**: The number of items that were read in from the feed. The number may vary from scan to scan but will never exceed MAX\_ITEMS\_PER\_CHANNEL, defined in the properties file.
5. **maxItems**: The maximum number of item fields to extract from the feed and place in the database. Each feed has one or more item fields (e.g. news article). The number of fields available in the feed may be more or less than this number. If there are more item fields in the feed, the excess is ignored. If this value is set to 0, then the feeder reads as many items as are available in the feed, subject to the maximum allowable number of items set by Feed Collector. The number of items placed in the database is stored in numItems. maxItems must be less than MAX\_ITEMS\_PER\_CHANNEL, which is defined in the properties file for FeedCollector.
6. **url**: The URL at which to obtain the feed.
7. **channelName**: This string is used to identify the feed to the FeedCollector parser. If either the "ruleName" or "processor" fields are null, then this name is used for these fields. This field must be unique within the channel table. For instance, if this feed requires custom field extraction, then the name of the custom parser rule file is named with the value in channelName, with ".xml" extension.
8. **ruleName**: This name is used as the name of the rule file, with ".xml" added, to use to parse this feed. If this field value is null, then channelName is used instead to look for the custom parse rule.

In addition, there is a text **status** field that is updated by FeedCollector on each scan, which is set to "OK" for normal process of this channel, or an exception message if there were problems processing this feed. Typical exceptions are: HTTP connection where the URL described in the channel cannot be accessed, or an XML SAX parser error, where the feed contains an invalid character or a syntax error.

## Systems and Methods for Presenting Information on Mobile Devices

The prepopulated values are listed below. The item id range is where to find the channel's items in the item table. ItemId range is calculated from the values channelId and numItems:

starting value = (channelId \* (MAX\_ITEMS\_PER\_CHANNEL + 1)) + 1

ending value = (channelId \* (MAX\_ITEMS\_PER\_CHANNEL + 1)) + numItems.

There can be a maximum of MAX\_ITEMS\_PER\_CHANNEL item records per channel. The itemId (channelId\*(MAX\_ITEMS\_PER\_CHANNEL+1)) for the channel is reserved for an agglomeration record.

Systems and Methods for Presenting Information on Mobile Devices

Item ID	Chan nel ID	scan Interval	num Items	channelName	URL
1001	1004	0	4	Reuters	http://www.microsite.reuters.com/rss/topNews;
2001	2003	0	3	BBCWorldNews	http://news.bbc.co.uk/rss/newsonline_world_edition/front_page/rss091.xml;
3001	3003	0	3	NewYorkTimes	http://www.nytimes.com/services/xml/rss/ny/HomePage.xml;
4001	4003	0	3	MarketWatch	http://www.marketwatch.com/rss/realtimeheadlines;
5001	5003	0	3	ReutersFeedRoom	http://xml.feedroom.com/public_rss/reuters_rss.xml?channel_id=6ffc0233e09c26943b265ef3abd575b2d872818b;
6001	6003	0	3	NewsIsFree	http://xml.newsisfree.com/feeds/34/634.xml;
7001	7003	0	3	BostonCeltics	http://www.boston.com/sports/basketball/celtics?mode=rss_10;
8001	8003	0	3	ESPN	http://sports.espn.go.com/espn/rss/news;
9001	9003	0	3	ESPNBA	http://sports.espn.go.com/espn/rss/nba/news;
10001	10003	0	3	BBSSports	http://news.bbc.co.uk/rss/sportonline_world_edition/motorsport/formula_one/rss091.xml;
11001	11003	0	3	FoxSports	http://msn.foxsports.com/feedout/syndicatedContent?categoryId=5;
12001	12003	0	3	BGNRacing	http://www.bgnracing.com/xml_news.php;
13001	13003	0	3	FoxSports2	http://msn.foxsports.com/feedout/syndicatedContent?categoryId=220;
14001	14003	0	3	ReutersEntertainment	http://www.microsite.reuters.com/rss/Entertainment;
15001	15002	0	2	AllRecipes	http://rss.allrecipes.com/2/3.xml;
16001	16001	0	1	DrinkNation	http://rss.drinknation.com/dotw.xml;
17001	17001	0	1	DrinkNation2	http://rss.drinknation.com/martini.xml;
18001	18003	0	3	Fodors	http://www.fodors.com/blog/index.rdf;
19001	19001	0	1	WeatherAlertsCA	http://weather.gov/alerts/ca.rss;
20001	20001	0	1	WeatherAlertsMA	http://weather.gov/alerts/ma.rss;
21001	21001	0	1	YahooWeatherGA	http://xml.weather.yahoo.com/forecasts?p=USGA0028&u=f;

Systems and Methods for Presenting Information on Mobile Devices

22001	22003	22	0	3	NetflixNews	<a href="http://rss.netflix.com/NewReleasesRSS;">http://rss.netflix.com/NewReleasesRSS;</a>
23001	23003	23	0	3	SlashDot	<a href="http://slashdot.org/index.rss;">http://slashdot.org/index.rss;</a>
24001	24003	24	0	3	ESPNMLB	<a href="http://sports.espn.go.com/espn/rss/mlb/news;">http://sports.espn.go.com/espn/rss/mlb/news;</a>
25001	25003	25	0	3	NASCAR	<a href="http://sports.yahoo.com/nascar/rss.xml;">http://sports.yahoo.com/nascar/rss.xml;</a>
26001	26004	26	0	4	MyGanav	<a href="http://www.myganav.com/www/rss/fnc;">http://www.myganav.com/www/rss/fnc;</a>
27001	27001	27	0	1	YahooWeatherMA	<a href="http://xml.weather.yahoo.com/forecastrss?p=USMA0400&amp;u=f;">http://xml.weather.yahoo.com/forecastrss?p=USMA0400&amp;u=f;</a>
28001	28001	28	0	1	YahooWeatherCA	<a href="http://xml.weather.yahoo.com/forecastrss?p=USCA1085&amp;u=f;">http://xml.weather.yahoo.com/forecastrss?p=USCA1085&amp;u=f;</a>
29001	29001	29	0	1	Traffic	<a href="http://rss.traffic.com/rss.xml?c=KSPeB7dWkCYVwH%2BgE2uzMy6LdPjgMjBRfUpiAgCqI%3D;">http://rss.traffic.com/rss.xml?c=KSPeB7dWkCYVwH%2BgE2uzMy6LdPjgMjBRfUpiAgCqI%3D;</a>
30001	30002	30	0	2	CBSSportsNFL	<a href="http://cbs.sportsline.com/partners/feeds/rss/inflayerupdates;">http://cbs.sportsline.com/partners/feeds/rss/inflayerupdates;</a>
31001	31001	31	0	1	CBSSportsFootballFantasy	<a href="http://cbs.sportsline.com/partners/feeds/rss/football_fantasy_news;">http://cbs.sportsline.com/partners/feeds/rss/football_fantasy_news;</a>
32001	32002	32	0	2	CBSSportsNBA	<a href="http://cbs.sportsline.com/partners/feeds/rss/nbaplayerupdates;">http://cbs.sportsline.com/partners/feeds/rss/nbaplayerupdates;</a>
33001	33001	33	0	1	CBSSportsBasketballFantasy	<a href="http://cbs.sportsline.com/partners/feeds/rss/basketball_fantasy_news;">http://cbs.sportsline.com/partners/feeds/rss/basketball_fantasy_news;</a>
34001	34002	34	0	2	SportsOnlineTennis	<a href="http://www.sportsline.com/partners/feeds/rss/tennis_news;">http://www.sportsline.com/partners/feeds/rss/tennis_news;</a>
35001	35001	35	0	1	TVGuide	<a href="http://www.tvguide.com/TVGO!/Syndication/rss.aspx?name=holflist;">http://www.tvguide.com/TVGO!/Syndication/rss.aspx?name=holflist;</a>
36001	36005	36	0	5	YahooMoviesThisWeek	<a href="http://rss.ent.yahoo.com/movies/thisweek.xml;">http://rss.ent.yahoo.com/movies/thisweek.xml;</a>
37001	37005	37	0	5	YahooMoviesTop25Trailers	<a href="http://rss.ent.yahoo.com/movies/top25trailers.xml;">http://rss.ent.yahoo.com/movies/top25trailers.xml;</a>
38001	38005	38	0	5	NetflixQueues	<a href="http://rss.netflix.com/QueueRSS?id=P1313492412131092048894861366467261;">http://rss.netflix.com/QueueRSS?id=P1313492412131092048894861366467261;</a>
39001	39008	39	0	8	NetflixTrackings	<a href="http://rss.netflix.com/TrackingRSS?id=P1313492412131092048894861366467261;">http://rss.netflix.com/TrackingRSS?id=P1313492412131092048894861366467261;</a>
40001	40004	40	0	4	NetflixRecommendations	<a href="http://rss.netflix.com/RecommendationsRSS?id=P1313492412131092048894861366467261;">http://rss.netflix.com/RecommendationsRSS?id=P1313492412131092048894861366467261;</a>

updated: 2006.12.01

Docket: XPR002PR

H - 9



## Systems and Methods for Presenting Information on Mobile Devices

### **3.2 Parsing Rules**

The parsing rules file follow xml 1.0 specification. The tag names and attribute names are at present not part of a namespace.

Parsing rules are kept in xml files residing on the same device as the FeedCollector program. See section 6. For the names and locations of the rule files.

The following is a summary of the tags and attributes

Tag Name (no. of occurrences)	Parent	Attributes ( <i>mandatory</i> )	Description	Restrictions & Comments
rule (1..1)	N/A	N/A	Root of the rule xml file	<u>feed</u> must immediately follow <u>rule</u> .
feed (1..n)	rules	<i>name</i>	Defines the name of feed, must match "ruleName" in DB, except for <u>_RSS_</u> , <u>_RDF_</u> , and <u>_ATOM_</u> which define the standard RSS, RDF and ATOM fields respectively	Multiple <u>feed</u> elements can be defined within a <u>rule</u> block (though for the custom rule files, it is one feed per file). Each <u>feed</u> must have a unique <u>name</u> attribute value.
channel (0..1)	feed	tagname, level	Channel data to be placed in the channel table	Only 1 <u>channel</u> definition per <u>feed</u>
item (0..1)	feed	tagname, level	item data to be placed in the item table.	Only 1 <u>item</u> definition per <u>feed</u>
field (0..n)	item, channel	<i>dbname</i>	field in the db to be populated	Multiple <u>field</u> elements per <u>item</u>
tag (1..n)	field, tag	<i>name, type, value, minOccurs, maxOccurs</i>	Tag in feed to process. Actual data may be in tag as attribute, in the text content or in nested tags.	Multiple and nested <u>tag</u> elements are permissible. Multiple <u>tags</u> at same level: extracted data is concatenated in order of tag definition.
parser (0..1)	tag	<i>type</i>	Switch to parser other than xml.	A switch to HTML parser w/o any tags will extract text stripped of all tags.
suffix (0..1)	tag, htmltag	n/a	Add this suffix to extracted data	Limitation: Only works within outermost <u>tag</u> .
prefix (0..1)	tag,	n/a	Add this prefix to	Limitation: Only

Systems and Methods for Presenting Information on Mobile Devices

	htmltag		extracted data	works within outermost tag.
htmltag (0..n)	parser, htmltag	name, type, value	Tags that occur in the HTML section.	Similar in behavior to tag.

The following table defines the various attributes

Attribute name	Applies to Tags	Description	Restrictions
name	feed	Name of the feed to which this rule is applied. The name corresponds to the channelName field in table channel.	mandatory attribute, and value must not be empty. Special values <u>_RSS_</u> , <u>_RDF_</u> , and <u>_ATOM_</u> is used to extract the standard fields for db.
name	tag, htmltag	Name of the tag to examine to extract data	mandatory attribute, and value must not be empty.
level	channel, item	Indicates the nesting level at which to find this tag, relative to the top level tag, e.g. <rss> = level 0.	At present not used. This is to differentiate tags that occur at different levels.
dbname	field	This is the name of the field to populate in the database.	
tagname	channel, item	This is the name of the tag that corresponds to either the channel or item element.	Optional attribute for tags "channel" and "item", used if the tag name is not "channel" and "item, respectively, as when parsing ATOM feed.
type	parser	Indicates the type of parser to switch to	At present the two acceptable choices are "html" and "regex". Regex parser is not implemented.
type	tag, htmltag	Indicates whether value is found in an attribute value or as text content.	Value must either be "attr" or "text". If nested tags are used, set this value to "text".
value	tag, htmltag	If the <u>type</u> attribute indicated "attr", the value of this attribute is the name of the attribute whose value is to be extracted.	<u>value</u> attribute must be present when <u>type</u> = "attr". The value must not be empty
minOccurs	tag	The minimum number of times this tag is expected to occur	Used for repeating elements, such as skipdays.
maxOccurs	tag	The maximum number of times this tag is expected to occur	Used for repeating elements, such as skipdays.

## Systems and Methods for Presenting Information on Mobile Devices

### 3.2.1 Rule Syntax

XML Schema for rule file. For now use tag and attribute descriptions in section 3.2 and the examples in the appendix.

## 3.3 Elements with Special Semantics

### 3.3.1 Special Elements to Support FeedLists

It is possible to store a tree structured multiple level feeds in the Feed Collector. The master node's items' link fields specify the URL of another feed. Such a feed is called "Feed List". Feed Lists can enumerate other Feed Lists, such that one can create a multilevel tree. FeedCollector will recognize a feed as being a Feed List if the following is true:

In the "channel" category, there is a "category" element whose value is "\_FeedList\_"

In addition, in order to be able to specify a custom parse rule for the sub feeds, one can specify the custom rule for each sub feed listed in the item category as follows:

In the "item" category, there is a "category" element whose value is "\_parseRule\_", where "parseRule.xml" would be the name of a custom rule file.

Each item in a field can have its own custom rule file, allowing heterogeneous feeds to be listed within a single master feed.

NOTE: channelId for all sub feeds are assigned dynamically by FeedCollector. There can be no assumptions made about the absolute value of a channelId other than for the master Feed List. itemId for sub feeds are dynamically assigned based on the channelId that is automatically generated, hence a value for any particular item cannot be determined a priori.

#### 3.3.1.1 FeedList Example

The following figure is an example of a Feed List type RSS feed:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<rss version="2.0">
  <channel>
    <title>Sample Feed List</title>
    <copyright>Copyright (c) 2006 Express Mobile Inc.</copyright>
    <link>http://xpresme.com/FeedCollector/test/cattree</link>
    <description>Master List of songs</description>
    <lastBuildDate>Mon, 13 Nov 2006 20:49:53 UTC</lastBuildDate>
    <category>_FeedList_</category>

    <item>
      <title>Most Popular</title>

      <link>http://xpresme.com/FeedCollector/test/mostpopularsample.xml</link>
      <description>Most Popular Category</description>
      <category> CSMCategory </category>
    </item>
  </channel>
</rss>
```

## Systems and Methods for Presenting Information on Mobile Devices

```
</item>
<item>
  <title>New Songs</title>
  <link>http://xpresme.com/FeedCollector/test/newsongsample.xml</link>
  <description>New Songs Uploaded into the system</description>
  <category>_CSMCategory_</category>
</item>
<item>
  <title>Charts</title>
  <link>http://xpresme.com/FeedCollector/test/chartsampler.xml</link>
  <description>Charts - Best Songs of the Month</description>
  <category>_CSMCategory_</category>
</item>
<item>
  <title>Summer Hits</title>
  <link>http://xpresme.com/FeedCollector/test/summerhitssample.xml</link>
  <description>Summer Hits Songs</description>
  <category>_CSMCategory_</category>
</item>
</channel>
</rss>
```

In this example Feed List, there are 4 sub feeds, each described in its own item element. Each item can have title, description or any other field. “link” is required, and this link points to another feed. Either description or title is required to meet the RSS specifications. The category element for each item has value “\_CSMCategory\_”. This indicates that this feed should use the custom rule file named “CSMCategory.xml” when parsing its content. “category” field is optional; if not present, then the default parse rule for the feed type will be used.

### **3.3.2 Other Fields that Affect FeedCollector Behavior**

Field “lastBuildDate” in a feed is used to minimize the updates required in the database, as well as to minimize network traffic for unnecessary scans of subfeeds for multi level Feed Lists. This is an optional field in the feed. If not present, the feed will be rescanned and the database is updated at the preset interval for this feed.

If this value changes from scan to scan, then the fields for this feed are updated. Conversely, if the value has not changed, the database is not updated for this feed channel.

For Feed Lists, if the value of “lastBuildDate” has changed in the master Feed List, all subfeeds are rescanned and the values updated in the database. Conversely, if the value in the master has not changed, then none of the sub feeds are scanned, nor is the database updated.

## Systems and Methods for Presenting Information on Mobile Devices

### **3.4 Properties File**

The properties file is a simple plain text name name-value pairs. The following is an example:

```
TIME_INTERVAL=1200
FEED_DB_URL=jdbc:mysql://localhost:3306/FeedCollector|publisherAdmin|publisherPassword
MAX_ITEMS_PER_CHANNEL=999
TITLE_EXCEPTION=Not Available
PUBLICATION_DATE_EXCEPTION=Not Available
LINK_EXCEPTION=Not Available
DESCRIPTION_EXCEPTION=Not Available

#Format is:
#Time interval is in seconds. This defines the default feed scan interval
#Feed URL string consists of URL, database username & password. Fields separated by |
#MAX ITEMS PER CHANNEL + 1 is distance between 1st items of 2 adjacent channels
```

The description of each field :

Property Name	Description	Comments
TIME_INTERVAL	Number of seconds between scans, whose value is greater than 0.	This is the default scan rate, which is used if scanInterval is set to 0 in the channel table for a particular channel
MAX_ITEMS_PER_CHANNEL	This sets the ceiling on the number of items each channel can record in the database.	To obtain the itemId of the 1 <sup>st</sup> item for a channel, itemId = channelId * (MAX_ITEMS_PER_CHANNEL + 1) + 1
RSS_DB_URL	This is the connection to the database, the username, and the password. The   is used as a field delimiter.	The account described by username/password must be able to freely manipulate FeedCollector database.
TITLE_EXCEPTION	String to display in the title field of an item if feed could not be read.	Not Used. Tagged for removal.
PUBLICATION_DATE_EXCEPTION	String to display in the pubDate field of an item if feed could not be read.	Not Used. Tagged for removal.
LINK_EXCEPTION	String to display in the link field of an item if feed could not be read.	Not Used. Tagged for removal.
DESCRIPTION_EXCEPTION	String to display in the description field of an item if feed could not be read.	Not Used. Tagged for removal.

Properties file is an external text file

### **3.5 Feeds**

The format of the feeds must be one of the following:

- a. RSS (various versions)
- b. ATOM 1.0 (Limited testing)
- c. RDF (Not tested)

## Systems and Methods for Presenting Information on Mobile Devices

Other formats could be handled using the custom feed rule mechanism.

### **4 Output Data**

All output, other than trace and error messages are written directly to the FeedCollector database. The database tables and fields are defined in section 3.1.

Tracing and recoverable errors are sent to stdout, which is usually redirected to a log file in the log folder.

Catastrophic errors will be posted on the screen in a red dialog box. A record of the error is also written out to stderr which is redirected to the debug.rss folder.

### **5 Control and Parameterization**

#### **5.1 Selecting Fields for Extraction**

Create or modify a custom parsing rule file to extract specific fields not covered in the default parser, or for modifying the value from that of the default parser.

#### **5.2 Changing the Scan Interval**

##### **5.2.1 Default Scan Interval**

The default scan interval is in the properties file, under key "TIME\_INTERVAL". The value is in seconds. Changing this value will change the scan interval for all channels that use the default scan interval, as specified by having scanInterval = 0 in the channel table. Once the value has been changed, stop and restart FeedCollector for the change to take effect.

##### **5.2.2 Individual Channel Scan Interval**

To change the scan interval for an individual channel, change the value of the scanInterval field in channel table in FeedCollector.sql. Then run the DDL with MySQL to recreate the database, e.g.:

```
MySQL -uroot -prootpassword < FeedCollector.sql
```

Restart FeedCollector for the changes to take effect.

#### **5.3 Changing the Database Access**

By default, the FeedCollector database on localhost is accessed. To change the location of the database, change the RSS\_DB\_URL property in rss.properties, then restart FeedCollector program.

### **6 Dependent Files and their Locations**

The following files are required for the Feed Collector to function properly. All locations are relative to PUBLISHER\_HOME, the root folder of the Publisher server system.

File Name	Location	Purpose
FeedCollector.bat	Install	Batch file to start up FeedCollector
feedcollector.sql	Install	Script to initialize FeedCollector

## Systems and Methods for Presenting Information on Mobile Devices

		database. Contains schemata for channel and item tables, and the initialization of the channel table with predefined feeds.
FeedCollector.jar	jar	Java program for feed collector
common.jar	jar	Dependent java library
htmlparser.jar	lib	Dependent java library
log4j-1.2.13.jar	lib	Dependent java library
mysql-connector-java-3.0.11-stable-bin.jar	lib	Dependent java library
soap.jar	lib	Dependent java library
rowset.jar	lib	Dependent java library
FeedCollector.properties	FeedCollector/conf	Properties file for FeedCollector
RSS .xml	FeedCollector/rule	Default parsing rule for RSS feed
ATOM .xml	FeedCollector/rule	Default parsing rule for ATOM feed
RDF .xml	FeedCollector/rule	Default parsing rule for RDF feed
<i>feedname.xml</i>	FeedCollector/rule	Parsing rule for a specific feed with the name " <i>feedname</i> ". This rule can override fields defined in one of the default parsing rule files.
FeedCollector.log	FeedCollector/log	Trace and log file for FeedCollector.jar (redirected from stdout)

## 7 External Document Links

The RSS specifications are available at:

<http://www.rss-specifications.com/rss-specifications.htm>

The RDF specifications are available at:

<http://www.w3.org/RDF/>

The ATOM 1.0 specifications are available at:

<http://atompub.org/2005/07/11/draft-ietf-atompub-format-10.html>

ROME (a generalized parser for RSS and ATOM formats) is available at:

<https://rome.dev.java.net/>

ROME was initially considered for use in FeedCollector, but was ultimately not used.

## 8 Appendix

### 8.1 Standard Fields Parsing Rule

The code below extracts the standard fields from an RSS feed into the database

```
<?xml version="1.0" encoding="UTF-8" ?>
<!--
    Definition of the default RSS Feed fields in the database FeedCollector
    =====
    Written by: Ken Brown
    (c) 2006 Express Mobile, Inc.
```

## Systems and Methods for Presenting Information on Mobile Devices

38 Washington St.  
Novato, CA 94947  
USA

```
-->
<rule>
  <feed name="_RSS_">
    <channel tagname="channel" level="1">
      <field dbname="title">
        <tag name="title" type="text">
          </tag>
        </field>
      <field dbname="pubDate">
        <tag name="pubDate" type="text">
          </tag>
        </field>
      <field dbname="link">
        <tag name="link" type="text">
          </tag>
        </field>
      <field dbname="description">
        <tag name="description" type="text" >
          </tag>
        </field>
      <field dbname="imageTitle">
        <tag name="image" type="text" >
          <tag name="title" type="text">
            </tag>
          </tag>
        </field>
      <field dbname="imageUrl">
        <tag name="image" type="text" >
          <tag name="url" type="text">
            </tag>
          </tag>
        </field>
      <field dbname="imageLink">
        <tag name="image" type="text" >
          <tag name="link" type="text">
            </tag>
          </tag>
        </field>
      <field dbname="imageDesc">
        <tag name="image" type="text" >
          <tag name="description" type="text">
            </tag>
          </tag>
        </field>
      <field dbname="imageWidth">
        <tag name="image" type="text" >
          <tag name="width" type="text">
            </tag>
          </tag>
        </field>
      <field dbname="imageHeight">
        <tag name="image" type="text" >
          <tag name="height" type="text">
            </tag>
          </tag>
        </field>
    </channel>
    <!-- ITEM STARTS HERE -->
    <item tagname="item" level="2">
      <field dbname="title">
        <tag name="title" type="text">
          </tag>
        </field>
      <field dbname="pubDate">
        <tag name="pubDate" type="text">
          </tag>
        </field>
      <field dbname="link">
```



## Systems and Methods for Presenting Information on Mobile Devices

```
        <tag name="link" type="text">
        </tag>
    </field>
    <field dbname="description">
        <tag name="description" type="text" >
            <!-- setting parser to html without any tags will create
                filtered text -->
            <parser type="html"></parser>
        </tag>
    </field>
</item>
</feed>
</rule>
```

### 8.2 Example Custom Fields Parsing Rule

This example is for parsing Yahoo! Weather RSS feed

```
<?xml version="1.0" encoding="UTF-8" ?>
<!--
    Example: rules for parsing Yahoo! Weather channel
    =====
    http://xml.weather.yahoo.com/forecastrss?p=USGA0028&u=f
-->
<rule>
<feed name="yahooweather">
<channel tagname="channel" level="2" ></channel>
<item tagname="item" level="2">
<!-- field dbname indicates the DB field name into which data will be stored-->
<!-- in this case, extract an image URL from the text of tag "description"-->
<field dbname="field01">
<!-- the information is to be found in the text portion of tag description-->
<!-- inside each item. The other option is type="attr" if value is found -->
<!-- in the attribute of this tag. In this case value="attrvalue" is present.-->
<tag name="description" type="text">
<!-- switch from xml to            to parse the text of "description"-->
<parser type="          ">
<!-- the information is the "          "            of the            tag-->
<htmltag                                  >
</htmltag>
</parser>
</tag>
</field>

<!-- extracts temperature and text from a custom tag yweather:condition-->
<!-- resulting field will read e.g., "                      Sunny." -->
<field dbname="field02">
<tag name="yweather:condition" type="attr" value="temp">
          
</tag>
<tag name="yweather:condition" type="attr" value="text">
          
</tag>
</field>

<!-- extract the weather description-->
<field dbname="field03">
<tag name="yweather:condition" type="attr" value="text">
</tag>
</field>
</item>
</rule>
```

## Systems and Methods for Presenting Information on Mobile Devices

### **8.3 Error messages**

FeedCollector runs as a “headless” application on the server. Except for catastrophic errors, it is silent. A trace file is created in the log subfolder, which can be useful for diagnostics. A catastrophic error will bring up a red popup screen with the stack trace and the exception detail. FeedCollector will terminate after a catastrophic error.

If a failure occurs during the reading of a feed, the “**status**” field in the channel table for the feed will be populated with the exception message. One can use a simple script to scan this field on a periodic basis to determine what, if any, feeds had problems.

# Appendix I

## Response File Redirector

### 1.0 Introduction

#### 1.1 Purpose

This document describes the basic design details for Response File Redirector, formerly known as JAD File Redirector, subcomponent of project Publisher 2.0. The purpose of this document is to provide a brief overview of the software architecture and the design goals of Response File Redirector.

This System Design document focuses on System level aspects / functionality of the Response File Redirector; this document shall be used as the basis for determining any system level considerations.

#### 1.2 Intended Audience

The audience for this SDD includes the employees/contractors of Akira Technologies playing roles in project management, the system architects and the developers who participate in the system design, and the developers who design/implement any Response File Redirector /related functionality in future.

#### 1.3 Scope

The scope of this document is limited to list the details of proposed Response File Redirector System's Design specifications.

Systems and Methods for Presenting Information on Mobile Devices

## 1.4 Assumptions, Dependencies & Risk

- **Database dependencies:**

Response File Redirector will fetch relevant data from following tables named **application\_record**, **response\_file\_record** and **WURFL database**. Any change in the specification of these tables may lead to design changes in the application

Table Name : application_record				
Column Name	Type	Not Null	Constraints	Index
application_id	INTEGER(10) unsigned	y	PK	
application_name	VARCHAR (50)	y		
category_name	VARCHAR (50)	y		
version	INTEGER(10) unsigned	y		
application_UUID_high	bigint(20)	Y		
application_UUID_low	bigint(20)	Y		

Table Name : response_file_record				
Column Name	Type	Not Null	Constraints	Index
response_file_id	INTEGER(10) unsigned	y	PK	
application_id	INTEGER(10) unsigned	y	FK	
response_file_path	VARCHAR (255)	y		
certificate_id	Integer	Y		
jad_property_flags	Integer	Y	Default = 0	
operator_flags	Integer	Y	Default = 0	
screen_h	INTEGER(10) unsigned	y		
screen_v	INTEGER(10) unsigned	y		
isEmbedded	BOOLEAN	y		
hasVideo	BOOLEAN	y		
video_format	VARCHAR(45)	Y		
type	VARCHAR(45)			

**Description of Fields:**

- a. **certificate\_id** Int – ID of certificate info in another table (this won't be used right now – for future expansion)
- b. **jad\_property\_flags**
  - a. Bit 0 (LSB) – former is\_embedded
  - b. Bit 1 – former has\_video
  - c. Bit 2 – has\_certificate - determines if the jad file contains a certificate
  - d. Bit 3 – has\_JPEG\_decoder (for Palm support)
  - e. Bits 4-31 reserved
- c. **Operator\_flags:** These are operator specific flags. It is set by Publisher Server to indicate suitability of a jad file for a specific operator. Multiple bits may be set if a single jad file can support multiple operators. If a bit for the operator found in Step 1 is not set then other means are used to find the best fit jad file.
  - a. Bit 0 (LSB) – is\_ATT – true if jad for AT&T (Cingular).
  - b. Bit 1 – is\_Sprint – true if jad for Sprint PCS.
  - c. Bit 2 – is\_Verizon – true if jad for Verizon.
  - d. Bit 3 – is\_Tmobile – true if jad for TMobile.
  - e. Bits4-7 – reserved for additional operators

## Systems and Methods for Presenting Information on Mobile Devices

- f. Bit 8 – is\_VFJ true if jad for VFJ (Vodafone Japan)
  - g. Bit 9 – is\_DoCoMo – true if jad for NTT DoCoMo
  - h. Bit 10 – is\_JT – true for Japan Telecom
  - i. Bit 11 – is\_KDDI – true for KDDI
  - j. Bit 12 – is\_Willcom – true for Willcom
  - k. Bits 13-15 – reserved for additional operators
  - l. Bits 16-31 – reserved for additional operators
- 
- The suggested architecture is based on the current heuristic logic. Any change in it may lead to design changes.
  - The suggested architecture is dependent upon the publisher server and assumes that it creates .jad/.html files in accurate format. Any anomaly in its behaviors may affect the performance of Jad file redirector and may lead to design changes.
  - The suggested architecture is based on the catering all the aspect of current implementation. Any changes from the current implementation which impacts the Architecture will be catered with a change request.
  - The architecture takes care of only two types of clients i.e. web and J2ME, and gives priority to J2ME client in providing service when there is a doubt.
  - The architecture is based upon the correctness of WURFL database. Certain devices may not be supported due to outdated database.
  - The architecture uses third party Java API's provided along with WURFL database. Any error occurring due to bugs in Java API needs to be separately handled.
  - The architecture is based upon the present information about correctness and formatting of Maxmind GeoIP database.
  - The architecture uses third party Java API's provided along with Maxmind GeoIP database. Any error occurring due to bugs in these API's needs to be separately handled.
  - The third party WURFL database needs to be periodically updated. Instructions for this task will be delivered along with other documents.
  - Only 3 palm devices are supported for now and only two resolutions (QVGA (240x320) format, OR QCIF (176x208) format) are supported for other MIDP devices. Any extension to include more clients or introduction of finer category checks will need a code change through a change request. (recognize palm device but send QVGA midp file )
  - Initially, only 30 most common PC browsers, OS pairs are to be supported by Jad file redirector. Any, future requirement to support a new 'web-browser, OS' pair would need database append. Also, The PC browser identification algorithm assumes that majority of web-browser religiously send their default user-agent, any change in their default user-agent may lead to non compliance with Jad-file redirector. (design modified)
  - In future, Introduction of a standard related to user-agent field's format may lead to design changes.
  - This application does not implement any user authentication schema.
  - No Load testing and Performance criteria is bench marked for this design ( connection pooling implemented to increase performance)

## Systems and Methods for Presenting Information on Mobile Devices

- Any database changes or functional changes out of scope will be taken care through a change request mechanism.
- This Design assumes Tomcat 5.0 as Servlet Container and Mysqi 4.0.18 as database.

## 2.0 Physical data design

Table Name : transaction_record				
Column Name	Type	Not Null	Constraint s	Index
transaction_id	INTEGER(10)	Y	PK, auto increment	
transaction_date	TIMESTAMP	Y		
requested_app_id	Int(10) unsigned			
device_id	VARCHAR(64)			
response_file	VARCHAR(255)			
IP_address	VARCHAR(32)	Y		
telno	VARCHAR (32)			
player_id_high	Bigint(20)	Y		
player_id_low	Bigint(20)	Y		

## 2.1 Program Architecture Design

### 2.1.1 Flow Diagrams

Diagram 1: The Following diagram describes the updated Work flow in JAD File Redirector

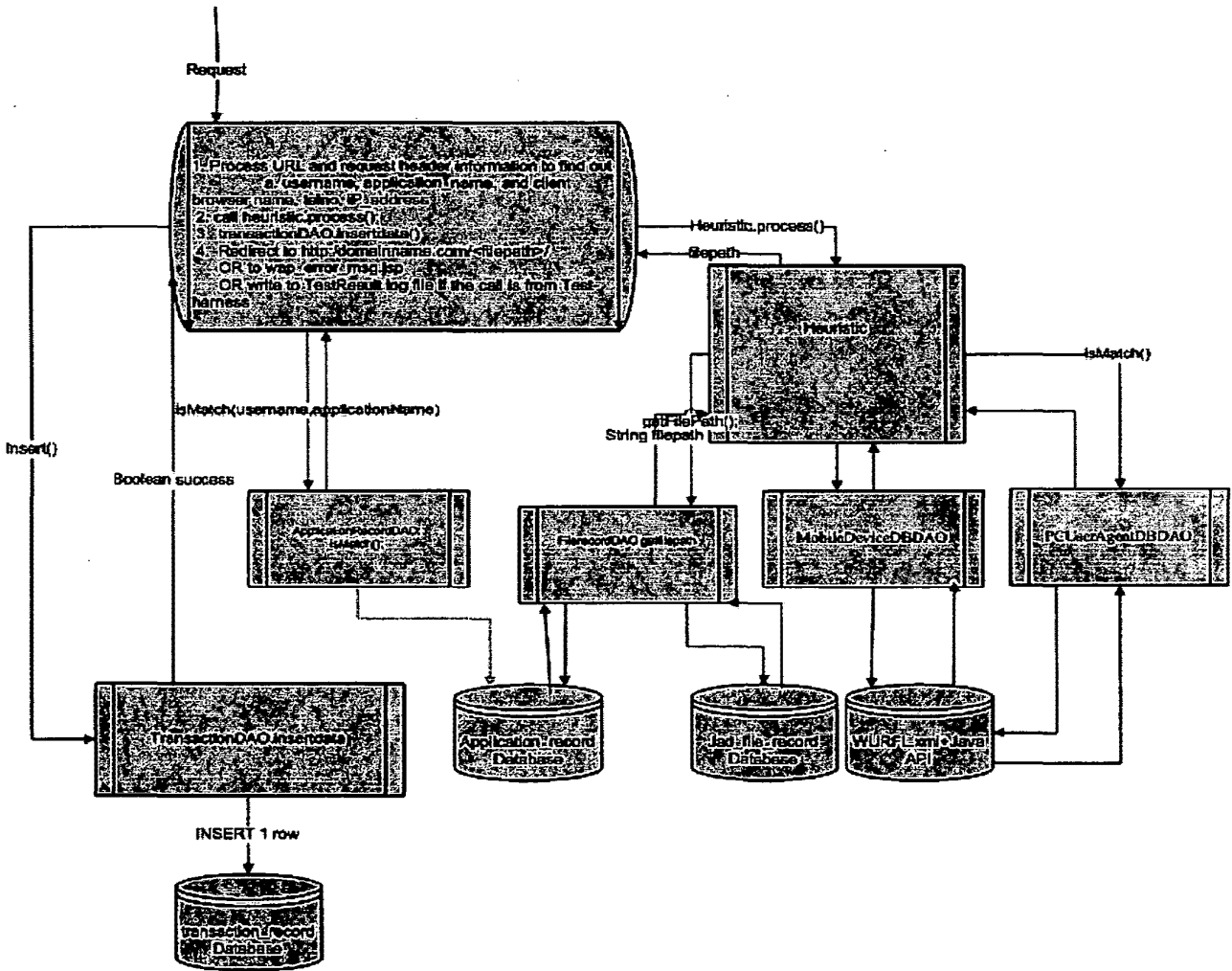


Diagram 2: The Following diagram describes the Heuristic decision flow in JAD File Redirector





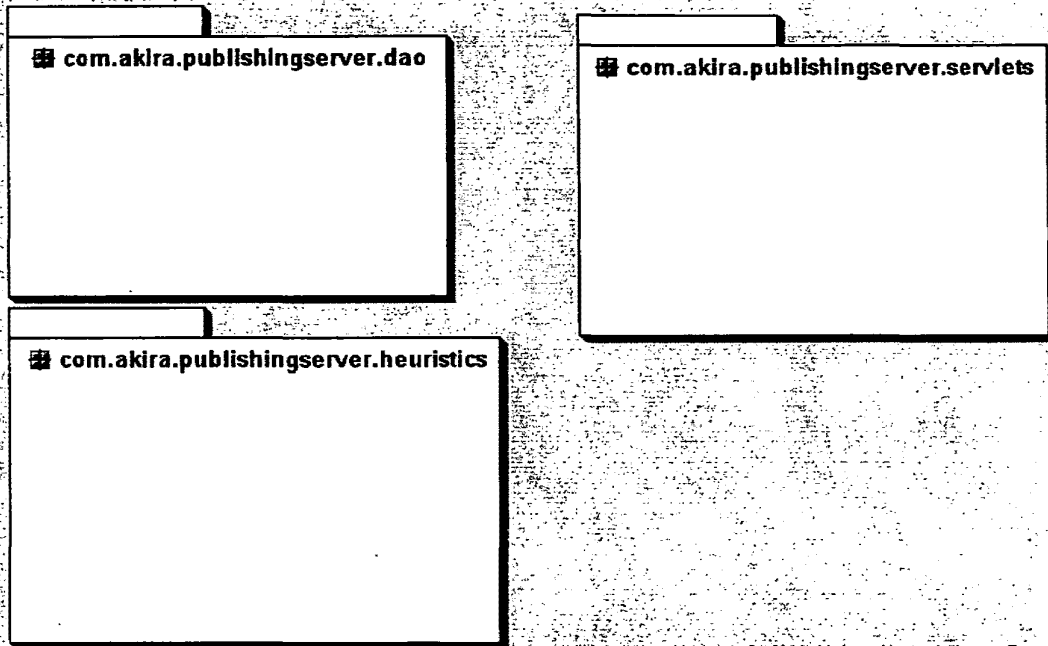
## Systems and Methods for Presenting Information on Mobile Devices

device as Palm Treo 650.

### 2.1.2 Physical Programming units

#### 2.1.2.1. Package Structure

JadFileRedirector architecture contains following packages: (new project created with Name jadredirector )



#### 2.1.2.2. Web Framework:

Web Framework contains the basic framework classes needed to manage a request from client (Player), process it and redirect the client system to appropriate .ad / html file or to an error message.

**JadRedirect:** It is a servlet which would wait for the client request. Upon request arrival, its task is to validate the requested application. If validated the appropriate file's path has to be found using heuristic, to record the transaction details and hereafter to redirect the application to the file's URL

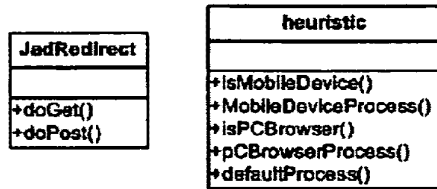
**Heuristic:** This class uses various decisions to find out most appropriate URL where the client browser should be redirected. (As described in the decision flow diagram earlier)

**Util:** utility classes

Docket: XPR002PR

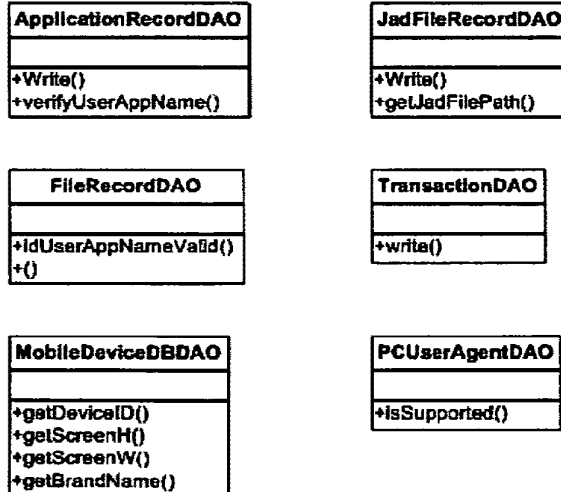
1 - 9

## Systems and Methods for Presenting Information on Mobile Devices



### 2.1.2.3. Database Access Framework

Following classes form the Database Access Framework Mainly to act as an abstraction layer.



BaseDAO & ConnectionFactory for connection pooling

Classes which form the database abstraction layer

### 2.1.2.4. Configuration files

1. JadRedirect.properties
  - Path of wurfl.xml (wurfl)

## Systems and Methods for Presenting Information on Mobile Devices

- Supported Palm device list, ~~supported screen resolutions list~~ Screen resolution to be added once there would be more specific decision related to screen resolutions. Currently 320X240 & 208X176 are hardcoded.
- Other global variables related to deployment & functionalities
  - http\_context
  - web\_inf\_path
  -

2. web.xml
  - Servlet's address Mapping(Jadredirect, PropertiesController, TestHarness, wap\_error\_msg.jsp)
3. build.xml :
  - Compile & deploy settings
4. AkiraLog4JConfig.xml
  - \* Log4J configuration file
5. get.xml
  - configuration file for connection pooling
6. error.properties
  - Error messages for .wml error file

## Appendix J

# Layout by Pixel Specification

### A: Background

In previous versions of XMO Publisher all Form objects currently are sized, in terms of height, by the line height as generated by the selected font size. Submit, and Clear buttons have their width defined by the width of the character string, as also determined by the selected font metrics.

Although there is a provision to “don’t scale” in many cases, it still means that the designer must control the actual layout through the font metric settings.

The difficulty is that MIDP does not have a font generator, and the fonts that are supported on phones can have very different font metrics, for the same logical font size.

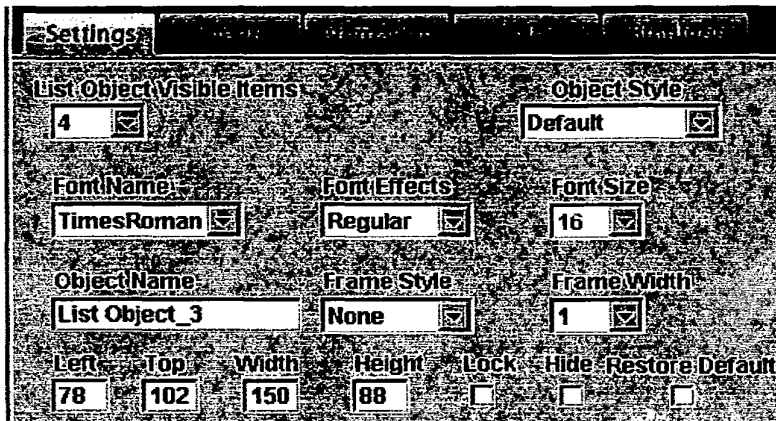
Since all other platforms do have compatible font metrics, the current method will remain, but there will be a new selection under the “Project” menu, “Define Form Objects by Pixels”. This will become a user preference and will be remembered between editing sessions. It can be changed at any time, even while editing an application.

## Systems and Methods for Presenting Information on Mobile Devices

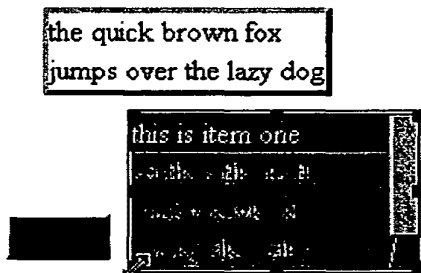
### **B: UI Implementation**

Font Size will be a text field showing the calculated value and will not be selectable

Both the width and height fields will always be selectable. The height field will still represent the total height of the Object. For Text Areas and Lists the Line Height will always be the Object Height divided by the Number of Visible Lines.



In addition, all Form objects, other than a check box, will now have 8 drag points so that the object can be resized with the mouse. Upon releasing the mouse a new font, on a best fit basis, will be selected and the text will then be reformatted based on this new font. Only a change in the height will affect the font, a width change will only change the line lengths that will be drawn.



For Submit and Clear buttons there will remain a "Don't Scale" Check Box. If scaling is enabled then the width of the button will be dynamically calculated on the phone based on the actual length of the Text String after a best-fit font is selected.

## Systems and Methods for Presenting Information on Mobile Devices

### **C: Effect on Vertical Positioning**

All text objects, including text buttons and paragraphs will be affected.

The only exceptions are Lists as they were previously vertically aligned by the top of the bounding box of the first item in the list.

When in “Define Form Objects by Pixels” mode the following objects will be vertically aligned based on the top of the bounding box of the first text string.

- Paragraph
- Text Button
- Submit Button
- Clear button
- Text Field
- Text Area
- Drop Down (Choice) List.

When in the normal “Font Typographic mode these objects will be vertically aligned on the baseline of the first text string.

### **D: MIDP Player Implementation**

All objects are now aligned in the same way, and their location and size is fixed in terms of pixels, with the exceptions of Text Buttons and Paragraphs, as well as Submit or Clear Buttons if the “scaling” option is chosen.

In all cases font sizes are picked on a best fit basis. This will determine what the line height of the object will be, even though the total height of the object will not change.

The Publisher, when in “Define Form Objects by Pixels” mode, will set the value of ObjectInteger [2] to the optimal line height based on the author’s selection.

This value will, as the PDL is read, be compared against the available font choices that the phone provides. A font will be selected so that the selected font’s line height will be the smallest possible, but equal to or greater than the object’s the optimal line height.

## Systems and Methods for Presenting Information on Mobile Devices

### **B: PDL Specification**

ObjectIntegerAttribute[2]

Text Objects: Vertical Offset

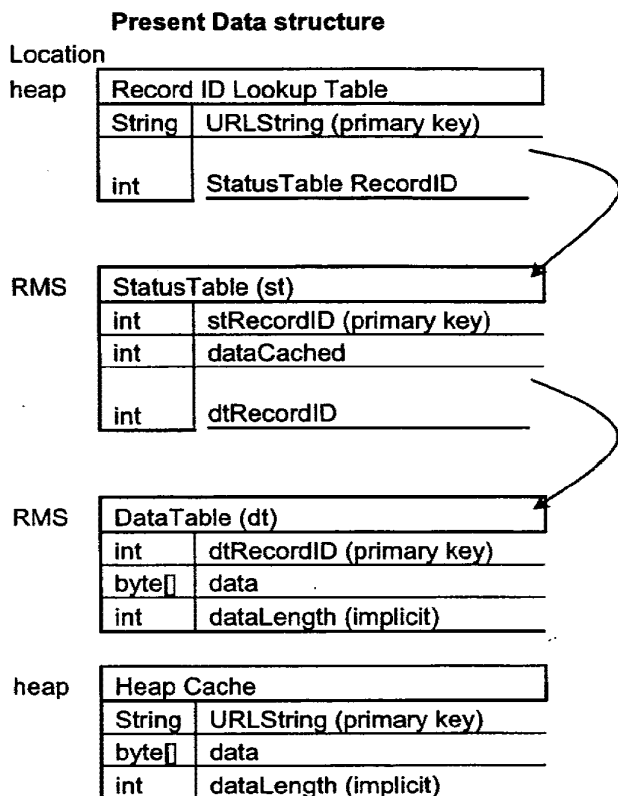
Text Objects Pixel Mode: Line Height in Pixels

ObjectIntegerAttribute[5]

YValue (For Text Objects it represents the baseline of the top line).

Text Objects Pixel Mode: Top of the first line's bounding box.

## Appendix G Data Structures

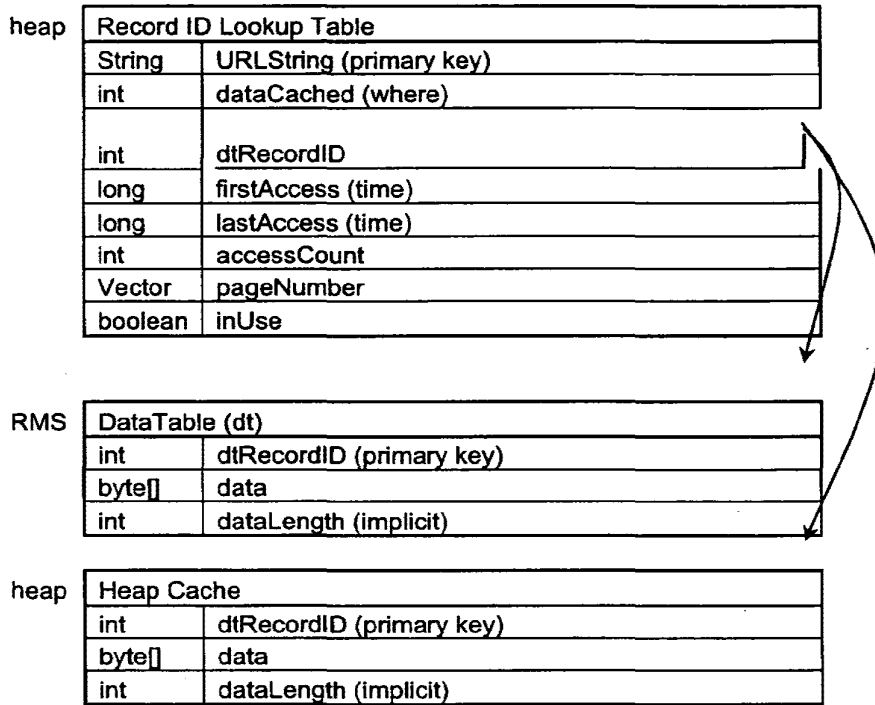


Using the present structures, once a URL request comes in, it requires 2 look ups, once from the RecordID lookup Table to the Status Table, and then again from the Status Table to the Data Table or Heap Cache, to finally obtain the data reference. Heap data requires the same number of lookups because the value of dataCached must be ascertained. Also, 4 tables are required to maintain the cache. Lastly, there is a redundancy in the storage of URLString in two locations.



## Systems and Methods for Presenting Information on Mobile Devices

### Proposed Data structure



Using the proposed structure, only one lookup is necessary to obtain the data reference, which can either be in RMS or heap. It not only simplifies the lookup but also reduces the number of tables to maintain. Also, the RMS and heap data tables have orthogonal structures, insofar as the elements are concerned. Finally, the URLString is stored only once reducing the overall overhead of cache maintenance. Additional fields in the Lookup table in the proposal are anticipated for developing a cached object release algorithm.

#### Additional proposed fields:

**firstAccess** is the timestamp of when the data was first accessed, in milliseconds since Jan 1 1970. session. The time may be from an earlier session.

**lastAccess** is the last access time in milliseconds since Jan 1 1970. The time may be from an earlier session.

**accessCount** is the number of times this object was requested. This value is reset to 0 if the application is restarted with a preserved cache. Only requests for the current session are counted.

## Systems and Methods for Presenting Information on Mobile Devices

**pageNumber** indicates the page(s) on which the reference to this object exists. Items on the current page take higher priority than items on other pages. MultilevelCache will maintain an instance variable containing the current page number. The current page number will be stored in this Vector at the time this object is cached and on each reference to this object.

**inUse** is set when the cache memory cannot be released for this object. This happens during the interval that the cached data is obtained by `getData` until the time the obtains data is closed. The `close` will set the object's `inUse` flag to false.

## Appendix F

# Enhanced Event Specification

### **I: Discussion:**

There are currently 28 possible exit events that can be assigned to an XMO Object, which have made the UI very complicated and unwieldy. Although internally the exit events will be stored as per Appendix A, for the UI, the following changes are proposed.

In addition, there are now two new Events:

1: **Send IM Alert (Push Rich Content Messaging)**

For Sending an IM Alert Message. This can occur by:

- firing on a basic object that has a logical link to a text field or text area
- firing on a List or Dropdown Item .
- firing on a slide with an attached phlog (chat room)

An IM Alert Message will generate, from the Content Server, an IM Alert Page (I may use the System Alert Object if the Alert Page cannot be found.

An IM Alert will have a Text Message defined, which will be extracted from either a designated text field or text area. Optionally, an Avatar, which could be an Image or Video, can be linked to the Alert Page. For Chat, the Avatar will be the Slide Media Object. If the Player is open the IM Alert will immediately Appear as a Modal Alert with Fire enabled for dismissing. If the player is not active, when the player later becomes active there will be an indication of the number of pending IM alerts. The method or methods for browsing the pending alerts is yet to be defined.

In the case of a phlog (chat), all members of the chat group, if not already active and have not been previously alerted to this chat during the current session, will receive the IM Alert.

For Chat, some additional specification work needs to be done. Two of the issues are:

- How will we define a chat "owner"?
- When a Chat is initiated by the Chat owner, what should the welcoming message be? This could be defined in the Master Page or at the SlideShow Level.
- How will the Chat Group be defined?

2: **Set Starting Page (already implemented)**

For personalizing the starting page by the consumer. This can occur by firing on a basic object or firing on a List or Dropdown Item. For any subsequent uses of the application, or any "Change View" events in the current session, if the "Change View to Starting Page" is defined, then the consumer-chosen "starting page" will appear.

## Systems and Methods for Presenting Information on Mobile Devices

### **II: User Interface:**

There will be up to six dropdown menus for defining an Event. A subset of these choices will be available, depending upon the type of Object that is selected.

- 1: Alerts:
  - None
  - Display on Fire
  - Display with Synchronization on Fire
  - Display/Hide OnFocus/OffFocus
  
- 2: Communication
  - None
  - Place Phone Call
  - Send SMS Message
  - Send IM alert
  
- 3 Share:
  - None
  - Share peer to peer
  - Share multiple selection
  - Share all
  
- 4 Personalization:
  - Add Icon to Icon Strip
  - Remove Icon from Icon Strip
  - Set Logical First (Starting) Page
  
- 5 Change View
  - None (next slide)
  - Goto External Page (WAP or HTML)
    - o Change View options
      - Replace Current View
      - Launch New Window
      - Replace Top View
  - Goto Internal web Page
    - o Change View Options
      - Next Internal Page
      - Specific Internal Page
      - Specific Internal Page w/Slide Number
      - Goto Widget
      - Goto Logical First (Starting) Page
  - Execute an XpressMo Application
  - Exit Application
  - Exit Player
  - Execute Scripting Method
  - Pause/Resume Page Timeout

## Systems and Methods for Presenting Information on Mobile Devices

### III: Object Specific Implementations:

As before, events can be assigned to three different object classes. For each class, the following events would be available.

- A: All Object Classes and Page Timeout  
0=None (or Next Slide)  
1=Goto External Page (WAP or HTML)  
2=Goto Internal web Page  
3=Execute an XpressMo Application  
27=Send IM Alert
- B: The Basic Object  
4=Place Phone Call  
5=Send SMS Message  
6=Exit Application  
7=Exit Player  
8=Execute Scripting Method  
9=Pause/Resume Page Timeout  
28=Set Starting Page
- C: An Item in a List or Dropdown Object  
4=Place Phone Call  
5=Send SMS Message  
10= Dropdown: Add Icon to Icon Strip  
List: Display/Hide Icon toggle  
11= Dropdown: Remove Icon from Icon Strip  
28=Set Starting Page
- D: A Slide in a Slide Show  
12=Goto Specific Slide
- E: Web Page  
8=Execute Scripting Method

“Goto New View Options” will only be available for the following Exit Events:

- 1=Goto External Page (WAP/HTML)
- 2=Goto Internal web Page

“Share Options” will only be available for the following Exit Events:

- 4=Place Phone Call
- 5=Send SMS Message

Systems and Methods for Presenting Information on Mobile Devices

**IV: Mapping to Current Player Event Manager:**

The Table Below describes what choices will be available by exit event for the six dropdown option menus, and how that maps to the possible list of 28 events.

UI Exit Event	Alert Options	Change View Options	Personalize	Communicate	Share	Internal Exit Event
None	None Display w/Fire Display OnFocus Hide OffFocus Display w/Sync			Send IM Phone Call Phone Call Phone Call Send SMS Send SMS Send SMS	None Share All None Share All	17
						19
						22
						27
						12
						24
						26
						18
						23
						25
						13
						14
						28
Goto External Page		Replace Current View Launch New Window Replace Top View				1
						2
						4
Goto Internal Page	None Display w/Sync None None	Specific/Next Page Specific/Next Page Logical 1 <sup>st</sup> Page Specific w/Slide #				3,5
						15
						21
						9
Execute XMO Application-Widget	None Display None					8
						20
						16
Exit Application						10
Exit Player						11
Execute Script						6
Pause/Resume Page						7
Goto Slide						10
Send IM Alert						27

## Systems and Methods for Presenting Information on Mobile Devices

### **Appendix A: PDL Specification for Exit Events**

#### **Common Exit Event Definitions**

- 0=None (Next Slide)
- 1=Goto External Web Page replacing Current Frame
- 2=Goto External Web Page Launched in a New Window
- 3=Goto a specific Internal Web Page
- 4=Goto External Web Page replacing the Top Frame
- 5=Goto the next Internal Web Page
- 6=Execute JavaScript Method
- 7=Pause/Resume Page Timeout
- 8=Execute an XpressMo Application
- 9= Goto a specific Internal Web Page with setting starting slide
- 10=Exit Application
  - Add Icon (List Object)
  - Go to a Specific Slide (Slide show Object)
- 11=Exit Player
- 12=Place PhoneCall
- 13=Add Icon from ChoiceObject invisible List to ChoiceObject visible List
- 14=Remove Icon from ChoiceObject visible List to ChoiceObject invisible List
- 15=Goto a specific Internal Web Page with Alert. "Backend Synchronization"
- 16=Goto Widget Object
- 17=Generate Alert. "With a Fire Event"
- 18=Send SMS Message
- 19=Toggle Alert. "Display OnFocus, Hide OffFocus"
- 20= Execute an XMO Application with Alert. "With a Fire Event"
- 21=Goto Logical First Page
- 22= Generate Alert with Backend Synchronization
- 23=Send SMS Message with Share (Player Download)
- 24= Place PhoneCall with Share (Player Download)
- 25=Send SMS Message with Share All (Player Download)
  - List Only
- 26= Place PhoneCall with Share All (Player Download)
  - List Only
- 27=Send IM Alert
- 28=Set Starting Page

## Systems and Methods for Presenting Information on Mobile Devices

### **Primitive Object Definitions**

#### **ObjectIntegerAttribute[12] (Exit Event)**

- 0=None
- 1=Goto External Web Page replacing Current Frame
- 2=Goto External Web Page Launched in a New Window
- 3=Goto a specific Internal Web Page
- 4=Goto External Web Page replacing the Top Frame
- 5=Goto the next Internal Web Page
- 6=Execute JavaScript Method
- 7=Pause/Resume Page Timeout
- 8=Execute an XpressMo Application
- 9= Goto a specific Internal Web Page with setting starting slide
- 10=Exit Application
- 11=Exit Player
- 12=Place PhoneCall from linked Text Field
- 13=N/A
- 14=N/A
- 15=Goto a specific Internal Web Page with Alert. "Backend Synchronization"
- 16=Goto Widget Object
- 17=Generate Alert. "With a Fire Event"
- 18=Send SMS Message from Linked Text Field
- 19=Toggle Alert. "Display OnFocus, Hide OffFocus"
- 20= Execute an XMO Application with Alert. "With a Fire Event"
- 21=Goto Logical First Page
- 22= Generate Alert with Backend Synchronization
- 23=Send SMS Message with Share (Player Download)
- 24= Place PhoneCall from linked Text Field with Share (Player Download)
- 25= N/A
- 26=N/A
- 27=Send IM Alert from linked Text Field or Text Area
- 28=Set Starting Page

#### **ObjectString[2]**

- exitUrl (1,2,4)
- Goto PageName (3,9,15)
- Goto Widget(16)
- XMO Application Name (8,20)
- JavaScript method Name (6)
- Starting PageName (28)
- Linked Text Object name with text message for:
  - sending SMS Message (18,23)
  - sending IM Alert (27)

#### **ObjectString[3]**

- Selected SlideShow Object Name for Exit Event to a Specific Slide (9)
- Linked Text Field name with phone number for:
  - placing a phone call (12,24)
  - sending SMS Message (18,23)



## Systems and Methods for Presenting Information on Mobile Devices

sending IM Alert (27)  
Or Starting Page for ExitMode=8 and 20 (Execute an XpressMo Application)  
Or Alert POPUp Page for:  
    ExitMode=15 (Goto Specific Internal Page & Display Alert)  
    ExitMode=17 (Display Alert w/Fire)  
    ExitMode=19 (Toggle Alert. "Display OnFocus, Hide OffFocus")  
    ExitMode=22 (Generate Alert with Backend Synchronization)  
ObjectString[6] Alert POPUp Page for ExitMode=20 and 27  
    ExitEvent=27:                      Avatar URL

### List Item Definitions

ItemExitEvent[NumberOfItems]

- 0=None
- 1=Goto External Web Page replacing Current Frame
- 2=Goto External Web Page Launched in a New Window
- 3=Goto a specific Internal Web Page
- 4=Goto External Web Page replacing the Top Frame
- 5=Goto the next Internal Web Page
- 6=N/A
- 7=N/A
- 8=Execute an XpressMo Application
- 9= Goto a specific Internal Web Page with setting starting slide
- 10=Set by IconStrip. Display Icon
- 11=N/A
- 12=Place Phone Call
- 13=Add Icon
- 14=Remove Icon
- 15=Goto a specific Internal Web Page with Alert. "Backend Synchronization"
- 16=Goto Widget Object
- 17=Generate Alert. "With a Fire Event"
- 18=Send SMS Message
- 19=Toggle Alert. "Display OnFocus, Hide OffFocus"
- 20= Goto an XMO Application with Alert. "With a Fire Event"
- 21=Goto Logical First Page
- 22= Generate Alert with Backend Synchronization
- 23=Send SMS Message with Share (Player Download)
- 24= Place PhoneCall with Share (Player Download)
- 25=Send SMS Message with Share All (Player Download)
- 26= Place PhoneCall with Share All (Player Download)
- 27=Send IM Alert
- 28=Set Starting Page

ListItemParameter [NumberOfItems]

- |                      |                                     |
|----------------------|-------------------------------------|
| ExitEvent=0:         | Parameter                           |
| ExitEvent=13,14:     | Widget IconName                     |
| ExitEvent=18, 23,25: | Linked TextObj Name for SMS Message |
| ExitEvent=20:        | AlertPageName                       |

## Systems and Methods for Presenting Information on Mobile Devices

ExitEvent=27:	Avatar URL
ListItemPageName [NumberOfItems]	
ExitEvent=1,2,4:	External URL
ExitEvent=12,18, 23-27:	PhoneNumber
ExitEvent=16:	Widget URL
ExitEvent=8,20:	ApplicationName
ExitEvent=28:	Starting Page Name
ExitEvent=3,9,15:	GoToPageName
ItemSlideShowName[NumberOfItems]	
ExitEvent=8,20:	StartingPageName
ExitEvent=9:	SlideShowObjectName
ExitEvent=15,17,19,22,27	AlertPageName
ExitEvent=13,14	IconStripName

### **Slide Show Slide Definitions**

SlideIntegerAttribute[6] Exit Slide Mode

- 0=Next Slide
- 1=Goto External Web Page replacing Current Frame
- 2=Goto External Web Page Launched in a New Window
- 3=Goto a specific Internal Web Page
- 4=Goto External Web Page replacing the Top Frame
- 5=Goto the next Internal Web Page
- 6=N/A
- 7=N/A
- 8=Execute an XpressMo Application
- 9= Goto a specific Internal Web Page with setting starting slide
- 10=Go to a Specific Slide
- 11=N/A
- 12=N/A
- 13=N/A
- 14=N/A
- 15=Goto a specific Internal Web Page with Alert. "Backend Synchronization"
- 16=Goto Widget Object
- 17=Generate Alert. "With a Fire Event"
- 18=N/A
- 19=Toggle Alert. "Display OnFocus, Hide OffFocus"
- 20= Execute an XMO Application with Alert. "With a Fire Event"
- 21=Goto Logical First Page
- 22= Generate Alert with Backend Synchronization
- 23=N/A
- 26=N/A
- 24=N/A
- 26=N/A
- 27=Send IM Alert to Chat Members
- 28=N/A

Systems and Methods for Presenting Information on Mobile Devices

SlideString[3]

ExitEvent=1,2,4: External URL  
ExitEvent=16: Widget URL  
ExitEvent=8,20: ApplicationName  
ExitEvent=3,9,15: GoToPageName  
ExitEvent=27: Symbolic Name for Chat Group Members

SlideString[4]

Selected SlideShow Object Name for Exit Event to a Specific Slide  
Starting Page for ExitMode=8 and 20 (Execute an XpressMo Application)  
ExitEvent=27: Linked TextObj Name for IM Alert Message  
Alert POPUp Page for:  
ExitMode=15 (Goto Specific Internal Page & Display Alert)  
ExitMode=17 (Display Alert w/Fire)  
ExitMode=19 (Toggle Alert)  
ExitMode=20 (Execute an XMO Application)  
ExitMode=22 (Generate Alert with Backend Synchronization)

SlideString[5] ExitEvent=27: Alert Page Name

SlideString[0] Used for exit event=27 for Avatar URL

## Appendix E

# Direct Text Entry Specification

### **A: Background**

For languages that can be sufficiently supported using Unicode characters 33 through 126 (this excludes, among others, Chinese, Japanese and Korean) it is possible to support direct text entry as long as the device returns to Java the keycode that was touched. Currently all phones with numeric key pads will be supported. MIDP 2.0 devices with hard or soft keyboards currently will have to continue to use the indirect "Form-based" text entry model. Also, all CDC and J2SE devices will also be supported.

For MIDP 2.0 devices with hard or soft keyboards we probably will need to rely on the Response Director to inform the player that it must use the indirect "Form-based" text entry model. The first implementation will assume a numeric keypad so it will not work on MIDP 2.0 devices with keyboards.

### **B: Supported Objects**

The following objects will be supported in this first implementation:

- Text Fields
- Chat Messages
- Selection Lists

Text Areas will be deferred to the next release.

## Systems and Methods for Presenting Information on Mobile Devices

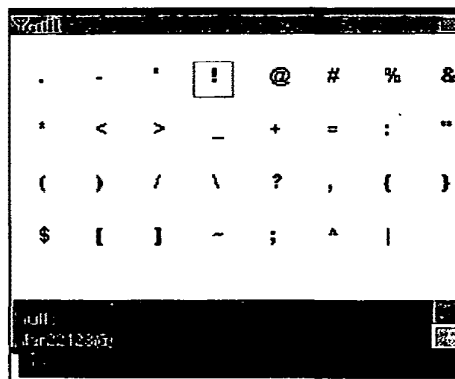
### C: Behaviors and UI

When a supported object is selected it will appear as a text field with a blinking cursor.



A text insertion point will be active and will blink 2.5 times per second. There are four possible text entry states which can be selected through soft key commands. There is a default state depending upon the type of text field..

- **Alpha**  
In this state touching a numeric key will cause an immediate echo of the first character assigned to that key. As long as the delay before touching that same key is less than 1.6 seconds then the insertion point will not move and the next character assigned to that key will be echoed. If the number of touches exceed the number of characters assigned to that key then the choices will be repeated. If either a different key is touched or the same key is touched after a 1.6 second delay, then the insertion point moves one character to the right, previous character is shown, and the new character, as appropriate, is echoed at the insertion point. Capitalization will occur if the character is an alpha character and it's the first character in the string, or the character is immediately preceded by a period and a space character.  
The current implementation is identical to that of the Sprint implementation for Samsung phones. (See appendix A)
- **ALPHA**  
Same behaviors as with "Alpha" except that all characters entered will be in uppercase.
- **Numbers**  
In this state touching a numeric key will cause an immediate echo of the numeric character assigned to that key and the insertion point will move one character to the right.
- **Symbols**  
When in this state a palette of symbols will be drawn on the top half of the screen, with a selection rectangle drawn either around the symbol in the upper left, or the last symbol selected. Navigation will

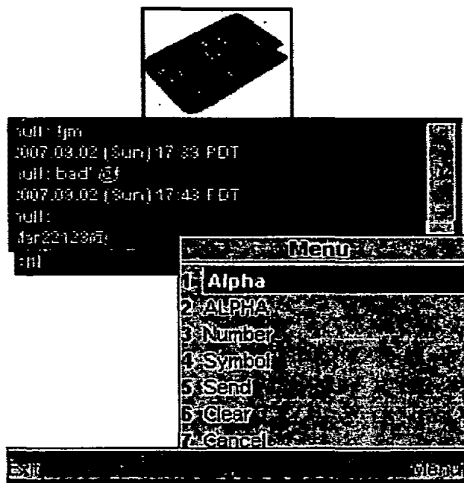


## Systems and Methods for Presenting Information on Mobile Devices

work in both dimensions, with recycling when hitting a symbol palette boundary. Once the desired symbol is selected either FIRE or the softkey "Send" command will place the symbol in the text field at the insertion point and the symbol palette will be erased.

The following Soft Key Commands will be available.

For Chat Messages and Selection Lists:



- "Alpha"
- "ALPHA"
- "Symbol"
- "Number"
- "Clear"

Clear will delete the character to the left of the insertion point but will only be available when not on the leftmost character

- "Send"

For a Chat Message the current Chat Message will be sent (and will appear with a time and userID stamp in the Chat Thread, and the chat message field will be cleared for the next message.

For a selection list the currently non-unique item entry will be selected and the list will close (See special behaviors below).

- "Cancel"  
In both cases the field will be cleared of all characters.
- For Text Fields "Alpha", "ALPHA", "Symbol", "Number" and "Clear" will be supported.

The following Navigation Keys will be supported.

- LEFT  
This will move the insertion point left one character. If already at the left most character then this will cause the focus to leave this object and the next appropriate object will be selected..
- RIGHT  
This will move the insertion point right one character. If already at the right most character then this will cause the focus to leave this object.
- FIRE  
For Chat Messages and Selection lists FIRE will behave in the same manner as the "Send" soft key command. For Text fields the Exit Event, if any, will be executed.
- BACK  
If available, it will delete the character to the left of the insertion point. However,

## Systems and Methods for Presenting Information on Mobile Devices

when not on the leftmost character, the “Back One View” operation will be executed.

- UP and DOWN  
They remain as object selection keys when on a text field. If on a Chat Message or List then their meaning has already been defined in the appropriate specification.

### Special Behaviors for Selection Lists.

When unselected, the Selection List Object will draw in a manner similar to that of a Text Field. When selected the following actions will occur:

- As soon as the first character is entered the Selector List will open.
- Characters can be deleted and/or inserted.
- If the entered character is valid (the current search string has at least one partial match to a search list item) then:
  - If there is only one unique match then the unique search list item will be selected and the Selection List will close.
  - If there is more than one list item partial match then the search list items drawn below the Text Field will be redrawn, starting now from the first valid partial match of the current search string.
- If the character entered defines a string that does not match any search list items the character will be inserted but the selection list items will not be redrawn.
- While the selection list is open the UP and DOWN operations will be supported.
- If FIRE is touched:
  - If there is a first valid partial match item selected item, or the item selected after navigating via the UP and DOWN operations, that item is selected and the list will close.
  - If there is no match then the list will close with the current non-matching entry will remain visible.

## Systems and Methods for Presenting Information on Mobile Devices

### Appendix A

```
KEY_NUM0:
    if (Count==0) code=" ";
    if (Count==1) code="0";
    if (Count==2) code="+";
KEY_NUM1:
    if (Count==0) code=".";
    if (Count==1) code=",";
    if (Count==2) code="@";
    if (Count==3) code="!";
    if (Count==4) code="?";
    if (Count==5) code="!";
    if (Count==6) code="*";
    if (Count==7) code="#";
    if (Count==8) code="/";
KEY_NUM2:
    if (Count==0) code="a";
    if (Count==1) code="b";
    if (Count==2) code="c";
    if (Count==3) code="2";
KEY_NUM3:
    if (Count==0) code="d";
    if (Count==1) code="e";
    if (Count==2) code="f";
    if (Count==3) code="3";
KEY_NUM4:
    if (Count==0) code="g";
    if (Count==1) code="h";
    if (Count==2) code="i";
    if (Count==3) code="4";
KEY_NUM5:
    if (Count==0) code="j";
    if (Count==1) code="k";
    if (Count==2) code="l";
    if (Count==3) code="5";
KEY_NUM6:
    if (Count==0) code="m";
    if (Count==1) code="n";
    if (Count==2) code="o";
    if (Count==3) code="6";
KEY_NUM7:
    if (Count==0) code="p";
    if (Count==1) code="q";
    if (Count==2) code="r";
    if (Count==3) code="s";
    if (Count==4) code="7";
KEY_NUM8:
    if (Count==0) code="t";
    if (Count==1) code="u";
    if (Count==2) code="v";
    if (Count==3) code="8";
KEY_NUM9:
    if (Count==0) code="w";
```



Systems and Methods for Presenting Information on Mobile Devices

```
if (Count==1) code="x";  
if (Count==2) code="y";  
if (Count==3) code="z";  
if (Count==4) code="9";
```

# Appendix K

## Master Page Specification

### **A: Discussion**

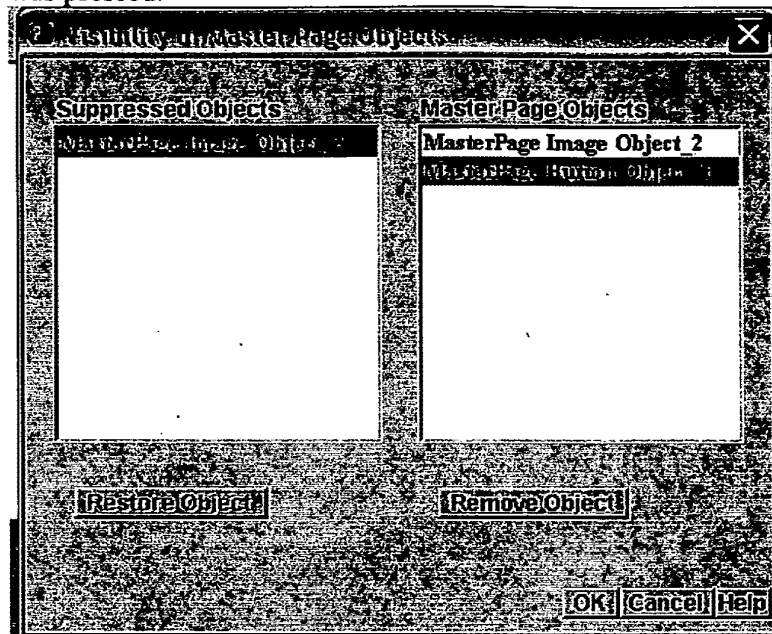
A Master Page will now become part of both the AKP Specification (replacing the Style Page in the Style Template” and the page default page in the Application. It will also become an extension of the PDL Header for the PDL Specification.

The characteristics of the Master Page will be:

1. **Defines Default Settings (at the page level) for all new pages.**  
This means that all the attributes that can be set in the “Edit Page” dialog box and/or the Page Resource Inspector will have their initial default settings defined by the Master Page.
2. **Repeating Objects.**  
Any objects defined on the Master Page will appear on every application page. The drawing order for an application page will be:
  - a. The first objects drawn will be from the Master Page, and drawn in the order defined in the master page.
  - b. All the application page specific objects will then be drawn in their drawing order.
  - c. Finally, if defined in the Master Page, the Icon Strip(s) will be drawn.
3. **Editing Operations:**
  - a. **Editing the Master Page:**  
To work with the Master Page select the “Master Page...” command under the file menu. The currently viewed page will now be the “Master Page. All operations that are available to an application page will be available to the Master Page. Any additions, deletions, or changes to objects on the Master Page will immediately take effect for all application pages in the application.
  - b. **Editing Master Page Objects on an application page.**  
Only Icon Strips can be selected and edited. All the other objects from the Master Page will be drawn but not be selectable.
  - c. **Applying a Current Page as the Master Page**  
Select “Set Master Page” under the pages menu. The current page’s objects will be appended to the Master Page in their current drawing order. Any page attributes that the current page has set to override the Master Page’s attributes will be transferred to the Master Page.
  - d. **Selectively Removing Master Page Objects form an Application Page.**  
When the “Master Page Objects” button is selected from the Resource Inspector the Visibility of Master Page Objects Dialog Box will appear.

## Systems and Methods for Presenting Information on Mobile Devices

All existing Master Page Objects will appear in the List on the right, and any Suppressed Objects will appear in the List on the left. The Suppressed Objects can be restored by selecting the one for restoration and pressing the "Restore Object" button. The diagram below shows what happens when a Master Page Object was selected and the "Remove Object" button was pressed.



- e. **Current Master Page Restrictions.**  
Soft Key Commands are disabled for Master Pages. In a later release they will be enabled and will be inherited for application pages in a similar manner as Master Page Objects. The Exit and Back commands would typically be assigned to the Master Page. As with Master Page Objects, it will be possible to selectively remove an inherited Master Page Soft Key Command (including the Back Command) from any given application page.

- 4. **Effect on Icon Strips.**  
Icon Strips will now be part of the Master Page. The only difference between Icon Strips and other repeating objects are that the Uber-application's Icon Strips will be inherited as repeating objects for all sub-applications. Any icon Strips defined in a sub-application will inherit all the icons from the uber-application, with its own icons appended. These sub-application specific icons will only be available while the sub-application is the current application for the player. The Icon Strip attributes if the sub-application will, while the sub-application is alive and are different than those of the uber-application, will take precedence.

## Systems and Methods for Presenting Information on Mobile Devices

### 5. Other Master Page Functions.

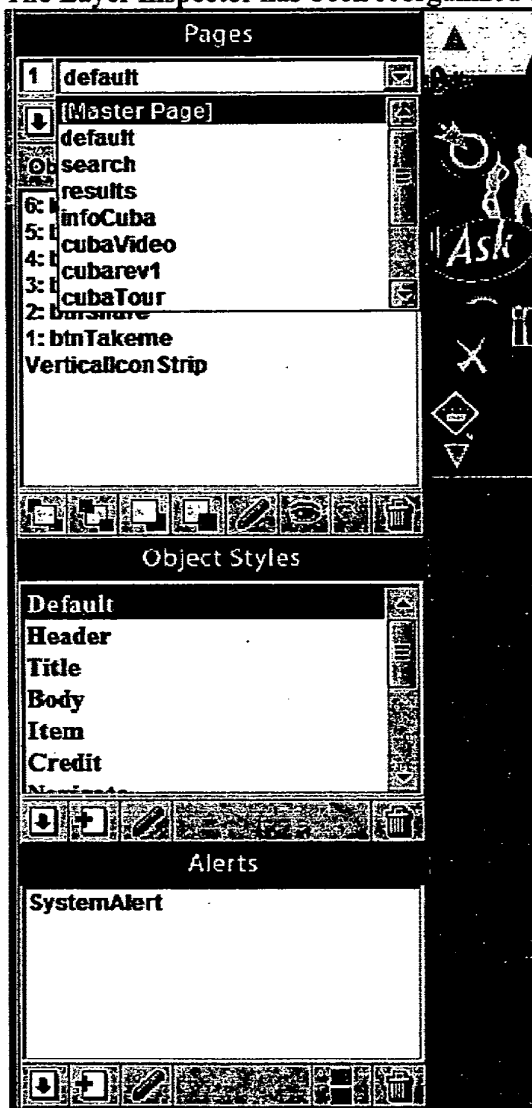
Eventually there may be many, but initially there will only be one.

#### a. Selection Rectangle Color.

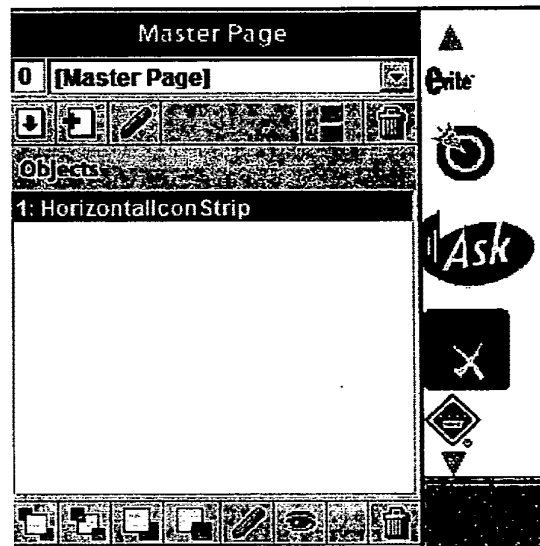
This will be a Master Page attribute. The current application's setting will take precedence.

## B: UI Implementation

The Layer Inspector has been reorganized to accommodate Master Pages



The Master Page can be selected by Clicking on the [Master Page] entry in the Page List or by Selecting “Master Page...” command under the file menu. The result will be the graphic below.



To return to an application page or an Alert Page either fire on the “Master Page” or “Alerts” Icons or by selecting a page or alert from their respective dropdown menus.

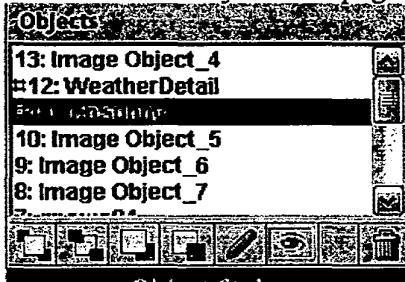
All icons below for changing the drawing layer, importing, creating, deleting and editing work as expected. An alternative method for editing an Icon Style is simply fire on the “Object Styles” Icon and the Edit Dialog Box for the currently selected style will appear.

The up and down arrow icons permit the navigation through application pages and alert pages respectively.

## Systems and Methods for Presenting Information on Mobile Devices

Selectively hiding an Application Page Object has been modified as follows:

1. Hiding Objects  
Select the object and then fire on the “Eye” Icon. A non-alpha character similar to a lozenge character will appear to the left of the object name. When a hidden object is selected the “Eye” Icon appears in a grayed out state. (See diagram below)
2. Restoring Hidden Objects  
Fire on the grayed out icon as shown below.
3. Effect of Hiding Objects  
A hidden object will only have effect during the current editing session. To hide a Master Object for a page during execution refer to “3.d” above.



## Systems and Methods for Presenting Information on Mobile Devices

### **B: PDL and AKP Specification**

The following page IntegerAttribute[] definitions below will now only apply to a master page.

- 13= Horizontal IconStrip Mode
  - 0=Convert to SlideShow
  - 1=Top Static
  - 2=Bottom Static
  - 3=Top Scrolls
  - 4=Bottom Scrolls
- 14 = Vertical IconStrip Mode
  - 0= Convert to SlideShow
  - 1=Left Static
  - 2=Right Static
  - 3=Left Scrolls
  - 4=Right Scrolls
- 15= Frame delay time (in milliseconds)
- 16= Number of Animation Frames
- 17=Number of Repeating Objects
- 18= Number of Icons on Horizontal IconStrip that are initially being set to visible (all or up to 20)
  - 0=all;
- 19= Number of Icons on Vertical IconStrip that are initially being set to visible (all or up to 20)
  - 0=all;

The following Vector will be added to the AKP and PDL for each Application Page.

```
Vector <Boolean> isObjectVisible= new Vector (IntegerAttribute[17])  
// If a Master Page generated object is deleted on a application page, its  
corresponding index in the isObjectVisible Vector will be set to false.
```

## Appendix C

# Complex List Specification

### **A: Complex List Types.**

Lists can be of the following types:

- **Dropdown Choice**
  - Default
  - Active Channel Personalization
  - Inactive Channel Personalization
  - Search From List
  - Widget Selection List from User's Library
  - Phone List
  - SMS List
  - Widget Selection list from Runtime Widget Library
  - Phone List from PIM Data
  - SMS List from PIM Data
  - Search Response list
  - RSS List
- **Selection List**
  - Default
  - Multi-Select
  - Category List
  - Channel Personalization
  - Widget Selection List From User's Library
  - Phone List
  - SMS List
  - Widget Selection list from Runtime Widget Library
  - Phone List from PIM Data
  - SMS List from PIM Data
  - Search Response list
  - RSS List

Web services are supplied to support the following:

- **Personalization of the Channel Launch Strip**


Dropdown lists have list types for both adding and removing channels (icons and their related intelligence) from the Channel Launch Strip. Selection lists have a Multi-Select implementation where by the Channel Launch Strip will be populated based on only those channels that are selected.
- **Widget Selection from User's Library**

The list object will automatically populate with the icons and the widget names that exist in the author's private widget library. No new items can be added or existing widgets deleted, but other attributes for each widget list item can be



## Systems and Methods for Presenting Information on Mobile Devices

added.

- **Widget Selection from Runtime Widget Library**  
Both Lists have Widget Selection Lists that are automatically populated with an entry for each Widget that is available to that player. Each Widget, at authoring time, can be assigned a unique icon.   
If an item in a Widget Selection list has a “Goto Widget Object” Exit Event then a Widget Icon Will be drawn. If the Widget Page has already had a Page Icon defined then it will be drawn. Otherwise the default Widget Icon will be drawn.  
At runtime when firing on an Item in a Widget Selection List that Widget will automatically be loaded and executed on the device.
  
- **Xpressmo Server Page Selection**  
For both Lists that are set to their default type, if there are any of the following exit events associated with a List or Choice Item:
  - Goto a specific Internal Web Page
  - Goto the next Internal Web Page
  - Goto a specific Internal Web Page with setting starting slide
  - Goto a specific Internal Web Page with Alert
  - Goto Logical First Pageand that internal WebPage has a Page Icon defined then it will be drawn. Otherwise, no icon will be drawn.
  
- **Automatic Population and Execution of Phone and SMS Lists:**  
Both Lists can be set to be automatically populated with the PIM data defined in the phone’s contact list.
  - If the device supports and permits access to the PIM content list data, then when firing on a Phone List Item the following will occur. If the phone supports simultaneous use of the phone with an application then the Xpressmo application will continue, otherwise the user will be prompted whether they wish to exit the Xpressmo application and then place the phone call
  - If the device supports and permits access to the PIM content list data, then when firing on an SMS List Item the SMS message will immediately be sent to that contact.

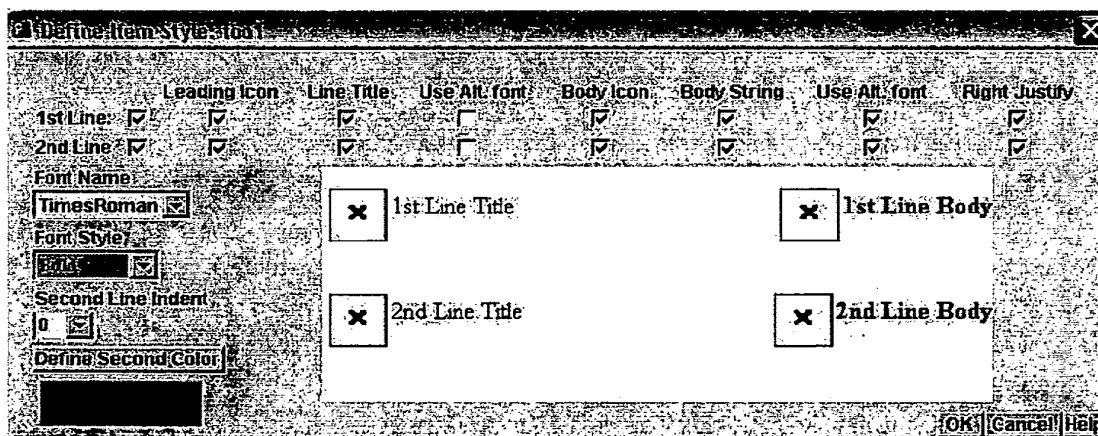
## Systems and Methods for Presenting Information on Mobile Devices

### **B List Layout Styles.**

All object styles now have an associated List Layout Style. The author can create as many List Layout Styles as desired by touching the "Create List Button". If so, then that new List Layout Style will become the associated List layout Style for that Object Style. The author can always change which is the associated List layout Style for a given Object Style by changing the selected List object Style.



If the author wants to change the definition for a given List Layout Style then firing on the "List Layout Style" in the Object style's Settings list will display the dialog box below.



The List Layout Style has the following controls:

a: First Line:

- Title Icon (optional)
- Title String (mandatory)
- Body Icon (Optional)
- Body Text (Optional)

The body icon and text can be left or right justified.

The Title Text and body text can have a selection of either 2 different fonts, including color. The font size will always be derived by the Object Style Font Size for both fonts.

## Systems and Methods for Presenting Information on Mobile Devices

### b: Second line (Optional)

- An indent can be defined relative to the first line.
- All other settings available for the first line are available and optional for the second line.

### C: Working with Selection Lists.

Selection Lists now have “Settings” attributes that are defined at three distinct levels:

- Object Level
- List Level
- List Item Level

You can switch between attributes levels by firing on the desired tab.

#### Object Attributes

The screenshot shows the 'Object Attributes' settings panel for a Selection List object. The panel is organized into several sections:

- List Visible Items:** A numeric input field set to '6'.
- Draw Frame:** A checkbox that is currently unchecked.
- Object Style:** A dropdown menu set to 'Default'.
- Font Name:** A dropdown menu set to 'TimesRoman'.
- Font Effects:** A dropdown menu set to 'Bold'.
- Font Size:** A numeric input field set to '16'.
- Object Name:** A text input field containing 'channel list'.
- Frame Style:** A dropdown menu set to 'None'.
- Frame Width:** A numeric input field set to '1'.
- Dimensions and Properties:** A row of input fields for 'Left' (10), 'Top' (10), 'Width' (200), and 'Height' (131). To the right are checkboxes for 'Lock', 'Hide', and 'Restore Default', all of which are currently unchecked.

Object Attributes deal with the basic Selection list object, including the Object's font Style, number of List Items that are visible (which either directly or indirectly define the height of the List object).

## Systems and Methods for Presenting Information on Mobile Devices

### List Attributes

The screenshot shows the 'List Attributes' configuration screen. It is organized into three columns: 'List Items', 'List Styles', and 'List Item Layout'.  
- **List Items:** Includes a text input field with '49ers win big!', a dropdown menu set to 'default', and buttons for 'Add Item', 'Remove Item', and 'Remove All'.  
- **List Styles:** Features a 'List Selector Type' dropdown menu with options like 'Multiple Select', 'Category List', 'Personalize Channels', 'Select Widgets from Use', 'Phone List', 'SMS List', 'Select Widgets from Run', and 'Phone List from PIM'. Below this are fields for 'Alternate Font Name' (set to 'Arial') and 'Alternate Font Style' (set to 'Bold').  
- **List Item Layout:** Contains checkboxes for '1st Line', '2nd Line', 'Leading Icon', 'Line Title', 'Use Alt. Font', 'Body Icon', 'Body String', and 'Right Justify'. It also has a 'Second Line Indent' field set to '0' and a 'Define Second Color' button.  
- **Unselected Item Icon:** A dropdown menu set to 'None'.

List items can be added by touching the “Add Item” button. The initial Item name, and its corresponding “Title String” will be set to a default value of “ItemNumber-#” where # will be the number in the list that the new Item was placed in, and immediately appear in the List Items Dropdown menu.

The settings for Define List Item Layout are initially inherited from the List Layout Style that is assigned to the current Object Style when the Selection List Object was created. Any other List Style can now be selected if desired by selecting a list Style from the List Style Dropdown menu.

Any changes made to the “List Item Layout” will override the inherited settings. Any changes in terms of the availability of Icons and/or Strings at the list item level will be reflected in the “Item Attributes” screen when a new Item is created. At the List Attributes level the following controls are available

1. For Selection Lists of types Default, Search Response, RSS Lists, Multiple Select, Category, Select Channels and Select Widgets, the following elements can be defined..
  - a. First Line:
    - i. Title Icon (optional)
    - ii. Title String (mandatory)
    - iii. Body Icon (Optional)
    - iv. Body Text (Optional)

## Systems and Methods for Presenting Information on Mobile Devices

- v. The body icon and text can be left or right justified.
  - vi. The Title Text and body text can have a selection of either 2 different fonts and/or colors.
  - b. Second line (Optional)
    - i. An indent can be defined relative to the first line.
    - ii. All other settings available for the first line are available and optional for the second line.
2. For Multiple Select Lists a palette of both selection and unselected vectors and icons are available.
  3. For Category Lists a palette of vectors and icons are available.
  4. For Search Response Lists:

The author can create a fully defined Search Response List form the Text Dropdown Menu.

Four objects will be created:

- Search Response List
- Text Field
- Text Area
- Slide Show

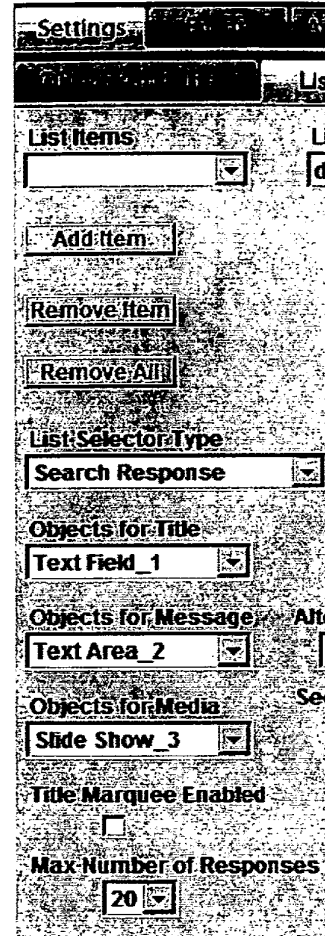
They will be logically linked together. The author will then be presented with three dropdown lists for possible defining the logical relationship with any Search Response List Item and other related objects on the page that could display metadata that is associated with that response. The maximum number of possible Search Response List Items can be set.



## Systems and Methods for Presenting Information on Mobile Devices

The Three Logically Linked Objects are:

- Title Object  
This could be either a text button or a text field, the difference being whether the width is fixed with possible scrolling or the width adapts itself to the length of the title string. If a text field, then the “title Marquee Enabled” check box becomes enabled. If selected, then, if the string for the title is too long for the width of the text field, a marquee animation will begin when this search result is selected.
- Message Object  
This could be either a paragraph or a text area, the difference being whether the number of visible lines is fixed with possible scrolling or the height adapts itself to the number of lines after the text string has been formatted to the width of the text area
- Media Object  
This must be a Slide Show. Any image or video metadata will be attached to slide that is in the same relative position as the list item.



During runtime, the object names for the Title, Message and Media object will be used to dynamically update their content as each item in the Response List is visited. As with all list items, any exit event can be assigned.

When the Search Response is received by the Search Response List, it will send the following listener event to the Web Service:

```
getParentObjectSelected("PageName","Search Response List Object  
Name","Maximum Number of Responses", true,"Object X  
Origin","Object Y Origin");
```

```
where ObjectIntegerAttribute[19]=Maximum Number of Responses
```

Note that the object names are assigned to the following object string attributes:

- ObjectString[1]=Media Object Name
- ObjectString[2]=Title Object Name
- ObjectString[3]=Message Object Name

## Systems and Methods for Presenting Information on Mobile Devices

When populating the Search Response List, either through backend code or by utilizing a web service, the following list item attributes will override the following attributes of the logically related objects:

- ItemSlideShowName=Media Object Slide URL (SlideString[0])
- ListItemSlideNo= SlideShow Slide Index Number
- ListItemPageName =Title Visible String (ObjectString[0])
- ListItemParameter=Message Visible String (ObjectString[0])

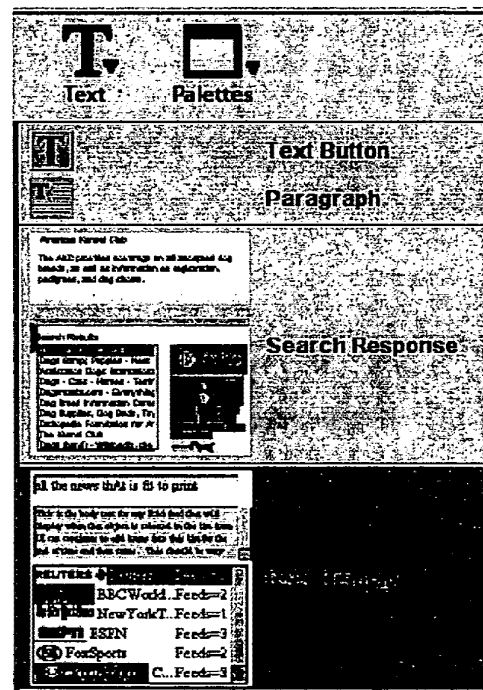
ObjectBooleanAttribute[7] contains the state for the Title Marquee Animation.

### 5: For RSS Display Lists:

The author can create an RSS Display List in a manner very similar to that of a Search Response list.

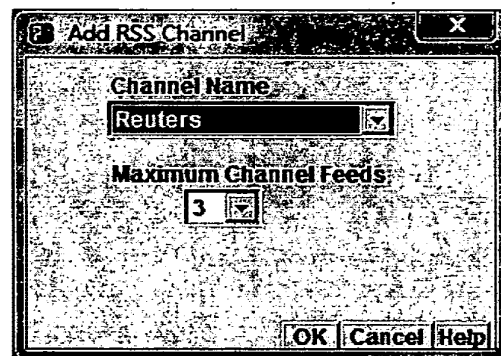
In the Template available to the right, The Title and Message (abstract) Objects are defined and logically connected to the RSS Display Lists.

This template also places, if available, the Channel's Image Icon as the title icon. The Icon is automatically sized so that its height will be equal to the Item's height, and the width is then calculated to maintain the image's aspect ratio.



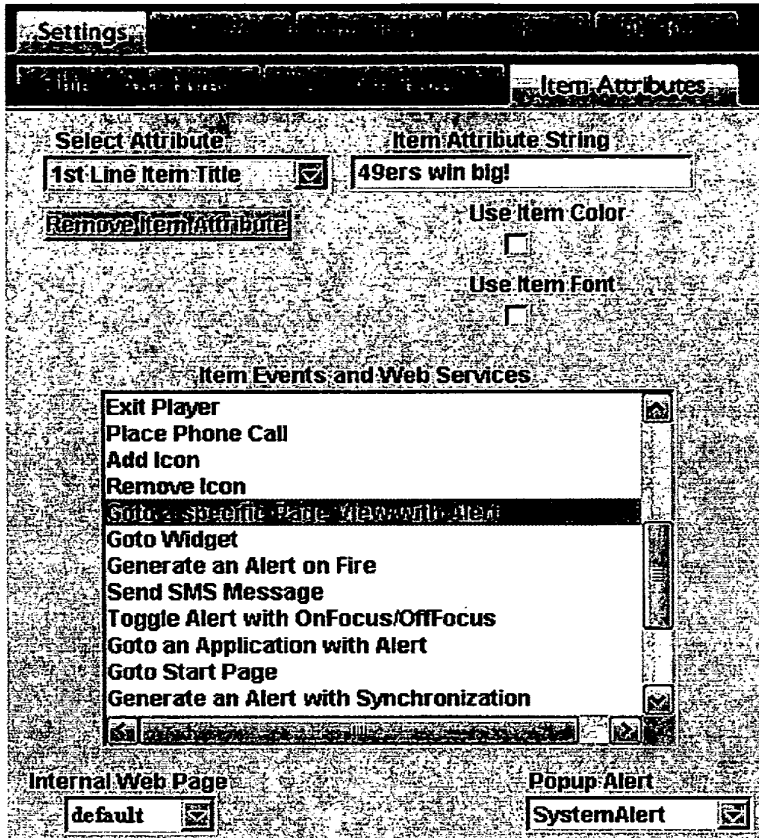
The author adds channels by touching the "Add Item" button under the "List Attributes" tab of the Resource Inspector. In addition to selecting the RSS Channel, the author can set the maximum number of feeds that will be extracted and displayed on the mobile device.

The author can remove any of the channels or remove all of the channels if desired.

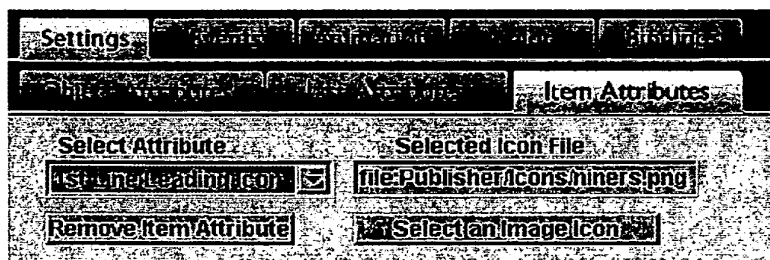


Systems and Methods for Presenting Information on Mobile Devices

**Item Attributes**



The graphic to the left shows the choices if a String item element is selected. In that case there are choices on whether to apply the Alternate Item Color and/or the Alternate Item font. Any choices will override the defaults that were inherited from the List Layout. The “Item Attribute String” can be edited and will immediately appear in the Workspace’s Selected List Object. If the “Selected Attribute” is the 1<sup>st</sup> Line Item Title then this will become the List Item name as displayed in the “List Item” Dropdown menu under the List Attribute Tab.



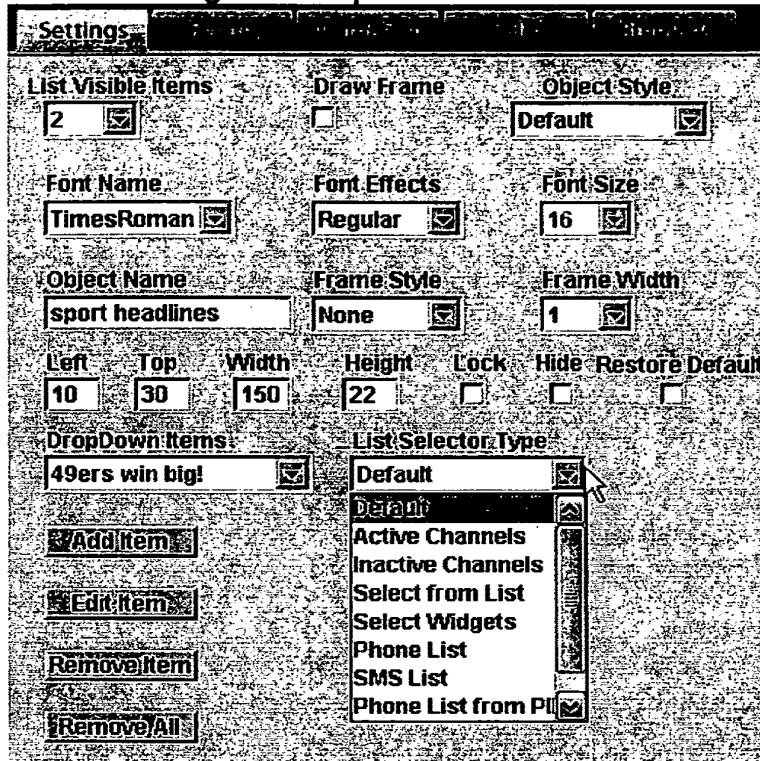
The graphic to the left shows the choices if an Icon element is selected. If “Select an Image Icon” is touched then the “Icon Management” dialog box is displayed.

Note that the 1<sup>st</sup> Line Title attribute can never be removed.



## Systems and Methods for Presenting Information on Mobile Devices

### D: Working with Dropdown Choice Lists.



Touching the "Add Item" button will display a dialog box where the item's attributes can be fully defined. If these attributes are to be changed, then select that item in the "DropDown Items" Dropdown menu, and touch the "Edit Item" button.

The differences in List Selector Type from a Selection list are as follows:

- Multiple Select and Category Selector Types are not available.
- Personalization of the Channel Launch Strip.  
The "Select Channels" selector type is replaced by "Active Channels" and "Inactive Channels".
- Search From List  
The "Search From List" Object can be populated in the same ways as other lists. When unselected, the "Search From List" Object will draw in a manner similar to that of a Text Field. When selected the following actions will occur:  
For the indirect Text Entry Mode:
  - On pressing Fire, or a text or numeric key the page view will switch as in a manner similar to that of a Text Field with the Text Field on the top of the screen representing the last successful search, if any. The search list items, starting from the first valid match (in alphabetical order) will be drawn below the Text Field.

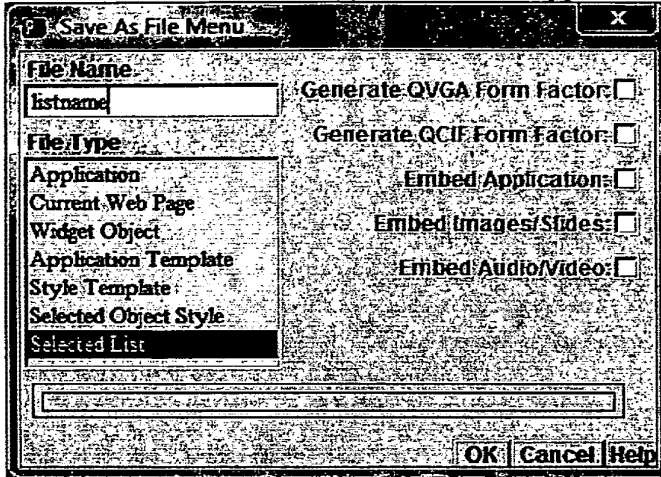
## Systems and Methods for Presenting Information on Mobile Devices

- Characters can be deleted and/or entered.
  - As soon as the first character is entered the Selector List will open.
  - If the entered character is valid (the current search string has at least one partial match to a search list item) then:
    - If there is only one unique match then:
      - The page view will return to the original page view
      - The unique search list item will be selected.
    - If there is more than one list item partial match then the search list items drawn below the Text Field will be redrawn, starting now from the first valid partial match of the current search string..
  - If the character entered defines a string that does not match any search list items, then that illegal character will be removed.
  - There will be two soft key commands.
    - Cancel:  
The page view will return to the original page view with no change.
    - Commit  
The following behaviors will be supported:
      - If the string entered is completely valid and matches, at least in part, a minimum of two items in the list, then:
        - The first item that has a successful partial match will be selected.
        - The dropdown will be open for navigation to other possible search list items.
        - The search list items in the dropdown will start from the currently selected item.
  - Search is not case sensitive.
- For the direct Text Entry Mode see the “Direct Text Entry Specification”

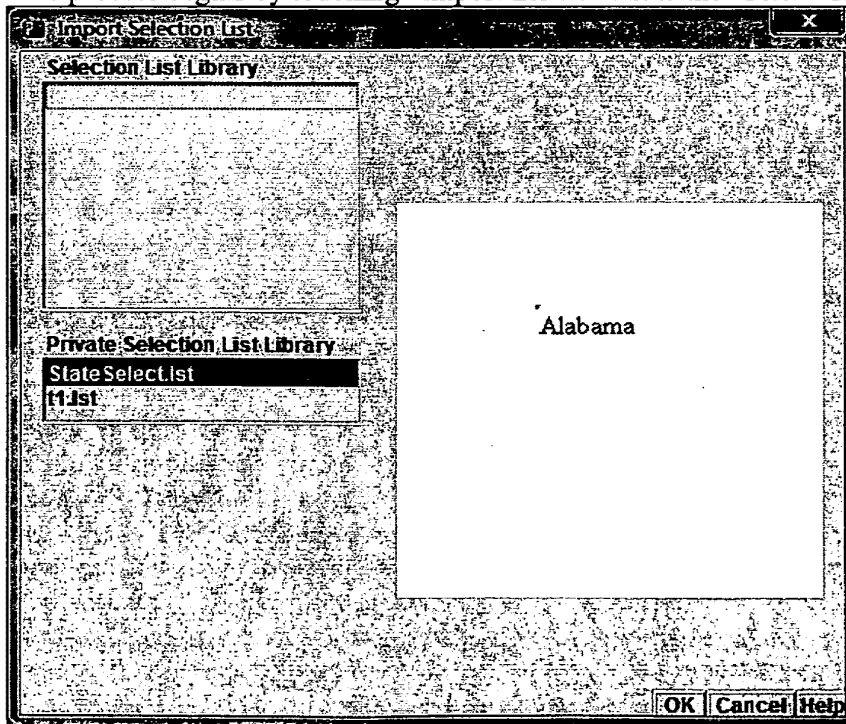
Systems and Methods for Presenting Information on Mobile Devices

**E: Working with Selection List Libraries.**

If a dropdown list or selection list is selected and a “Save As” operation is requested, then there will be a new Choice under File Types.



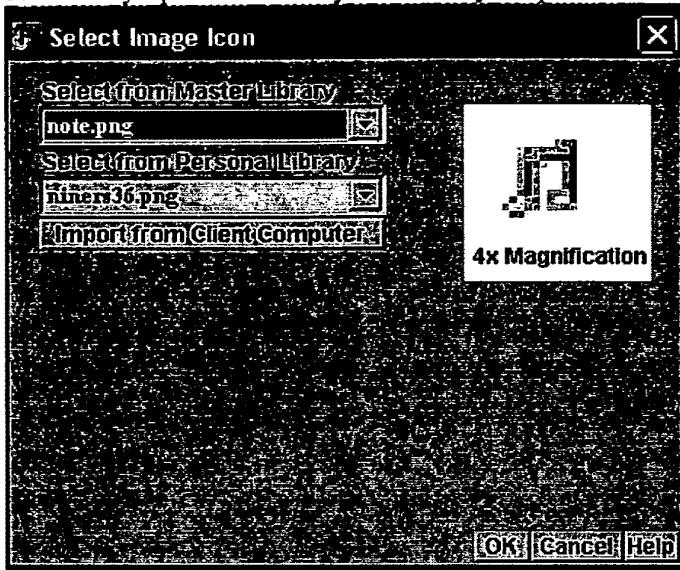
These Saved Lists, including all their icons and dynamic binding, can now be imported into an application, either from a public Xpressmo library or from the User's List Library. This process begins by touching “Import List...” under the “File” Menu.



## Systems and Methods for Presenting Information on Mobile Devices

### **F: Icon management.**

Icons are similar to images in that they have both private and public libraries and can be seamlessly uploaded from your PC to your private library.



### **G: Suppress Frame**

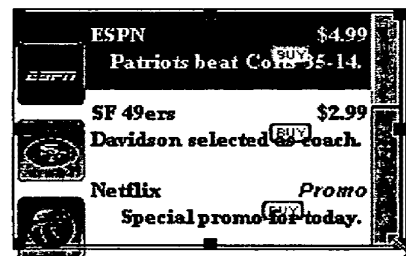
Only applies to List Objects.

If this checkbox is Selected and the number of items in the list are less than or equal to the number that are visible, then the Frame drawing will be suppressed.

### **H: Non-scaling Lists**

By choosing the "Define Form Objects by Pixels" under the "Project Menu the List will now be defined similar to that of an Image. The location and size are specifically defined, and they can be manipulated by dragging an attachment point.

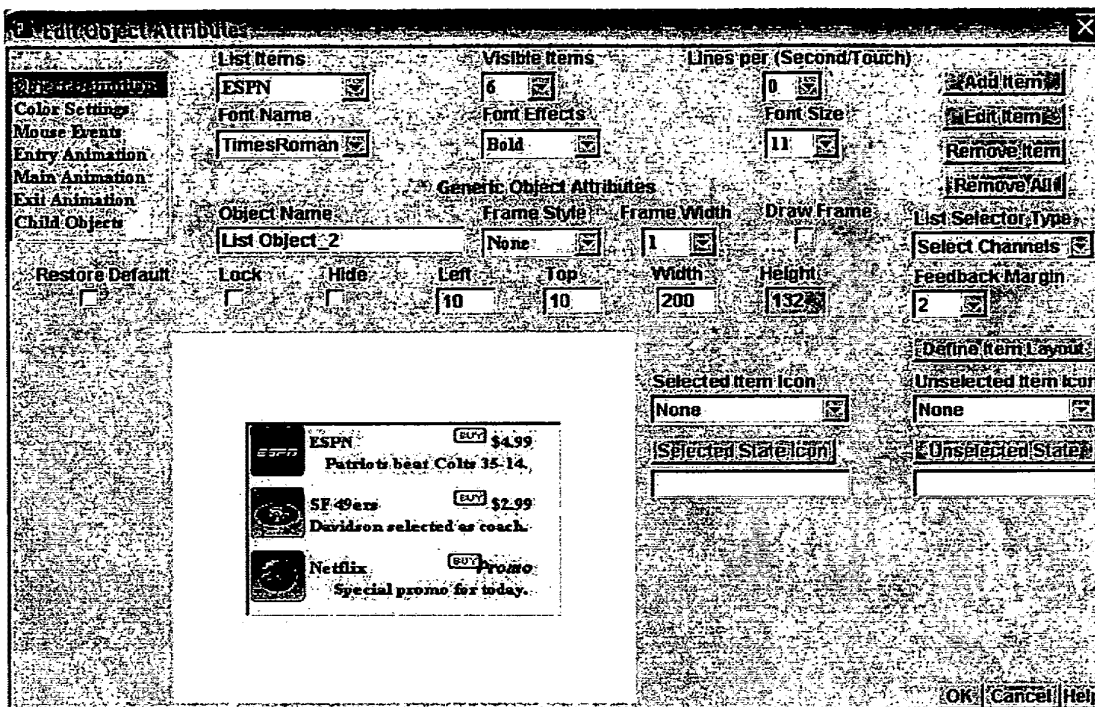
The font size, when in "Define Form Objects by Pixels" mode, will be calculated based on the number of items that are visible and the height of the List object.



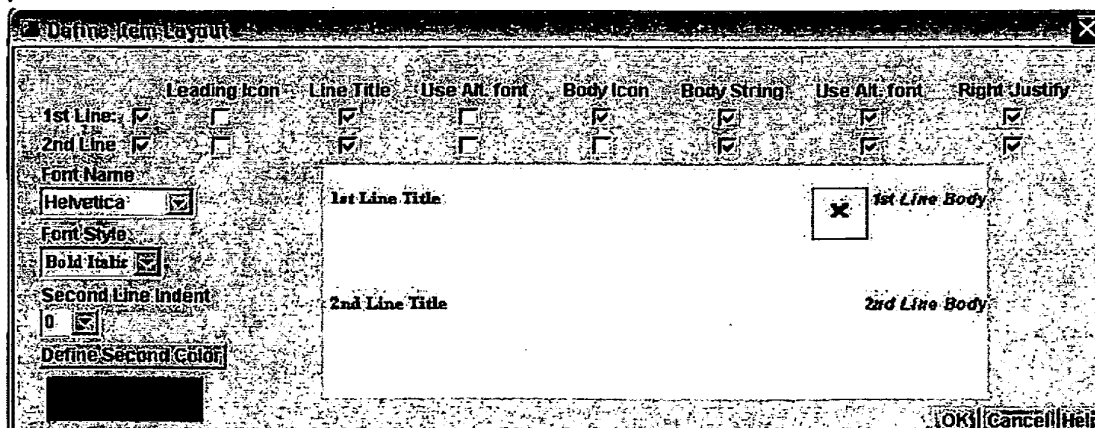
Systems and Methods for Presenting Information on Mobile Devices

**I: Working with the Edit Dialog Box.**

You can work with a Selection List utilizing the Edit Dialog Box which is available from the Layer Inspector or by right clicking over the Selection list Object.

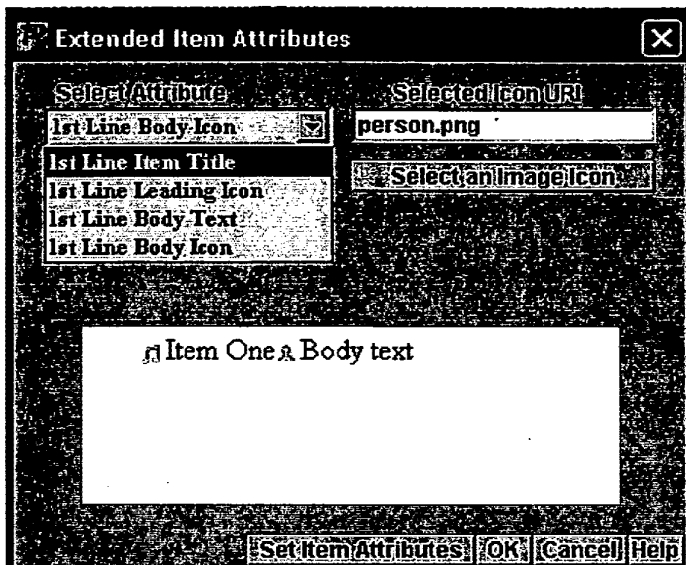


If the “Define Item Layout” button is touched then, in a manner identical to the “List Attributes” screen in the resource inspector, the Item Layout settings, which were inherited from the Object Style, can be changed.



## Systems and Methods for Presenting Information on Mobile Devices

When touching either the “Add Item” or edit Item” buttons, the dialog box will have an “Extended Item Attributes” button. This will display the “Extended Item Attributes” dialog box which will work in a manner similar to that of the “Item Attributes” screen in the resource inspector.



Systems and Methods for Presenting Information on Mobile Devices

**Appendix A: PDL Specification for Complex Lists**

ObjectBooleanAttribute[10]

If true (default) draw the frame.

ObjectIntegerAttribute[15]

ChoiceObject

0=default

1=: Lists visible icons/channels from generic icon strip

2=: Lists invisible icons/channels from generic icon strip

3=: Lists all icons/channels from generic icon strip

4=: WidgetSelection

5=Phone Selection List

6=SMS Selection List

7=Phone Selection List From Pim

8=SMS Selection List From Pim

ListObject

0=default

1=: Multiple Select

2=: Category

3=: Lists all icons/channels from generic icon strip

4=: WidgetSelection

5=Phone Selection List

6=SMS Selection List

7=Phone Selection List From Pim

8=SMS Selection List From Pim

Complex List Generic Attributes

Color ItemColor

Font ItemFont

IconType[0]: For defining the selected icon where:

0=none.

1=custom image icon

2=Checkmark

3=X

4=Checkmarked Box

5=X in Box

6=Filled Circle

IconType[1]: For defining the unselected icon where:

0=none.

1=custom image icon

2=empty box

3=empty circle

If IconType[0]==1:

isIconDefined[0]=true

## Systems and Methods for Presenting Information on Mobile Devices

IconURL[0]=the Selected Icon File Name  
If IconType[1]=1:  
  isIconDefined1]=true  
  IconURL[1]=the Unselected Icon File Name

boolean hasSecondLine: Enables and draws second line attributes

### Complex List Layout Attributes

short secondLineIndent: If 2<sup>nd</sup> line # of pixel indent  
short DefaultItemNumber: List item # for setting default values  
boolean[4] isItemStringDefined:  
boolean[4] isItemImageDefined:

Each Complex List item may have up to 2 lines, and each line may have up to 2 elements. If set to true the actual items will have these attributes available for definition. 1<sup>st</sup> Line Body is always available. The elements are:

1<sup>st</sup> Line Title

  Optional Image Icon

  Mandatory String

1<sup>st</sup> Line Body

  Optional Image Icon

  Optional String

2<sup>nd</sup> Line Title

  Optional Image Icon

  Optional String

2<sup>nd</sup> Line Body

  Optional Image Icon

  Optional String

boolean[2] RightJustify:

Each line may have the body element right justified if its related RightJustify[line number]=true.

### Basic Choice Object Item Attributes

ItemString[0]=ItemLabel

ListItemParameter

ListItemExitEvent

0=None

1=Goto External Web Page replacing Current Frame

2=Goto External Web Page Launched in a New Window

3=Goto a specific Internal Web Page

5=Goto the next Internal Web Page

4=Goto External Web Page replacing the Top Frame

6=Execute JavaScript Method

7=Pause/Resume Page Timeout

8=Execute an Akira Application

9= Goto a specific Internal Web Page with setting starting slide

10=Set by IconStrip. Display Icon

11=Goto Widget Object



## Systems and Methods for Presenting Information on Mobile Devices

ListItemPageName

ListItemSlideShowName

For ExitEvent=9 it is the selected SlideShow Object Name

For ExitEvent=8 it is the starting Pagename

Fro ExitEvent=3 it is the application name

ListItemSlideNo

### Complex List Item Attributes

For each item in a complex list, there are definitions for all of the possible elements that were enabled by the respective Layout Booleans.

#### **For Strings:**

boolean[4] isItemStringDefined:

If set to true the actual item will have these attributes defined and displayed. 1<sup>st</sup> Line Body is always defined.

The String elements are:

String[4] ItemString: The string for the element:

boolean[4] UseItemColor: True means use ItemColor

boolean[4] UseItemFont: True means use ItemFont

where:

0=1<sup>st</sup> Line Title

1=1<sup>st</sup> Line Body

2=2nd<sup>t</sup> Line Title

3=2nd Line Body

#### **For Image Icons:**

boolean[4] isItemImageDefined:

If set to true the actual item will have these attributes defined and displayed.

The Icon Element is defined by:

String[4] ImageURL: Image Icon File Name

short[4] ItemImageWidth: Image Icon width

where:

0=1<sup>st</sup> Line Title

1=1<sup>st</sup> Line Body

2=2nd<sup>t</sup> Line Title

3=2nd Line Body

### Runtime Attributes for dynamic content.

String IconPathName: transmitted to the List Object from the populateComplexChoiceObject API.

Boolean Append: transmitted to DynamicDataConnection from the populateComplexChoiceObject API to either append or replace the existing items in the List object.

For each List Item:

boolean[4] DownloadIcon

If true then the IconPathName: will be used for an http request. If false, then the ImageURL[index] must be prepended by either

**Systems and Methods for Presenting Information on Mobile Devices**

“Icons/” “Images/” or “Slides/”, and the image that is to be used must already have been defined in the PDL.

where:

0=1<sup>st</sup> Line Title

1=1<sup>st</sup> Line Body

2=2<sup>nd</sup> Line Title

3=2<sup>nd</sup> Line Body

## Systems and Methods for Presenting Information on Mobile Devices

### **Appendix B: Backend API support for Choice and List Objects**

```
/**
 * Appends or replaces items to an Akira Complex List (Choice) menu.
 *
 * @param pageSource
 *     is the location indicator for the page.
 * @param pageName
 *     is the page name that the player will be directed to visit.
 * @param objectName
 *     is the name of this object.
 * @param IconPathName
 *     is the pathanme for any icons that will be downloaded dynamically.
 *
 * @param ObjectNumber
 *     is the object number for this object. This value is ignored if
 *     operation is APPEND.
 * @param numberOfItems
 *     is the number of items to be appended.
 * @param ListItems
 *     is fully populated vector of listItem objects. For convenience this object can be created
 *     through the ListItemRun constructor.
 */
public void populateComplexChoiceObject(short PageSource, boolean append, String
pageName, String objectName, String IconPathName, short reserved1, short numberOfItems,
Vector<ListItemRun> ListItems) {
```

## Systems and Methods for Presenting Information on Mobile Devices

```
/**
 * Appends or replaces items to a Choice menu.
 *
 * @param pageSource
 *     is the location indicator for the page.
 * @param isDropDown
 *     true=ObjectType=38, false=ObjectType=39.
 * @param append
 *     if true means the items will be appended to the current list.
 * @param pageName
 *     is the page name that the player will be directed to visit.
 * @param objectName
 *     is the name of this object.
 * @param ObjectNumber
 *     is the object number for this object. This value is ignored if
 *     operation is APPEND.
 * @param numberOfItems
 *     is the number of items to be appended.
 * @param ItemExitEvent
 *     is an array of byte values which can have the values of
 *     OBJECT_VISIT_A_NEW_WEB_PAGE,
OBJECT_VISIT_A_WEB_PAGE_IN_A_WINDOW,
 *     OBJECT_EXIT_TO_A_SPECIFIC_PAGE_NUMBER,
 *     OBJECT_REPLACE_TOP_FRAME_WITH_NEW_WEB_PAGE,
 *     OBJECT_EXIT_TO_THE_NEXT_WEB_PAGE,
 *     OBJECT_EXIT_TO_AN_AKIRA_APPLICATION,
 *
OBJECT_EXIT_TO_A_SPECIFIC_PAGE_NUMBER_WITH_SETTING_INITIAL_SLIDE;
 * @param ItemStringAttributes
 *     is an array of String values which describe all of the String
 *     attribute values for each item in a DROPDOWN_OBJECT. The first
 *     index is the item Number.
 * @param ItemParameterAttributes
 *     is an array of String values which describe all of the String
 *     attribute values for each item in a DROPDOWN_OBJECT. The first
 *     index is the item Number.
 * @param ItemPageName
 *     an array of page names to jump to for go to page exit event
 * @param ItemSSNames
 *     an array of slideshow names when go to specific page in slide show
 *     exit event is used.
 * @param ItemSlideNumber
 *     is an array of short values that define the slide number if the
 *     exit event was.
 *
OBJECT_EXIT_TO_A_SPECIFIC_PAGE_NUMBER_WITH_SETTING_INITIAL_SLIDE.
 */
public void populateChoiceObject(short PageSource, boolean isDropDown, boolean append,
String pageName, String objectName, short reserved1, short numberOfItems, byte[] ItemExitEvent,
String[] ItemStringAttributes, boolean[] isItemSelected, String[] ItemParameterAttributes, String[]
ItemPageNames, String[] ItemSSNames, short[] ItemSlideNumber) {
```

## Systems and Methods for Presenting Information on Mobile Devices

```
/**
 * Instantiates a complex list item for populating a Complex List menu.
 * @param itemString is an array of strings for the four possible string
elements(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) for this Item
 * @param imageURL is an array of strings for the URLs four possible image icon
elements(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) for this Item
 * @param ItemStringDefined is an array of booleans defining which of the four possible string
elements(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) are to be
created.
 * @param UseItemColor is an array of booleans defining which of the four possible string
elements(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) should adopt
the alternate item color.
 * @param UseItemFont is an array of booleans defining which of the four possible string
elements(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) should adopt
the alternate item font.
 * @param ItemImageDefined is an array of booleans defining which of the four possible image
icons(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) are to be
created.
 * @param DownloadIcon is an array of booleans defining whether the four possible image
icons(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) should be
downloaded or selected from existing application image content. If selected from existing content,
the imagefilename must be prepended with "Icons/" or "/Images".
 * @param itemImageWidth is an array of shorts defining the width of the four possible image
icons(LOFN_1ST_LINE_TITLE_ELEMENT, LOFN_1ST_LINE_BODY_ELEMENT,
LOFN_2ND_LINE_TITLE_ELEMENT or LOFN_2ND_LINE_BODY_ELEMENT) are to be
created.
 * @param exitEvent - the exit events for these items
 * @param slideNumber - for goto page name the starting slide number of the slide show that the
rolodex points to
 * @param itemLabel - the visible strings for these items
 * @param itemParameter - the parameter strings for these items
 * @param pageName - the goto page name for these items if goto an internal page is an exit event or
the exit URL
 * @param SlideShowName - for goto page name the name of the slide show that the rolodex points
to
 */
public ListItemRun( String[] itemString,String[] imageURL, boolean[] ItemStringDefined, boolean[]
useItemColor, boolean[] useItemFont, boolean[] ItemImageDefined, boolean[] downloadIcon, short[]
itemImageWidth, short ExitEvent, short SlideNumber, String PageName, String ItemParameter,
String SlideShowName) {
```

## **Appendix D**

### **Publisher 2.0 PublisherUsers Database**

#### **1 Introduction**

This document describes types of database tables that are maintained by the Publisher 2.0 system's PublisherUsers database. The Publisher system maintains information about a variety of transactions. This document enumerates this database and its contents.

#### **2 Audience**

This document is intended for:

- a. Back End programmers who need to use the database information as an essential part of the business logic of Publisher-generated Applications.
- b. Partners and Customers that require analytics on the usage of Publisher-generated Applications.
- c. Managers and Administrators that need to maintain Publisher Users and information related to Publisher Users.
- d. Engineering staff, for maintaining and augmenting the player, authoring tool and server code.

#### **3 Database**

There are several databases in use by Publisher System. This document covers the database known as "**PublisherUsers**". This database contains all of the transaction logs maintained by the server for Publisher system.

The relational database management system used by Publisher is MySQL 5.0. The default data storage location is in folder "C:\Program Files\MySQL\MySQL Server 5.0\data\publisherusers" for x86 systems under Windows.

Systems and Methods for Presenting Information on Mobile Devices

**4 Tables**

The following tables are currently available. Details will be presented in subsections following this introduction.

Table Name	Description	Category
application record	List of all applications created by all users of a particular jurisdiction system	Response Director Internal
response file record	Information on the various responses, e.g. call logs, interviews that are available via application	Response Director Internal
application table	List of applications that have back end java code	Back end code support
file table	Java source code for the back end code	Back end code support
clientinfo	List of Publisher users (authors)	Publisher Author Tracking
transactions	Log in (and log out) information on Publisher authors	Publisher Author Tracking
application record	The list of all applications created by all users of a particular jurisdiction system	Response Director Internal
response file record	Information on the various responses, e.g. call logs, interviews that are available via application	Response Director Internal
application table	List of applications that have back end java code	Back end code support
file table	Java source code for the back end code	Back end code support
clientinfo	List of Publisher users (authors)	Publisher Author Tracking
transactions	Log in (and log out) information on Publisher authors	Publisher Author Tracking

Table 1- List of tables in database PublisherUsers

The tables will be discussed in the categories listed in the table.

## Systems and Methods for Presenting Information on Mobile Devices

### **4.1 End User Tracking**

The tables in this category contain information about end users and their behavior. There are three types of events that are tracked. The table that is used to record the events is in parentheses.

1. Request for a Publisher-generated Application via the Response Director (transaction\_record)
2. Install or Delete Notifications from a Java ME application (notification\_record)
3. The sign in and other events of an application tracked by the Content Server (content\_server\_log)



## Systems and Methods for Presenting Information on Mobile Devices

### 4.1.1 transaction\_record

This table is populated by the Response Director, when it receives a request for an application. The Response Director will record all requests, whether successful or not, regardless of what response the end user is directed to.

Field	Type	Null	Key	Default	Extra
transaction_id	int(10) unsigned	NO	PRI	NULL	auto_increment
transaction_date	timestamp(14)	NO		CURRENT_TIMESTAMP	
requested_app_id	int(10) unsigned	YES		NULL	
device_id	varchar(64)	YES		NULL	
response_file	varchar(255)	YES		NULL	
IP_address	varchar(32)	NO			
telno	varchar(32)	YES		NULL	
operator_name	varchar(127)	NO			
incoming_url	varchar(255)	NO			
country_code	char(2)	NO			
player_id_high	bigint(20)	NO		0	
player_id_low	bigint(20)	NO		0	

Table 2 - transaction\_record table definition in MySQL

Systems and Methods for Presenting Information on Mobile Devices

Field	Description	Comments
transaction_id	The primary key for this table. Each new row is assigned a unique number.	This field is guaranteed to ascend in chronological order
transaction_date	The date and time to the second when the request for an application was received.	This records the server's local time, not the end user's time.
requested_app_id	If a suitable Publisher-generated application was found, this number indicates the application_id for that application in the table application_record.	If the requested application is not found, this field will be null.
device_id	The descriptive make and model of the device.	For PC's, this will describe the PC's browser type, e.g. Opera.
response_file	The URL to which the end user was directed.	The source of this URL could be the response_file_record or it could be constructed by the Response Director heuristic method.
IP_address	The IP address from where the request came from, in the form xxx.xxx.xxx.xxx, where xxx is a number between 0 and 255.	This address may be the actual devices IP address, or it may be that for an intermediate device such as a gateway or proxy server.
telno	The mobile phone telephone number. This value will be available if the SMS message, to which a mobile device user responded, contained the telephone number as a parameter.	Numeric digits only. Must begin with country code, e.g. 1 for US.
operator_name	The name of the network operator that was identified from the IP address. For PC's this may simply be the ISP's name.	There may be cases where the operator may not be identifiable (e.g. when user is using an anonymous proxy)
incoming_url	IP address of the device at the time of the connection.	
country_code	The country code identified from examining the IP address.	
player_id_high	The most significant 64 bits of the player ID. The player ID is a UUID that is issued by the Response Director to each Publisher-generated Application request.	Multiple requests from the same device will generate distinct player ID's. Player ID field may be all 0's which indicates that this request for an application
player_id_low	The least significant 64 bits of the player ID.	

## Systems and Methods for Presenting Information on Mobile Devices

	did not issue a PDL Player.
--	-----------------------------

Table 3 - *transaction\_record* field descriptions

### 4.1.2 notification\_record

This table is populated when a PDL Player is installed on or deleted from a Java ME device. The jad file of a PDL Player contains special URL strings that are issued when the downloaded player is installed. The installation may indicate success or failure, and in the case of failure, the approximate cause of failure. In addition, when an end user deletes the application from the device memory, a notification is sent to the server.

By analyzing transactions in this table, one can better determine which devices are likely to have the PDL Player installed. It can also be used for diagnostics, as the device will return a failure code if the installation is not successful. Used in conjunction with the *transaction\_record* table, one can determine, for instance, how long it took to install the player, or how long the user kept the player on their device.

Field	Type	Null	Key	Default	Extra
notify_id	int(11)	NO	PRI	NULL	auto_increment
notify_type	int(11)	NO			
notify_date	timestamp(14)	NO		CURRENT_TIMESTAMP	
notify_message	varchar(50)	YES		NULL	
app_id	int(11)	NO			
version	int(11)	NO			
player_id_high	bigint(20)	NO			
player_id_low	bigint(20)	NO			

Table 4 - *notification\_record* table definition in MySQL

**Systems and Methods for Presenting Information on Mobile Devices**

<b>Field</b>	<b>Description</b>	<b>Comments</b>
notify_id	The primary key for this table. Each new row is assigned a unique number.	This field is guaranteed to ascend in chronological order
notify_type	Either 0 for install or 1 for delete	This field is the value of parameter <u>type</u> in the URL
notify_message	The response string that is contained in the BODY of the http POST.	The possible messages are listed in Table 6 below.
app_id	The application id for this player, as stored in the application_record application_id field.	This field is the value of parameter <u>AppID</u> in the URL.
version	The version number of the application that was installed or deleted.	Corresponds to field <i>version</i> in table <i>application_record</i> .
player_id_high	The most significant 64 bits of the player ID. The player ID is a UUID that is issued by the Response Director to each Publisher-generated Application request.	This field is the value of the parameter <u>PlayerID</u> in the URL.
player_id_low	The least significant 64 bits of the player ID.	

*Table 5 - Notify table field descriptions*

<b>Code</b>	<b>Message</b>
900	Success
901	Insufficient Memory
902	User Canceled
903	Loss of Service
904	JAR size mismatch
905	Attribute mismatch
906	Invalid Descriptor
907	Invalid JAR
908	Incompatible Configuration or Profile
909	Application authentication failure
910	Application authorization failure
911	Push registration failure
912	Deletion notification
913	Required package not supported by the device

*Table 6 - Codes and messages returned by device on installation or deletion of Player*

## Systems and Methods for Presenting Information on Mobile Devices

### 4.1.3 content\_server\_log

This information is logged whenever an application which has content server enabled is started by the user. It could be used to determine the frequency with which a particular application is run, for instance. At present, only sign in is logged. In the near future, sign out will also be logged. Right now the sign out is implied when the session times out.

Field	Type	Null	Key	Default	Extra
content_server_log_id	bigint(20)	NO	PRI	NULL	auto_increment
application_uuid_high	bigint(20)	NO		0	
application_uuid_low	bigint(20)	NO		0	
application_name	varchar(30)	YES		NULL	
author_name	varchar(30)	YES		NULL	
version	int(11)	NO		0	
player_id_high	bigint(20)	NO		0	
player_id_low	bigint(20)	NO		0	
event_type	int(11)	NO		0	
event_dt	timestamp(14)	NO		CURRENT_TIMESTAMP	
event_page_name	varchar(30)	YES		NULL	
event_object_name	varchar(30)	YES		NULL	
event_object_subcategory	int(11)	NO		0	

Table 7 - content\_server\_log definition in MySQL

Systems and Methods for Presenting Information on Mobile Devices

Field	Description	Comments
content_server_log_id	The primary key for this table. Each new row is assigned a unique number.	This field is guaranteed to ascend in chronological order.
application_uuid_high	The most significant 64 bits of the application UUID, which identifies which application this is.	The application UUID can be looked up in table <i>application_record</i> with the same-named fields for more information on this application.
application_uuid_low	The least significant 64 bits of the application UUID.	
application_name	The application name assigned by the author.	Corresponds to field <i>application_name</i> in table <i>application_record</i> .
author_name	The username of the Publisher author that created this application.	Corresponds to field <i>username</i> in table <i>clientinfo</i> .
version	The version number of the application that invoked the content server.	Corresponds to field <i>version</i> in table <i>application_record</i> .
player_id_high	The most significant 64 bits of the player ID. The player ID is a UUID that is issued by the Response Redirector to each Publisher-generated Application request.	Looking this value up in table <i>transaction_record</i> will provide more details about the end user.
player_id_low	The least significant 64 bits of the player ID.	
event_type	The type of event that is being captured: 1 : Sign In 2 : Sign Out 3: Enter Page 4: Exit Page 5: Object Selection	At present, only 1: Sign In is supported automatically. 2: Sign Out is not supported. 3, 4, & 5 can be captured using custom back end java code.
event_dt	The date and time to the second when the end-user event is captured.	The time is stored as the server's local time.
event_page_name	The symbolic name of the page on which the event occurs	The page name is defined by the Publisher author at application creation time.
event_object_name	The symbolic name of the object on which the event occurs	The object name is defined by the Publisher author at application creation time.
event_object_subcategory	The subcategory of the object on which the event occurs. This is primarily to record the slide number, if the object is a slide show.	This value is 0 if the object of the event does not have a subcategory.

Table 8 - content\_server\_log field descriptions

**Systems and Methods for Presenting Information on Mobile Devices**

## Systems and Methods for Presenting Information on Mobile Devices

### **4.2 Publisher Author Tracking**

These tables are used to administer and manage the Publisher Authoring Tool.

#### **4.2.1 clientinfo**

This table contains the username and password for each of the registered users of Publisher for a given site. The word "author" will denote a person that uses Publisher Authoring Tool to create applications. The word "user" in this context refers to "author".

Field	Type	Null	Key	Default	Extra
ID	int(11)		PRI	NULL	auto_increment
UserName	char(30)				
Password	char(30)				
Email	char(30)	YES		NULL	
FirstName	char(30)	YES		NULL	
LastName	char(30)	YES		NULL	
RegDate	date	YES		NULL	
ExpDate	date	YES		NULL	
LastTrans	timestamp(14)	YES		NULL	
Session	int(11)			0	

Table 9 - clientinfo table definition in MySQL



Systems and Methods for Presenting Information on Mobile Devices

Field	Description	Comments
ID	The primary key for this table. Each new row is assigned a unique number.	This field is guaranteed to ascend in chronological order of assignment of user account.
UserName	The user name required to sign in to Publisher Authoring Tool. This value is also saved with each application that this author creates.	Can only contain alphanumeric character and "-", "_", and "+". Guaranteed to be unique within a site.
Password	The password required to sign in to Publisher Authoring Tool.	This field must be populated with a non-empty string.
Email	E-mail address of user, as entered during registration	May be null if no e-mail address was entered.
FirstName	First given name of the user	May be null if no first name was entered.
LastName	Last name of the user	May be null if no last name was entered.
RegDate	The date on which this user account was created.	
ExpDate	The date on which this account expire.	This field is set to 1 year after the registration date during registration.
LastTrans	The last date and time to the second, on which the user signed in.	If the account has never been used, this is the date and time on which the account was created.
Session	The number of times the user signed in to the Publisher Authoring Tool.	0 if the account was created but never used.

Table 10 - clientinfo table field descriptions

Systems and Methods for Presenting Information on Mobile Devices

**4.2.2 transactions**

This table records each session of the Publisher Authoring Tool.

Field	Type	Null	Key	Default	Extra
TransID	int(11)		PRI	NULL	auto_increment
UserIPAddress	char(30)				
TransDate	timestamp(14)	YES		NULL	
Duration	int(11)	YES		1	

Table 11 - transactions table definition in MySQL

Field	Description	Comments
TransID	The primary key for this table. Each new row is assigned a unique number.	This field is guaranteed to ascend in chronological order of user sign in
UserIPAddress	<del>Records the IP address from where the Publisher Server was accessed.</del> This records the username of the user signing in.	There is confusion as to the naming of this field. ??
TransDate	The date and time to the second on which the user signed in.	Records the time in server's local time zone.
Duration	The time during which the user was logged in.	At present, this field is not populated, and always has the default value of 1.

Table 12 - transactions table field descriptions

## Systems and Methods for Presenting Information on Mobile Devices

### 4.3 Back End Code Support

The back end code is necessary for certain application's business logic. The back end code is written in java. The application writer uploads the java code using the Content Management SDK to store the relevant information in a database. Two tables are required to track the back end code: application\_table and file\_table.

#### 4.3.1 application\_table

The application table records the applications that have back end code.

Field	Type	Null	Key	Default	Extra
APPLICATION_ID	float		PRI	NULL	auto_increment
USER_ID	int(20)	YES		NULL	
APPLICATION_NAME	char(50)	YES		NULL	
LAST_COMPILED	double	YES		NULL	

Table 13 - application\_table definition in MySQL

Field	Description	Comments
APPLICATION_ID	The primary key for this table. Each new row is assigned a unique number.	
USER_ID	This records the user ID of the user signing in to the Content Management SDK.	Corresponds to field <i>ID</i> in table <i>clientinfo</i> .
APPLICATION_NAME	Name of the application.	Corresponds to field <i>application_name</i> in table <i>application_record</i> .
LAST_COMPILED	The time at which the back end code for this application was last compiled.	Milliseconds elapsed since midnight, January 1, 1970 UTC.

Systems and Methods for Presenting Information on Mobile Devices

**4.3.2 file\_table**

The file table records the java archive (jar) files that contain the java source code for the back end custom code.

Field	Type	Null	Key	Default	Extra
FILE_ID	float		PRI	NULL	auto_increment
USER_ID	int(20)	YES		NULL	
APPLICATION_ID	int(20)	YES		NULL	
FILE_NAME	varchar(100)	YES		NULL	
FILE	blob	YES		NULL	
LAST_MODIFIED	double	YES		NULL	

Table 14 - file\_table definition in MySQL

Field	Description	Comments
FILE_ID	The primary key for this table. Each new row is assigned a unique number.	
USER_ID	This records the user ID of the user signing in to the Content Management SDK.	Corresponds to field ID in table <i>clientinfo</i> .
APPLICATION_ID	The value of field <i>APPLICATION_ID</i> in <i>application_table</i> which corresponds to the application to which this file belongs.	
FILE_NAME	The filename of the file containing the back end java source code	Java code is stored in one jar file per application.
FILE	The contents of the file.	
LAST_MODIFIED	The date and time of the last modification	Milliseconds elapsed since midnight, January 1, 1970 UTC.

Table 15 - file\_table field descriptions

#### 4.4 Response Director Internal

The Response Director relies on these two tables as part of its heuristics for finding the best response to an application request.

##### 4.4.1 application\_record

The application\_record table contains the list of all applications created at a given site. The combination of category\_name and application\_name comprise a unique identifier for an application at a given site. The application UUID on the other hand provides a globally unique identifier for the application – no two applications in the world will have the same UUID.

Field	Type	Null	Key	Default	Extra
application_id	int(10) unsigned		PRI	NULL	auto_increment
application_name	varchar(50)				
category_name	varchar(50)				
version	int(10) unsigned			0	
application_UUID_high	bigint(20)			0	
application_UUID_low	bigint(20)			0	

Table 16 - application\_record definition in MySQL

Field	Description	Comments
application_id	The primary key for this table. Each new row is assigned a unique number.	
application_name	The name of the application group. This is not necessarily the name of the jad/html file.	Corresponds to field <i>ID</i> in table <i>clientinfo</i> .
category_name	The category name == author of the application.	Corresponds to field <i>UserName</i> in table <i>clientinfo</i> .
version	The most recent version number of the application. Applications in the field may have lower version number.	This version number is attached to the player's jad file. This is NOT the pdl version number.
application_UUID_high	The most significant 64 bits of the application UUID, which identifies which application this is.	Application UUID is assigned by the Publisher Authoring Tool, and is maintained in the .pdl and .akp files.
application_UUID_low	The least significant 64 bits of the application UUID.	

Table 17 - application\_record field descriptions

## Systems and Methods for Presenting Information on Mobile Devices

### 4.4.2 response\_file\_record

The response\_file\_record table contains the “responses” that are generated by the Publisher Authoring Tool server. “Response” is some executable or displayable code that is returned to an end user when the end user requests an application. Publisher automatically creates several variants of both jad and html files for each application. This information is stored in this table.

Field	Type	Null	Key	Default	Extra
response_file_id	int(10) unsigned		PRI	NULL	auto_increment
application_id	int(10) unsigned			0	
response_file_path	varchar(255)				
certificate_id	int(10)			0	
jad_property_flags	varchar(32)			0	
operator_flags	varchar(32)			0	
screen_h	int(10) unsigned			0	
screen_v	int(10) unsigned			0	
video_format	varchar(45)	YES		NULL	
type	varchar(45)				

Table 18 - response\_file\_record definition in MySQL

Field	Description	Comments
response_file_id	The primary key for this table. Each new row is assigned a unique number.	
application_id	The id number of the application group.	Corresponds to field application_id in table application_record.
response_file_path	The URL, relative to PUBLISHER_HOME, where the response file resides	The file may be generated by other than Publisher, e.g. a WML deck.
certificate_id	An id that indicates the certificate, if any, that is embedded within this response. Certificates may be attached to jad files.	At present, only one type of certificate (Verisign Class 3) is available, so this field is not used, and its value is 0.
jad_property_flags	A bit flag, stored as a variable length string, indicating the properties of a java application. Each bit is either character “0” or “1”.	The definition of the bits is described in Table 20. Only used with jad files. For other types, this value is “0”.

Systems and Methods for Presenting Information on Mobile Devices

Field	Description	Comments
operator_flags	A bit flag, stored as a variable length string, indicating that this jad file is appropriate or customized for a specific network operator.	These are “hints” for the Response Director, and are not absolute indicators. See Table 21.
screen_h	The horizontal pixel size of the display for which this “response” was designed.	The screen size is used as a “hint” to the Response Director to find the most appropriate sized “response”.
screen_v	The vertical pixel size of the display for which this “response was designed	
video_format	The video format, as a mime type, that is used by this “response”.	The supported mime types are listed in Table 22. This field may be null or empty.
type	String representing different general categories of jad, html and wml files	See Table 23 for description of values.

*Table 19 - response\_file\_record field descriptions*

## Systems and Methods for Presenting Information on Mobile Devices

### 4.4.2.1 jad\_property\_flags

The property flags are numbered 0 through 31, with 0 being the least significant bit. The flags are stored in string representation, with each bit occupying one character, either a "0" or a "1". It is stored as a variable length string, with no leading zeros. For instance if only bit 2 and bit 0 were set, the value stored is "101"; if no bits are set "0" is stored.

Bit number	Description
0	When set, it indicates that the media files and/or pdl and/or video files are embedded in the jar file.
1	When set, it indicates that this application contains a video.
2	When set, it indicates that this jad file contains a digital certificate.
3-31	Reserved, and must be 0.

Table 20 - jad\_property\_flags bit definitions

### 4.4.2.2 operator\_flags

The operator flags are numbered 0 through 31, with 0 being the least significant bit. The flags are stored in string representation, with each bit occupying one character, either a "0" or a "1". It is stored as a variable length string, with no leading zeros. For instance if only bit 2 and bit 0 were set, the value stored is "101"; if no bits are set "0" is stored.

The operator flags are hints provided by the Publisher to the Response Director, to indicate which operators this response is suitable for. It is not an absolute indicator: if the operator bit is not set, the Response Director may still select this response based on other criteria, and similarly, even if the bit is set, the Response Director may select another response.

The values that are currently defined are the operators that the company anticipates it will be supporting in the near future. Others operators will be added as requirements arise.

Bit number	Description
0	If set, hinted for AT&T Wireless network
1	If set, hinted for Sprint PCS network
2	If set, hinted for Verizon Wireless network.
3	If set, hinted for T-Mobile USA
4-7	Reserved, and must be 0
8	If set, hinted for Softbank Mobile
9	If set, hinted for NTT DoCoMo (Japan)
10	If set, hinted for Japan Telecom
11	If set, hinted for KDDI au (Japan)
12	If set, hinted for Willcom (Japan)
13-15	Reserved, and must be 0.
16	If set, hinted for Vodafone (UK)
17	If set, hinted for T-Mobile (UK)
18	If set, hinted for Three (UK)



Systems and Methods for Presenting Information on Mobile Devices

<b>Bit number</b>	<b>Description</b>
19	If set, hinted for Orange (UK)
20	If set, hinted for O2 (Telefónica) (UK)
21	If set, hinted for Movistar (Telefónica) (Spain)
22 – 28	Reserved, and must be 0.
29	If set, hinted for CTI (Argentina)
30	If set, hinted for Personal (Argentina)
31	If set, hinted for Movistar (Argentina)

*Table 21 - operator\_flags bit definitions*

## Systems and Methods for Presenting Information on Mobile Devices

### 4.4.2.3 Video formats

The player recognizes the video formats listed in the following table.

Video mime type
video/3gpp
video/3gpp2
video/mp4
video/h263-2000
video/h263
video/h264
video/quicktime

Table 22 - Supported video mime types

### 4.4.2.4 'type' field values

The type field is used as a hint to the Response Director as to what the response is intended for. This is not an absolute indicator, and the Response Director may override this hint. This field is optional, and may be empty or null.

Value	Description
jad midp2 0 qvga	This response is appropriate for QVGA (320x240) sized MIDP 2.0 devices
jad midp2 0 qcif	This response is appropriate for QCIF (176x208) sized MIDP 2.0 devices
jad midp2 0 vga	This response is appropriate for VGA (480x640) sized MIDP 2.0 devices
jad midp2 0 palm	This response is appropriate for Palm devices with 320x320 screen size.
jad midp2 0 generic	This response is appropriate for devices other than the above MIDP 2.0 devices
generic html	This response is appropriate for PC type browsers
generic pda	This response is appropriate for PDA (e.g. iPAQ) browsers
wap generic device	This response is appropriate for WAP capable devices
(empty or null)	This response is not hinted with a 'type' value.

Table 23 - 'type' values

<End of document>

# Appendix L

## Generic Player Update Plan

### 1 Audience

This document's intended audiences are:

- Developers interested in the implementation methodology of the J2SE Generic player, for further development or for debugging.
- QA engineers, to write test plans for the Generic player.
- Technical managers and project managers that schedule resources for further development or debugging of the Generic player or for extending the same implementation paradigm to create players for other platforms.

The information contained in this document is confidential and proprietary to Express Mobile.

### 2 Objectives

Since the beginning of 2007, the focus of development has been on the MIDP player of Publisher 2.0. The MIDP player had achieved many new functions while the so-called Generic player, which runs as an Applet under a browser, had languished.

The primary cause of not keeping the Generic player up to date was cost. Because the two players differed significantly in source code base, when a change in the MIDP player was made, a similar change was also required to be placed manually into the Generic player code. This effort to keep the Generic player code base up to date was significant enough that it was abandoned while the focus was placed on the MIDP player.

The current effort then, has two primary objectives:

1. Bring the Generic player functionality up to the level of the MIDP player.
2. Minimize the code changes between Generic player and MIDP player to reduce cost of maintaining both players.

### 3 Strategy

Both Generic and MIDP player are written in the Java language. Generic uses J2SE platform while MIDP uses the J2ME/CLDC/MIDP platform. This is a salient feature which will be fully exploited.

MIDP Java standard libraries (cldcapi11, midpapi20) offer a subset of the API's available under J2SE. It is not a perfect subset, as there are slight differences in the method signatures and semantics of certain parameters.

The strategy for implementation involves using the MIDP player source code with minimal changes and to build a support library layer of code which appear identical to the MIDP API's yet perform under the J2SE environment.

## Systems and Methods for Presenting Information on Mobile Devices

The goal is to have a single code base that can be used for both MIDP and Generic player generation. Any modification to the code base will immediately be reflected in both players without any additional effort on the part of the programmer.

### **4 Details of Execution**

The existing MIDP player source code is the master source code from which to derive the Generic player code. The entire MIDP player source code is considered the “business logic” of the player. Support code was therefore any layer outside of the player code proper. Since the calls to the outside of the player code are made exclusively through the MIDP/CLDC standard API's (with the exception of JSR-75 support), the so-called VM abstraction layer took the MIDP API specifications and implemented a subset to run under an Applet environment of J2SE.

N.B.: The current so-called VMAbstraction code which implements the MIDP API under J2SE is a subset of the MIDP API. Many classes are omitted. Classes with identical signatures and semantics between J2SE and J2ME have not been re-implemented (e.g. java.util.Vector). Even within the classes that are implemented, not all methods, fields, classes, and constants may be implemented. What is implemented? The subset that is required for the MIDP player *as it exists now* to run under J2SE Applet environment. If in future, the MIDP Player uses additional API methods not currently supported, VMAbstraction code will require augmentation.

For any logic that is specific to one implementation or the other, special “glue” code is added so that the main business logic need not be concerned about the differences. In addition, a runtime variable (isNotMIDP) is available to the player to determine which environment it is running under, so that simple differences in logic that do not require “glue” code can be made inline using an if-else type branching logic.

#### **4.1 MIDP Code Excluded from Generic**

The following code that exists in the MIDP player has been excluded from the Generic player:

<b>Class excluded from Generic</b>	<b>Reason</b>
CacheStore_RS	Generic player relies on the browser cache. Having a player level cache would be redundant.
MediaCache	Same reason as above.
PimAPIs	There is no accessible PIM under the browser.
pim	Same reason as above.

The code is excluded at build time, when the Generic player code is extracted from the Player\_Master code.

#### **4.2 Keyboard Mapping**

##### **4.2.1 Control Keys**

The MIDlet keys for navigation and other control keys are mapped as follows:

## Systems and Methods for Presenting Information on Mobile Devices

Phone Key	Generic Mapping	Notes
left joystick/arrow	left arrow key	Standalone left arrow key, not numeric left arrow keypad with NumLock ON.
up joystick/arrow	up arrow key	Standalone up arrow key, not numeric up arrow keypad with NumLock ON.
right joystick/arrow	right arrow key	Standalone right arrow key, not numeric right arrow keypad with NumLock ON.
down joystick/arrow	down arrow key	Standalone down arrow key, not numeric down arrow keypad with NumLock ON.
fire/select	Enter key	Any keyboard key that generates a newline '\n' character.
back/clear key	Backspace key	
left soft key	n/a	Mapped using command panel button. See 4.2.3 <i>Soft Key Emulation</i>
right soft key	n/a	Mapped using command panel button. See 4.2.3 <i>Soft Key Emulation</i>

### 4.2.2 Text Entry Keys

Text entry method for Applet uses the natively available text entry method. There has been no attempt to emulate the text entry method available on the phone. All printable characters are accepted as text.

The following have specific mapping in that they retain the game key values that are used in the MIDlet code:

- a. Numbers 0 through 9 are on the MAIN keyboard cluster are mapped to the phone's numeric keypad 0 through 9. The PC keyboard numeric keypad keys 0 through 9 with NumLock ON are not. The latter can be used for regular text entry but are not recognized for numeric fields.
- b. The asterisk ("\*") is mapped to the asterisk symbol found as SHIFT-8 on a QWERTY keyboard.
- c. The hash ("#") is mapped to the hash symbol found as SHIFT-3 on a QWERTY keyboard.

### 4.2.3 Soft Key Emulation

Soft keys are emulated in the Generic player using a row of keys at the bottom of the applet window. There are 2 such buttons available, depending on the number of soft keys that have been defined. There is at a minimum always one soft key defined, for the Exit command.

The arrangement of the soft keys is described here:

- a. If only one command exists, it will be the right button and the left button will be blank and non-functional.
- b. If two commands exist, the higher priority item shall occupy the right button and the lesser priority the left.

## Systems and Methods for Presenting Information on Mobile Devices

- c. If three or more commands exist, the highest priority command shall occupy the right button, and the left button shall be named "Options", which, when clicked will open up a popup menu. The popup menu shall contain the rest of the commands ordered by priority, with higher priority commands toward the top of the menu.
- d. The priority is determined by the priority integer value assigned to the command when the Command object is created, which is passed to addCommand().

### **4.3 Mouse / Pointer Device Mapping**

Three events from Applet's mouse are mapped to the available MIDlet API's

Mouse Pressed (any button) → Pointer Pressed

Mouse Released (any button) → Pointer Released

Mouse Dragged (with any button down) → Pointer Dragged

### **4.4 UI component Mapping**

The basic hierarchy of the MIDlet UI objects has been retained in the abstraction layer.

The class Displayable is the parent of many of the UI components.

Displayable is implemented as a wrapper around the J2SE Component class. All subclasses are also wrappers of a subclass of Component, as they inherit the Component wrapped in Displayable. The mappings are as follows:

<b>MIDP UI subclasses of Displayable</b>	<b>J2SE Component subclass</b>
javax.microedition.lcdiui.Displayable	java.awt.Component
javax.microedition.lcdiui.Canvas	java.awt.Canvas
javax.microedition.lcdiui.TextBox	java.awt.TextArea
javax.microedition.lcdiui.Form	java.awt.Panel
javax.microedition.lcdiui.Alert	java.awt.Panel

The subcomponents of Form are all subclasses of Item, as in MIDP. Item is also implemented as a wrapper of J2SE Component. The Item subclasses are mapped as follows:

<b>MIDP UI subclasses of Item</b>	<b>J2SE Component subclass</b>
javax.microedition.lcdiui.Item	java.awt.Component
javax.microedition.lcdiui.Gauge	java.awt.Label (static text, no animation)
javax.microedition.lcdiui.ImageItem	java.awt.Component (subclass ImageComponent based on Image)
javax.microedition.lcdiui.StringItem	java.awt.TextArea
javax.microedition.lcdiui.TextField	java.awt.TextField

#### **4.4.1 Current Limitations**

As of 2007.06.13 the following limitations exist in the UI mapping from MIDlet to J2SE:

1. Alert class is not implemented, just a stub.
2. TextBox behavior is not completely emulated. See section 5 *MIDP and Generic Player Functionality Differences*.

## Systems and Methods for Presenting Information on Mobile Devices

3. Gauge is implemented as a single fixed label that reads "Loading. Please wait...". There is no animation of a gauge.

### **4.5 Audio / Video Mapping**

The implementation requires a QuickTime decoder on the client PC. The Manager is essentially a factory for Player. Player is an interface which each distinct player implements. The QuickTime Player implementation code is found in Player\_QT.

QuickTime has a special Component QTComponent which is created to hold the video image. This Component is set up with the URL of the video file (http, or rtsp protocols), the location and size of the video. Because the Player code does not have any access to the Canvas within which space it must draw, a listener interface is created such that when Display.setCurrent() is called to switch pages, Player\_QT is notified so that it may alter the Applet container.

The order of components in the Container (in this case the Applet) is important for proper layering. The video Component must appear BEFORE the canvas Component which draws the rest of the page. The displayChanged listener is called with a reference to the Container. Normally the order is thus prior to any changes:

Index	Contents
0	MyAwtCanvas (extension of java.awt.Canvas)
1	CommandPanel (buttons for softkey emulation)

The displayChanged listener in Player\_QT modifies this as follows:

Index	Contents
0	CommandPanel (buttons for softkey emulation)
1	QTComponent (video component)
2	MyAwtCanvas (extension of java.awt.Canvas)

More <TBD> as this is under construction. Audio is left.

### **4.6 MIDlet to Applet Mapping**

The implementation makes the MIDP MIDlet class a subclass of Applet. The following are the method mappings used.

MIDlet method	Applet equivalent or other function
destroyApp()	Mapped to destroy().
startApp()	Mapped to start().
pauseApp()	Mapped to stop().
checkPermission()	Not mapped, always returns -1 indicating "unknown".
getAppProperty()	Mapped to getParameter().
notifyDestroyed()	This will exit the applet via System.exit(0).
notifyPaused()	Not mapped.
platformRequest()	Mapped to getAppletContext().showDocument(). It will always return false.
resumeRequest()	Not mapped.

## Systems and Methods for Presenting Information on Mobile Devices

### **4.7 Font Mapping**

The Font class is a hybrid of java.awt.Font and javax.microedition.lcdui.Font. There are two implementations, one for MIDP (J2ME) and one for Generic (J2SE).

#### **4.7.1 J2SE Abstraction Layer**

The abstraction Font class is mapped as a subclass of java.awt.Font. Most of the Font methods required in the player, map one-to-one to java.awt.Font. The call to getFont() is a hybrid of javax.microedition.lcdui.Font and java.awt.Font, in that it takes a face name (String), style (int) and point size (int). The javax.microedition.lcdui.Font getFont() takes face (int), style (int) and size (int). The player logic already deals with face names and point sizes, and this maps much closer to what J2SE supports. Specifically, getFont() in the abstraction layer is defined for both MIDP and Generic as:

```
public static Font getFont(String face, int style, int size)
```

The "style" attributes in MIDlet is retained. This has three attribute bits: BOLD, ITALIC and UNDERLINE, and a default PLAIN (with no bits set). PLAIN, BOLD and ITALIC are supported by the java.awt.Font(String name, int style, int size) constructor, but UNDERLINE is not. To support UNDERLINE attribute, getFont() in the abstraction layer calls the alternative constructor java.awt.Font(Map attributes), which allows detailed specification of the font, including underlining.

The abstraction also supports Font methods that are unique to J2ME: methods charWidth(), getHeight() and getStringWidth() are supported by wrapping java.awt.FontMetrics into the Font class and calling this wrapped FontMetrics to obtain values for these methods.

#### **4.7.2 J2ME Abstraction Layer**

The getFont() method in the abstraction layer differs from the native javax.microedition.lcdui.Font version in that it uses a String face name and a point size value for indicating the font size:

```
public static Font getFont(String face, int style, int size)
```

To support this alternative getFont(), which is used by the MIDP player logic, the abstraction getFont() for J2ME/MIDP maps the face as follows:

<b>AWT face name</b>	<b>MIDP face id (constant value)</b>
Dialog	FACE_SYSTEM (0)
Monospaced	FACE_MONOSPACE (32)
Serif	FACE_PROPORTIONAL (64)
<any other name>	FACE_PROPORTIONAL (64)



## Systems and Methods for Presenting Information on Mobile Devices

Font size mapping is obtained by first mapping the face to one of the three available MIDP faces, and then obtaining the line height of each of the three MIDP sizes available for that face. Then line height is calculated by comparing the line height (including leading) that the original designer specified, based on the chosen point size., with the available device dependent fonts, and a best match calculation is performed. This target line height operation occurred in the Publishing Client using the J2SE API:

```
getFontMetrics("Java Assigned Font Object").getHeight();
```

Let  $h_i$  be the original line height and  $h_L$ ,  $h_M$ , and  $h_S$  be the font height for the LARGE, MEDIUM, and SMALL fonts, respectively, available on MIDP. The mapping of font size is as follows:

point size	MIDP font size (constant value)
$h_i > h_{L-2}$	SIZE LARGE (16)
$h_{M-2} > h_i \geq h_{L-2}$	SIZE MEDIUM (0)
$h_i \leq h_{M-2}$	SIZE SMALL (8)

### 4.8 Image Mapping

The MIDlet API Image wraps a java.awt.Image object. The various createImage() calls are mapped as follows:

MIDlet method	Applet equiv. or other function
createImage() for mutable image	Mapped to BufferedImage, a subclass of java.awt.Image.
createImage(java.io.InputStream)	Mapped to a custom PNG decoder. NOT RECOMMENDED FOR USE. This limits the image file to PNG file only, and the implementation does not currently even support full PNG. Use createImage(String) instead.
createImage(String)	Mapped either to getClass().getResource(name) for embedded resources, or to Applet.getImage(url) for external network resources.
createImage(int[] rgb...)	Mapped to BufferedImage with the rgb data copied into it.
createImage(Image)	Not implemented. (incompletion)

### 4.9 Display Implementation

Class Display's primary purpose is to switch display contexts. This is what is implemented in the Applet code. When Display.setCurrent(Displayable) is called, the current top level Components attached to the Applet (Container) are removed and replaced with the incoming Displayable. The Displayable consists of a main component, such as Canvas or Form, and a subsidiary component called CommandPanel. The latter

## Systems and Methods for Presenting Information on Mobile Devices

implements the soft key emulation as buttons on the bottom of the applet panel window. Both the main Component and the CommandPanel are attached to the applet Panel, made visible, and redrawn.

NOTE: The method

```
setCurrent(Alert alert, Displayable nextDisplayable)
```

is not fully implemented. Although it will switch to the Alert, it will not switch to the next Displayable when the Alert is timed out or dismissed. This is an incompleteness.

### **4.10 Graphics Mapping**

The MIDP API of Graphics wraps the java.awt.Graphics. Otherwise, the methods of the two are very similar. Graphics could not be implemented as a subclass of java.awt.Graphics because of an incompatible return value for the method getColor(). In MIDP this returns type int, and in AWT it returns type Color.

### **4.11 Connection Mapping**

In MIDP, class Connection is used for connecting to the network server, primarily the Content Server. Class Connector is defined as a factory class, even in MIDP. It delivers any subclass of Connection. Connection is the super-super class for HttpConnection, an interface which defines the methods for HTTP connections. This is implemented in a new class Connection\_HTTP which is essentially a wrapper around java.net.HttpURLConnection.

### **4.12 RecordStore Implementation**

The MIDP player relies on RecordStore (aka Record Management System or RMS) memory to store data that is persistent, i.e., whose value persists between sessions of running the player. There is no equivalent class in J2SE under the Applet environment.

An emulation of RecordStore is implemented in the Generic player using a combination of heap memory and cookies. The RecordStore records are mapped to a Hashtable in heap whose key is the record ID, a unique integer, and the value is the data for that record. This Hashtable's content is read from a cookie in the browser upon initialization and is written out to a cookie at termination of the player. If cookies are not available to the applet either due to browser's user preference or because the applet is not running in an environment that supports cookies, such as the Applet Viewer, then only the heap memory is used to store the Hashtable. In such case, the data stored in the Hashtable will only last until the end of session, at which time they are destroyed. This will still allow consistency within a session but not between sessions.

The primary users of RecordStore are the PlayerID code and the PlayerPersistentStorage code, to store the application start page and other user defined variables.

## **5 MIDP and Generic Player Functionality Differences**

The following are known differences between MIDP and Generic.

## Systems and Methods for Presenting Information on Mobile Devices

Function	MIDP	Generic
URL to access player/app	<a href="http://domain.com/websites/appName/bin/appName.jad">http://domain.com/websites/appName/bin/appName.jad</a>	<a href="http://domain.com/websites/appName/bin/appName.html">http://domain.com/websites/appName/bin/appName.html</a>
Pointer support	Only on devices with pointer (1% of target devices).	Devices with a mouse or other pointing device (99% of PCs).
Soft key commands	Use device-provided keys and Application Management Software (AMS) provided logic for assignment and order of commands	Emulated using two buttons components at bottom of applet window. Can only be accessed via mouse click. Assignment logic is described in § 4.2.3.
Runtime Environment	Runs as a MIDlet within a MIDlet suite launched by AMS.	Runs as applet launched from within a browser via HTML
Slideshow animation	Runs at the frames per second which the device can support.	Because the processor is likely to be much more powerful, the animation may appear different due to timing between frames.
Text Box	<ul style="list-style-type: none"> <li>• Supports automatic line wrapping.</li> <li>• Limited to number of characters specified in constructor.</li> <li>• Does not allow input of NEWLINE character.</li> </ul>	<ul style="list-style-type: none"> <li>• Does not automatically wrap lines. Scroll bars available for horizontal scroll when line length exceeds display width.</li> <li>• No practical limit to number of characters.</li> <li>• Supports input of NEWLINE character.</li> </ul>
PIM support	Supported if underlying device supports JSR-75 and permission is granted	Not supported. There are no client side PIM data available under a browser.

## 6 Appendix

### 6.1 List of Partially or Fully Implemented MIDP API Classes

Alert  
 AlertType  
 Canvas  
 Command  
 CommandListener  
 Connection  
 ConnectionNotFoundException  
 Connector  
 Control  
 Controllable

## Systems and Methods for Presenting Information on Mobile Devices

Display  
Displayable  
Font  
Form  
Gauge  
Graphics  
HttpConnection  
Image  
ImageItem  
InputConnection  
InvalidRecordIDException  
Item  
Manager  
MediaException  
MIDlet  
MIDletStateChangeException  
OutputConnection  
Player  
PlayerListener  
RecordComparator  
RecordEnumeration  
RecordFilter.java  
RecordStore  
RecordStoreException  
RecordStoreFullException  
RecordStoreNotFoundException  
RecordStoreNotOpenException  
StringItem  
TextBox  
TextField  
VideoControl  
VolumeControl

## Appendix V

# Uber Icon Strip and Inter-Application Navigation Specification

Icon Strips are now, from a UI point-of-view, distinctly different objects from Slide shows. Slide Shows, on the other hand, can morph into various Chat Rooms utilizing our Blogging Technology.

### A: Creating an Icon Strip

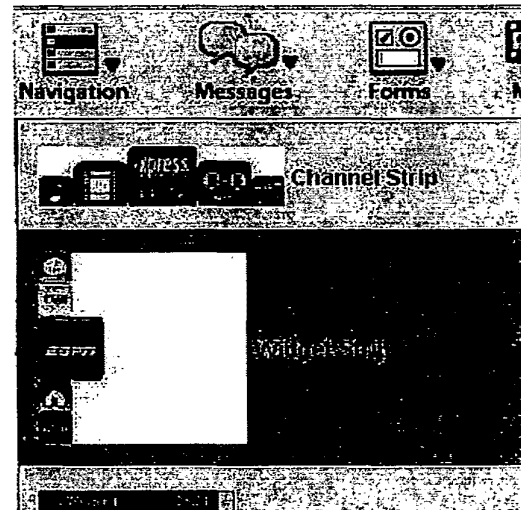
There are 2 starting points for Icon Strips as shown to the right. Once created, there are no restrictions in terms of working with these attributes.

The only differences in these two choices are:

- 1: Orientation
- 2: Type of Icon Strip

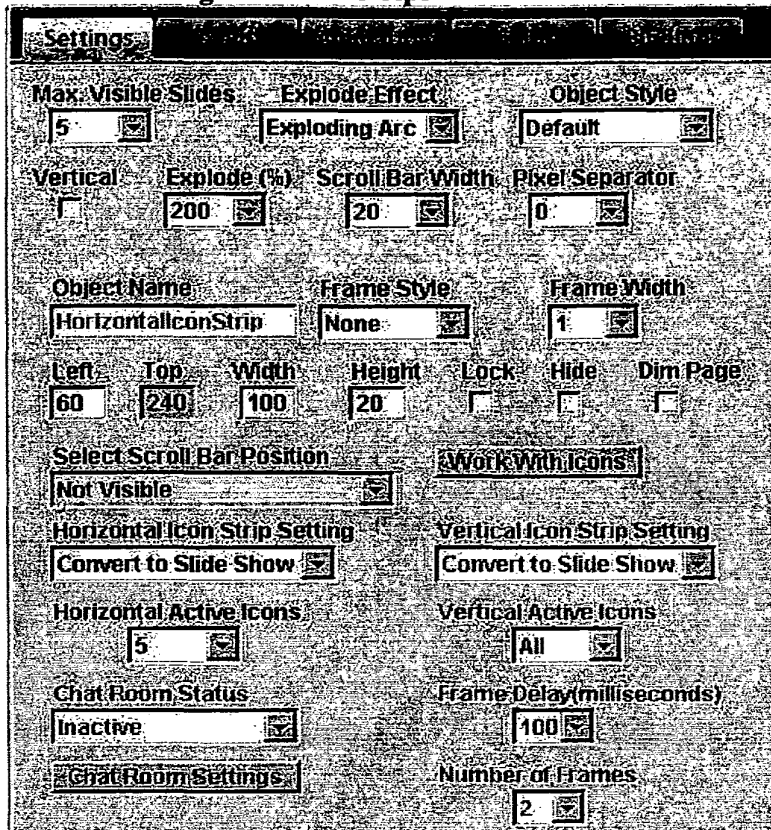
The Channel Strip utilizes the Mac-Like UI, while the Widget Strip resembles the ESPN implementation but you can also add a slide transition and auto sound when navigating between Icons.

It is possible to have both a vertical and horizontal icon strip defined.



## Systems and Methods for Presenting Information on Mobile Devices

### B: Working with Icon Strips



Although it is possible to still use the old Edit Dialog Box, going forward we should only be using the Inspector as shown above.

1: Maximum Number of Visible Slides

This setting will display, in the orientation selected, the number of actual slides, or the Visible Slides setting, whichever is smaller.

2: Explode Effect

Setting to none is a way to morph the icon strip into a slideshow (at least in theory).

3: Vertical

Switches orientation.

4: Explode Percent.

This defines the relationship between the Selected Icon Size and that of the smallest Icon. For Exploding Arc, the intermediate icons are the average between the two.

5: Scroll Bar Width and Select Scroll Bar Position

These settings control the presence, width, and placement for a Scroll Bar.

6: Pixel Separator:

## Systems and Methods for Presenting Information on Mobile Devices

This is the number of pixels, from 0 to 15, that will define the separation between slides.

### 7: ObjectName

For both vertical and horizontal icon strips this name is fixed and is not editable. We can, however, if you don't like the name a chose, change it in the UI.

### 8: Frame Style and Width:

This choice still remains and is available for all XpressMo Objects.

### 9: Left, Top, Width and Height.

These values should generally not be changed, as the positioning and size is being calculated automatically by the number of visible slides, the size of the original images, the Explode Percentage Setting, the Exploding Effect, the Page Size, and the Orientation and placement selection. However, to adjust the value that controls centering from "centered within the page" to something else based on aesthetics is available.

### 10: Lock and Hide.

These settings still remain and are available for all XpressMo Objects.

### 11: Dim Page

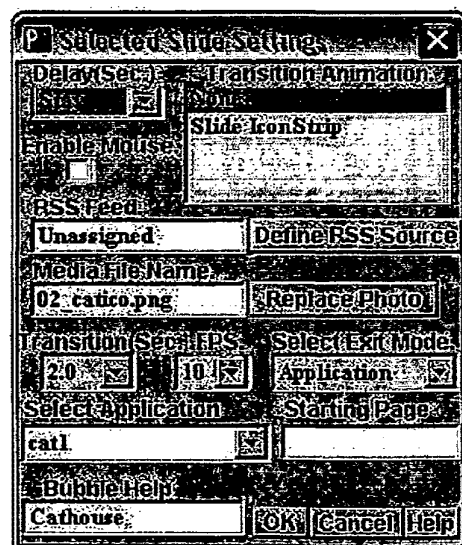
This determines whether the page should be dimmed with the Icon Strip is visible and active.

### 12: Work With Icons

This permits the insertion, deletion, and setting attributes for specific icons.

### 13: Setting Navigation for Starting an Application on a Specific Page

If the Execute Application Exit Event is chosen, then both a dropdown list for the available applications in your workspace will become available for selection, as well as a text field that lets you determine the page name that you want to be the initial page. As a refinement, I will eventually change this to a drop down menu so that you can select from the available pages for a given application.



## Systems and Methods for Presenting Information on Mobile Devices

### 14: Horizontal Icon Strip Setting (or Vertical Icon Strip Setting)

This determines where the icon strip will be placed based on its orientation, whether you want it always visible or enter from a page edge, or whether you want this Icon strip to become a slide show.

### 15: Horizontal Active Icons (or Vertical Active Icons)

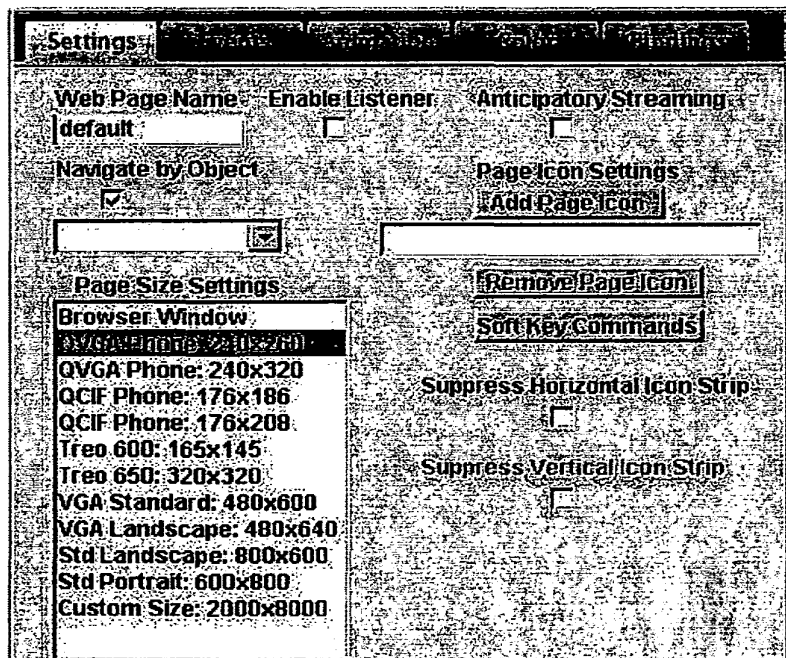
This sets which Icons, based on the number selected and their order, will immediately be available for selection. Moving icons from a selected to/from an unselected state is managed by the Icon Selection Complex List object.

### 16: Frame Delay and Number of Frames

These settings are only available for the "Exploding Icon Type". This sets the speed and granularity for the Explosion and transition effect for this type of Icon Strip.

### 17: Selective Suppression of Icon Strip Availability

It is possible, at the page level, to selectively disable either the Vertical Icon Strip, the Horizontal icon Strip, or both.





## Systems and Methods for Presenting Information on Mobile Devices

### **C: Inter-Application Navigation**

Similar to B:13 above, it is now possible to assign a particular starting page when utilizing the ExitEvents for all XpressMo Objects and from a particular Choice Object Item or a Complex List item.

There are up to 2 Uber Icon Strips that can be assigned during a Player Session, a Vertical Uber Icon Strip and/or a Horizontal Uber Icon Strip.

The first Uber Icon Strip-enabled application that is visited by the Player during a given session will application will be the owner of its Uber Icon Strip. This could be the Uber-Portal, or, assuming that the Uber Portal only has one Uber Icon Strip defined, then the first application that defines the other type of Uber Icon Strip will be the Owner.

Ownership as defined as having a set of Icons that remains persistent throughout the session. These Icons may be visible or invisible, but they will be persistent.

When a subordinate application is visited that has an Uber Icon Strip whose type already has an owner, then the subordinate applications Icons will be appended to the Uber Icon Strip. This process can continue for any number of Application Layers.

When the owner of an Uber Icon Strip is visited, all subordinate icons are immediately removed.

## Appendix U

### Specialized Text Field and Text Button Specification

Text Fields now have an additional type for sending SMS messages.

If Text Fields are of the types:

- 1: Numeric
- 2: Phone Number
- 3: SMS Number

The text entry model is now direct. Only numeric character can be entered. Cursor movement, editing and deleting within the string are also supported. When a direct entry text field is selected there will be a soft key command available "clear" for deleting the character to the right of the insertion point.

In the event that the Text Field has the attribute Phone Number, then, under the condition that 10 numeric characters have been entered, when leaving the Text Field the numbers will be formatted as:

(xxx) xxx-xxxx.

In the event that the Text Field has the attribute SMS Number, then:

1: under the condition that 10 numeric characters have been entered, when leaving the Text Field the number "1" will be prepended .

In either case, the number will be formatted as:

1-(xxx) xxx-xxxx.

When entering these text fields the formatting will be removed and just the numeric characters will display.

Note: For all other text editing choices the indirect method of opening up a MIDP Text area will remain in order to support foreign languages and T9.

Submit, Clear and Text buttons now have a new state "Don't Scale". If enabled the MIDP player will maintain the original width of the button object. This is useful if you are superimposing these type of buttons over rasterized text or images.

Systems and Methods for Presenting Information on Mobile Devices

**Appendix A: PDL Specification for Specialized Text Fields**

ObjectIntegerAttribute[18]

Text Field:	Type
0:	default
1:	Password
2:	Numeric
3:	Phone Number
4:	SMS Number

## Appendix T

# Soft Key Commands Specification

Any number of Softkey commands can be assigned to a page. In addition, any of the 10 numeric keys can be assigned to a soft key commands and will act as a synonym.

Authoring Softkey commands will have a similar UI as authoring Lists.

The designer will be able to add, edit and remove a soft key from the EditPage Dialog Box in a manner similar to that of working with list items in the ListEdit dialog box.

The default Back Command will appear and cannot be erased. However, it can be reset to "Goto Page" if desired.

When adding or Editing Soft key commands the following attributes will be assignable.

- 1: A drop down for selection of the attached command. The 8 choices initially will be:
- Goto Page
  - Back (this is defined by default)
  - Goto Page & Report
  - Back 1 View & Report
  - Goto Start Page
  - Goto Start Page & Report
  - Login
  - Logout

If a command with "Report" is selected, then the normal getParentObjectSelected interface method will be called, with a boolean identifying this as a soft key command. The command name is used as the object name and the command number is used as the object number.

2: A text field to define the string that will appear as the soft key list (which, on many phones, may appear as a scrolling pane after a certain number of commands are defined.

3: If Goto Page is selected then a dropdown will appear to select the page name for this operation.

4: If Login or Logout is selected, then a dropdown of the currently available variable names will appear. Currently there is only \$username\$.

---

## **Systems and Methods for Presenting Information on Mobile Devices**

5: A drop down for any of the remaining unassigned numeric keys as well as "none". Note that the numeric keys of 2, 4, 6 and 8 are already assigned as navigation synonyms but can be overridden if desired by the designer. (not yet implemented)

## Systems and Methods for Presenting Information on Mobile Devices

### **Appendix A: PDL Specification for Soft Key Commands**

#### Soft Key Command Attributes

NumberOfCommands: short

Vector SoftKeyCommand()  
Vector of Shorts where:  
0=none.  
1=goto page  
2=back

Vector SoftKeyCommandString()  
Vector of Strings that will appear in the soft key menu

Vector SoftKeyParameter()  
Vector of String Parameters for executing the command.  
For "goto page" it is the pagename.

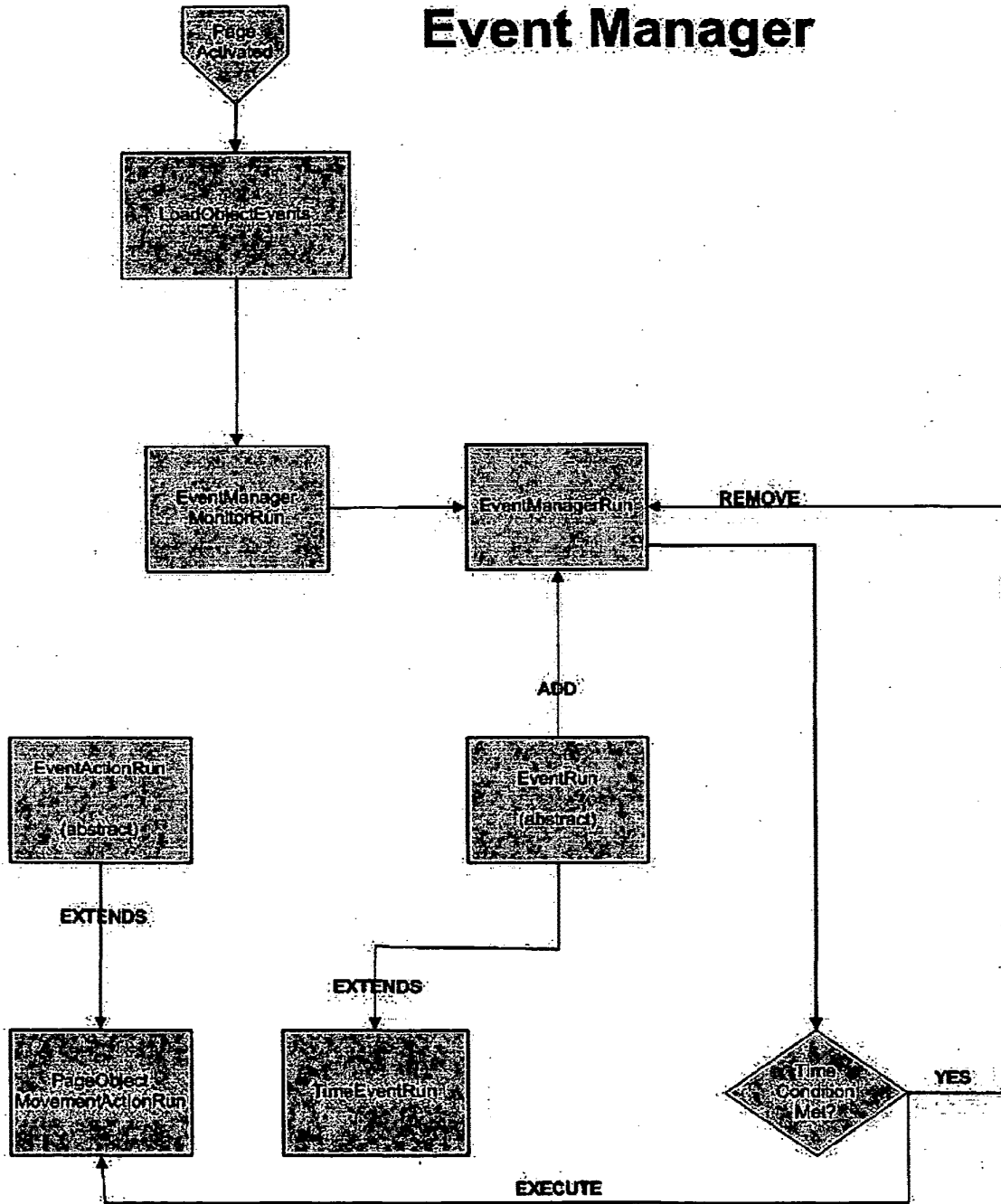
#### Numeric Key Binding Attributes

boolean[] hasNumericKeyBindings= new boolean[10]  
State of whether the numeric key, as defined by the array index,  
is bound.

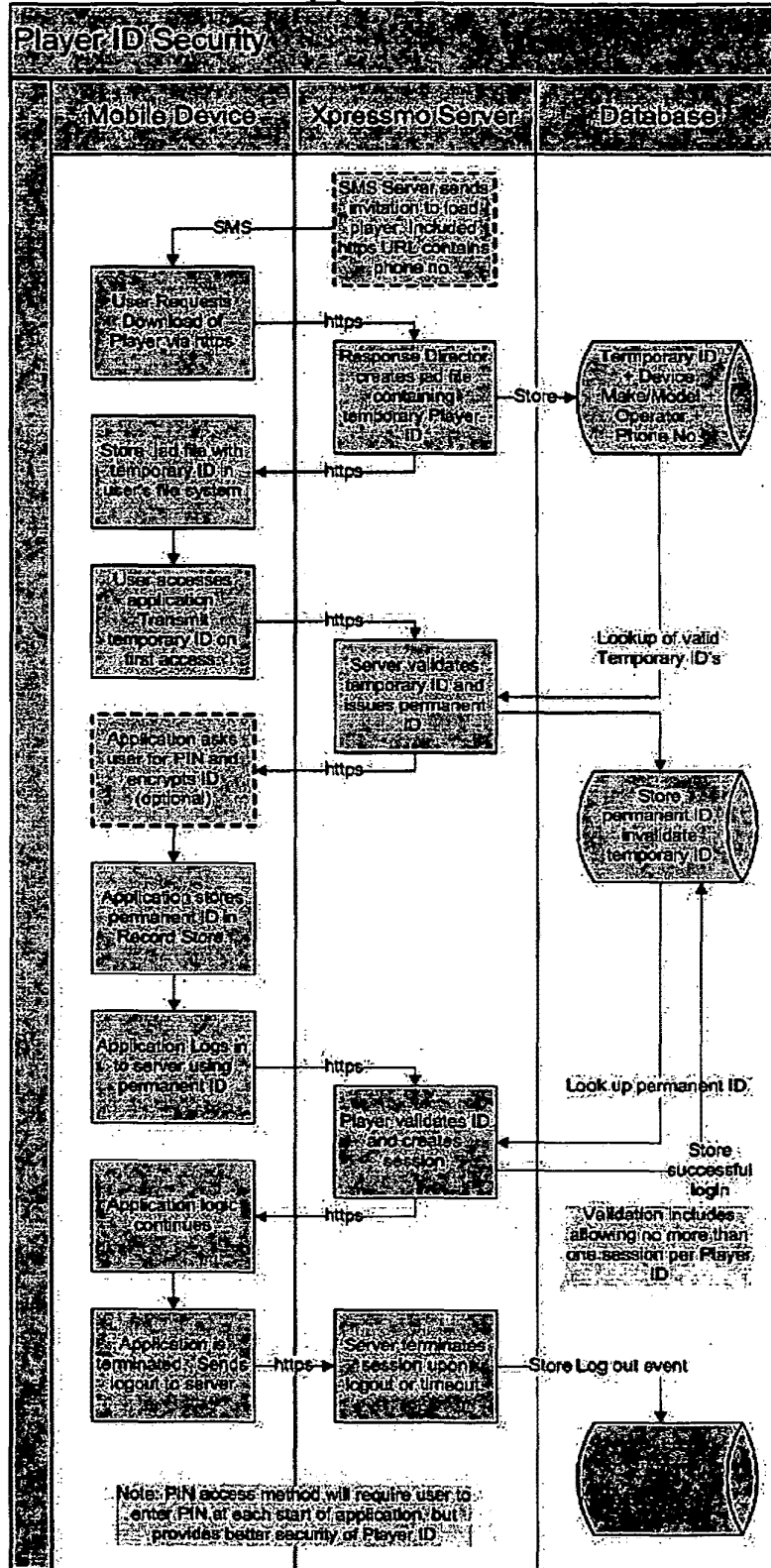
byte[] NumericKeyBindings= new byte[10]  
A byte pointer into the SoftKeyCommand() Vector to bind that key  
to a particular SoftKey Command.

## Appendix M

# Event Manager



### Appendix N





# Appendix O

## Popup Alert Specification

### A: Overview

There will be 3 classes of Alert Popup Pages, all which will have the same coordinates as regular pages but with a transparent background.

- 1: Application Specific
- 2: Author Specific
- 3: System Default

Alert Pages will essentially create a multi-layered drawing model for pages, which, in later implementations, could become a "windowing" Platform with overlapping windows.

There will be author-defined exit events to display an Alert Popup Page. It will be permissible for Alert Pages to call and display another Alert Page.

The Player will first look in the PDL Image Table for the Alert Popup page. If not found it will look at the Author Specific Alert Popup Page Library (much like an Xpressmo Server Page).

System Alerts, there will be a default System Alert Template Page available for all applications. This Page will have defined text objects for the actual test message that will be displayed based on the real-time condition that triggered the System Alert. The author will be able to add other objects (branding) as well as control the look of the Text Objects.

There will be three types of Popup Alerts.

- 1: Non-modal  
The Alert will draw. All functions for underlying page will be active.  
A non-modal alert can either be discarded by:
  - a: A timeout assigned to the Alert Page
  - b: An Exit Event
  - c: A backend API
- 2: Modal (User dismissible)  
The Alert will draw. All functions for underlying page will be disabled except for Exit application. A dimPage mask will be drawn over the underlying page.  
A User Dismissible Modal alert will be discarded on the first "Fire" Key Event.
- 3: Modal (Event Driven)  
The Alert will draw. All functions for underlying page will be disabled except for Exit application. A dimPage mask will be drawn over the underlying page.  
An Event Driven Modal alert can either be discarded by:
  - a: An Exit Event

## Systems and Methods for Presenting Information on Mobile Devices

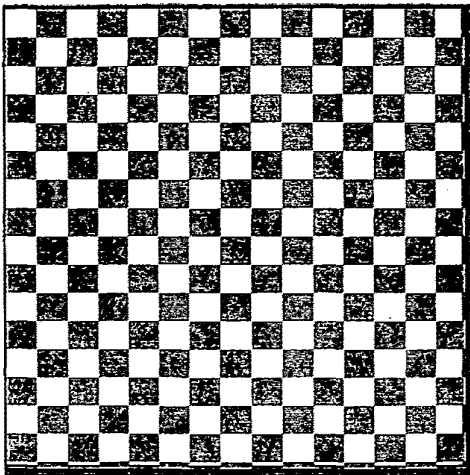
### b: A backend API

For the first version, “Fire” and Author –defined Soft Key Commands will be the only interactive events supported. They will apply to the entire Popup Alert object, and it will only be effective for Popup Alert Type 2. The Events and Animation Tabs of the Resource Director will not be available. No navigation within the Alert will be supported. Export and Import of Alerts from the Author and/or System Libraries will not be supported in this version.

Alert Pages of all three types can be displayed or disposed by backend API's.

### **B: Authoring UI**

When an Alert is created or selected the background will draw as transparent page.

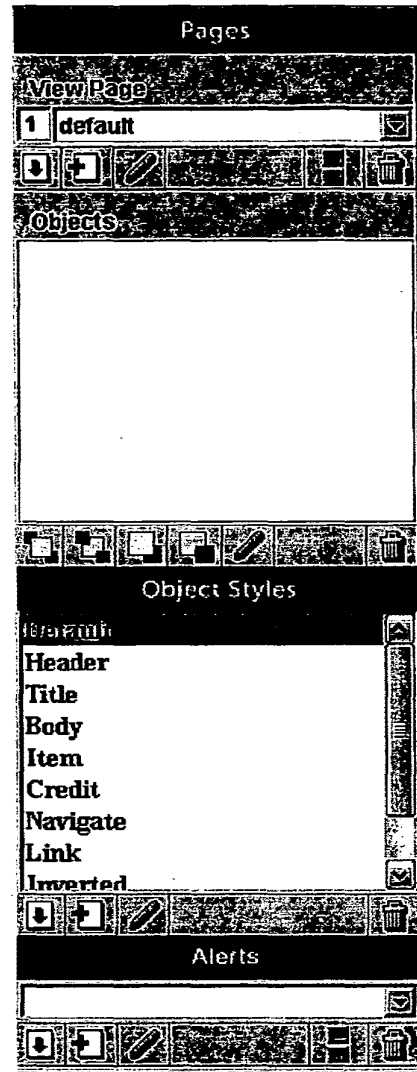


When an Alert is created, there will be a test to make sure that its name is unique, both in terms of other application specific alerts and/or application pages.

## Systems and Methods for Presenting Information on Mobile Devices

Alerts will appear as a choice under “Messages”, under the Objects-Messages... menu bar, and in the Layer Inspector as shown to the right.

On the Layer Inspector there will be a dropdown showing the current alerts defined for the application. It will work like the Pages dropdown.



When an Alert is selected, selection occurring in an identical manner as a Page, the Resource Inspector with a “Settings” tab available:

There will be a list with 3 possible selections.

- 1: Non-modal  
If selected there will be a timeout selection, including “stay”. Stay would mean that this non-modal alert can only be dismissed by:
  - a: A backend API
  - b: A “hide alert” Exit Event.Otherwise, if not yet dismissed, the timeout will dispose of the Alert.
- 2: Modal (User dismissible)
- 3: Modal (Event Driven)

## Systems and Methods for Presenting Information on Mobile Devices

There will also be the same choices that a Page has in terms of defining Soft Key Commands.

The Alert Object Name will be available for renaming, and, for convenience, the x,y location of the cursor will be displayed as well as the size of the underlying page.

The Objects that can be defined for an Alert will be Forms (non-interactive), Media, Shapes and/or Text. No child objects will be available. Overlapping Objects will be supported.

## Systems and Methods for Presenting Information on Mobile Devices

### **C: Internal Authoring Architecture**

There will be a Hashtable

```
public Hashtable<String, WebPage> Alerts_ = new Hashtable<String, WebPage>();
```

Alert pages will be appended to both the AKP and PDL files as page definitions in the normal way. When Creating, Importing and/or Opening files that may contain both Pages and Alerts, the objects will be placed in their respective Hashtables.

### **D: Invocation Methods**

- 1: By API
- 2: By Error Condition
- 3: By Exit Event
  - Slide
  - ListItem
  - Object

There will be 6 Exit Events that will relate to Alerts:

- a: ExitMode=15 (Goto Specific Internal Page & Display Alert)  
Invoked by a "Fire" Action.
- b: ExitMode=17 (Display Alert)  
Invoked by a "Fire" Action.
- c: ExitMode=19 (Hide Alert)  
Invoked by a "Fire" Action.
- d: ExitMode=20 (Goto an XMO Application & Display Alert)  
Invoked by a "Fire" Action.
- e: ExitMode=21 (Display Alert)  
Invoked by a "Selection" (i.e MouseOver) Action.
- f: ExitMode=22 (Hide Alert)  
Invoked by a "Selection" (i.e MouseOver) Action.

If any of these are selected, a dropdown menu will appear displaying the available Alerts for this application. In the first release it will only include those Alerts that are application specific.

## Systems and Methods for Presenting Information on Mobile Devices

### **E: Internal Player Architecture.**

WebPageRun will become usable for both Application Pages and Alerts.

Alerts, for the purposes of the Cache and Record Store, will be treated as Application WebPageRun objects, and will, other than the state of IntegerAttribute[6], have an identical PDL representation.

There will be a second WebPageRun Object defined in RunTimeEngine for an Alert Object. This object will normally be null, however, on the invocation of a Popup Alert, this object will be instantiated through the getPDLPage() method in RunTimeEngine.

The paint(Graphics g) method in WebPageRun will be modified to immediately call a new method drawThisPage(Graphics g). This should have no impact on the drawing of a regular page.

#### **Invocation of an Alert:**

RuntimeEvents or DynamicDataConnection will be the only 2 classes that can invoke an Alert.

They will perform the following tasks;

- 1: Instantiate the Alert WebPageRun Object in RuntimeEngine with getPDLPage()
- 2: Set the int TypeOfAlert in RunTimeEngine to the type of Alert Exit Event.
- 3: Set the necessary states based on the exit type. This will affect:
  - keyPressed() in WebPageRun
  - pointerPressed() in WebPageRun
  - drawThisPage(Graphics g) in WebPageRun
- 4: Complete the "GotoPage" or "Goto Application" operations if necessary.
- 5: If the Alert is Modal then switch the Command action Listener to the soft key commands associated with the Alert.
- 6: Set Engine.EngineBoolean[15] to true.

#### **Drawing an Alert:**

The code in DrawThisPage(Graphics g) will have the same test on Engine.EngineBoolean[15]

If Engine.EngineBoolean[15]=true, and Engine.Alert\_!=null, then:

- 1: Draw DimPage if the Alert is Modal
- 2: Pass the graphics context to Engine.Alert\_ with:  
"Engine.Alert.drawThisPage(g);

#### **Discarding an Alert:**

- 1: Engine.EngineBoolean[15]=false
- 2: Discard all Media Objects and close all related Alert\_ threads.
- 3: Set Engine.Alert\_=null.

Systems and Methods for Presenting Information on Mobile Devices

**Appendix A: PDL Specification for Popup Alerts**

Page Definition

IntegerAttribute[6] (Page Type)

0=Normal Application Page (resides in the Apps PDL)

1=Alert Page Non-Modal

2=Alert Page Non-Modal with Timeout

3= Alert Page Modal

4= Alert Page Modal, Fire Enabled

**Primitive Object Definitions**

ObjectIntegerAttribute[12] (Exit Event)

15= Goto a specific Internal Web Page with Alert

Invoked by a "Fire" Action.

17=Generate Alert

Invoked by a "Fire" Action.

19=Hide Alert

Invoked by a "Fire" Action.

20= Goto an XMO Application with Alert

Invoked by a "Fire" Action.

21=Generate Alert

Invoked by a "Selection" Action.

22=Hide Alert

Invoked by a "Selection" Action.

ObjectString[3]

Alert Popup Page URL for:

ExitMode=15, 17, 19 ,20, 21 and 22.

**List Item Definitions**

ListItemExitEvent

15= Goto a specific Internal Web Page with Alert

Invoked by a "Fire" Action.

17=Generate Alert

Invoked by a "Fire" Action.

19=Hide Alert

Invoked by a "Fire" Action.

20= Goto an XMO Application with Alert

Invoked by a "Fire" Action.

21=Generate Alert

Invoked by a "Selection" Action.

22=Hide Alert

Invoked by a "Selection" Action.

ListItemSlideShowName

Alert Popup Page URL for:

ExitMode=15, 17, 19 ,20, 21 and 22.

## Systems and Methods for Presenting Information on Mobile Devices

### **Slide Show Slide Definitions**

SlideIntegerAttribute[6] Exit Slide Mode

10= Goto a specific Internal Web Page with Alert

    Invoked by a "Fire" Action.

11=Generate Alert

    Invoked by a "Fire" Action.

12=Hide Alert

    Invoked by a "Fire" Action.

13= Goto an XMO Application with Alert

    Invoked by a "Fire" Action.

14=Generate Alert

    Invoked by a "Selection" Action.

15=Hide Alert

    Invoked by a "Selection" Action.

SlideString[4]

    Alert Popup Page URL for Slide Exit Mode= 10 thru 15.

### **RunTimeEngine Definitions**

protected WebPageRun Alert\_ =null;

protected int TypeOfAlert (may not be required);



## Appendix P

# Publisher 2.0 Server File System Structure

### Overview

When installed, the Publisher 2.0 system creates files under three different roots:

1. Under the Apache HTTP server folder, typically at **C:\Program Files\Apache Group\Apache2**.
2. Under the Tomcat server folder, typically at **C:\Program Files\Apache Software Foundation\Tomcat 5.0**.
3. Under the Publisher server folder, typically at **C:\Program Files\Publisher2.0**.

The files created or modified in the Apache HTTP server folder are minimal. In fact, the only files that are modified are configurations files:

- httpd.conf** – the main configuration file for Apache HTTP server
- mime.types** – the mapping of file extensions to MIME types.

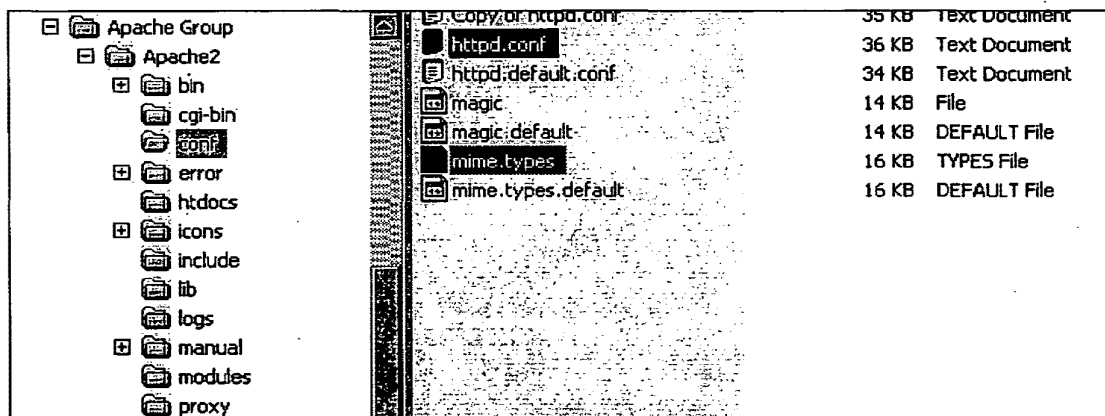


Figure 1 - Files modified under the Apache HTTP Server folder (highlighted)

The files created or modified in the Tomcat server folder are more extensive. These will be covered below.

The bulk of the installed files reside under the **Publisher2.0** folder. The installer creates an environment variable with this folder's absolute path, so that at most locations, on a command line or in Explorer, one can type **%publisher\_home%** to refer to this folder.

### Publisher Root Folder

The publisher root folder is typically at **C:\Program Files\Publisher2.0**. On 64-bit Windows 2003 R2, this is at **C:\Program Files (x86)\Publisher2.0**. It may be different if the install folder was modified during installation, or if your system's "Program Files"

## Systems and Methods for Presenting Information on Mobile Devices

folder is on a different drive. In any case, the environment variable "publisher\_home" is always set to the folder regardless of location, so you can refer to this folder as `%publisher_home%`.

The root folder contains the following subfolders. Immediately after install time, not all of these may be present. Some of these folders are created by the processes that are run as part of the publisher server.

## Systems and Methods for Presenting Information on Mobile Devices



### **Tilde (~) folders**

Notice that there are several folders that begin with the “~” (tilde) character. These folders are user (author) accounts. Each one of these accounts was created by the Publisher Registration utility. Each one contains the author’s workspace for Publisher 2.0. There are as many tilde folders as there are Publisher accounts. The details of these accounts are discussed later.

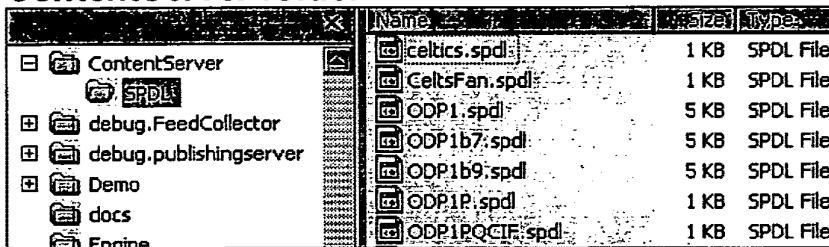
### **config folder**

Folder “config” contains the properties files for Publisher Server. PublisherServer.properties contains configurable parameters for such items as: Streaming Server, Digital Certificates. PublishingServerError.properties is empty right now, but it is for storing error message text strings.

Name	Size	Type
config		
ContentServer		
PublishingServer.properties	2 KB	PROPERTIES File
PublishingServerError.properties	1 KB	PROPERTIES File

Systems and Methods for Presenting Information on Mobile Devices

**ContentServer folder**



This folder contains one subfolder “SPDL”, which contains “.spdl” files. “SPDL” stands for “Server-side PDL”, and “PDL” is the term used for the storage of application data in the Publisher system. SPDL files contain information for Dynamic Data Binding. For each application that contains Data Binding, there will be a corresponding .spdl file in this folder.

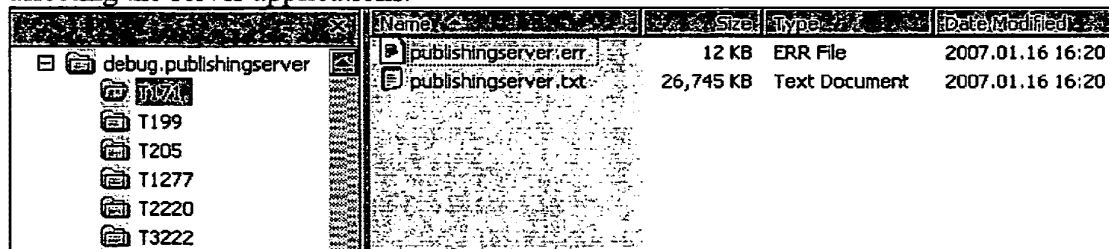
*NOTE: Currently, there is a restriction that the application name of any application using dynamic binding must be unique across all users, as the .spdl from all users and all applications are stored here. This restriction will be lifted in the near future, when the .spdl files will be converted into a database table.*

**debug.\* folders**

Notice that there are two folders in this example that begin with “debug.”. These folders contain snapshots of the trace and error log file at the time of a system exception. These are used by xpressmo Development to diagnose and debug the system. The name following “debug.” is the name of the batch (.bat or .cmd) file that invoked the application that is being monitored. The possibilities are:

publishingserver	Publisher Server
publishingclient	Publisher Client program
registration	Registration Utility
rssfeeder	RSS Feeder program
FeedCollector	Feed Collector program

If an error never occurred in any of these programs, the “debug.” folder may not exist. These “debug.” folders are not removed during uninstall and the update installation process, and will accumulate indefinitely. They can be manually removed without affecting the server applications.



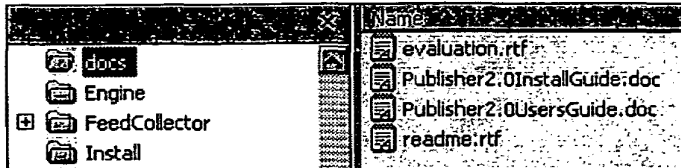
Within the debug folders are subfolders beginning with the letter “T” followed by 1 to 4 digits. The digits are generated randomly and do not have significance, other than to provide a unique name to the “T” folders. It is best to sort the debug folder by date of last modification so that the latest files are either at the top or bottom of the list.

## Systems and Methods for Presenting Information on Mobile Devices

Within the “T” folder are 2 files which are used for diagnostic purposes. For the publishingserver

### **Docs folder**

The “docs” folder contains static documents that placed here by the installer.



- **evaluation.rtf** is the license agreement that is displayed in the installer.
- **Publisher2.0InstallGuide.doc** is the installation document covering both the

server and client installations. This document also provides information on the essential parameters that need to be set up in property files etc.

- **Publisher2.0UsersGuide.doc** is the Users Guide to Publisher tool. *NOTE: the content of this document is antiquated. It was last updated in 2005.12.*
- **readme.rtf** is the readme file that is displayed during installation.

### **Demo folder**

This folder is not in active use at this time. It is used to contain demo applications, so that the URL to a Publisher-generated application is not long, such as

<http://xpresmo.com/Demo/foo.jad> rather than

<http://xpresmo.com/~user/WebSites/foo/bin/foo.jad>.

### **Engine folder**

This folder contains the executable components for the generic (PC) version of the player.

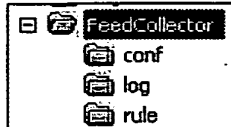
Name	Size	Type
akessAnimationrun.jar	23 KB	Executable Jar File
akessAPI.jar	52 KB	Executable Jar File
Akessformrun.jar	22 KB	Executable Jar File
Akessgisrun.jar	51 KB	Executable Jar File
Akesslibrun.jar	8 KB	Executable Jar File
akessphlogrun.jar	8 KB	Executable Jar File
akessrun.jar	35 KB	Executable Jar File
Akessruninit.jar	16 KB	Executable Jar File
akessserver.jar	19 KB	Executable Jar File
akesssliderun.jar	25 KB	Executable Jar File
Akessvectorrun.jar	22 KB	Executable Jar File
Akessvrrun.jar	21 KB	Executable Jar File
dimage.png	2 KB	PNG Image
tabbottom.png	1 KB	PNG Image
tableft.png	1 KB	PNG Image
tabright.png	1 KB	PNG Image
tabtop.png	1 KB	PNG Image

All executable .jar files in this folder are code-signed using Verisign Digital Certificates. These files are downloaded to the browser when the PC player is invoked via an html file created for a specific application.

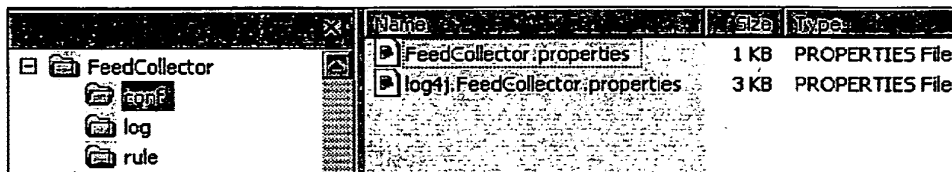
## Systems and Methods for Presenting Information on Mobile Devices

### **FeedCollector folder**

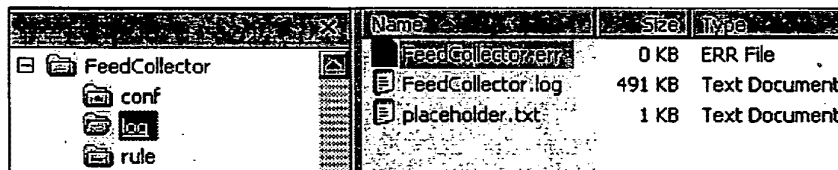
This folder contains the data and configuration files for the FeedCollector program.



- conf contains configuration files
- log contains log files
- rule contains the parse rules



- **FeedCollector.properties** contains modifiable parameters for the FeedCollector program, such as the default scan interval.
- **log4j.FeedCollector.properties** contains the logger parameters.



- **FeedCollector.err** contains error trace information in case FeedCollector has a system exception.
- **FeedCollector.log** contains the log, including debug information, generated by FeedCollector.
- **placeholder.txt** is only used during installation and is not required thereafter. It may be deleted if desired, with no affect to the applications.



- **\_ATOM\_.xml** is the default parse rule for reading ATOM format feeds
- **\_RSS\_.xml** is the default parse rule for reading RSS format feeds
- The other xml files are custom rule files for specific applications. For instance, **newtrailers.xml** is a custom rule file for a FeedCollector channel, which has the name "newtrailers". See the FeedCollector documentation for how to create custom parse rules.

### **Install folder**

This folder contains, files used during the installation and uninstall processes, as well as some utilities for debugging.

## Systems and Methods for Presenting Information on Mobile Devices

Name	Size	Type
pskill.exe	92 KB	Application
pslist.exe	84 KB	Application
tail.exe	291 KB	Application
publishingclient.ba-	2 KB	BA- File
publishingserver.ba-	2 KB	BA- File
registration.ba-	1 KB	BA- File
test.html	1 KB	HTML Document
Feedcollector.ico	25 KB	Icon
publishingclient.ico	25 KB	Icon
publishingserver.ico	25 KB	Icon
registration.ico	25 KB	Icon
akirausers.sql	8 KB	SQL File
appvariable.sql	2 KB	SQL File
FeedCollector.sql	16 KB	SQL File
Grant.sql	1 KB	SQL File
phlogdb.sql	2 KB	SQL File
Player.sql	3 KB	SQL File
prospects.sql	1 KB	SQL File
rssdb.sql	7 KB	SQL File
exturl.txt	1 KB	Text Document
CreateDB.bat	4 KB	Windows Batch File
FeedCollector.bat	1 KB	Windows Batch File
Kill_Client.bat	1 KB	Windows Batch File
Publisher2.0Setup2.bat	9 KB	Windows Batch File
Publisher2.0Uninstall.bat	11 KB	Windows Batch File
rss.bat	1 KB	Windows Batch File
SetExturl.bat	5 KB	Windows Batch File
x64.zip	10 KB	WinZip File

**pskill.exe** is a utility to kill processes. It is used by the uninstall process.

**pslist.exe** is a utility to list processes. It is here as a debugging tool. It is not used by any process.

**tail.exe** is a utility to list out the tail end of a text file. It is a debugging tool, useful for monitoring log files in real time. It is not used by any process.

The “\*.ba-“ files are batch file templates for the respective “.bat” files. The batch files generated using these “.ba-“ files are placed in the Publisher root folder.

**test.html** is for debugging. It is used to test the accessibility of files under the Publisher folder via http.

The “\*.ico” files are the icon files for the respective applications, corresponding to the batch file name which invoke the application.

The “\*.sql” files are MySQL scripts to generate the respectively named databases, except for “Grant.sql, which does not create a database but sets up the access rights for each of the databases.

**exturl.txt** contains that last setting of the server IP address and the Build ID. This file persists between an uninstall and update install cycle.

The \*.bat files are either files used during installation or uninstallation, or are the batch files to start the installed applications:

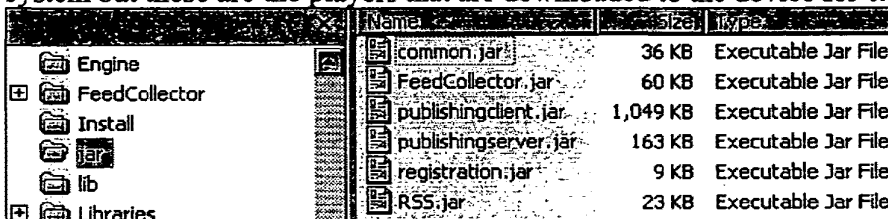
## Systems and Methods for Presenting Information on Mobile Devices

- Used during install: **CreateDB.bat, Publisher2.0Setup2.bat, SetExturl.bat**
- Accessible after install: **FeedCollector.bat, Kill\_Client.bat, rss.bat, SetExturl.bat**
- Used for Uninstall: **Publisher2.0Uninstall.bat**

**x64.zip** (which starts out as **x64.zi-** and is renamed **x64.zip** during install process, if needed) contains a set of properties and configuration files that replace the default properties and configuration files, when the server device is running 64-bit Windows 2003 R2. This is required because the default “Program Files” folder for 64-bit Windows is “Program Files (x86)” for applications written for 32-bit Windows. The files in this zip file substitute all occurrences of “Program Files” for “Program Files (x86)” in the properties and configuration files.

### **jar folder**

This folder contains all of the .jar executable files that are part of the Publisher server system, excluding those that are installed under Tomcat. There are other .jar files in the system but these are the players that are downloaded to the device for execution there.

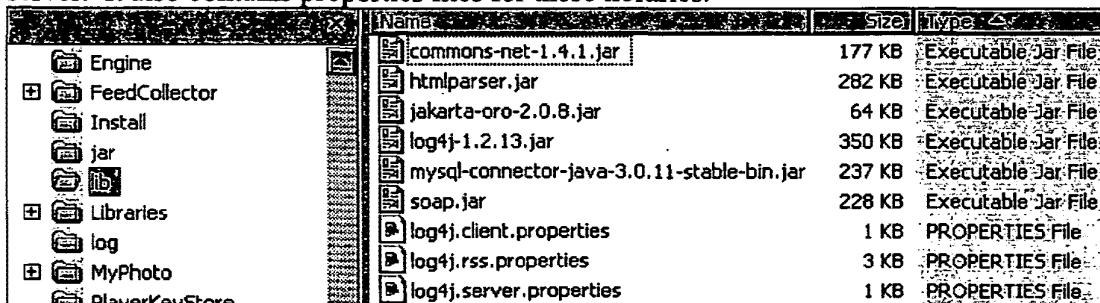


Name	Size	Type
common.jar	36 KB	Executable Jar File
FeedCollector.jar	60 KB	Executable Jar File
publishingclient.jar	1,049 KB	Executable Jar File
publishingserver.jar	163 KB	Executable Jar File
registration.jar	9 KB	Executable Jar File
RSS.jar	23 KB	Executable Jar File

- **common.jar** contains common classes among the various applications.
- **FeedCollector.jar** is the main executable for the FeedCollector program
- **publishingclient.jar** is the Publisher client application. This is installed both in the server and client installations.
- **publishingserver.jar** is the server side component of the Publisher tool.
- **registration.jar** is the Publisher user account registration utility.
- **RSS.jar** is the RSS feeder program. This is the predecessor of FeedCollector but is kept alive for compatibility with older applications that may refer to its database of RSS feeds.

### **lib folder**

This folder contains 3<sup>rd</sup> party libraries used by Publisher and other applications on the server. It also contains properties files for these libraries.



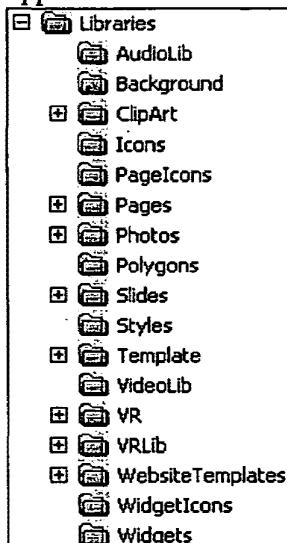
Name	Size	Type
commons-net-1.4.1.jar	177 KB	Executable Jar File
httparser.jar	282 KB	Executable Jar File
jakarta-oro-2.0.8.jar	64 KB	Executable Jar File
log4j-1.2.13.jar	350 KB	Executable Jar File
mysql-connector-java-3.0.11-stable-bin.jar	237 KB	Executable Jar File
soap.jar	228 KB	Executable Jar File
log4j.client.properties	1 KB	PROPERTIES File
log4j.rss.properties	3 KB	PROPERTIES File
log4j.server.properties	1 KB	PROPERTIES File



## Systems and Methods for Presenting Information on Mobile Devices

### **Libraries folder**

This folder contains multimedia resources that are available to all users and all applications.



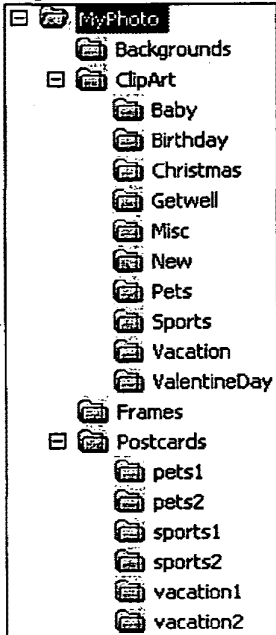
- **AudioLib** contains audio files that are accessible in Publisher
- **Background** contains background images
- **ClipArt** contains clip art, all .gif file.
- **Icons** contains icons that can be used in list boxes
- **PageIcons** contains icons for pages
- **Pages** contains pre-made pages that can be inserted into an application inside Publisher
- **Photos** contain stock photo image files all .jpg format. It is divided into the following categories, each in their own subfolders: Animals, Business, Food, Holiday, Landscapes, Misc, Nature, Sports, and Vehicles. Each photo has 4 sizes defined: full size (no suffix), medium (suffix m), small (suffix sm), and thumbnail (suffix th). E.g. if the full size image name is Eagle.jpg, the others are Eaglem.jpg, Eaglesm.jpg, and Eagleth.jpg, respectively.
- **Polygons** contains vector graphics stored in the proprietary .avg format, a subset of the PDL format.
- **Slides** contains pre-made slide shows. Each slide show consists of a proprietary .avd format file, and a folder containing the slide images.
- **Template** contains the default template **default.tpp**, plus any additional application templates
- **VideoLib** contains video files
- **VR** contains Virtual Room samples
- **VRLib** contains images that can be used in a Virtual Room.
- **WebsiteTemplates** contain pre-made applications that can be used as a starting point for new applications.
- **WidgetIcons** contains icons to represent widgets.
- **Widget** contains the same thing as WidgetIcons

### **log folder**

This folder holds the log file **AkiraLogger.log** for the RSS feeder application. The RSS feeder is slated for removal, as its functionality is superseded by FeedCollector, so this folder will not be used in the near future.

## Systems and Methods for Presenting Information on Mobile Devices

### **MyPhoto folder**



This folder contains additional stock photos and clipart for composing photo postcards etc.

The structure of the subfolders is shown on the left.

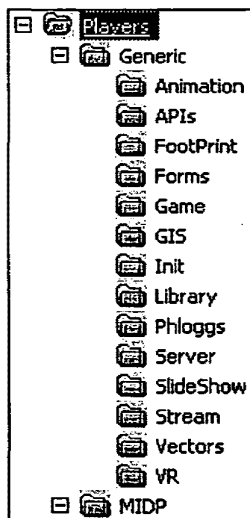
- **Backgrounds** contains .gif files.
- **ClipArt** contains subfolders, each contains .gif files.
- **Frames** contains .gif files.
- **Postcards** contains .gif files as well as subfolders, each also containing .gif files.

### **PlayerKeystore folder**

This folder contains the keystore which contains the certificate used for digitally signing player .jar files. It contains one file, **player.sks**, which contains the certificate chain from Express Mobile to Verisign.

The keystore must reside in this folder. However, the keystore file name and its contents can be modified. The parameters which control the keystore filename, keystore password, certificate alias and certificate password used by Publisher Server are stored in the PublishingServer.properties file.

### **Players folder**



This folder contains the .jar files for generating the generic/PDA player as well as the MIDP player. Each subfolder under Generic contains a single .jar file component of the player. The MIDP player subfolder has the same structure as the Generic folder and so is omitted from the image on the left.

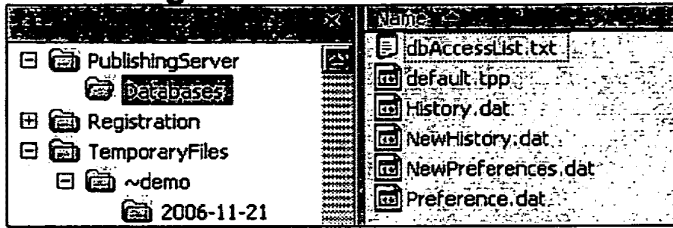
In addition, for MIDP only, the MIDP folder under Players contains two "universal" players: **UniversalPlayer.jar** and **UniversalPlayerNF.jar**. The latter is a subset of the former, with file system and PIM accessing code removed. The "NF" suffix stands for "No File".

## Systems and Methods for Presenting Information on Mobile Devices

### **Publisher folder**

This folder is not used in the present implementation.

### **PublishingServer folder**



This folder contains one subfolder, **Databases**, which in turn contains files that are copied into the user workspaces when the workspaces are created by the Registration utility.

dbAccessList.txt is the default list of databases that are accessible by a user. It is a text file containing parameters for the databases.

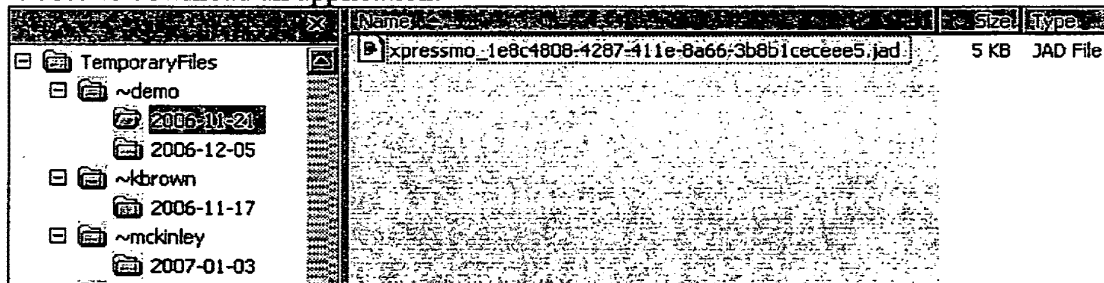
- **default.tpp** is the default template which must be present in the user workspaces for Publisher to start up.
- **History.dat** is the legacy default history (MRUD) list file.
- **NewHistory.dat** is the current default history (MRUD) list file.
- **NewPreferences.dat** is the current default Publisher user preferences file.
- **Preference.dat** is the legacy default Publisher user preferences file.

### **Registration folder**

This folder contains one subfolder, **icons**, which contain icons and images used by the Registration utility. There is also an html file, **register.html** which is not used.

### **TemporaryFiles folder**

This folder does not exist after a clean install but is created when the Response Director is used to download an application.

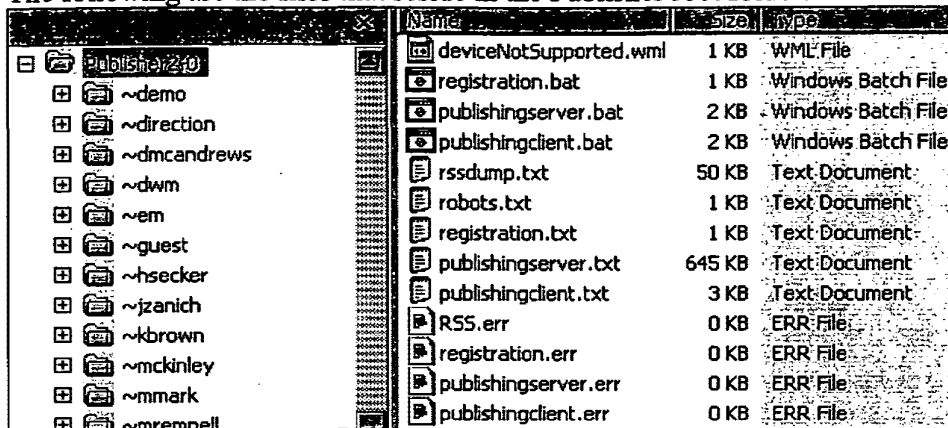


Response Director generates .jad files with properties that are specific to an individual device. It uses this folder to create this .jad file from one of the .jad files that is generated by Publisher. The structure of this folder is thus:

- The first level indicates the user account from which the application was generated.
- The second level is the date on which the .jad files were generated. All .jad files generated for a specific application are placed in the same subfolder for that date.
- The .jad file has the name of the application, followed by an underscore, followed by the player id, a UUID in 8-4-4-4-12 format, which was generated for this particular request for a player.

### **Publisher root folder files**

The following are the files that reside in the Publisher root folder.



Name	Size	Type
deviceNotSupported.wml	1 KB	WML File
registration.bat	1 KB	Windows Batch File
publishingserver.bat	2 KB	Windows Batch File
publishingclient.bat	2 KB	Windows Batch File
rssdump.txt	50 KB	Text Document
robots.txt	1 KB	Text Document
registration.txt	1 KB	Text Document
publishingserver.txt	645 KB	Text Document
publishingclient.txt	3 KB	Text Document
RSS.err	0 KB	ERR File
registration.err	0 KB	ERR File
publishingserver.err	0 KB	ERR File
publishingclient.err	0 KB	ERR File

- **deviceNotSupported.wml** is used by the Response Director. It is sent in place of a download when Response Director determines that the device requesting the application cannot accept any of the existing variants of the application executables. For instance, this may happen if the device only supports MIDP1.0, because the J2ME players require MIDP 2.0.
- **registration.bat**, **publishingserver.bat**, and **publishingclient.bat** start up their respective java applications. These files are generated by running SetExturl.bat. They are not part of the original installed files.
- **rssdump.txt** is a periodic (every 10 minute) text dump of the RSS database. This is not part of the standard install, and needs separate scheduling.
- **robots.txt** is a file to inform spiders and robots to not scan this and all subfolders.
- **registration.txt**, **publishingserver.txt**, and **publishingclient.txt** are the log files for their respective applications. A snapshot of this file is copied to the debug.xxx folder (where xxx is either *registration*, *publishingserver* or *publishingclient*) when a system exception occurs. These are cleared every time the application is restarted.
- **registration.err**, **publishingserver.err**, and **publishingclient.err** are the error logs for their respective applications. A snapshot of this file is copied to the debug.xxx folder (where xxx is either *registration*, *publishingserver* or *publishingclient*) when a system exception occurs. These are cleared every time the application is restarted.

### **Tilde folders revisited**

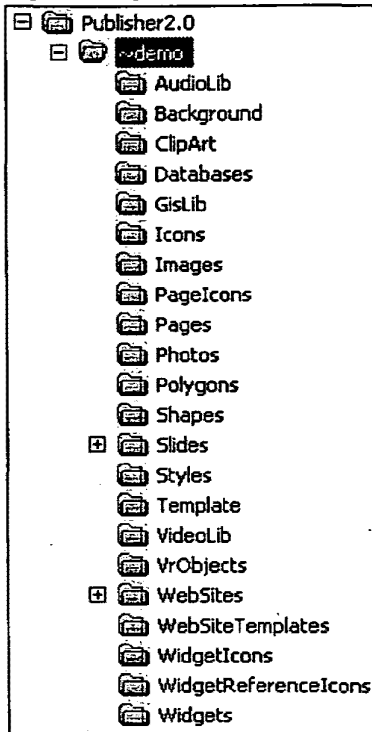
Each folder starting with a tilde character contains the Publisher workspace for a single Publisher account. For instance, if a Publisher user was registered as JSmith, then the workspace folder is ~JSmith.

All workspace folders have identical structure.

Many of the subfolders of the workspace are repositories of media which can be accessed by all applications in the workspace. The repositories are automatically populated

## Systems and Methods for Presenting Information on Mobile Devices

whenever a media file is uploaded to the server from the user's hard disk. This avoids repeated uploads for media files that are used frequently.



- **AudioLib** may contain audio files that were imported from the user's hard drive.
- **Background** may contain background images that are shared among the user's applications.
- **ClipArt** may contain clip art that are shared among the user's applications.



- **Databases** contains the database schema for the databases that are accessible by this user. Initially, the file **dbAccessList.txt** contains a default set of databases. From this list are generated **\*.db** files, which are schema definitions that

can be read by Publisher. If a user wishes to add or remove databases that are available for Dynamic Binding, he can modify **dbAccessList.txt**. The next time Publisher client is started, the **\*.db** files will be regenerated.

- **Images** may contain image files that can be shared among the user's applications.
- **Pages** may contain pages that were used by earlier applications that can now be shared among all user's applications.
- **Polygons** may contain polygon files which can be shared among the user's applications.
- **Slides** may contain slide shows that can be shared among the user's applications
- **VideoLib** may contain video files which were uploaded, which can be shared among the user's applications.

**VrObjects** may contain Virtual Room objects which can be shared among the user's applications.

**WebSites** contains the applications generated by using Publisher tool. Please see detail explanation below.

**WebsiteTemplates** may contain templates that can be shared among user's applications.

### **WebSites folder**

Here is a typical user account's WebSites folder.

Name	Size	Type
VideoLib		File Folder
VrObjects		File Folder
WebSites		File Folder
ODP1		File Folder
WebSiteTemplates		File Folder
History.dat	1 KB	DAT File
ODP1.akp	155 KB	AKP File
ODP1.spdl	5 KB	SPDL File
Preference.dat	1 KB	DAT File

## Systems and Methods for Presenting Information on Mobile Devices

A WebSites folder contains the files **History.dat** and **Preference.dat**, for legacy versions, and **NewHistory.dat** and **NewPreferences.dat** for newer systems. All 4 files may co-exist if both a legacy Publisher or new Publisher were used on the same account.

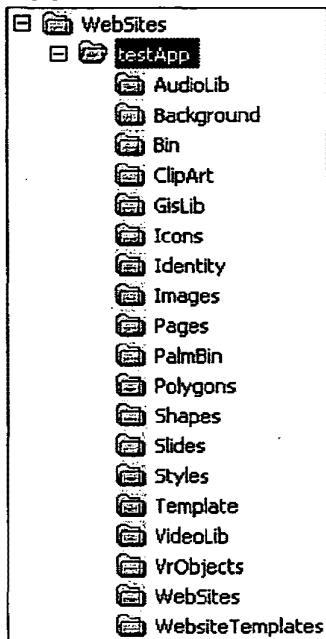
In this example, the WebSites folder contains one application named "ODP1". For each application, the following file and folder will be present (substitute applicationName with the name of the actual application):

- **applicationName.akp** – This is the application file format used by Publisher
- **applicationName folder** – This contains the resources required by the application.

The following files may be present:

- **applicationName.spdl** – This file is present if the application has Dynamic Data Binding. This file contains the bindings of an attribute within the application to a database. This file is copied to **ContentServer/SPDL** folder each time the application is saved.
- **applicationName.akp.bak** – (not shown in above illustration) This is a backup of the previous version of the .akp file, and will be present if the file was saved more than once. This allows a one-level restore of the application by manually copying the .akp.bak file into the .akp file of the same name.

### Application folder within WebSites folder



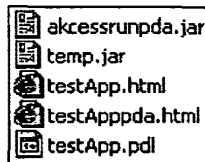
As mentioned above, each application has a folder under WebSites which contains the resource files required for the application. The structure of the folder is shown on the left. It is very similar to the user workspace folder structure. Each of the resource folders, e.g. AudioLib, are exclusive to the application.



The **Bin** folder is one folder not available at the user workspace level. It holds the executable files for J2ME MIDP player. There may be multiple versions of both the jar files (which are the binary java executable files) and the jad files (which are property files describing the environment under which the jar files run). For instance, in this example, there are 2 versions of jar and 4 versions of jad file.

The **Identity** folder contains two image files that can be replaced with suitable versions for the application. These are the **MIDPSplashLogo.png** which shows up in the splash screen, currently the xpressmo logo, and the **playericon.png**, which is the icon that shows up on the menus of some phones

to represent the application. If these are modified, the next time Publisher saves the application these will be included in the player built for this application.

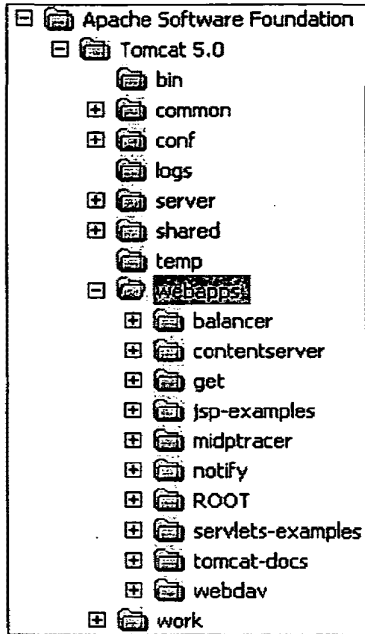


The application folder itself also contains files.

- **akessrunpda.jar** is the PDA version of the player.
- **temp.jar** is a temporary file used during the generation of MIDP jar files.
- **applicationName.html** is the loader for the PC browser version of the player.
- **applicationNamepda.html** is the loader for the PDA version of the player.
- **applicationName.pdl** is the application logic that is common to all players.

## Files under Apache Tomcat

The server installer installs four servlets under Tomcat's **webapps** folder:



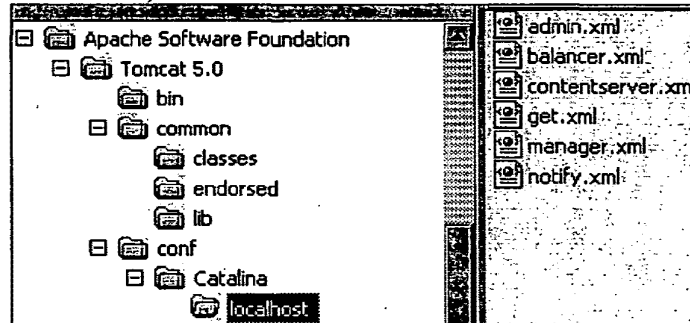
**contentserver** contains the Content Server, the main player runtime server.

**get** contains the Response Director.

**midptracer** contains a utility to log trace statements from the MIDP player into individual log files per player ID. These log files are created in Tomcat's **logs** folder.

**notify** handles download notifications from the MIDP player and can be considered part of the Response Director.

Besides these locations, there are configuration files in the **conf** folder, as shown below.

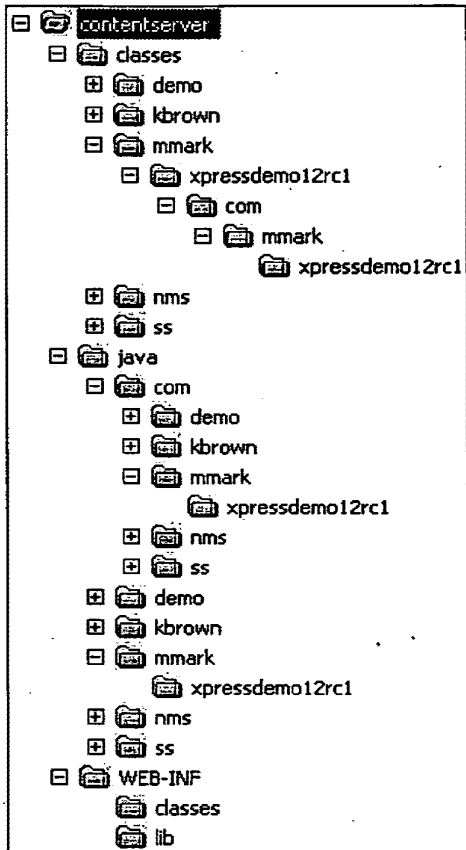


The 3 configurations files installed by Publisher are: **contentserver.xml**, **get.xml**, and **notify.xml**.



## Systems and Methods for Presenting Information on Mobile Devices

### **contentserver folder**



The structure is shown to the left.

This contains not only the runtime binaries for content server but also serve as a staging area for the back end code.

The **classes** and **java** folders under contentserver are used for compiling and running the back end code. In this illustration, user **mmark** had created back end code for application **xpressdemo12rc1**. The **classes** folder holds the compiled classes generated from the java code that is uploaded to the server. The convention is that each application is in a package named *com.username.applicationname*. You can see this pattern in both the **classes** folder and **java** folder. Both **classes** and **java** folders are managed by the Content Management System. This provides UI to upload application back end code, verifies that it compiles, and then dynamically hooks the back end code into the content server.

Both **classes** and **java** are staging areas. The master copy of the source code is kept in a separate database table. When Tomcat is started, all applications back end code is reloaded from the database. If your system has many applications with back end code, Tomcat may take some time to start up, because each of the back end application code must be compiled and

loaded.



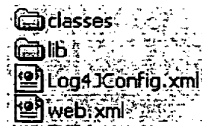
contentserver also contains files.

**build.xml** is an ant script used to compile the backend java code.

The rest of the files (**design.css**, **header.jsp**, **Main\_login.jsp**, **ViewApp.jsp**, and **ViewFile.jsp**) are part of the Content Management System web UI.

### **WEB-INF folder**

The code for content server proper resides under the **WEB-INF** folder.



It contains two folders and two configurations files.

**Log4JConfig.xml** contains configuration parameters for the logger.

This determines the format of the trace statements.

**web.xml** configures the web interface for the Content Management

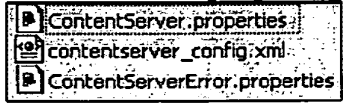
System.

## Systems and Methods for Presenting Information on Mobile Devices

### **classes folder under WEB-INF**

This contains additional configuration and properties files.

**ContentServer.properties** contains configurable properties for the content server.



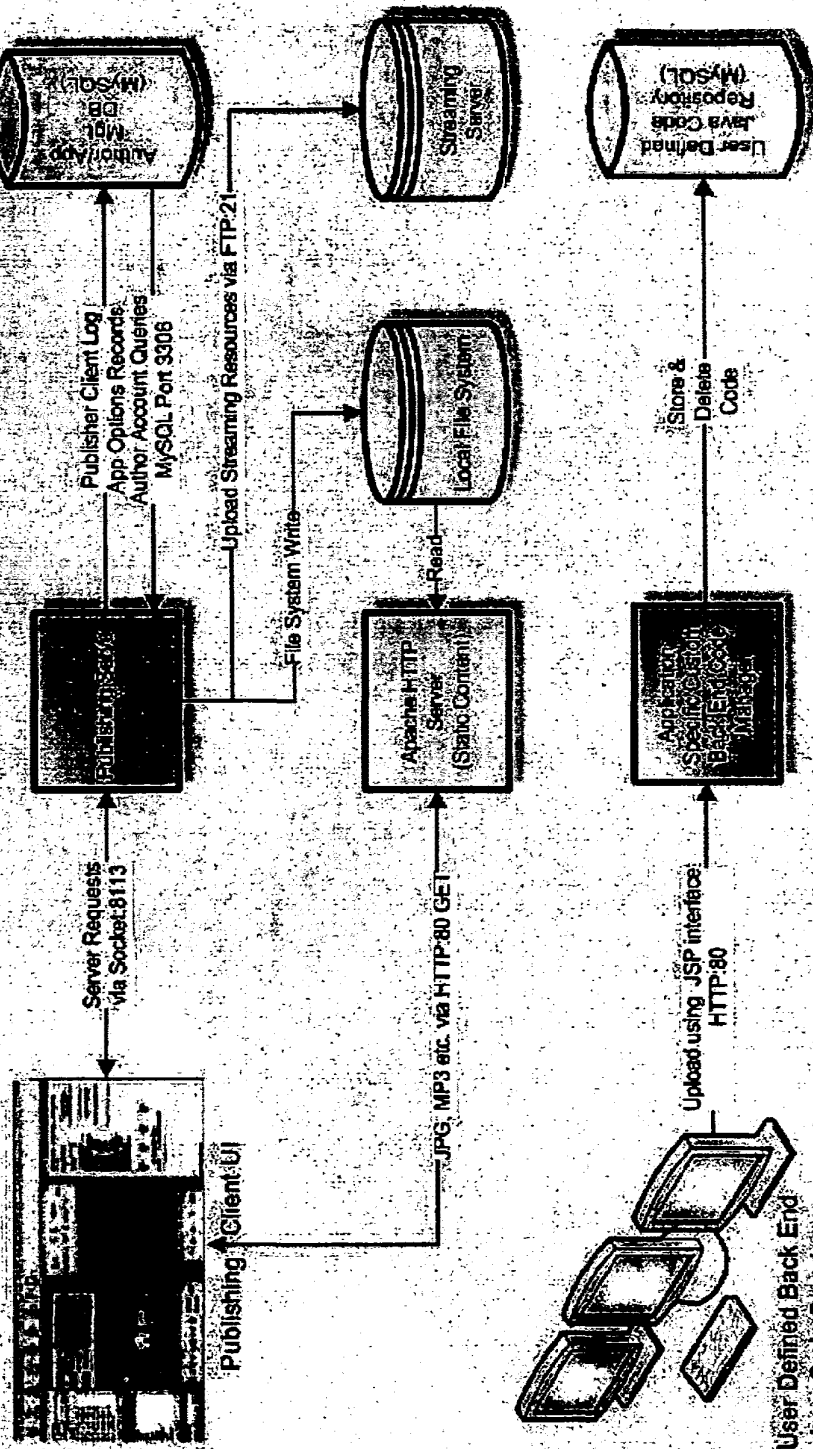
**contentserver\_config.xml** defines the pages and actions in a struts-like structure. There are no user configurable items here. This file is modified, for instance, if a new state is added in the player-content server communications.

**ContentServerError.properties** contains message strings.

### **lib folder under WEB-INF**

This folder contains all the binary .jar files that content server AND the back end code uses.

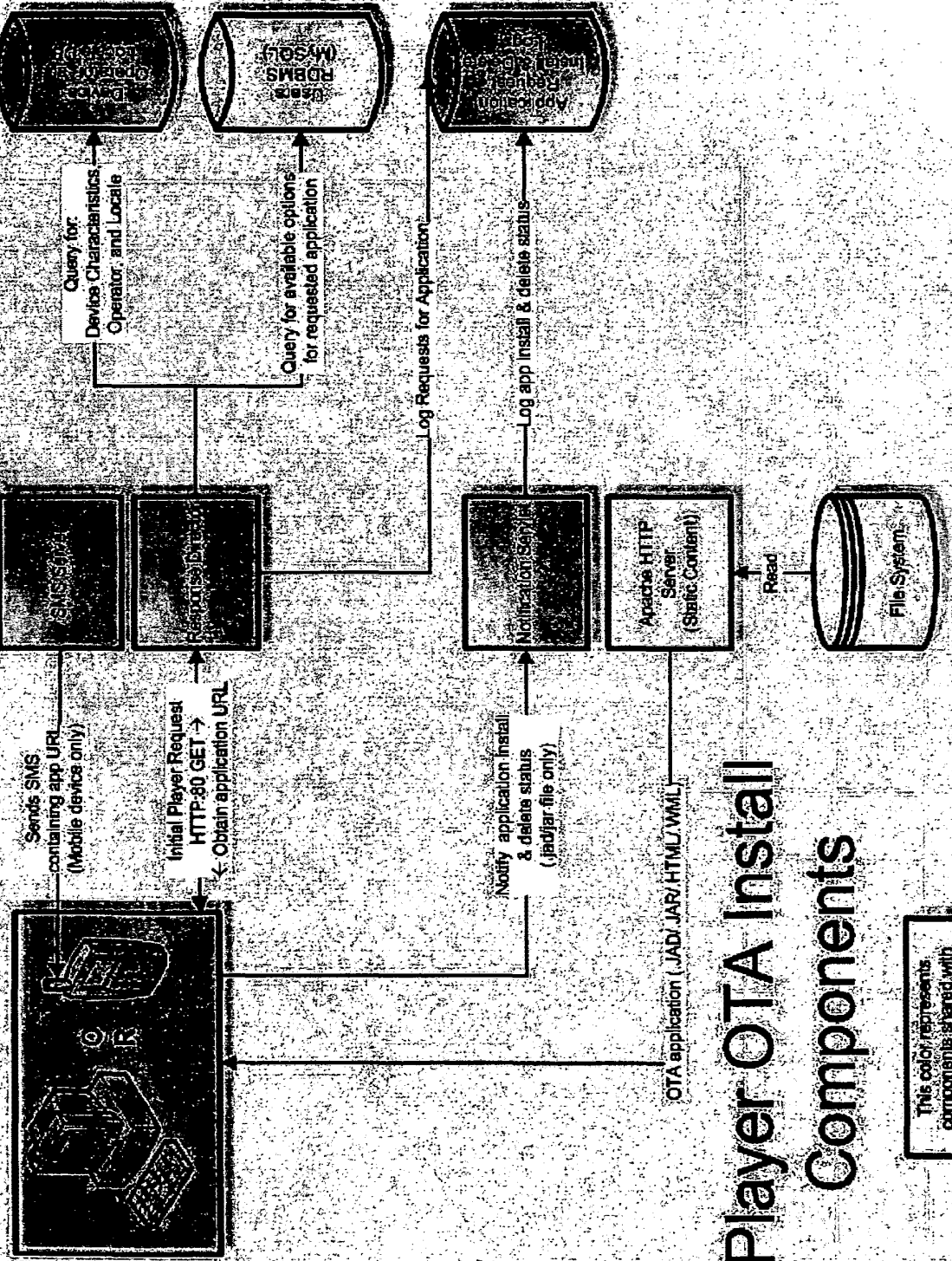
# Client Server Services



## Authoring Time Components

This color represents components shared with player installation/run time.

# Client Server Services

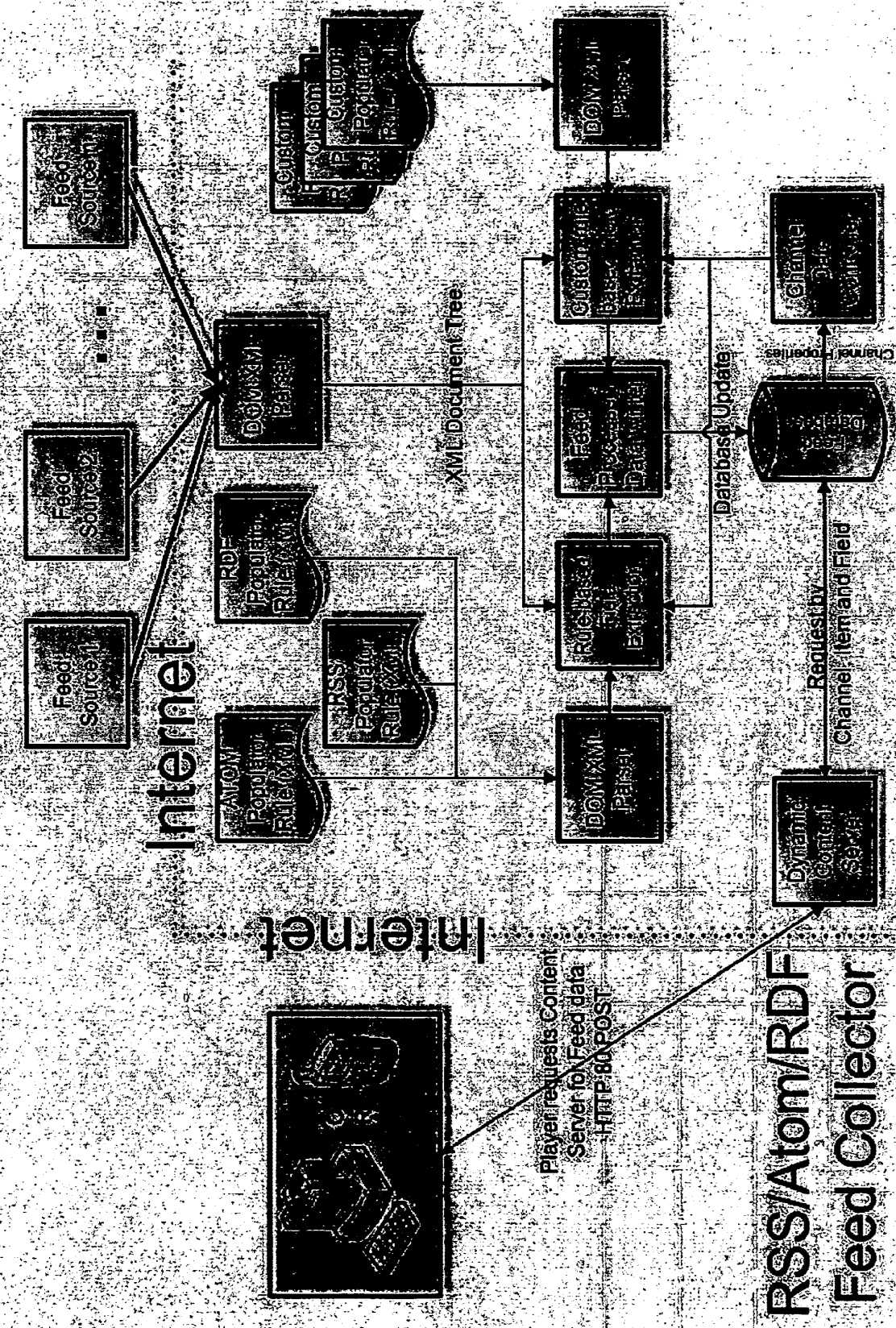


## Player OTA Install Components

This color represents components shared with authoring and/or run time



# Client Server Services



# Appendix R

## Response Director Validation

### 1 Introduction

This document describes the requirements of validating the Response Director for given market segments in which this product will be deployed. The requirements listed here assume a large-scale deployment of the Response Director.

### 2 Audience

This document is intended for:

- a. QA Department for preparing Test Suites for Response Director.
- b. Partners and Customers that must validate the Response Director for the devices within their domain.
- c. Third party testing services that will validate the use of the Response Director for devices.
- d. Akira Engineering staff, for maintaining and augmenting the Response Director.
- e. Akira Management, for determining the cost and resource requirements for improvements and enhancements to the Resource Director.

### 3 Structure

The Response Director has the following connections:

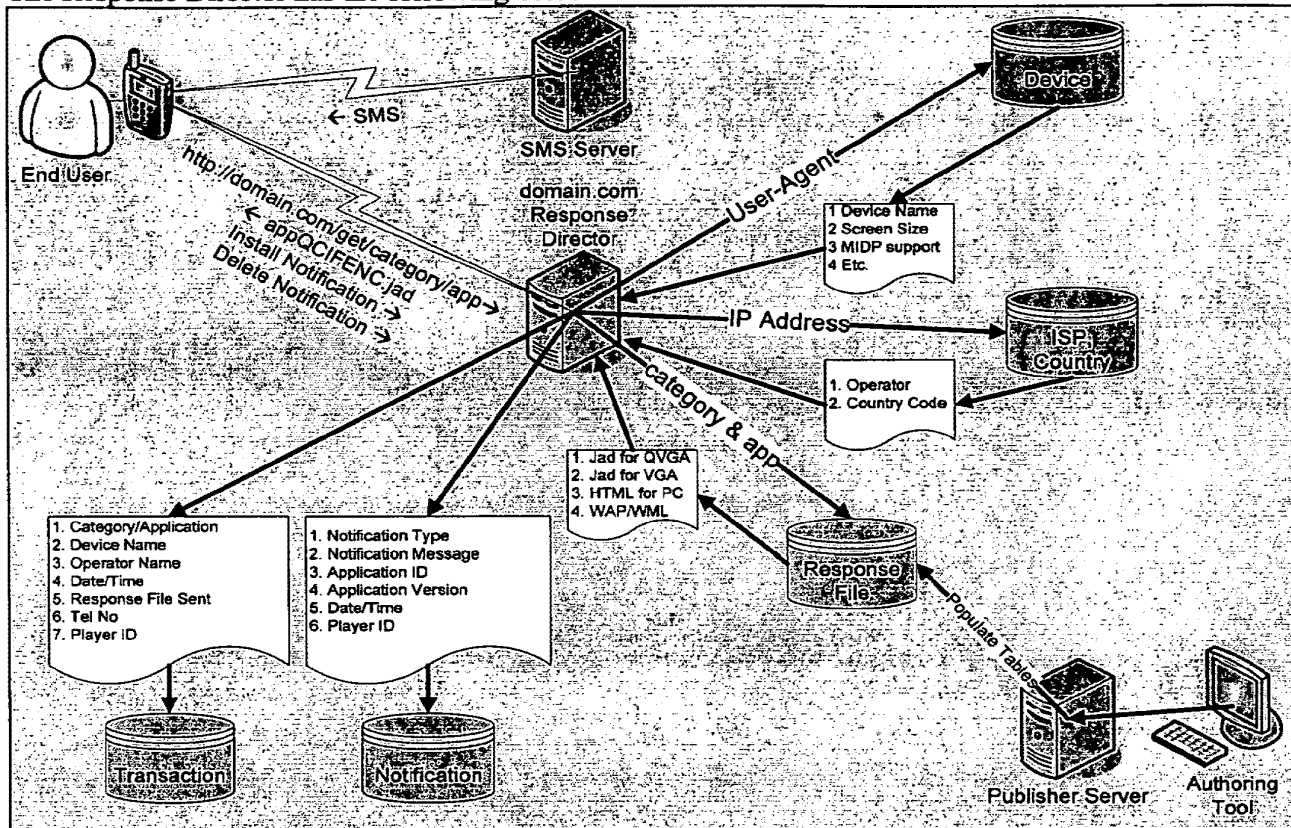


Figure 1 - Response Director Connections and Interactions

In the above diagram, all except the SMS Server and Publisher Server are considered part of the Response Director. However, both will need to be tested in conjunction with Response Director. The main variables are the handset device that the user has and the network on which the handset is used. There are 5 database tables involved, 3 that are for lookup only and 2 that are write only, as far as Response Director is concerned.

The above diagram assumes that the user interaction is initiated by an SMS message that is sent to the end user's handset via an SMS server. The SMS message contains a URL which if the user selects to go to, will be sent to the response director. The URL has the form:

<http://domain.com/get/category/app?telno=xxxxxxxxxxxx>

where

Docket: XPR002PR

R - 2



## Systems and Methods for Presenting Information on Mobile Devices

domain.com	is the domain name of the server
get	is the name of the Tomcat service that handles Response Director
category	is an application category, as defined in the response file database table
app	is the application name, as defined in the response file database table
telno	is a parameter keyword that indicates a telephone number follows
xxxxxxxxxxxx	is a normalized telephone number, starting with country code. For USA, this is 11-digits long, starting with 1, followed by area code and number.

## 4 Validation Requirements

1. Handset Device capabilities.
  - a. Define the set of devices to be supported.
  - b. For each device, list the capabilities (Device Capabilities List).
  - c. For each device, determine the appropriate response required by Response Director. In other words, what is the response object to send back to the device?
  - d. Confirm that expected response is obtained by handset device.
  - e. Confirm that handset device can run the test application properly.
2. SMS Service
  - a. Validate that SMS text messages with http addresses can be sent to devices in the supported operating network. (Some operators may restrict SMS messages from certain origins).
  - b. Confirm that the device can take action on the URL that is sent as part of the message.
3. Data Service
  - a. Confirm that the handset has Data send/receive capability and determine the expected average bandwidth.
    - i. Variables: network operator, network type, handset type
4. Notification Capabilities
  - a. For each java capable phone, *and on each operator network*, determine if the notification features work.
5. SMS Push Capabilities
  - a. For each java capable phone, *and on each operator network*, determine if SMS Push feature works. (This could be extended to WAP capable devices, as WAP also has push capability.)
6. Response Director Server Capacity
  - a. Determine how many incoming and outgoing messages can be handled with reasonable delays on a single 64-bit architecture CPU-based PC.
  - b. Determine the estimated capacity required for deployment.
7. Infrastructure and Hosting
  - a. Where to host the Response Director
  - b. How much bandwidth must be guaranteed to it?
  - c. How many devices are required in the server farm to handle estimated volume?
  - d. Load balancing and distribution
  - e. SMS gateway and services (e.g. broadcasting)

## 5 Additional Work

1. Database Enhancements

## Systems and Methods for Presenting Information on Mobile Devices

- a. For any devices that are not in the device database or that have insufficient data, augment the database using the device database patch mechanism.
  - b. Upgrade RDBMS to latest version of MySQL (or Oracle, MSSQL, or other system).
2. For any devices that do not work as expected, there will need to be engineering resources allocated to determine whether the problem operation can be overcome.
3. For each of the operators to be supported, a developer relations must be established to obtain information specific to the operators, for optimal operation under each operator
4. For each operator, obtain a digital certificate and/or certification that will allow Trusted Third Party Status for Akira Player.
5. WAP/WML support
  - a. Option 1: Separately write WML deck that is appropriate for each of the applications to support. Then include this in the Response Director database so that it will be sent to devices that are not capable of Java MIDP 2.0.
  - b. Option 2: Create an automated method of translating Akira Application into a WML deck. Publisher itself will populate the Response Director database appropriately to point to the WML content, just as it already does for JAD and HTML.
6. Provide Utility to manage the Response Director, including:
  - a. Updating the Response File Table, such as adding an entry for WAP, or removing an unused application.
  - b. Modifying the Heuristics module for customized heuristics.
  - c. Obtain reports on transactions and notifications.
  - d. Monitor performance
7. Device Capability Test Applications
  - a. Define an Akira application test suite that tests the various features of the phones to validate that the features are indeed functional and not just a line in the specification. Areas to test include (not all inclusive):
    - i. Real time streaming of video and audio
    - ii. Video playback including formats supported
    - iii. Video controls available
    - iv. Audio playback, including formats supported
    - v. Audio controls available
    - vi. Audio capture capability
    - vii. Video capture capability
    - viii. Still image capture capability
    - ix. Record store size and accessibility
    - x. File System accessibility
    - xi. Slide show
    - xii. Animation
    - xiii. Run external web site (and whether it can return to app)
    - xiv. Heap size
    - xv. Font metrics
    - xvi. Display buffering
    - xvii. Still image formats support (especially JPEG)
8. Remote testing tool (nice to have!)
  - a. Design a physical test harness that allows remote operation of actual handsets. Such a device will be comprised of:
    - i. A digital camera (a webcam) to capture the display content

## Systems and Methods for Presenting Information on Mobile Devices

- ii. A microphone to capture the sound. If the device has a sound output jack, the sound should be captured using this jack.
- iii. "Fingers" to press the keys of the device
- iv. Camera output, sound output, and "finger" controls are connected to a PC, which contains software to monitor the camera and sound and activate the "fingers".
- v. PC connects to the internet to be able to communicate with the remote control software. Camera/sound output streamed; "finger" controls via sockets.
- vi. One "server" PC can control multiple harnesses simultaneously.
- vii. Remote control software to view camera output and control the "fingers"
- viii. The harness will be flexible so that different form factor phones can be attached to the harness.

b. Implement a prototype of the harness

<End of document>

# Appendix S

## Rich Media IM Specification

### A: Overview

#### 1: New Alert Type.

There will be a new alert type "Rich Media IM" which will have an automatically generated (and undeletable) text area for user generated text messages.

#### 2: Binding to Phone-based rich media.

For this type of Alert Page, there will be, for any Video, Image object, Slide Show, or audio Attribute, the ability to bind these objects to phone based files. In the event that our player is controlling the camera, this binding can be automatic. Otherwise, there would have to be a way to present to the user a list of valid media files that he can include in the "Rich Media IM"

#### 3: Binding to Server-based static and/or real-time content.

This will also be permitted.

#### 4: Group Management.

These "Rich Media IM" messages can be sent to an individual, a group, or selected members of a group.

#### 5: Modes of presentation.

The recipient will have the ability to grant instant access for these messages to selected friends and family. Otherwise there would be some indicator to alert the user that there is a pending alert(s).

When browsing through these alerts at least the phone number of the sender will be visible. If available and appropriate, the person's name and/or text message will also be visible while browsing.

When viewing the message, if its text and/or audio only, the message could be non-modal so that the user can continue with the underlying application. If images or video is included it probably would be modal, at least for current generation phones.

#### 6: Additional Content Generation options.

The user could have a mechanism for assembling the phone generated content into a slideshow, and send that as part of the message.

Much of the permissions required to control the camera and/or gain access to the file system should not be a problem if its an on-deck relationship. Off-deck there are likely to be many restrictions, depending upon the device and operator.

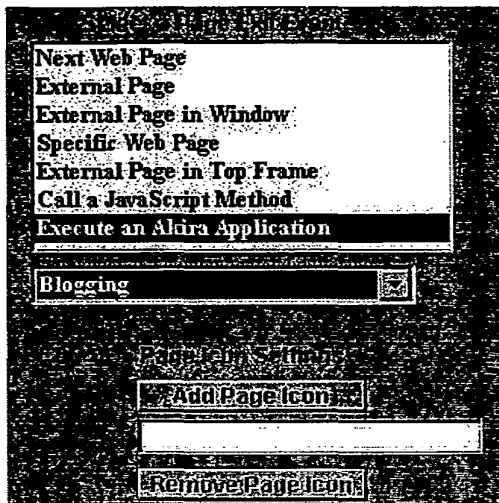
## Appendix X

# Widget Object Specification

### A: Authoring Widgets.

#### 1: Defining the Custom Widget Icon

After building the starting widget page, touch "Edit" with no object selected.



You will have a choice of Xpressmo standard Widget Icons, or the designer can select their own which will be uploaded automatically into their Widget Icon Library.

The publishing server will identify the following cases for aggregating the icons for:

- 1: First Page of an Application (release 2)
- 2: Page Objects (release 2)
- 3: Xmo Server Pages (release 2)
- 4: Widgets. (release 1)

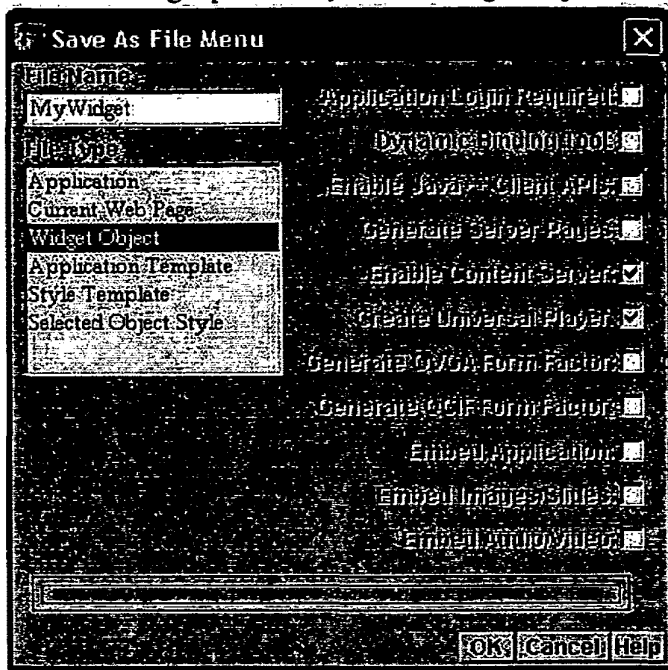
In order for there to be a symbolic link between the icon and the application, server page or widget that it represents the folder for these navigation icons will take on, in some manner, the name of the application that they represent. In release 1, the Widget Reference Icons will be placed in Author's Widget Reference Icon folder, and will have the file name that is the same as the Widget application name.

#### 2: Saving a Widget Object

The widget Object will be saved in the Designer's Personal Widget Object Library and thus become available as a widget (with its accompanying Icon) from the Xpressmo Widget Dashboard.

Systems and Methods for Presenting Information on Mobile Devices

There will also be an Import Widget Object choice under the File menu for editing a previously saved Widget Object.



## Systems and Methods for Presenting Information on Mobile Devices

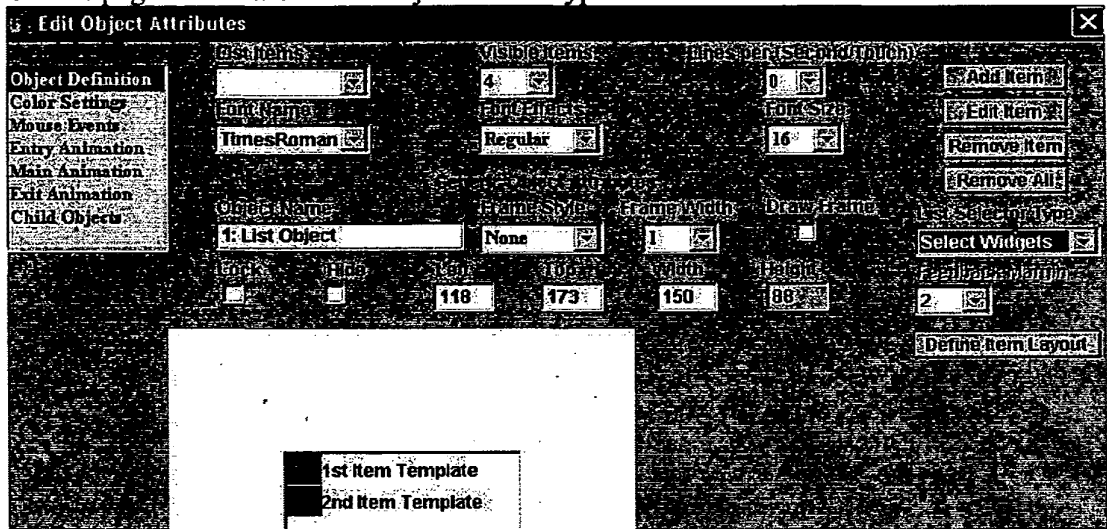
### 3: Defining a Widget Launch Page.

#### a: Widget Libraries

First create a library of Widget objects. For each of these Widget Objects it would be desirable that a custom widget Icon is selected or created.

#### b: Create A Widget Launch Page.

On this page one of more List Objects of the type



If there is a “Goto Widget Object” Exit Event attached to a list item, then a default Widget Icon will be drawn.

However, if the Widget Page has already had a Page Icon defined then it will be drawn.

#### c: Widget Categories

We probably want to define a get of categories for widgets, and, in fact, have a different default icon for each category. These widget icons would be available in the Xpressmo public Widget Icon Library.

The design of the Launch Page should probably take this into account, so that there is a “Select Widgets” List Object for each category. In the event that multiple categories are in the same list, then I would suggest that a “default” Complex List is used, in which the Widget Category Icon is the Title Icon and the Custom Widget Icon is the Body Icon.

#### d: Other Information that could be presented in the Launch Page.

Since we can have up to 2 lines with 5 icons and four text strings per widget item, this opens up things like:

## Systems and Methods for Presenting Information on Mobile Devices

- A Widget Rating Icon (1 to 5 stars)
- A Free, Share ware or fee based icon.
- Number of downloads for each widget.
- The Author's name and the Widget's name.
- Etc.

### 4: Libraries

#### a: Publisher Level (publicly available libraries)

- Pages
- Widgets
- Page Icons
- Widget Icons

#### b: Author Level

- Pages (Page Library)
- Widgets (Widget Library)
- Page Icons (all page icons that were uploaded by author)
- Widget Icons (all widget icons uploaded by author)
- Widget Reference Icons (see A.1)

#### c: Application Level

- Background (Page/Widget Icons)
- Icons (application icons)
- pages (Xmo Server Pages)



## Systems and Methods for Presenting Information on Mobile Devices

### **B: Player Widget Operations**

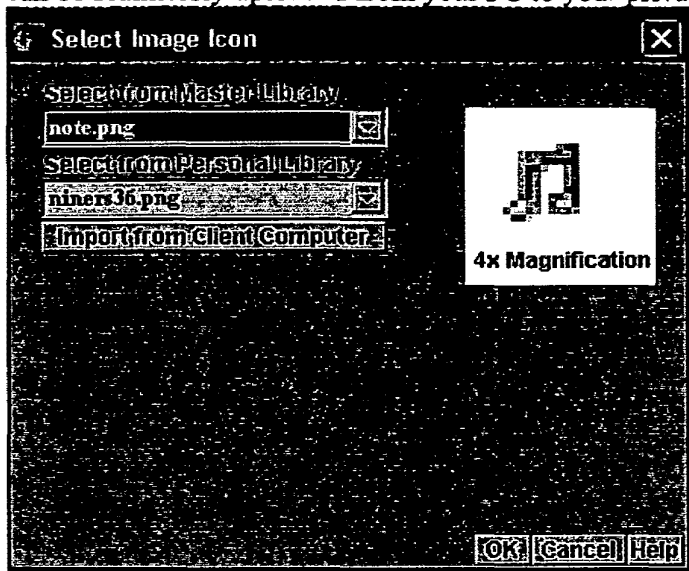
When a widget designer saves a widget it initially is placed in a personal Widget library, and the Custom Widget Page Icon, if any, is placed in both in the widget application Page Icon sub-folder as well as in the designer's personal Widget Icon Library Folder.

This specification does not cover what further procedures should be implemented during operations.

## Systems and Methods for Presenting Information on Mobile Devices

### **C: Widget Icon management.**

Widget Icons are similar to images in that they have both private and public libraries and can be seamlessly uploaded from your PC to your private library.



Systems and Methods for Presenting Information on Mobile Devices

**Appendix A: PDL Specification for Widget Objects**

**WebPage Definitions**

BooleanAttribute[2]= Page Icon Exists

IntegerAttribute[4]= Page Icon Width

IntegerAttribute[5]= Page Icon Height

**Primitive Object Definitions**

ObjectIntegerAttribute[12] (Exit Event)

16=Goto Widget Object

**List Item Definitions**

ListItemExitEvent

16=Goto Widget Object

**Slide Show and Icon Strip Objects**

SlideIntegerAttribute[6] (Slide Exit Event)

8=Goto Widget Object

## **Appendix W**

# **Version Control System Integration into Publisher Technical Documentation**

### **1 Introduction**

Publisher saves the working application and its resources in the OS file system on the server machine running Publisher Server. At present, it offers no back up\* or version control mechanisms for saved applications. This could lead to potential data loss in case the latest revision is corrupted, either due to program error or transmission problems over the network. At present, the author's only recourse is to Save-As in many different-named copies of the application and maintain a system for identifying revisions on a custom basis, a cumbersome and error prone method.

This proposal would provide a full version control system back end to Publisher so that an author is freed from manual management of versions; it also provides a convenient and reliable back up in case of data corruption due to errors.

### **2 Choice of Version Control System**

The first choice is whether to roll-our-own or purchase or re-use an existing version control system (VCS). The choice is obvious. Our domain expertise is not in developing a back up or versioning system. There are open-source versioning systems that are widely adopted by the development community. The first choice has been made: use an existing VCS.

The second choice is to select which versioning system to use. There are two widely adopted open source based VCS: CVS and Subversion (SVN). We use CVS internally for development of Publisher, and there is some familiarity. However Subversion can be thought of as the successor of CVS and fills in the gaps where CVS is weak: Subversion also has a well designed API for creating Subversion clients, including one in Java. Finally, it runs well on many popular platforms, including Linux and Win32 and Mac OS X. The second choice tilts in the favor of Subversion.

The most current version of Subversion as of this writing is 1.4.3.

First implementation to be based on CVS for expediency. Xpressmo already uses CVSNT, a variant of CVS, and there would not be any resources expended supporting another VCS.

---

\* Except for the .akp file going back just one version.

### 3 Phased Approach to Integration

The integration with VCS will take a phased approach to provide relief in the short term and ease of use and flexibility in later phases.

The proposed phases are outlined here, in the order of implementation:

Phase	Description	Time/Resources
1 Manual	<p>Modify the way Publisher stores files in the file system to make it easier to save and restore to VCS. Specifically, store all files related to a given application within an application folder. At present, the .akp and .spdl files reside outside of the application folder. These will be brought into the root level of the application folder.</p> <p>This will affect the following components:</p> <ul style="list-style-type: none"> <li>• Publisher Server</li> <li>• Content Server</li> </ul> <p>The author will create a repository with the name of the user and name of the application, e.g., if the root of the repository is /svn/repository then create /svn/repository/servername/username/appname/trunk to store the app folder.</p> <p>&lt;detailed SVN commands TBD&gt;</p>	<p>1 day for design and implementation + testing</p> <p>2 days for setting up test SVN server</p>
2 Minimal Integration	<p>Modify Publisher Server so that after a Save or Save-As operation has been fully committed, including background processing of image files and scaled versions, a script file with pre-determined name and location is called. The script file will be provided with the relevant information to allow a snapshot of the application to be stored in VCS:</p> <ul style="list-style-type: none"> <li>• server name</li> <li>• user/author name</li> <li>• application name</li> </ul> <p>A single Save or Save-As may invoke the script multiple times when multiple auto-scaled versions of the application are created.</p> <p>Modify Publisher Server so that at Open Application time, a script file is invoked after the user selects the application to edit but before the application is loaded into Publisher. The script file will have a pre-determined name and location, and its purpose is to load the latest revision of the saved snapshot of the application from the VCS repository. It will be provided with the same information as during Save.</p> <p>There are no changes to existing Publisher GUI.</p>	<p>1 week for design and implementation + testing</p> <p>2 days for setting up test SVN server</p>

Systems and Methods for Presenting Information on Mobile Devices

	Requires manual creation of VCS repository project, unless script can accommodate this.	
3 Full Integration	<p>Modify Publisher so that it becomes a GUI client for the VCS. Use the Java API's to tightly integrate with the VCS. Support using Java API's will include:</p> <ol style="list-style-type: none"> <li>CVS (via JavaCVS client library)</li> <li>SubVersion (via SVNKit library)</li> <li>Perforce (via P4Package library)</li> <li>Stub API for implementing to other VCS's</li> </ol> <p>As with the minimal integration phase, Publisher will access the VCS at Open Application and Save/Save As Application time. Provide additional functionality at the Publisher Client level:</p> <ul style="list-style-type: none"> <li>Selecting a version other than the latest version to edit at Open Application time, either by using tags or date and time &lt;tbid&gt; – implemented as a pull down list of available versions</li> <li>Allowing comments and tags to be stored with the version at Save/Save As Application time. (Tags are short keywords that will show up in the version list during Open Application. Comments are longer descriptions, which can be displayed separately during Open Application time by pressing a View Comment button.)</li> <li>Automatic creation of VCS repository project when a new application is saved</li> <li>Option to not load application from VCS during Open.</li> <li>Option to not save application to VCS during Save.</li> </ul>	<p>1 month for design and implementation + testing</p> <p>2 days for setting up test SVN server</p>

**4 Areas Not Covered by VCS Integration**

The above proposal will allow applications to be saved in their entirety. However there are some saves which fall outside the application. Those areas not covered are:

- \* Single Pages – in the Pages folder under the user folder
- \* Widget Objects – in the Widgets (?) folder
- \* Application Templates – in the Templates (?) folder
- \* Style Templates – in the Styles (?) folder
- \* Selected Object Styles in the (???) folder
- \* Back end code – in the database
- \* Application metadata such as the currently assigned application UUID – in the database.

These areas can also have VCS integration in a manner similar to Application, but this is not covered in the above proposal.

## 5 Application Life Cycle Management

The addition of version control system to Publisher greatly enhances its usability. However it does not address application management, which should be considered as well. At present the following functions are lacking in Publisher:

- \* Deletion of Application and its associated back end code and metadata.
- \* Copy of Application to different user workspace or different server
- \* Renaming of Application without requiring Opening and Save-As.

## 6 Configuration Files

Some of the parameters controlling Publisher's interaction with the VCS will be set in configuration file publishingserver.properties

Proposed parameters for phase 2, for CVS and SVN:

Parameter Name	*	Value Description (top level: SVN, bottom level CVS)
VCS_Repository_URL	-d	SVN: The URL to locate the repository root. This may be on a remote server.
		CVS: The domain name or TCP/IP address of the CVS server. This may be on a remote server.
VCS_Port	-o	SVN: Not used. Port is part of the repository URL.
		CVS: Port number for connection to CVS server.
VCS_UserName	-u	SVN: A username to access server. Default will be Publisher username.
		CVS: Same as above.
VCS_Password	-p	SVN: A password to access server. Default will be Publisher password.
		CVS: Same as above.
VCS_BaseFolder	-f	SVN: The folder under the repository root to place all user/appname folders.
		CVS: The path to the repository.
VCS_Trunk	-r	SVN: The folder under user/appname to place trunk branch
		CVS: Not used.
VCS_Tags	-t	SVN: The folder under user/appname to place tag copies
		CVS: Not used.
VCS_Branches	-b	SVN: The folder under user/appname to place branch copies
		CVS: Not used.
VCS_Open_Script		SVN: The location + file name of the script to run when user Opens an Application.
		CVS: Same as above.
VCS_Save_Script		SVN: The location + file name of the script to run when the user Saves or Save-As an Application.
		CVS: Same as above.

\* This is the parameter prefix used on a command line to send this information to the script file. E.g. to pass VCS\_UserName to the script use "-u username".

## 7 SVN Repository Set Up

<TBD>

## 8 CVS Repository Set Up

Tested system is CVSNT from <http://www.march-hare.com>. The version used was CVSNT 2.5.03 for both client and server.

CVSNT was selected because it runs under Windows as well as Linux servers. The base CVS code runs only under Linux and other Unix-like systems. CVSNT is of course based on the CVS code so there should be a high degree of compatibility. The test server is running under Windows 2000 Server. Previously it has been run on Windows 2003 on a 32-bit architecture machine. However on a 64-bit architecture machine, I have not had a successful configuration of CVSNT server.

To use CVSNT as the repository server, go to the march-hare website and download the latest version of the binaries for Windows. Install both client and server on the repository server machine. Follow the lucid guide at:

[http://www.mobilefish.com/developer/cvsnt/cvsnt\\_quickguide\\_usage.html](http://www.mobilefish.com/developer/cvsnt/cvsnt_quickguide_usage.html)

to install and configure the basics.

To use CVS with Publisher, install the CVSNT client on the machine running Publisher server, and verify that the command "cvs" is available from the command line in the Publisher root folder.

### 8.1 cvs wrappers

As described in the cvsnt installation guide above, the file named cvs wrappers must be edited to include all binary file extensions. A copy of this file that is used for our testing is listed in the Appendix section 10.1 cvs wrappers content. Use the method described in the guide to update this file BEFORE anything else is saved into the repository.

## 9 Phase 2 Implementation (updated)

Change: Phase 2 will use CVS rather than SVN as the target VCS. This change has the following advantages:

- a. Allows Xpressmo to use the feature in-house as part of on-going testing.
- b. Reduces development time by using an already running, already familiar VCS.
- c. Information gathered to integrate with CVS is applicable to SVN or other VCS.

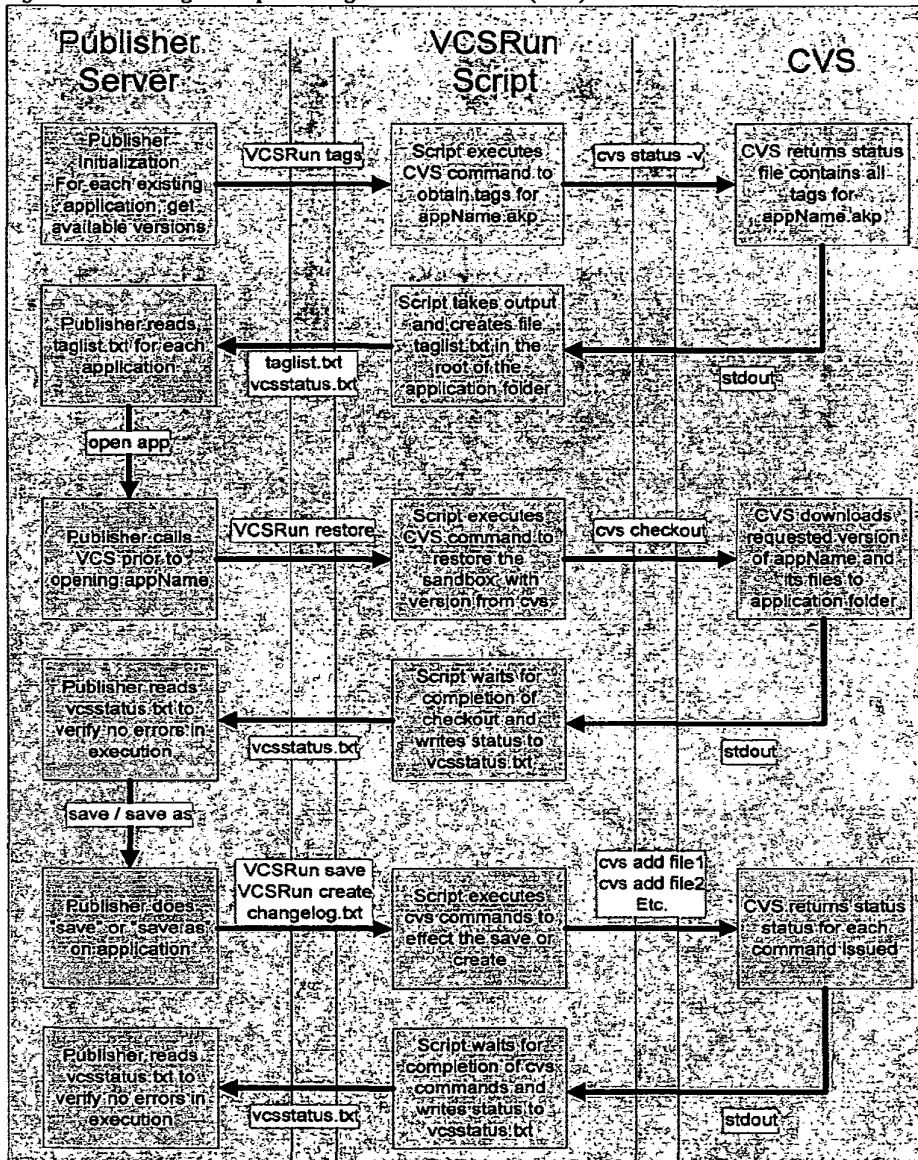
### 9.1 Flow Diagram

The following diagram shows the three functions in publisher that require calls to the VCS. Publisher communicates with the VCS via an intermediary script file and text files.



Systems and Methods for Presenting Information on Mobile Devices

Figure 1 - Flow diagram representing Publisher to VCS (CVS) communication



## Systems and Methods for Presenting Information on Mobile Devices

### **9.2 Commands used with CVS**

CVS requires more detail than envisioned for SVN. The distinct functions that CVS requires are:

<b>Function</b>	<b>CVS command</b>	<b>Explanation</b>
Create a new project	cvsc -C -n import project cvsc tag tagName *	CVS creates a new project repository folder and initializes it. The import must occur in the folder with the new project = application folder. All files in this folder and subfolder will be copied to the new project.
Add a new file	cvsc add filename cvsc commit	Each file that is added after the project has been created in the repository requires an add statement, followed by a commit. Multiple add, remove, and update commands can be grouped with one commit.
Add a new folder	cvsc add folder	Folders are added without needing a commit. Folders must be created before any content in that folder can be added.
Remove file	cvsc remove filename cvsc commit	Removes one or more files
Remove folder	cvsc remove * cvsc commit	Remove all content of the folder but <i>folder is not removed</i> in the repository.
Modify file	cvsc update filename	Any file that was open for file output that already existed.
Restore latest version	cvsc checkout project	This copies the latest stored versions of every file in the project into the 'sandbox'
Restore tagged version (not in Phase 2)	cvsc checkout -r tagname project	This copies the version of the project specified by the tagname into the 'sandbox'.
Rename file	cvsc remove filename cvsc add filename cvsc commit	There is no atomic operation for rename in CVS. The recommended method is to remove then add.
Get available tags for a project (not in Phase 2)	cvsc status -v someFileName	Obtain the list of tags in a status report. Use one of the existing files to obtain the tag. A safe one would be the .akp file for the project. Parsing out the tags

Docket: XPR002PR

W - 7

## Systems and Methods for Presenting Information on Mobile Devices

		is required as status contains other information.
--	--	---

### **9.3 VCSRRun API**

Java API to which Publisher users may connect their preferred VCS, other than CVS, SVN and Perforce.

<TBD> This will be provided in Phase 3

### **9.4 VCSRRun Script**

Publisher requires VCS access at three junctures. All three VCS accesses are performed by calling the script file named VCSRRun, located in the install folder under Publisher root on the server:

**Comment (k1): Any better location??**

#### **9.4.1 Publisher Initialization**

During Publisher initialization, it will request the tags that are available for each of the applications that are in the user folder by calling "VCSRRun tags". These tags are displayed to the user as choices. (Note: Not implemented in Phase 2).

#### **9.4.2 Application Open**

Once the user has selected an application to open for editing, but before any file is opened (other than for preview), "VCSRRun restore" is run. This copies the latest version (or version matching specific tag, in future) of all files in the application to the 'sandbox', which is the location where Publisher reads and writes the application data, e.g., C:/Program Files/Publisher2.0/~userName/WebSites/appName.

#### **9.4.3 Application Save**

When the user executes "Save" or "Save As" on an application, the following occur after all files related to the application have been written out:

- a. If the application is new, that is, does not exist, then a new project is created in the VCS by calling "VCSRRun create appName".
- b. If the application exists, then a list of all changes is accumulated in file changelog.txt. This file is passed as a parameter to "VCSRRun save appName"

### **9.5 Changelog.txt File Format**

Changelog.txt is generated by Publisher server to communicate to the VCSRRun script about changes that have been made to the application.

Changelog.txt is a text file with CR-LF delimited records and space delimited fields. Each line of changelog.txt is one record describing an action that the editor has taken on the application during the last editing period (between the last open and save, or between last save and save).

Each record consists of fields, separated by a space

- a. Action : header, add, remove, update, rename, mkdir or create (new project)

Docket: XPR002PR

W - 8

## Systems and Methods for Presenting Information on Mobile Devices

- b. Name of file or folder on which action was performed
- c. Name of file or folder (destination for rename)

The file separator is always a forward slash.

The action "header" is followed by the version of the changelog.txt file. Additionally it contains the username (that is the Publisher user account name), the application name, the tagname (the application version), and the number of following actions in changelog.txt. The header record is required to be the first record in changelog.txt. It is not intended to emit any code for VCS but can be used by the script for verification and to determine compatibility with its version.

Examples:

```
version 1 username appName v1_01_01 6
create appName
add images/foo.jpg
remove slideshow/foo.avg
update appName.akp
rename appName.akp appName.bak
mkdir slideshow/slide1/images
```

### **9.6 VCSRun script for CVS**

The Windows script is defined here.

REM Parameters

```
REM %1 Operation, one of "save", "restore", or "tags".
REM %2 Name of Publisher account, without the leading ~.
REM %3 Name of Publisher application which is to be saved,
REM restored, or get tags for.
REM %4 Full path to application, e.g.,
REM C:\Program Files\Publisher2.0\~userName\WebSites\appName.
REM %5 Version number (major_minor_micro) that can be used in tagging,
REM e.g. 1_01_01 for 1.01.01 (for save and restore).
REM Full path and filename into which to write the available tags
REM for this application (for tags).
REM %6 Comments attached to all saved items.
```

REM

REM All operations require parameters 1 through 5.

REM

REM Operations

```
REM save Saves the application identified by the path in parameter 2 & 3
REM to the repository. Parameters 1 through 5 are required.
REM Parameter 6 is optional. The file changelog.txt must be in folder
REM specified by parameter 4.
REM restore Restores the application to the sandbox specified in path in
REM parameter 4. Parameters 1 through 4 are required. Parameter 5,
REM if present, will retrieve the version tagged with this tag.
REM Parameter 6 is ignored.
```

Docket: XPR002PR

W - 9

## Systems and Methods for Presenting Information on Mobile Devices

REM tags Creates a list of tags that are available for this application.  
REM This list is intended to be presented to the author for selection  
REM of a version. The output file is specified in parameter 5.  
REM Parameters 1 through 4 are also required. Parameter 6 is ignored.  
REM tagfile format:  
REM Each record is a line delimited by CR-LF.  
REM there are exactly 3 fields per record. The record separator is  
REM the comma (.). The field 1 is the tag ID, field 2 is date/time of  
REM modification, and field 3 is comment

### **9.7 Changes to Publisher Server**

Two alternative approaches are proposed and outlined. Alternative 2 was selected for implementation.

#### **9.7.1 Alternative 1**

To capture the information required for CVS during a save operation, all instances of writing to the file system are captured and processed. I have identified the following two classes and their methods as requiring interception. These will be subclassed to add the appropriate code to record the changes that are being made to the application folder.

Intercepted method	Description/Purpose	Emitted to changelog.txt
java.io.File.<init>	Obtain the file or folder name of file being written to.	--nothing--
java.io.File.delete java.io.File.deleteOnExit	File or folder being deleted.	remove filename -or- remove foldername
java.io.File.mkdir	Folder being created	add foldername
java.io.File.mkdirs	One or more folders being created. E.g. for a/b/c →	mkdir a mkdir a/b mkdir a/b/c
java.io.File.renameTo	file being renamed	remove filename add filename
java.io.FileOutputStream.<init>	Obtain file or folder name; if file already exists emit A. otherwise emit B.	A. modify filename B. add filename

All file and folder names in changelog.txt are relative to the application folder. However there could be multiple applications saved in one invocation of "Save" or "Save As" in Publisher, to support scaled-down versions of the application. Internally, the tables will be implemented as Hashtable of Vector, where the Hashtable key is the application name and the value Vector consists of the commands to be placed in changelog.txt for the particular application.

## Systems and Methods for Presenting Information on Mobile Devices

### **Pros and Cons**

The advantage of this alternative over Alternative 2 is that the information is gathered without regard to the state of the Publisher. As long as the VCS command list is purged after a save, the Publisher server need not be even aware that a tally is being kept of the changes. Also, the lack of an inventory means it uses less memory – it only tracks changes, not the entire inventory.

The disadvantage of this alternative over Alternative 2 is that it must delve into the File I/O of each and every file output operation and determine if it belongs to an application folder.

Another disadvantage is that it must custom handle situations where an external application creates, updates or removes files or folders, where the tracking of File operations is not possible.

Yet another disadvantage is that it must handle situations where during the course of the “save” operation, a file may be created and destroyed multiple times. Each operation on the file is recorded, and the logic in the code that emits the commands to the VCS must determine the final state of the file and remove any extraneous commands that affect some non-final state of the file.

### **9.7.2 Alternative 2**

To capture the information required for CVS during a save operation, an inventory of the application folder is taken just after the “open” is completed of the application and just after a “save” is completed of the application. The differences between the “open” and the “save” inventories, or between the previous “save” and the current “save” inventories will determine what commands need to be sent to the VCS for the save.

The “save-as” operation has three possibilities:

- a. The “save-as” is to the same application name as the current name – in this case it behaves just like the “save” case and uses the inventory taken after “open” which is compared to the inventory after “save”.
- b. The destination is a new application that does not currently exist – in this case the earlier inventory will be empty. This is a degenerate case of c below.
- c. The destination differs from the currently edited application name, and the destination already exists – in this case the earlier inventory is taken just prior to the execution of the “save-as” operation.

The information is kept in a tree structure which will reflect the application folder tree. A degenerate instance of this tree is the null tree where not even the application folder exists. This would be the case when a new application is created using “save-as”.

The information that is kept is thus:

Application folder path, relative to Publisher root.

For each node that is a folder, the name of the folder (if it exists).

Docket: XPR002PR

W - 11

## Systems and Methods for Presenting Information on Mobile Devices

For each node that is a file, the file name and last modification date (if it exists).

Let A be the inventory prior to the "save" operation. Let B be the inventory after the "save" operation. The directives to the VCS will be created thus:

Condition	VCS directive
File does not exist in A but exists in B	add file
File exists in A but not in B	remove file
File exists in A and B with difference in modification date	modify file
Folder does not exist in A but exists in B	add folder
Folder exists in A but not in B	remove folder
Application folder does not exist in A	create appFolder

### **Pros and Cons**

The advantage of this alternative method over Alternative 1 is that one does not need to subclass file I/O methods.

Another advantage is that it will handle any files or folders that are created, modified or deleted by external applications such as the jarsigner or ImageMagick. In alternative 1, the files modified by the external applications must be each considered separately and handled.

Yet another advantage is that there is no need to worry or consider intermediate states of the application folder. In the course of editing and saving, a file may be created and deleted multiple times, but only the state of the file at the time of the inventory matters. In alternative 1, there must be logic to remove VCS commands associated with non-final states of the file since all file operations are tracked.

The disadvantage of this alternative over Alternative 1 is that the inventory must be retained at all times for all users and for all applications that being edited. In a multi-user environment this would require more memory than Alternative 1.

Another disadvantage is that it is reliant on the state of the Publisher to obtain its inventories.

### **9.8 Resource Requirements for Phase 2**

2 days: Design and Documentation  
1 day: modification to Publisher Server to send essential info to VCS  
1.5 day: creation of VCSRun script file  
0.5 day: modification to Publisher Server to respond to error and other status returned from VCS.  
1 day: Unit testing.

---

Total: 6 days.

Docket: XPR002PR

W - 12

## 10 Appendix

### 10.1 cvswrappers content

```
# This file affects handling of files based on their names.
#
# The -m option specifies whether CVS attempts to merge files.
#
# The -k option specifies keyword expansion (e.g. -kb for binary).
#
# The -t option overrides the default mime type.
#
# Format of wrapper file ($CVSROOT/CVSROOT/cvswrappers or .cvswrappers)
#
# wildcard [option value][option value]...
#
# where option is one of
# -k      expansion mode      value: b, o, kkv, etc.
#
# and value is a single-quote delimited value.
# For example:
#*.gif -kb
#
# Publisher specific
*.akp -k 'b'
*.app -k 'b'
*.aud -k 'b'
*.avd -k 'b'
*.avg -k 'b'
*.db -k 'b'
*.avr -k 'b'
*.dat -k 'b'
*.pdl -k 'b'
*.stp -k 'b'
*.tpp -k 'b'
*.vid -k 'b'
*.sid -k 'b'
# audio video formats
*.3g2 -k 'b'
*.3gp -k 'b'
*.3gpp -k 'b'
*.3gpp2 -k 'b'
*.aac -k 'b'
*.ac3 -k 'b'
*.aif -k 'b'
*.aiff -k 'b'
*.amr -k 'b'
*.amr_nb -k 'b'
*.amr_wb -k 'b'
*.ams -k 'b'
*.au -k 'b'
*.awb -k 'b'
*.band -k 'b'
*.gsm -k 'b'
*.h263 -k 'b'
```



Systems and Methods for Presenting Information on Mobile Devices

\*.h264 -k 'b'  
\*.iff -k 'b'  
\*.m3u -k 'b'  
\*.m4a -k 'b'  
\*.m4b -k 'b'  
\*.m4p -k 'b'  
\*.mdl -k 'b'  
\*.mid -k 'b'  
\*.midi -k 'b'  
\*.mp\_ -k 'b'  
\*.mp4 -k 'b'  
\*.mpa -k 'b'  
\*.mpu -k 'b'  
\*.mt2 -k 'b'  
\*.mtm -k 'b'  
\*.nrt -k 'b'  
\*.qcp -k 'b'  
\*.qt -k 'b'  
\*.ram -k 'b'  
\*.rm -k 'b'  
\*.rng -k 'b'  
\*.snd -k 'b'  
\*.thx -k 'b'  
\*.toc -k 'b'  
\*.ulaw -k 'b'  
\*.wma -k 'b'  
\*.xm -k 'b'  
\*.xmi -k 'b'  
# Other common  
\*.a -k 'b'  
\*.avi -k 'b'  
\*.bin -k 'b'  
\*.bmp -k 'b'  
\*.bz2 -k 'b'  
\*.cab -k 'b'  
\*.class -k 'b'  
\*.doc -k 'b'  
\*.dll -k 'b'  
\*.ear -k 'b'  
\*.exe -k 'b'  
\*.exp -k 'b'  
\*.fla -k 'b'  
\*.gif -k 'b'  
\*.gz -k 'b'  
\*.hqx -k 'b'  
\*.ilk -k 'b'  
\*.jar -k 'b'  
\*.jpeg -k 'b'  
\*.jpg -k 'b'  
\*.lib -k 'b'  
\*.mov -k 'b'  
\*.mpeg -k 'b'  
\*.mpg -k 'b'  
\*.mp3 -k 'b'  
\*.msi -k 'b'  
\*.mso -k 'b'  
\*.ncb -k 'b'

Docket: XPR002PR

W - 14

Systems and Methods for Presenting Information on Mobile Devices

\*.o -k 'b'  
\*.obj -k 'b'  
\*.ogg -k 'b'  
\*.pdb -k 'b'  
\*.pdf -k 'b'  
\*.pfx -k 'b'  
\*.png -k 'b'  
\*.ppt -k 'b'  
\*.res -k 'b'  
\*.rpm -k 'b'  
\*.sit -k 'b'  
\*.so -k 'b'  
\*.swf -k 'b'  
\*.tar -k 'b'  
\*.tgz -k 'b'  
\*.tif -k 'b'  
\*.tiff -k 'b'  
\*.tlb -k 'b'  
\*.vsd -k 'b'  
\*.xls -k 'b'  
\*.war -k 'b'  
\*.wmv -k 'b'  
\*.wmz -k 'b'  
\*.zip -k 'b'

<end of document>

# Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/US2009/039695

International filing date: 06 April 2009 (06.04.2009)

Document type: Certified copy of priority document

Document details: Country/Office: US  
Number: 61/166,651  
Filing date: 03 April 2009 (03.04.2009)

Date of receipt at the International Bureau: 30 April 2009 (30.04.2009)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)



World Intellectual Property Organization (WIPO) - Geneva, Switzerland  
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

160048

# THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS SHALL COME:

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

*April 29, 2009*

**THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A FILING DATE.**

**APPLICATION NUMBER: 61/166,651**

**FILING DATE: *April 03, 2009***

**RELATED PCT APPLICATION NUMBER: *PCT/US09/39695***

**THE COUNTRY CODE AND NUMBER OF YOUR PRIORITY APPLICATION, TO BE USED FOR FILING ABROAD UNDER THE PARIS CONVENTION, IS *US61/166,651***



Certified by

Under Secretary of Commerce  
for Intellectual Property  
and Director of the United States  
Patent and Trademark Office



## Transforming the Desktop Internet Experience to Mobile Devices

July 15, 2008

### The Xpressmo Platform:

#### Transforming the Desktop Internet Experience to Mobile devices

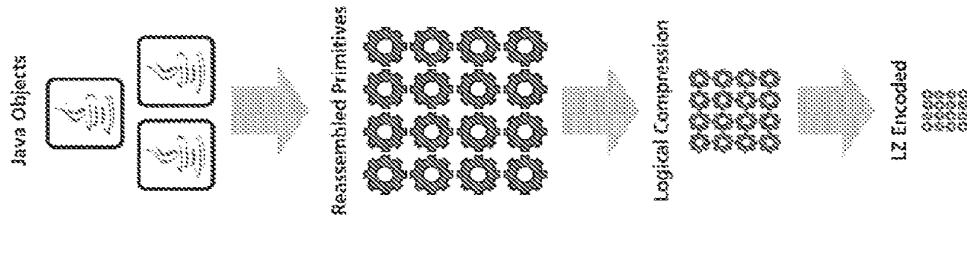
Apple's iPhone introduced one appealing implementation of transforming, in an appropriate way, the PC browser experience with the inherent uses and constraints of a mobile device. Going forward, the bar will continue to be raised as applications and services will appear that offer:

- ▶ **Comprehensive Application Model**
- ▶ **Professional Authoring Platform that enhance:**
  - Rapid application development
  - Much lower maintenance and upgrade costs
  - Much faster time to market
- ▶ **Web Components**
  - Powerful proprietary and 3<sup>rd</sup> party Web Services integration
  - Tight integration into Authoring Platform
- ▶ **Java APIs and the Xpressmo SDK**
- ▶ **Intuitive Navigation**
- ▶ **Advanced Feature Support**
  - Messaging and Social Networking Support
    - ▶ Full PC Instant Messaging Capability transformed to Mobile
    - ▶ Generated and Customized Content
  - Personalization and Temporal Adaption
    - ▶ Consumer Personalization Requests
    - ▶ Consumer usage patterns
    - ▶ Location
    - ▶ Time of day
  - Push Capability
  - Tight Integration with Phone Functions (PIM, Camera, etc.)
  - Mashups
  - Location Awareness integration with all relevant services
  - WebComponents -- easy access to XML web services
- ▶ **Maximum Distribution to Internet Connected Devices**
  - Removal of most device and network resource constraints
  - Solution for device fragmentation
- ▶ **Application Model Support for Operator Extensions and Advanced APIs**
- Robust and Extensible Cross-Platform Application Model
- ▶ **Complete Analytic Reporting**
- ▶ **Best Possible User Experience**

Xpressmo has developed a new authoring and delivery platform that enables its customers to rapidly develop and deploy, to a high percentage of internet connected devices, applications and services that will equal or exceed that which is available on the Desktop. As a result, the platform broadens distribution, reduces costs, improves user experience, and provides for high return on investment throughout the entire mobile chain.

Copyright © 2006, 2007, 2008 Express Mobile, Inc. All rights reserved. Express Mobile, Xpressmo and the Xpressmo logo are trademarks of Express Mobile, Inc. All other product names are trademarks of their respective companies.

Graphic 2: PDL Code and Data Compaction



**Portable Description Language**

The PDL can be conceptually viewed as a device, operating system and virtual machine agnostic representation of Java serialized objects. While either designing the application with the Authoring Tool, or programming with the SDK, the internal representation of the programming logic is Java.

At "publish" time this Java code gets translated into the PDL. This translation would also occur in real-time during the execution of any Web Services or backend business logic that interacts with the user.

**Some of the advantages of the PDL as opposed to other languages, are:**

**1: Code and Data Compaction**

Code, as well as vector, integer and Boolean data are compacted and then compressed resulting in a size reduction of 40 to 60 times that of the original Java serialized objects. This is important not only for performance over the network but for utilizing the virtual memory manager of the player more efficiently. See Graphic 2.

**2: Virtual Machine and Operating System Independence**

The reassembled primitives of the application no longer have any dependencies from the original programming language (Java) that they were defined in. The player architecture takes full advantage of this by abstracting all the virtual machine and/or operating system interfaces from the code that processes the PDL.

**3: Extensibility and Compatibility**

Because the PDL is defined by the means of nested arrays of primitives, there are no constraints in terms of extending the player seamlessly as market demands and device capabilities continue to grow. Compatibility with other languages is inherent based on the various player abstraction implementations.

**A: Comprehensive Application Model**

All languages, whether they are programming languages such as C, C#, C++, Java or BREW, scripting languages such as JavaScript, or markup languages such as WAP and HTML, all implement an application model, which is the abstraction of what capabilities the language was designed for.

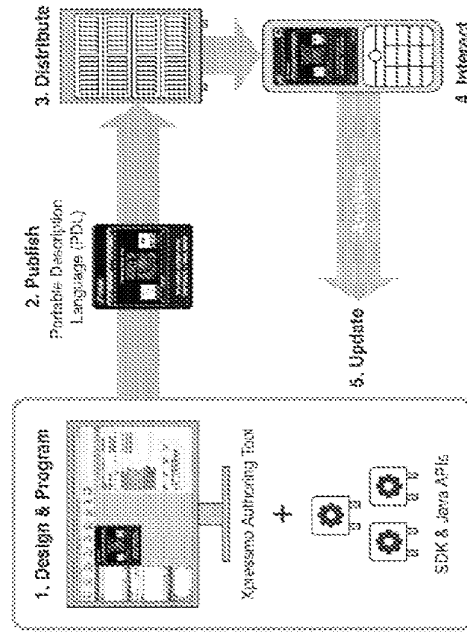
Productivity tools, such as word processors and spreadsheet, also have application models, although they usually have very limited extensibility as compared to programming languages. Typically they sacrifice the robustness of a programming language for ease-of-use.

Xpresso has implemented an authoring platform that supports a comprehensive application model that combines the ease-of-use of well designed productivity tools with the robustness of a programming language.

**Xpresso Application Model Life Cycle**

The implementation of the Xpresso application model is the Portable Description Language (PDL). As Graphic 1 depicts, the PDL is the common language for the Xpresso Authoring Tool, the Xpresso Players at the time the application is distributed, and the Xpresso SDK.

Graphic 1: Application Model Life Cycle



### 3: Advanced Interactive Event Model for Rich, Compelling Applications

The 21st Century Internet is a highly interactive and personalized platform for all consumer, business, and public sectors. Mobile devices must be as interactive and personal as PCs and gaming consoles. This requires timed events, smooth movements, and animations. The Xpressmo delivery platform has the following advanced features built into its authoring environment:

- ▶ User, time and/or location initiated events, which allow content developers to base interactivity on specific user interactions and/or instances in time and space.
- ▶ Timelines and animations, which are critical for timing of multiple events and for a myriad of animations when entering, on, or exiting pages of the application.
- ▶ Waypoints, which act similar to key frames to allow smooth movement of objects within pages of the application. Waypoints define positions on a page object's animation trajectory. When an object reaches a specific waypoint.
- Other object timelines can be initiated, creating location-sensitive multiple object interaction.
- Audio can be defined to play until the object reaches the next waypoint.

### 4: Full Style Inheritance

Both Master Page inheritance (for structural layout inheritance and repeating objects) and Object Styles (for both look and feel attribute inheritance) are supported.

After a style has been defined for an object, the object will inherit the style. Style attributes include both the look and the feel of an object, including mouse interaction, animations, and timelines. For example, if the content developer creates various text objects using a style that sets the font to red, the fonts of these objects will be red. Suppose the developer changes the font color of a specific button to green. If later, the developer changes the style to blue, all other text objects that were created with that style will become blue except for the button that had been specifically set to green.

### 5: Page View, Style, Object, Widget and Application Template Libraries

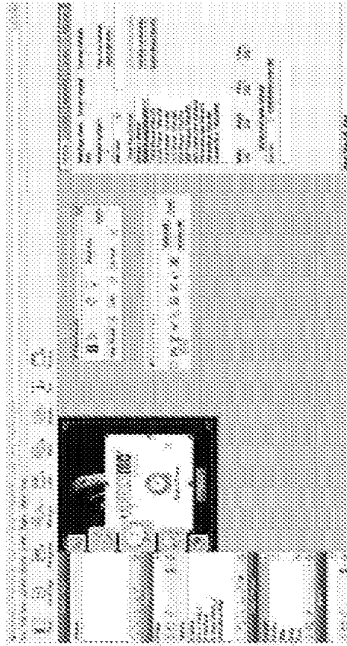
Templates can be created and placed in private libraries to make it easy to retain consistency throughout the application. There are also public libraries that can be made available to all developers. Templates can be used to define the look and feel of the entire application, specific pages, or specific slide shows and virtual tours.

## B: Authoring Platform

### 1. WYSIWYG Full-Featured Authoring Tool

The Xpressmo delivery platform authoring editor provides developers with a what-you-see-is-what-you-get (WYSIWYG) PC- and Mac-based full featured editor. Most of the complicated Java programming functions is presented with easy-to-use pull-down menus, dialog boxes, and buttons.

Graphic 3: Xpressmo Authoring Tool



### 2: Intelligent Interactive Object Libraries for Sophisticated Development Environment

Designers of the Xpressmo delivery platform have taken the most sophisticated and powerful features of advanced web and web services applications, in addition to location-based services (LBS) and photo services, and integrated them within the platform's authoring environment. Now, our solution providers can use the intelligent interactive object libraries to offer their customers and mobile users:

Slide shows, which incorporate images, video, audio, animated transitions, multiple chats, and mouse interaction.

Full 2-D vector graphics.

GIS (advanced LBS), including:

- Multiple raster and vector layers.
- Feature sensitive interactions.
- Location awareness.

Streaming and embedded audio/video.

Virtual tours.

Image processing and enhancement.

Widgets

Chile objects, which can be activated by user interaction and/or time. Uses include:

Mouse over (object selection), hover and fire events.

Launching of object-specific, rich-media experiences.

## C: Web Components

Both Xpressmo customers and third parties can register their web services directly into the Xpressmo Authoring Platform.

### 1: Xpressmo XML Web Component Registry

Customers and third parties can use XML to register their Web Services into the Xpressmo XML Web Component Registry. The Registry will contain:

- All Consumer Inputs related to the Web Service.
- Any Environmental data such as PIM, time or location values.
- Any persistent variable data.
- All outputs related to the Web Service.
- Optional Hinting for improving the Author's productivity.

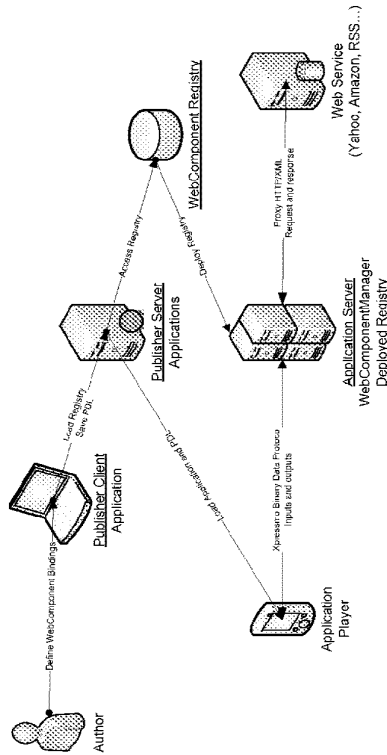


Figure 1 - WebComponentManager Workflow

### 2: Binding to the Author's Application

The XML Web Component Registry for all registered Web Components is loaded into the Publisher Client tool at login time. The author can then assign any of these Web Components to an application without any need to write code. As a Web Component is selected the authoring platform presents the author with WYSIWYG dialog boxes that enable the binding of all the inputs and outputs of the Web Component to GUI components of the application. Multiple Web Components can be assigned to any UI Object or Event in order to facilitate mashups. These UI Object and/or Event bindings, for each instance of a Web Component, are stored in the PDL. The

war file containing a WebComponentManager to handle data from player to web service automatically is deployed as a web application archive to any application server.

### 3: Execution of Web Components

The Xpressmo Player, upon detecting an event in which a Web Component has been defined, assembles and sends all related inputs to the WebComponentManager URL which proxies the request to the web service and returns outputs to the Xpressmo Player. The Xpressmo Player then takes the outputs and binds the data to the UI components in the application.

For example, if the author wanted to use an ATOM feed in their application, they would look at the UI Components available in the Publisher Client, pick the feed they want to use, and bind the output of the feed summary to a textbox. The bindings would be saved into the PDL and processed by the player at runtime. If the ATOM feed does not exist a new one can be added to the Registry that contains all the configuration data required, such as the actual feed URL, the WebComponentManager URL, and what output fields are available for binding.

### 4: The Growing Library Web Components

All Web Components are available either to the author or to the Xpressmo SDK and Java APIs.

Xpressmo offers an expanding set of Web Components. They include:

- Dynamic Binding of Real-time Content
  - ▶ Authors bind inputs and outputs of XML web services to GUI components
- Third Party Web Services
  - ▶ Search (Google and Yahoo)
  - ▶ Maps (MapQuest and Yahoo)
  - ▶ Storefronts (ThumbPlay)
  - ▶ SWS Share (olickale)
  - ▶ Stock Quotes
  - ▶ Facebook
  - ▶ Stock Quotes
  - ▶ Weather (Accuweather)
  - ▶ Movie Trailers
- Xpressmo Server Pages
- Web Services for Communication and Sharing Through Chats and Forums
- Rich Messaging Alerts
  - ▶ Set up Message Alerts which in turn could have Web Components defined within them, including the capture of consumer generated and Web Service supplied rich media and textual content.
- Widgets



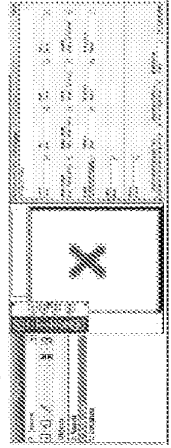
**b: Third Party Web Services**

**Example: Yahoo Maps**

The Xpressmo platform supports both free and premium mapping and traffic web services.

An Xpressmo designer can easily place Yahoo Maps into their application by binding any of Yahoo Map's Inputs and/or Outputs to appropriate Xpressmo UI Objects as seen below.

The graphic on the right depicts a possible result for Yahoo Maps, in which 12 zoom levels can be made available to a consumer with a single touch.

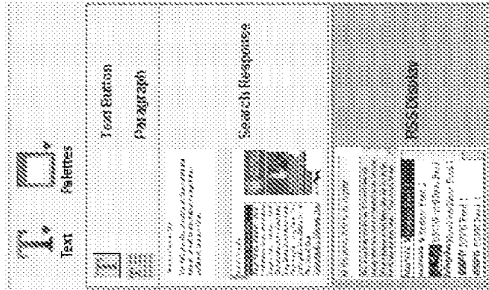


**Example: Web Services for Text, Image and Video Search**

The Xpressmo platform supports both open source and proprietary Search Engines. The author can place a Search Response List in various ways in an application. To the right is a proprietary automated "Search Response list" object that already has established logical links to expose 3 types of metadata. Logically attached to this List can be a large set of available metadata, including:

- Expanded Search
- Narrowed Search
- Search Suggestions
- Search Abstracts
- Image Search Results
- Video Search results

An example of a search response is shown here. Note that a Title, Abstract and a Thumbnail Image are displayed for each item in the Search Response List. This brings the PC desktop search experience to the mobile device.



In addition, the authoring tool contains an easy-to-use dialog box (graphic to the left) that dynamically links objects to data and feeds determined by RSS and chat databases. Any relevant attribute for a page view and/or object can be dynamically bound to a value in a server-side database. This includes elements within complex objects such as:

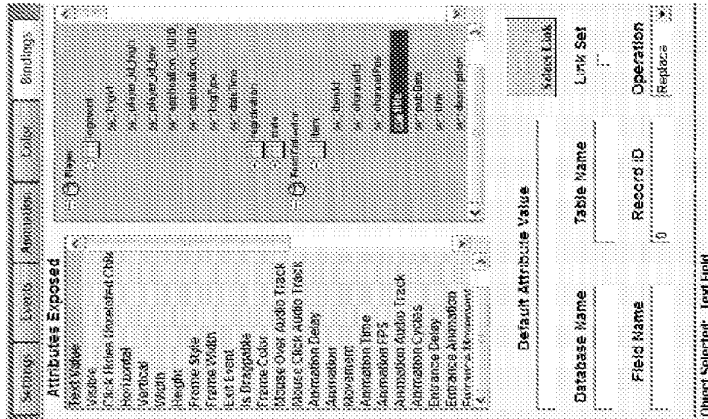
- Any Icon or Text Element within a Graphical List
- Any Icon within a Launch Strip
- Any Feature within any Geographical View of a GIS Service Object
- Any Virtual Room within a Virtual Tour.

**Graphic 6: Xpressmo Dynamic Data Binding Tool**

**a: Dynamic Binding of Real Time Content**

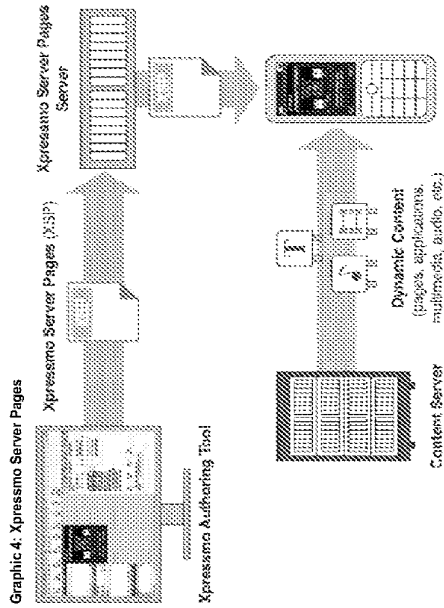
Inserting RSS feeds and other forms of dynamic content into mobile applications is easy with the Xpressmo delivery platform.

The authoring tool comes with a "RSS Display" List which allows to author to select RSS Channels and feeds from an extensible list of available dynamic content. Meta data, such as titles, abstracts and images can be revealed immediately by the user as they traverse this List, bringing the PC experience completely and conveniently to the mobile device world.



**c. Xpressmo Server Pages: Web Services for Real-Time, Shared, and Broadcast Content for Timely Distribution**

Magazines, newspapers, websites, and other forms of published material are based on templates and styles that offer a consistent, familiar look and feel to those viewing the material. These forms of publications offer content that can be accessed (i.e., read) in near real-time (depending upon the frequency of the publication), shared with others (i.e., showing an article to a friend), and broadcast to groups of people (i.e., subscribers). It should be no different for the mobile market. That's why the Xpressmo platform offers Xpressmo Server Pages (XSPs), which, like Java Server Pages, serve objects and dynamic content to individual pages (that have been set with styles and templates) without the need to open other pages.



However, unlike JSPs and ASPs, which are restricted to the functionality supported by the browser, the functionality of XSPs can be remarkably extended through Xpressmo's Java APIs. Also, unlike the slow performance inherent in browser operations, XSPs take full advantage of the compact technology of the Xpressmo platform, resulting in response times usually less than 2 seconds.

With the dynamic binding functionality of an XSP, a page can be saved as a page object in an author's "pages" library, and then can be dynamically populated with real-time content simultaneously as the page is downloaded to a given handset player based on a newly expanded API. Xpressmo Server Pages can also be produced programmatically, but in most cases the Xpressmo delivery platform authoring tool will be a much more efficient way to generate and maintain libraries of dynamically changing XSPs.

With XSPs, pages that have dynamic content built into them can be sent directly to the handsets. Without XSPs, content authors would have to define each page in the application. With the Xpressmo implementation, no pages need to be defined. For example, in a World Cup application, one page could represent real-time scores that change continuously on demand. With polling (for example, a prompt to the users asking who they predict will win a game), a back-end database would tabulate the information and then send the results dynamically to the handsets. With a bar chart, the application would use dynamic Portable Description Language (PDL) with scaling on the fly. For example, the server would recalculate the bar chart for every ten numbers

**d. Web Services for Communication and Sharing Through Chats and Forums**

User-generated content and chats have become the trend in 2006. For example, YouTube.com delivers more than 40 million video views every day with 35,000 new videos uploaded daily.

A USA Today article revealed that MySpace has reached well over 47.3 million members. The Blog Herald also reported that MySpace "is growing by an amazing 180,000 new users a day, according to MediaPost" (comScore Media Matrix, a division of comScore Networks, Inc.).

The Xpressmo platform has integrated a social networking message board technology within its solution. With the Xpressmo mobile chats, mobile handset users can communicate within groups of communities. Entrepreneurs and artists can create personalized Chat Rooms, and sport teams can create team chats. Other benefits target MVNOs and media companies.

**MVNOs**

- Increase average revenue per user (ARPU).
- Increase upgrades and premium service plans.
- Expand subscriber base with incremental users sharing photos and chats.
- Leverage photos and chats with seasonal marketing events.

**Media companies**

- Build community and capture new users with chatting.
- Increase consumer interaction with brands/properties.

A tight and seamless integration of all phone communication functions with any robust, mobile social networking application will be an essential prerequisite for market adoption. This means that communication functions should be:

**Easier**

The number of touches should be less (certainly no worse than equal) than that of using the phone communications UI directly.

**More Robust**

The communication choices should be an appropriate superset of what the phone already offers.

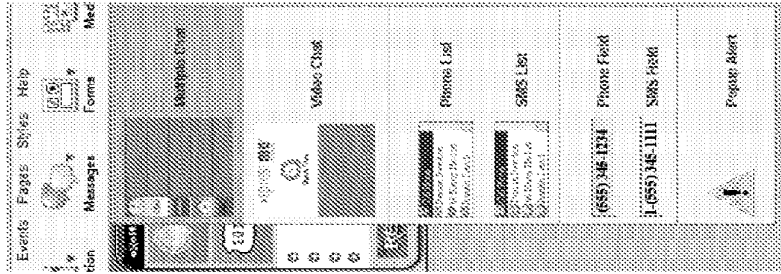
**More Entertaining**

The communication choices should offer appropriate imagery, video, text, and audio to make the user experience more compelling.

**Personalization**

The user should be able to adjust the communication metaphor to whatever they feel works best for them.

In addition, the content that is available for messaging and social networking should only be limited by the imagination of the user. This means that any content capturing functions on the device must be seamlessly supported. Such content can be customized as the user wishes. Messages can be easily assembled based on any content that is available to the user. Messages can easily be shared with any individual and/or group.



Communication metaphors must be state-of-the-art and easily available to both the authors and the users of the Xpressmo platform. Two examples of this are depicted in Graphic 11 above.

#### **Video Chat**

A chat session can be conducted simultaneously, inside an integrated page view, with a video or television station.

#### **Multiple Chat**

Multiple chat sessions, each with a designated individual and/or group, can be conducted simultaneously, with each of the chat threads visible inside an integrated page view.

#### **e. Web Services for Widgets**

An extension of an XSP is a Widget Object. Widgets are supplied with the following services:

Widgets can be developed from numerous sources:

- The Xpressmo Authoring Tool
- A Consumer Publishing Tool
- An XML to Widget Conversion Tool

The Xpressmo SDK Widget Libraries are automatically populated and managed.

Widget Selection Lists are available and can be populated with author defined icons.

## **D. Java APIs and the Xpressmo SDK**

The authoring platform also includes the following components:

- Xpressmo SDK for Custom Code Development.
- A full set of Java APIs are included in the Xpressmo System Development Kit (SDK) to make it easy to add extensions and functionality to mobile applications, and tie applications to back-end databases through the Xpressmo Content Server.
- As mentioned under Section C, Xpressmo offers an expanding set of Web Services, available through the Xpressmo SDK.
- A Web Services Interface to SOAP/XML enabled Web Services (Graphic 5)
- An RSS/Atom and RDF Feed Collector and Content Gateway (Graphic 6)

### **1: Xpressmo SDK and Java APIs**

The Xpressmo SDK will work with various popular IDEs including Eclipse. Available through the SDK is a large and powerful set of APIs and interfaces that permit the seamless extension of any application to back end business logic, web services, etc. These interfaces and APIs support:

#### **a. Listeners.**

There is a large set of listeners that expose both player-side events and dynamically linked server side data base events. Some examples are:

- Player-side Forms-based content
- Player-side User Interactions
- Player-side Object Status
- Server-side driven dynamic content events.

#### **b. Player-side Object Operations**

A superset of all authoring tool functionality is available through APIs. They include:

- Page View Level APIs for inserting, replacing, and/or modifying any page object.
- Object Level APIs for modifying any attribute of existing objects, adding definitions to attributes, and adding, hiding or replacing any object.



## E: Intuitive Navigation

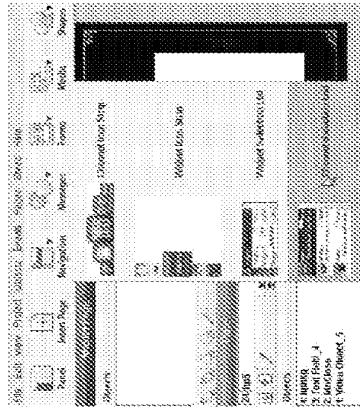
Efficient Platform User Interface with Touch Minimization for Optimizing Form Factors and Navigation  
The Xpressno delivery platform offers a very powerful and broad set of extensible navigation objects, as well as object- and pointer-navigation options to make it easy to populate small mobile device displays with content and to navigate easily.

among page views  
within applications  
between objects in a single page view of an application.

### Navigation Objects

Navigation objects include various types of launch strips, various intelligent and user-friendly text fields and scrolling text boxes, powerful graphical complex lists, as well as Desktop level business forms. In fact, every type of object can be used for navigation by assigning a navigation event to it.

The authoring Tool offers a list of navigation Object Templates, which then can be modified in numerous ways. (Graphic 6)



## Launch Strips and Graphical List Templates

### Launch Strips (Graphic 9)

Launch strips can be designed by the author with a most, no restrictions. They can be stationary or appear on command from any edge of the device, their size, style, audio feedback and animations can be freely defined to create highly compelling experiences.

In the example to the left a portal type Launch Strip becomes visible from the left edge when requested, while a channel type Launch Strip, which may be application specific, becomes visible from the bottom when requested. The channel type Launch strip could have an appropriate sound effect for each channel when being selected, as well as popup bubble help.

### Graphical Lists (Graphic 10)

As seen in the examples below, graphical lists can contain items with many possible text and image elements. Each element can be defined at authoring time and/or populated dynamically through either Xpressno's Web Services or APIs.

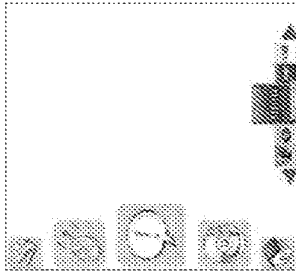
### Assignable Navigation Events

All objects, and/or all elements within an object, can be assigned navigation events that can be extended to Xpressno's Web Services or APIs. For example, a Rolodex type of navigation event can dynamically set the starting slide of the targeted page view (or the starting view of a targeted application).

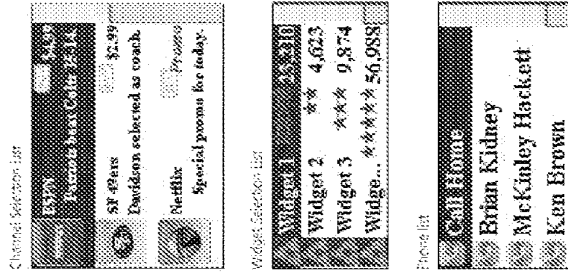
### Intuitive Object and Pointer Navigation

The Xpressno platform has created options to choose between object and Mouse navigation for other devices. Object Navigation permits selection of objects with a Joy Stick and/or Cursor Keys in all 4 directions. When within a complex object the navigation system automatically adapts to the navigation needs for that object. For coordinate sensitive objects such as geographical information services (GIS) and location-based services (LBS) or virtual tours a soft cursor appears. For Lists, scrolling text areas and chats, Launch strips, and slide shows the navigation process permits intuitive selection of elements within the object. Scroll bars and elevators are optionally available for feedback. If the device has a pointing mechanism then scroll bars are active and simulate the desktop experience.

Graphic 9: Launch Strips



Graphic 10: Graphical Lists



## F: Advanced Feature Support

### 1: Personalization and Temporal Adoption

Personalization has become a prerequisite for any comprehensive consumer-facing mobile solution. It consists of two customization methodologies.

#### Adoption

The choices, navigation options, etc. are based on user usage patterns.

#### Customization

The user can select which skins, choices, layouts, dynamic content, widgets, etc. are available either through a customization on the phone or one that is on the desktop but dynamically linked to the user's other internet connected devices.

Personalization also must be extended to include temporal adoption. That means that the skins, choices, layouts, content, widgets, etc. are further influenced by:

#### Location

Information requirements will significantly change based on whether the user is at home, at the office or when traveling.

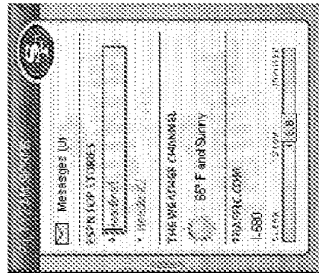
#### Time of Day

Information requirements could also significantly change by the day of the week, which season of the year, and the time of day.

For example, a possible phone top layout for a weekday when a user wakes up might look like Graphic 12 to the left. Any messages received overnight would be flagged, the user's favorite RSS sports feeds would be visible, today's weather forecast would be available, and the current traffic conditions between the user's home and office would be graphically depicted.

#### Persistent Storage

In order to support many personalization functions there must be a convenient method for maintaining, both within a user's session, and between sessions, memory about various user choices and events.



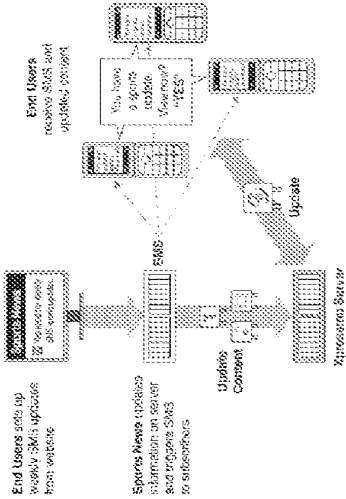
Graphic 12: Temporal Example

### 2: Push Capable

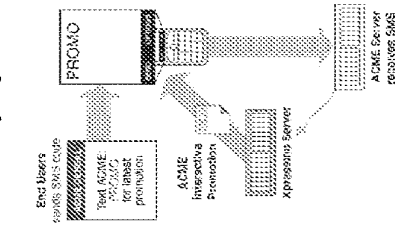
Most agencies and brands want more than just rich, interactive content. They want to reach, acquire, and retain customers through the mobile channel. This requires a sophisticated back-end technology. The Xpressmo platform works with both.

For example, short codes can be applied to cereal boxes and beverage containers, and SMS text fields can be applied to promotional websites. In either case, a consumer can text to an Xpressmo SMS server, which then serves the appropriate application link back to the consumer. (Graphics 13A and 13B)

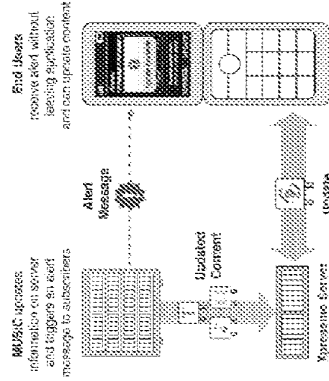
Graphic 13A: Content Pushing using SMS Example 1



Graphic 13B: Content Pushing using SMS



Graphic 14: Content Pushing using Rich Media Multi-layer Messaging



Example 2


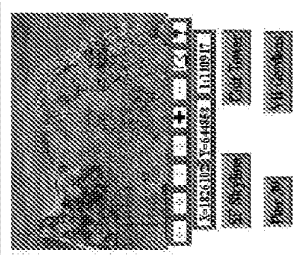

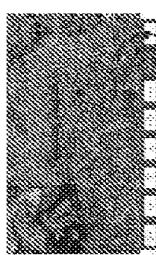

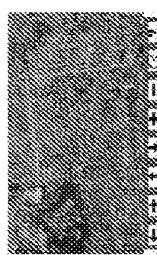
(Graphic 14) A more powerful example is utilizing the Rich Media Multi-layer Messaging Architecture so that powerful and entertaining messages can be broadcast to any logical group of devices. The consumer is notified in real-time of the pending alert and can view and interact with the message without interrupting the current application and view.

**Example: Location Awareness Integration with all Relevant Services**

More and more social networking applications require the use of geographical information services (GIS) and location-based services (LBS) as integral components. For example, a social networking application may detect the location of "friends" within a certain radius of the mobile device using GIS and phone triangulation (location based on the three nearest mobile cell towers near the respective phone) or through the GPS receiver on the phone. GIS images may be used for mapping to theaters or restaurants, detect optimal traveling routes based on traveling conditions, etc. A broad array of mobile GIS web services have been built into the Xpressmo delivery platform for location-aware applications.

Graphic 16A and Graphic 16B are two examples of possible implementations.

**Graphic 16A: LBS Optimal Travel Route**

<p><b>Step 1</b> Select the  icon or fire after moving a soft cursor to zoom in to the desired geography.</p>	
<p><b>Step 2</b> Request directions from two User-Defined geospatial points. Select the  icon and move cursor to a desired start point 'A'. Move your cursor to a desired end-point 'B'.</p>	
<p><b>Step 3</b> To get the directions, select  icon again.</p>	

**Result**  
You will start off with a raster image of a map with two points selected. The optimal route will be displayed as a green vector graphic on top of the raster image of the map. You can zoom in closer to the optimized directions for a closer look. You can also see points of interest, by viewing a 360-degree virtual tour. See Graphic 17: LBS Enhanced Services

**3: Tight Integration with Phone Functions**

As described above as a prerequisite for messaging and social networking, there are a large and expanding set of phone based resources that must be tightly integrated with any consumer facing application to be a viable robust solution. This list will continue to grow, but it already includes:

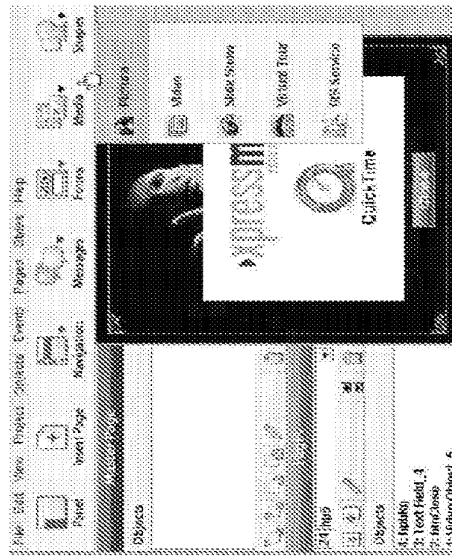
- Placing and receiving phone calls
- Access to the phone's PIM data
- Access to the phone's camera

**4: Mashups**

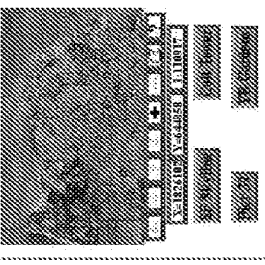
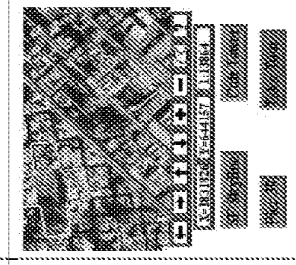
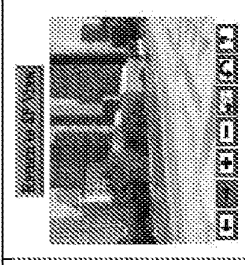
The efficient and robust integration of different web services to create a value-added seamless user experience is an inherent attribute of the Xpressmo Authoring Platform. Some of the facilities that support mashups are:

- The Feed Collector**  
Utilizing the Xpressmo supplied rules based populator, dynamic content from different feeds and/or web services can be logically consolidated.
- Extensible Attributes for Complex Objects**  
Slide shows, virtual tours, GIS Service objects, and other objects, can consolidate content in a logical way, whether it be text, video, audio, pictures and/or vectors.
- Xpressmo Supplied Web Services**  
Services such as dynamic binding and Xpressmo Server Pages further help facilitate the mashup of logically related dynamic content.

**Graphic 15: Web Services Objects**



Graphic 16B: LBS Enhanced Services Demo

<p><b>Step 1</b> The user could, as a short cut, select a point of interest through a button or any other navigation object.</p>	
<p><b>Step 2</b> Alternatively, the user could select the icon or fire after moving a soft cursor to zoom in to the desired geography so that the point of interest becomes visible and is selected. Or the user could enter in or select an address. The user could then fire on the button labeled "Take Tour".</p>	
<p><b>Step 3</b> Immediately, a virtual tour of the point of interest becomes available in which the user can now pan and zoom throughout the virtual representation of the point of interest. Alternatively, if a video cam is operational, a live video feed could be displayed.</p>	

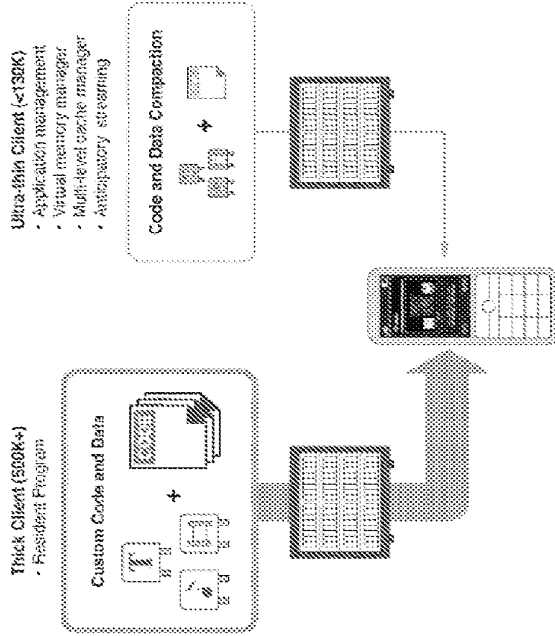
## G: Maximum Distribution to Internet Connected Devices

### 1: Removal of Most Device & Network Constraints

#### Thick vs. Thin Client

Most current implementations that support rich and highly interactive user experiences are computer programs that are written either to the Operating System (OS) or Virtual Machine (VM) of the device. From a development point-of-view this is a far quicker path to take. However, it means that at significant changes to the application require reprogramming the application and then complete the required O/A cycle, a process that can take several months. Also, for feature phones, the entire application must be downloaded and placed into the limited heap of the device. For any robust and full featured application this design will, at best, quickly restrict its use to a very limited number of devices.

Graphic 17: Thick vs. Thin Client





**Graphic 18: Ultra-thin Client Engine and its extensions**

The Xpressmo Platform is based on an eight year R&D effort, and utilizes an ultra-thin client architecture. The Xpressmo Client is light weight and extends the OS and/or VM of the device to:

- Extend the capability of the device to that of a desktop computer
- Manage all applications and application upgrades
- Rescue device, OS, VM and language fragmentation

A thin client architecture requires that a language be adopted that manages resources efficiently, is extensible, supports a robust application model, and has no device specific dependencies.

**Virtual Memory and Multi-Level Cache Architecture**

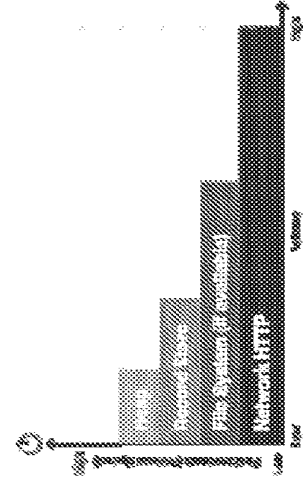
**Virtual Memory**

Desktop systems all support virtual memory by using a physical page model with supporting hardware. Xpressmo has implemented a logical page virtual manager. This architecture requires no supporting hardware and works efficiently with constrained devices. All page view images, which could span multiple applications, are placed in a table as highly compacted and compressed code. A typical page view will range from 500 bytes up to about 1,500 bytes. When rolled into the heap and instantiated this code increases to the more typical 50,000 up to 250,000 bytes. Additional alert pages may also be rolled into the heap and superimposed on the current page view. Any changes to any page currently downloaded are placed in a tightly compact change vector for each page, and rolled out when the page is discarded. Note that whenever an application is visited that had previously been placed in virtual memory the Server is interrogated to see if a more current version is available, and, if so, downloads it. This means that application logic can be changed in real-time and the results immediately available to mobile devices.

**Anticipatory Streaming and Multi-Level Caching**  
To operate efficiently with the bandwidth constraints of mobile devices, the Xpressmo delivery platform also features anticipatory streaming™ and multi-level caching. Anticipatory streaming looks ahead in the current application to see if there is content that is likely to be visited in yet untouched page views.

With the multi-level caching system, mobile applications are bounded so that the handset don't run out of memory. Multi-level caching is a memory management system with results similar to embedding, without the overhead of instantiating the content. In other words, with multi-level caching, handset users get an "embedded" performance without the embedded download. Multi-level caching determines the handset's heap through an API, and also looks at the record store to see how much memory is resident. This content is placed in record store and/or the file system, and may, if there is available heap, also place the content there as well. Note that when content is flagged as cacheable and is placed in persistent storage, a digital rights management (DRM) solution will be used.

**Graphic 20: Memory Speed vs. Volume**

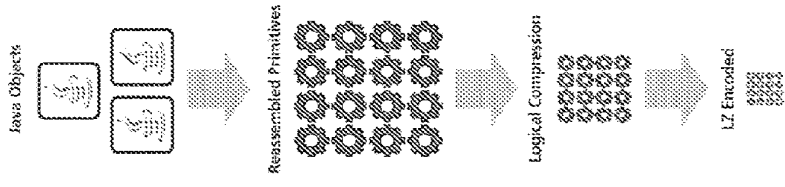


**Code and Data Compaction**

The Xpressmo delivery platform uses compaction to transform the code and data in an intelligent way while preserving all of the original classes, methods and attributes. This requires both an intelligent server engine and client (handset) player, both of which fully understand what the data means and how it will be used. The patentec Xpressmo compaction technology includes transformation algorithms that deconstruct the logic and data into their most primitive representations, and then reassembles them in a way that can be optimally digested by further compression processing. This reassembled set of primitive representations define the Xpressmo PDL.

Prior to compression the code has already been transformed so that:  
There are no dependencies on the original programming language (.java)  
The code and data have been reduced by 4 to 10 times

**Graphic 21: Code and Data Compaction**



Compression has two distinct phases. The first takes advantage of how the primitive representations had been assembled, while the second utilizes standard LZ encoding.

The final result is an overall reduction of 40 to 100 times the original size as represented by Java serialized objects.

The player, when preparing a page view for execution, will decompress and then regenerate the original objects, but this time in compliance with the programming APIs exposed by that device.

## 2. Solution for Device Fragmentation

### Response Director for wide coverage of the wireless device market

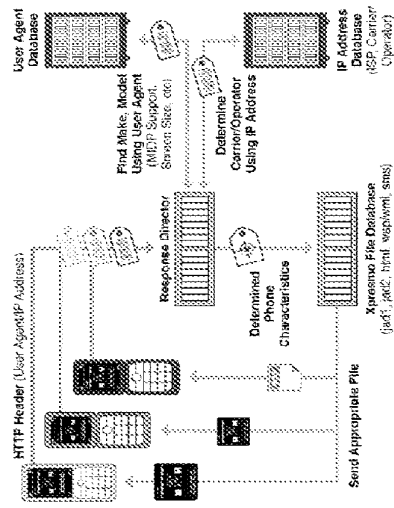
The heart of the Xpressmo deployment platform resides in its Response Director, which determines a user's handset, fetches the correct application from different databases, and delivers a respective highly compressed application in a Postscript-like Portable Description Language (PDL) format over the air (OTA).

With the Response Director, any mobile device can be serviced, and the most appropriate application for the device will be delivered to the device, based on the characteristics of the device.

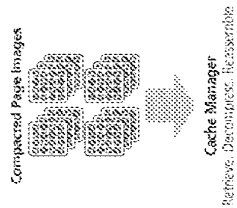
#### How the Response Director Works

Let's take a look at how it works. First, an SMS message is generated to the mobile device, which then automatically sends an http/7 stream that includes handset information and its phone number to the Response Director. The Response Director then looks at a field in the http header (which includes the user agent and IP address) that identifies the browser (i.e., the User Agent). The User Agent prompts a database lookup table that brings back a slew of data (for example, make, model, attributes, MIDP 1.0 MIDP 2.0, WAP). The database also distinguishes the same models from different countries. There is another IP address database, which identifies the carrier/operator. A decision tree determines which application to fetch and send to the handset.

Graphic 23: Response Director



Graphic 22: Decompression



Another Xpressmo database contains the data types such as jad1, jad2, html, wmlwap2, and so on. A list of available applications are returned to the decision tree, which then returns, to the handset, the application that is appropriate for the respective handset. For each file type, there is an attributes list (e.g., streaming video, embedded video, streaming audio, etc.) so that there is enough information to determine what to send to the handset.

If there is an application that has a data type that the handset cannot support, for example, video, the Response Director sends an alternative application to the handset, for example one that has a slide show instead. If the device cannot support a slide show, an application might have text and images and display a message that indicates it does not support video.

Another powerful feature of the Xpressmo Response Director is its exposed API from the decision tree that permits Xpressmo solution providers to override the default output of the decision tree. Xpressmo solution providers will be given a choice of applications and then can decide to use the defaults or force other applications.

### Device Scaling for Reduced Fragmentation

One of the most visible forms of fragmentation resides in the various form factors of wireless, and particularly mobile, devices, which range from 128x128, 176x208, 240x260, 320x320, and many other customized sizes in between. Often, developers must create hundreds of builds for one mobile application.

The Xpressmo platform automatically scales applications at publishing time to various form factors to reduce the amount of fragmentation among devices, and the Response Director serves the appropriately scaled version to the device. For example, a QVGA application will automatically scale to the QCIF form factor.

### Player Abstraction and Device Adaptation for Eliminating Fragmentation

The Xpressmo player architecture involves an abstraction interface that separates all device, operating system and virtual machine dependencies from the player's application model business logic. The advantages of this abstraction interface are:

- Porting to other operating systems and virtual machines is far more efficient.
- Adding extensions to the application model and PDL can be implemented once and then seamlessly propagated across all platform implementations.
- Less robust platforms can be augmented by extending higher end capabilities inside that platform's abstraction interface implementation.

The players also extend the power of the Response Director by adapting the application to the resources and limitations of any particular device.

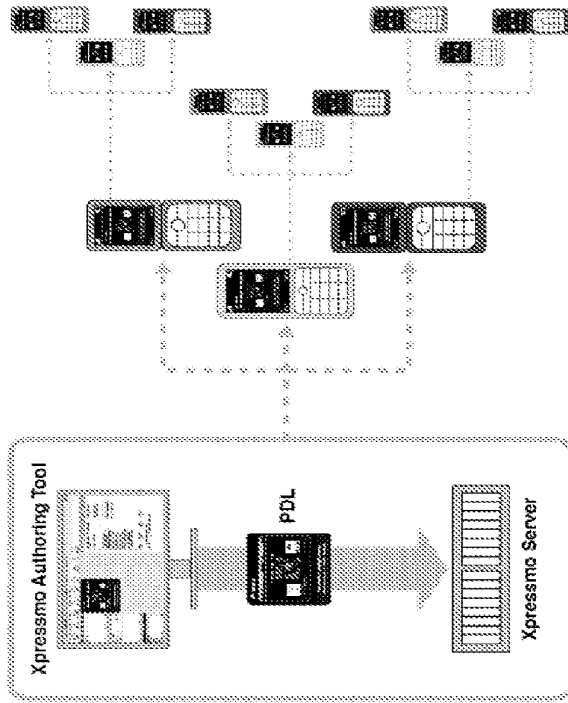
Some of these areas of adaptation include:

- The speed of the device's microprocessor.
  - The presence of device resources such as cameras and touch screens.
  - The Heap, Record Store and File System memory constraints.
- For example, the player will automatically throttle down an animation to the frame rate that the device can handle so that the best possible user experience is preserved.

**Automatic Deployment Around the Globe**

Finally, the Response Director can automate the deployment of applications across national boundaries, supporting both traditional and double-byte Asiatic languages.

**Graphic 24. Reduce Fragmentation**



**3: Application Model Support for Operator Extensions and Advanced JSRs**

Over time, the operators, device manufacturers, operating system and virtual machine vendors, responding to market demand, will increase the robustness of the IP Stack. This is already being seen as numerous optional JSRs have been released that extend the power of the Java MIDP virtual machine, and operators such as DoCoMo, Vodafone and Sprint have released numerous extensions that are specific to their networks.

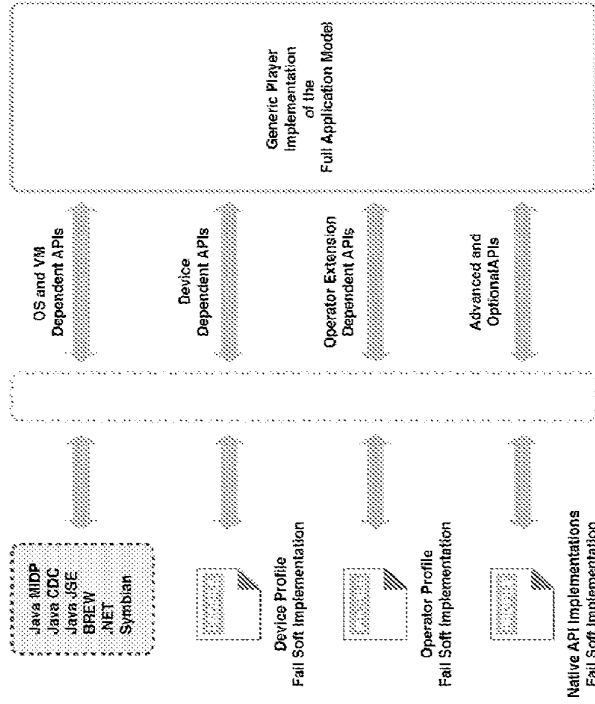
Xpressmo has a facility, when it makes business sense, to extend its application model to support these new capabilities, without fragmenting application deployment over legacy devices that do not support these capabilities. Generally, this is accomplished by using the following criteria to determine which new capabilities to add to the application model.

Is this capability useful to a broad segment of the market?

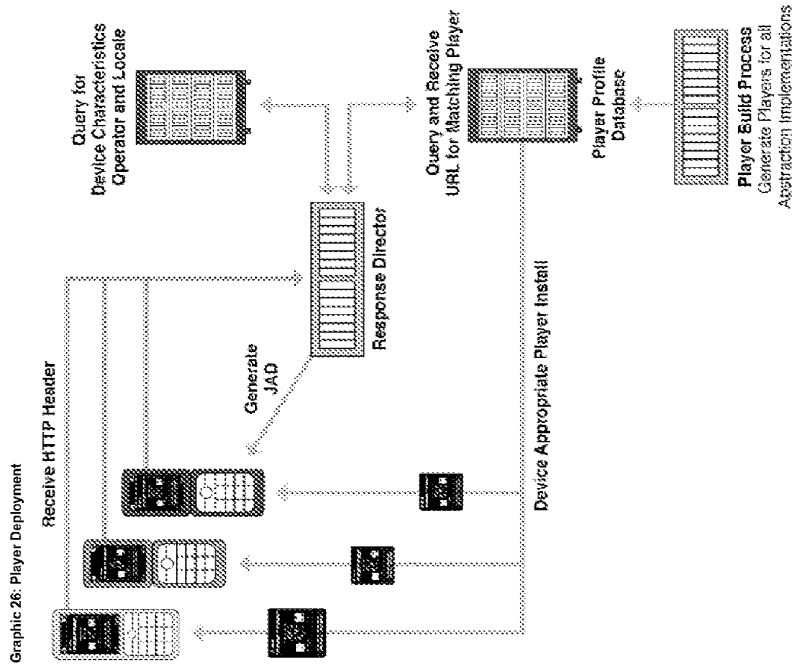
Is there a reasonable comparable or fail-safe implementation that can be added to the players for devices that do not support this capability?

The processes below are largely automated in order to efficiently generate and support an expanding data base of Player Profiles.

**Graphic 25. Player Profiles**



As discussed previously, the Response Director manages the deployment of all players.

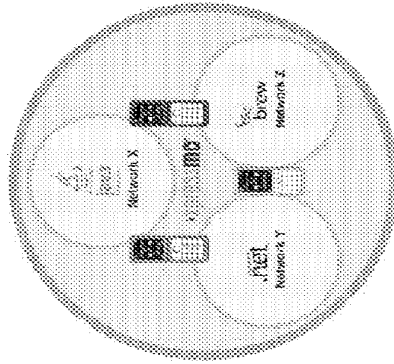


This does not preclude the support for advanced capabilities when there is no reasonable way to implement some reasonable representation on legacy devices. It does, however, infer that Legacy devices will, when receiving PDL instructions that it cannot support, ignore those instructions. Xpressmo, or its customers, may release advanced versions of the Xpressmo Platform targeted for these more capable environments.

#### 4: Robust and Extensible Cross-Platform Application Model

it is not possible, with modern technology, to have a solution for deploying rich intelligent content to the wide array of 21st Century Internet applications that span set-top boxes, game consoles, PCs, and mobile devices without the implementation of a well conceived Application Model as the conceptual framework for all applications and services.

Graphic 27: Cross Platform Application Model



## H: Complete Analytic Reporting

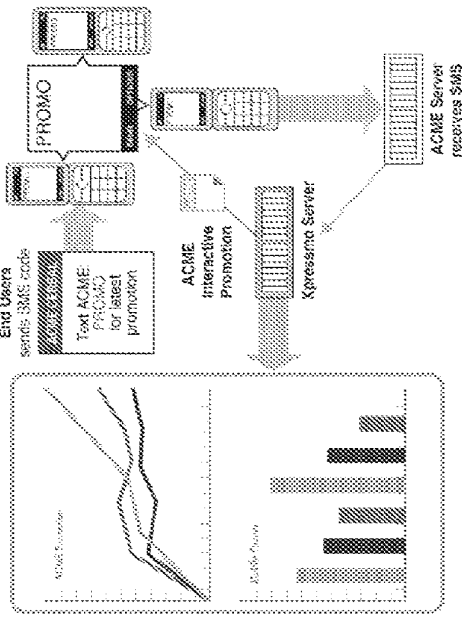
Analytics can be tracked and reported by:

- Application
- User Demographic
- Time of Day
- Location

In addition, the type of analytic content is really only limited by which listeners have been activated for which objects and for which pages. Some of the choices are:

- Player-side Forms-based content
- Player-side User Interactions
- Player-side Object Status
- Server-side driven dynamic content events

**Graphic 26: Reporting and Management**



## I: Enhanced User Experience

The quality of the user experience will be determined by a number of factors. All of the sections above address various components that will affect the user experience. In summary, they are:

### Minimized Handset Response Time Delay

This has been addressed by the Xpressmo Cache and Virtual Memory Manager, anticipatory streaming, compaction of code and data, and the design of the Xpressmo player.

### Navigation

This includes the various launch strips, graphical lists, intelligent test objects, and the overall power the Xpressmo Event Manager Model.

### Entertaining Presentation Layer

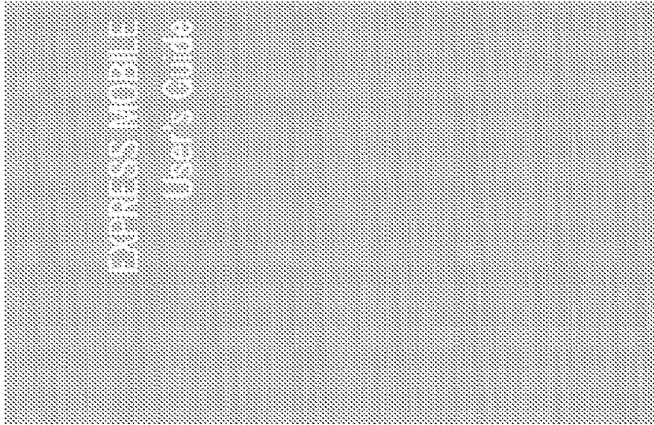
By empowering the designer and UI Engineer with the Xpressmo Publishing Tool, by offering, through massive analytic reporting the behaviors of the consumer, and with the rapid iteration capability to continue to adapt the presentation layer to consumer needs, the likelihood of success goes way up while the risk of failure is greatly minimized.

### Personalization

With both extensive customization choices available to the consumer on both the phone and desktop, and with the adaptive technologies based on actual usage, the consumer will find the experience familiar, pleasing and intimate.

### Coverage

Although this discussion has been primarily focused on the feature phone, the consumer already is connected to the internet with office and home PCs, through the Television and its STE, and possibly through other mobile devices such as laptops, PDAs, Blackberries, etc. Since the Xpressmo platform can operate on all these devices, and can create an integrated and adaptive experience that will follow the consumer anywhere in this interconnected world, the internet world becomes human centric, not device centric.



## Getting Started

Installing the Publisher	01.04
Log In	01.05

## Chapter 1: Overview of the Interface

A: Menu Bar	01.05
B: Toolbar	01.05
C: Canvas	01.12
D: Layer Inspector	01.12
E: Resource Inspector	01.16
F: Page Viewer	01.17
G: Palette Docking	01.21

## Chapter 2: Working with Pages

A: Master Page	02.04
B: UI Implementation	02.06
C: Application Pages	02.08
D: Alerts	02.14
E: Libraries	02.16
F: Saving Files	02.17
G: Enhanced Navigation Specification	02.18

### Chapter 3: Introduction to Objects

A:	How to Create	03.05
B:	Keyboard Operations with UI Objects	03.05
C:	Persistent Variables	03.06
D:	Layout by Pixel vs Layout by Font	03.15
E:	Suppression of Selection Rectangle	03.17
F:	Using Object Libraries	03.18

### Chapter 4: Working with Navigation Objects

A:	Creating a Launch Strip	04.04
B:	Working with Launch Strips	04.04
C:	Working with Complex Lists	04.07
D:	Inter-Application Navigation	04.08
E:	Bubble Help Color	04.08

### Chapter 5: Working with Message Objects

A:	Working with Chat	5.04
B:	Working with Phone Calls	5.07
C:	Working with SMS Messages	5.09
D:	Working with Rich Media Alerts	5.11

### Chapter 6: Working with Business Forms

A:	Introduction	06.04
B:	Text Fields	06.05
C:	Always On Text Fields	06.06
D:	Lists and Dropdowns	06.09
E:	Text Entry	06.23

### Chapter 7: Working with Media Objects

A:	Pictures (Bitmap Images)	07.04
B:	How to Edit Using Photo Enhance	07.05
C:	Slide Shows	07.09
D:	Video	07.16
E:	LBS Maps	07.19

### Chapter 8: Working with Vector Shapes

A:	How to Create	08.04
B:	How to Create a Group	08.06
C:	How to Edit	08.07

**Chapter 9: Working with Text Objects**

- A: How to Create 09.04
- B: How to Edit 09.05

**Chapter 10: Working with Child Objects**

- A: How to Create 10.04
- B: How to Select 10.06
- C: Behaviors 10.07

**Chapter 11: Working with Web Components**

- A: Supported Objects 11.05
- B: How it Works 11.07
- C: Behaviors and UI 11.10
- D: Adding a Web Component 11.11
- E: Completing a Web Component with the "Generate UI" Option 11.13
- F: Completing a Web Component with the Existing UI Objects 11.19
- G: Persistent Variables 11.23
- H: Enhancing Web Components with Auto-Generated Persistent Variables 11.25
- I: Restrictions for Editing UI Objects related to Web Components 11.29

xpressmo.com

- J: Working with Control Lists 11.31
- K: Working with Collection Lists 11.38
- L: Collection List Keys and Complex List Indexes 11.40
- M: Sharing a Web Component 11.42
- N: How to create a Web Component Model 11.47

**Chapter 12: Working with Object Styles**

- A: Introduction 12.04
- B: Object Styles 12.05

**Chapter 13: Animations & Timelines**

- A: Introduction 13.04
- B: How to Define Animations & Behaviors 13.06
- C: Child Object Animations & Behaviors 13.10
- D: Defining Object Entry Timelines 13.11
- E: Defining Object Exit Timelines 13.12
- F: Tips 13.13

xpressmo.com



## Chapter 14: Working with Events

A: Child Object Events	14.04
B: Exit Events	14.05
C: Select and Fire Events	14.10
D: Waypoints	14.13

## Chapter 15: Testing Your Application

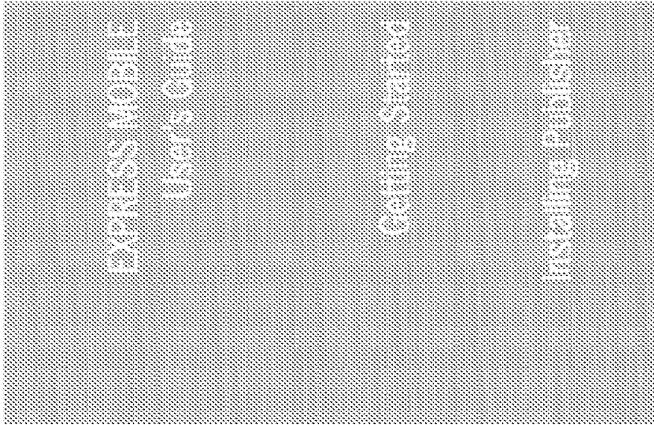
Testing an Application created in Publisher	15.04
--	-------

## Appendix A: Mapping for Direct Text Entry

## Appendix B: Object Animations

## Appendix C: Web Component Models

## Appendix D: Landable Attributes



## Getting Started

Installing the Publisher	01.04
Log In	01.05

## Getting Started : Glossary

JRE	Java Runtime Environment (JRE)
QuickTime	Apple's popular and comprehensive Video and Audio Player. Compatible with Java.

## Installing Publisher

Publisher Client can be installed on Windows XP SP2 or higher, and on Mac OS X 10.1 or higher. Please consult your System Administrator about the compatibility of your machine's OS. If you wish to run Publisher on a machine that does not already have Publisher installed, obtain a copy of the Publisher Client installer from your System Administrator. Also, confirm that the account you are using on your machine has the privileges required to install a new application.

Make sure that the following support software is installed and is the correct version for the Publisher Client which you plan to install:

- Java Runtime Environment (JRE)
- QuickTime
- Operating System

For details on installation requirements and procedures, please refer to the document "**Publisher 2.1 Client Install Guide**".

## Log In

Once Publisher Client is installed, open Publisher by finding the desktop icon named "Publisher Client" or by selecting from the taskbar Start → All Programs → Publisher Client → Publisher 2.1. You should see a log in window as shown here.

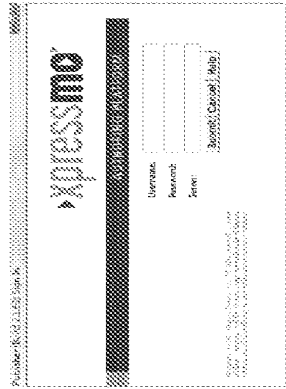


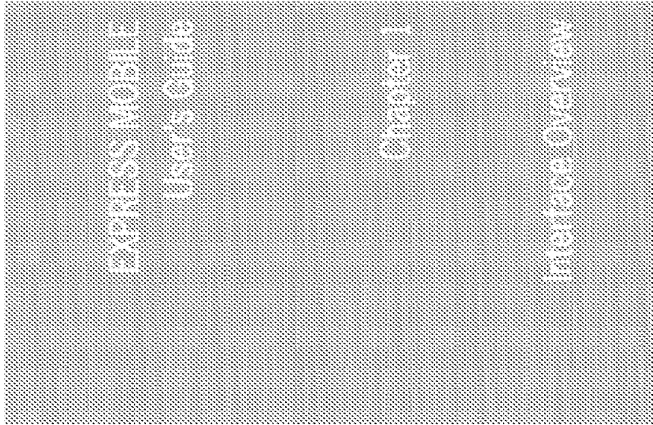
Fig A1

Obtain the following information from your System Administrator prior to first log in:

- Username
- Password
- Server to which the client connects

Select "Submit" once you have entered the required information. In a few seconds, you should see the Publisher canvas page.

On subsequent log ins, Publisher client will remember the last used username and server, so all you will need to type in is your password.



### Chapter 1: Overview of the Interface

A:	Menu Bar	01.04
B:	Toolbar	01.04
C:	Canvas	01.11
D:	Layer Inspector	01.11
E:	Resource Inspector	01.15
F:	Page Viewer	01.16
G:	Palette Docking	01.20

## Chapter 1: Glossary

**Application Page**

A page that is available to be visited through any navigation event. Application Pages inherit all the attributes of the Master Page, unless that attribute is specifically changed during an editing session.

**Master Page**

The initial definition for all application pages. Any additions or changes to the master page will be immediately reflected in all application pages, unless prior editing of that application page modified that particular attribute.

**UI Parent object**

Any object that is not a child object. Parent Objects can be selected when running the application.

**UI Child Object**

Any object that was created by first selecting a Parent UI Object and then creating a Child Object. Child objects are always part of the same drawing layer as its parent object, but will draw first. Child objects themselves are not directly selectable when running the application.

**Author's Private Page Library**

The author can at any time save any application page into their Private Page Library. This library is restricted to the author that created this library, and is available during any subsequent editing session and these pages can be imported into any application.

**Xpressmo Public Page Library**

This library contains pages that are available to all authors of the organization that is licensing the Express Mobile Publisher Platform.

**Page List**

The Page List is part of the Layer Inspector. It displays all the pages currently available in the application.

**Edit Page Dialog Box**

This dialog box offers all the choices that is available through the Resource Inspector of a Page, but these changes are not applied to the page until the Edit Page Dialog Box operation is completed.

**Style Object**

Style objects contain the initial default values that will be applied to any UI Parent or UI Child Object when it is first created. Any changes to a style object will be reflected in all the UI Objects that are attached to this style object, unless, during an editing session, that particular attribute had been changed in the UI Object.

xpressmo.com

**Dynamic binding**

This is the technique that permits the author to connect many attributes of UI Objects to a particular data base field on a Server. When running the application, the current value in the referenced data base will be immediately applied. During the application session, any other real time changes to these values in the referenced data base will again be immediately displayed.

**Resource Inspector**

The primary interface Panel for interactively working with UI objects that have been placed on the Canvas.

**Page Viewer**

One or several floating palettes that provide direct feedback and interactions with a set of objects and functions. The Page Viewer will, in various ways, draw one, some, or all the current application pages at the same time as the current canvas is available.

**Hot Zone**

This is the space in the current defined view port (Page Size), that, if Page Scroll Bars were drawn, they would lie in this hot zone. This helps the author to place objects so that the need for Page Scroll Bars is minimized.

xpressmo.com

### Publisher Interface

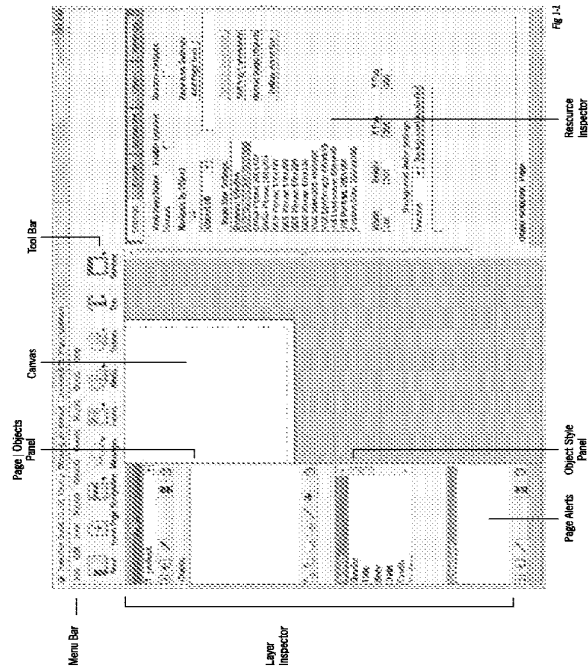


Fig 1.4

#### A. Menu Bar

The Menu Bar offers a comprehensive way to work with the Publisher.

#### B. Toolbar

The Choices under the Toolbar, including the floating palettes, represent a convenient way of accomplishing the same functions as the Menu Bar.

1. Add pages to the Canvas

With your mouse Select the Insert Page Icon to bring up the "Insert Page" dialog box. If Append is not checked then the page will be inserted in front of the current page that is in the Canvas. The Page name must be unique for the current application. If not, then the name will turn red until it becomes unique again.

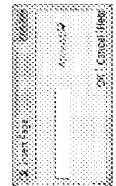


Fig 1.2

2. Add UI Objects to any page in the canvas

The Toolbar exposes various categories of UI Objects, all of which can be selectively placed on the Canvas.

#### a) Navigation

Select the Navigation Icon to access Navigation Objects that become a starting point for designing a navigation model for your application. In addition to changing the many attributes for any of these Navigation Objects to create very different results, the Events Model (See Chapter 1.4) can also be used. Also see Working with Navigation Objects in Chapter 4.1;

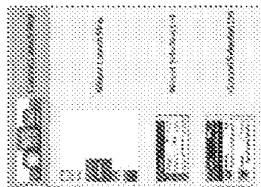


Fig 1.3

#### b) Messages

Select the Messages icon to view the "Message Objects" menu. There is a very large set of message Objects that can be used for communication through content such as:

- 1) Chat
- 2) IM
- 3) Placing Phone Calls from Lists that could have been populated from the phone address book or some internet-based Contact Management System.
- 4) Sending SMS messages to one recipient, or a group of recipients, again from lists that been populated from the phone address book or some internet-based Contact Management System.
- 5) Sending Rich Media Alert Messages that could include any page and/or widget that is available to that consumer.

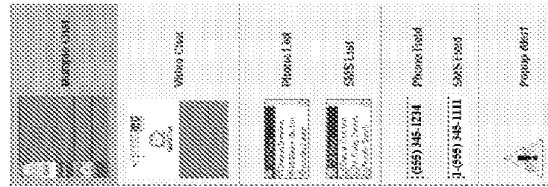


Fig 1.4





g) Palettes.

Select the Palettes icon to see the Palettes Objects menu.  
 Any of up to 5 floating palettes can be placed on the display, in any desired location. Once placed, there will be memory so that if you move the Publisher Window, or if you exit and then restart the Publisher, the Floating Palettes will remember their location relative to the Publisher's Window.

The Page Viewer is discussed later in this chapter. Section A.1.1.a

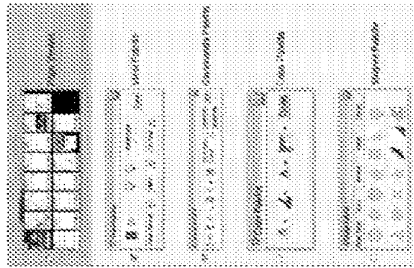


Fig. 1.12

1. The View Palette:

The View Palette permits the following functions to be performed.

a: Edit or Preview:

The default is Edit. Selecting the Play icon will close the various Panels and Inspectors and immediately begin to preview the application inside the Publishing tool. (In this mode certain functions, such as Exit Events, are not available)



Fig. 1.13

Selecting the Stop icon will return the Publisher to the Edit State.

The author can also selectively preview the current page or preview the entire application by selecting either "Go: Current Page" or "Go: All Pages" respectively. If the author selects "Go: Launch" a browser window will open and the application will play in a manner similar to how it would work on a phone, except that the mouse is available as a selection tool. If the author wishes to simultaneously create a Desktop version of the Application, then Launch will behave identically as it will in these environments. The default browser that you are using will be selected but be sure that the Java J2SE 6.0 or higher plug in is available.

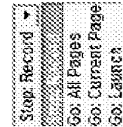


Fig. 1.14

b: Zooming:

The default zoom level is 100%. Touching the + or - icons will increase or decrease the zoom in increments of 10%. Holding down the mouse while on these icons will cause a continuous zoom until the mouse is released. Alternatively, the author can pick any zoom level from 20% to 800% by selecting the desired zoom level from the dropdown menu.

c: The Snap-to Grid

If one of the 3 grid settings is selected from the Zoom Grid Drop Down, then all objects will snap its upper left hand corner of its bounding box to the nearest grid point. The graphic to the right shows Medium (40 pixel) grid setting. Additional Grid controls are available by selecting "Grid..." under "View" in the Menu Bar.

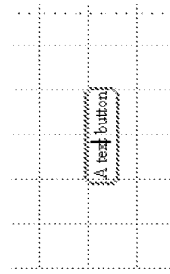


Fig. 1.15

d: Show Hot Zone

See Chapter 2 (Working With Application Pages).

e: Zoom Page Viewer

See "The Enhanced View Palette" discussion, in the Page Viewer Section later in this chapter.

2. The Command Palette:

This palette is a convenient shortcut for the most common commands that are also available under "File" and "Edit" from the Menu Bar. The "Group" and its counterpart "Ungroup" icons become available as soon as more than one object is selected at the same time. This is accomplished by holding down the "Shift" key and then selecting the desired objects that you wish to group.

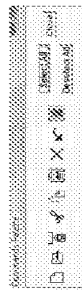


Fig. 1.16

3. The Color Palette:

The Color Palette, in its default state, shows the currently selected color that can be applied, with a single touch, to up to four color attributes for the currently selected object.

- Fill
- Line
- Text
- Frame

Selecting any of the four arrows will expand the color palette as shown on the right. These additional controls permit the change of the Color for that attribute. These choices will be remembered between editing sessions.

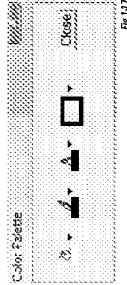


Fig 1.17

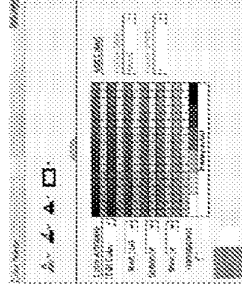


Fig 1.18

4. The Shape Palette:

The Shape Palette offers another convenient way for placing shapes on the Canvas (See Shapes in Section B.2.e)

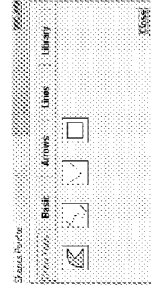


Fig 1.19

C: Canvas

The canvas is where all the work is performed for building or modifying an application. You can work with any object on the canvas directly with the mouse, and indirectly through the Inspector Panel or the Page/Objects Panel. The Canvas can be set to any zoom level from 20% to up to 800% of the actual size. If the page becomes larger than the visible canvas then the scroll bars become enabled.

D: Layer Inspector

The Layer Inspector controls:

1. The Page (Master Page, Application Page or Alert Page) you want to work with.
2. Permits various operations for Parent and Child Objects that are assigned to that Page:
3. Permits various operations for Object Styles:
4. Navigating between Page types and Object Styles
  - a. Toggling the Pages icon:  
This will switch between the Master Page and Application Pages. It will also deselect Object Styles and Alerts
  - b. Touching the Object Styles icon:  
This will open the Edit Styles Dialog Box and deselect any Master, Application or Alert Page.
  - c. Touching the Alert icon  
This will select an Alert Page and deselect any Master Page or Application Page.

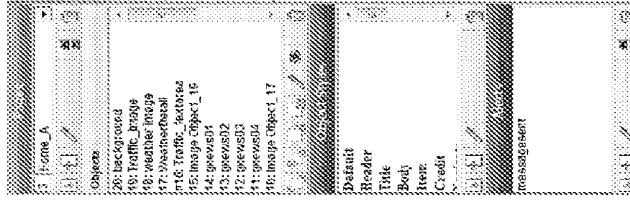


Fig 1.20

5. In addition, touching any object or icon in the related list for:

- a. Pages/Object Panel
- b. Object Styles Panel
- c. Alerts Panel

will automatically select that panel.

a. Pages/Objects Panel

Page Operations:

1. All Selection activities for:
  - a. Any Application Page
  - b. The Master Page
  - c. Any UI Parent object
  - d. Any UI Child Object

2. Common Page Operations

- a. Import from either the Author's Private Page Library or from the Xpressmo Public Page Library.
- b. Insert a new page, inheriting all the attributes of the Master Page, and place this new page at any location in the Page List.

c. Edit the currently selected page, by working with the Edit Page Dialog Box. All the functions of the Resource Inspector are available, but are not applied to the actual page until exiting the Edit Page Dialog Box.

d. Delete the selected Page.

3. Common UI Object Operations

a. Change the drawing order layer to:

- i. Bring to the front
- ii. Send to the back
- iii. Bring to the front one layer
- iv. Send to the back one layer.

b. Edit the currently selected page, by working with the Edit Page Dialog Box. All the functions of the Resource Inspector are available, but are not applied to the actual page until completing the entire editing process.

c. Hide (and then reshow) the Selected Object. This facilitates working with objects whose display is, at least in part, obstructed by another UI Object.

d. Delete the Selected UI Page Object

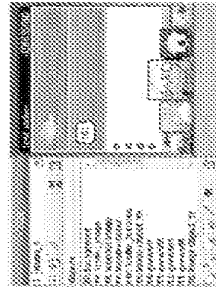


Fig. 1.21

b. Object Styles Panel

Similar to working with Pages, Object style Operations include:

1. Import from either the Author's Private Object Style Library or from the Xpressmo Public Object Style Library

2. Insert a new Object Style.

The styles can be inherited from the currently selected object (as shown to the right with the "grows01" UI Object selected, or from a previously defined Style Object.)

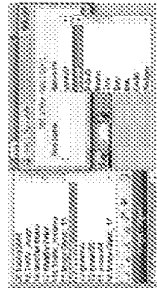


Fig. 1.22

3. Edit the currently selected Object Style by working with the Edit Style Dialog Box.

Style attributes can be assigned many attributes, including the look, and behavior of any object that inherits these objects. In addition, List Layout Styles can be created or changed (See the Section on Complex Lists)

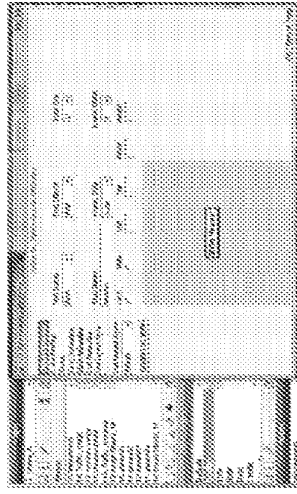


Fig. 1.23

c: Alerts Panel

Alert Pages can have many of the attributes of Application Pages, but they are only activated through an Event (See Working with Events in Chapter 14 ) and will be superimposed onto whatever the current Application is. (See the section "rich media Alert Messages" in Chapter 5).

1. Alert Pages all have transparent backgrounds, and they function as an template overlay.
2. Alert Pages can have dynamic binding to real time content.

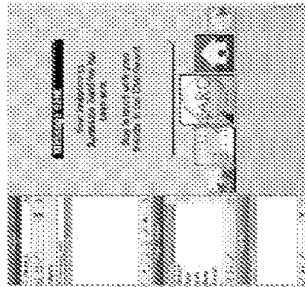


FIG 124

E: Resource Inspector

The Resource Inspector, whether working with pages or UI Page Objects, has five Standard Tab Selections.

1. The Settings Tab:

When activated, the settings dialog permits basic configuration of the selected object. These include:

- a. Name
- b. Size
- c. Location
- d. Navigation and Visual Settings
- e. Other

Depending upon the type of object, numerous other attributes could be exposed.

2. The Events Tab:

This includes all end user interactions and time based operations. (See Chapter 14)

3. The Animation Tab:

This includes all animations and timelines. (See Chapter 13)

4. The Color Tab:

This includes all the possible color attributes. These vary significantly by object type.

5. The Bindings Tab:

This is where all Web Component Operations are defined and all dynamic binding settings. (See Chapter 11)

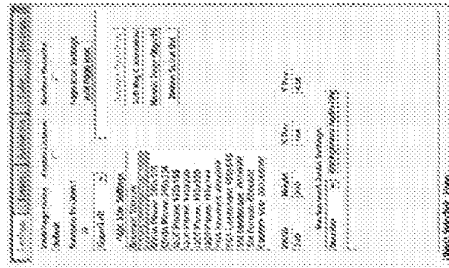


FIG 125

## F: Page Viewer

### 1: Activation:

Methods to select the page viewer palette:

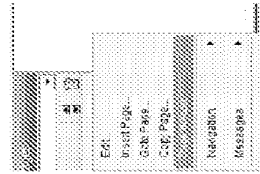


Fig. 1.26

Screen shot to the left shows the choice if there is a right click when on the working canvas.

Screen shot to the right shows the choice from the "View" menu.

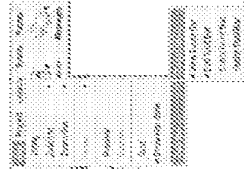


Fig. 1.27

Finally the Page Viewer can be activate the Palettes menu.

In addition, if the Page Viewer had been activate when the last Publishing Session was terminated, then the Page Viewer will automatically appear, with the following memory:

- The Position and Size of the Page Viewer Dialog Box.
- The Zoom Level that had last been selected.

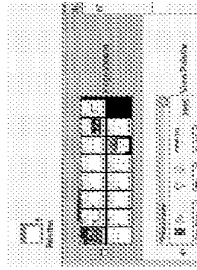


Fig. 1.28

### 2: Dismissal:

There are four ways to dismiss the Page Viewer:

- a. From the Palettes Menu.
- b. From the View Menu.
- c. By Touching the Close Icon.
- d. By Closing the Page Viewer Dialog Box.

### 3: Positioning and Zooming:

The Page Viewer will act like a floating palette, much like the other palettes. However, the user can freely resize the Page Viewer. Immediately the Page Viewer will follow the page thumbnails to the changed container size. If the height of the Page Viewer is less than 150% of the height of a page thumbnail, then a Horizontal Scroll Bar will be displayed (if necessary) and all the thumbnails will line up in a single row. See the example below which has set the zoom level of the page thumbnails set to 30% of the original, and has resized the Page Viewer container into a horizontal strip.

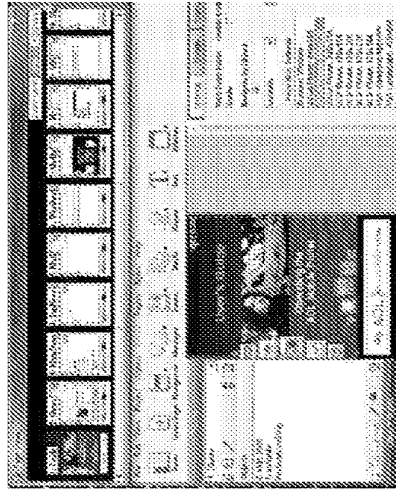


Fig. 1.29

The zoom level can be changed as required from inside the Page Viewer.

If the height of the Page Viewer is greater than 150% of the height of a page thumbnail then the thumbnails will display as a grid, with a Vertical Scrollerbar available.

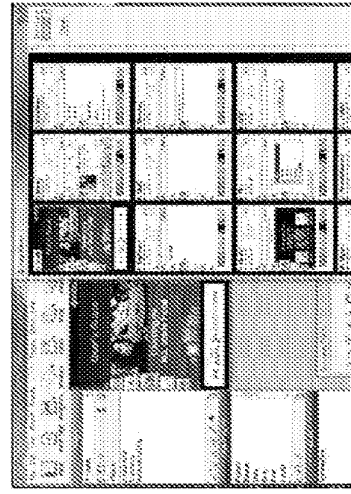


Fig. 1.30

The Author can change the zoom level to 200% or even up to 300% so that the Page Viewer acts as a magnifying lens.

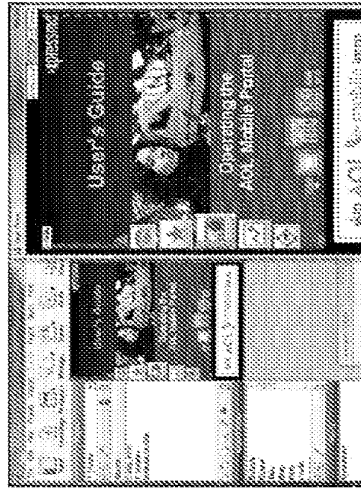


Fig 1.31

4: Preferred Size:

When opening Publisher, if the Page Viewer is visible, it will resize itself to a preferred size. Specifically, the size will adapt itself to the appropriate size to display the default page. When additional pages are added to the application, the Page Viewer will resize itself to the latest page dimension.

5 : View Palette:

The View Palette now has 2 new entries:

1. Show Hot Zone
2. Zoom Page Viewer

Changing the Show Hot Zone setting or the Page Viewer Zoom Level will be immediately reflected in the Page Viewer Dialog Box.

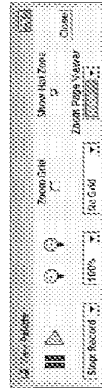


Fig 1.32

6: Editing with the Page Viewer activated:

All visible editing operations will immediately become visible in the Page Viewer.

7: Using the Page Viewer for Navigation:

The page number and page name will usually draw over the upper left hand corner of the Page thumbnail. However, when the mouse is over a particular page thumbnail, the page number and name will attach itself to the cursor.

Selecting a page image within Page Viewer will switch the canvas to that page.

## G: Palette Docking

### 1: Palette Docking

There are currently 5 palettes available in the Pub Tool.

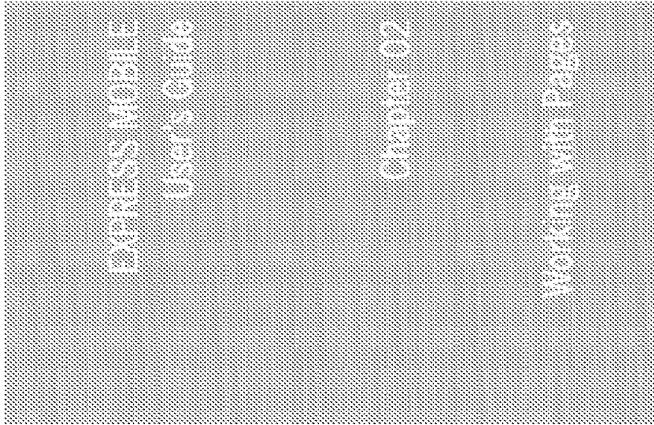
- Page Viewer
- Command Palette
- View Palette
- Color Palette
- Vector Palette

All of these palettes can be dragged to any position on the screen. They can selectively set to visible or hidden.

The Page Viewer Palette can also be resized.

When dragging or resizing the Publisher windows, all visible palettes will move in synchronization with the Pub Palette Frame (they are effectively docked to a fixed position in or relative to the Publisher Palette Frame). Any dragging and/or resizing of palettes will have no effect on the position or size of the Pub Tool Frame.

As with the Publisher frame itself, the location and size of all the visible palettes will be remembered when the Author exits from the Publisher. These positions and sizes will be restored when the Author returns to the Publisher.



## Chapter 2: Working with Pages

A:	Master Page	02.04
B:	UI Implementation	02.06
C:	Application Pages	02.08
D:	Alerts	02.14
E:	Libraries	02.16
F:	Saving Files	02.17
G:	Enhanced Navigation Specification	02.18



## Chapter 2: Glossary

Launch Strip Attribute	The settings for each icon in the Launch Strip. Usually they are links to different pages or applications.
User-Application	The first application that is executed that activated by the phone. Usually this is defined by the author.
Sub-Application	An application is called either from the Uber-Application or from another Sub-Application.
Alert	A type of page that is superimposed over the current application page.
Soft Key Command	These are commands that become available through touching a Soft Key button on the mobile device.
getParentObject:Selected	An example of an Interface Method that is called, in this case, when an Object is selected. These Interface Methods are very useful for working with Web Components, Express Mobile APIs, and for collecting analytics.
PDL Image Table	A table of all the Application and Alert Pages that have currently been downloaded and are available.
dmfPage	A screen that is drawn over the page when the page is temporarily inactive. This often occurs when a launch strip is active or an Alert page is active.

## A: Master Page

1. Defines Default Settings (at the page level) for all new pages.  
This means that all the attributes that can be set in the "Edit Page" dialog box and/or the Page Resource Inspector will have their initial default settings defined by the Master Page.
2. Repeating Objects.  
Any objects defined on the Master Page will appear on every application page. The drawing order for an application page will be:
  - a. The first objects drawn will be from the Master Page, and drawn in the order defined in the master page.
  - b. All the application page specific objects will then be drawn in their drawing order.
  - c. Finally, if defined in the Master Page, the Launch Strips will be drawn.
3. Editing Operations:
  - a. Editing the Master Page:  
To work with the Master Page select the "Master Page" entry in the Page List or by Selecting the "Master Page" command under the file menu.. The currently viewed page will now be the "Master Page. All operations that are available to an application page will be available to the Master Page. Any additions, deletions, or changes to objects on the Master Page will immediately take effect for all application pages in the application.
  - b. Editing Master Page Objects on an Application Page.  
Only Launch Strips can be selected and edited. All the other objects from the Master Page will be drawn but not be selectable.
  - c. Applying a Current Page as the Master Page  
Select "Set Master Page" under the Pages menu. The current page's objects will be appended to the Master Page in their current drawing order. Any page attributes that the current page has set to override the Master Page's attributes will be transferred to the Master Page.

d. Selectively Removing Master Page Objects from an Application Page.

When the "Master Page Objects" button is selected from the Resource Inspector, the Visibility of Master Page Objects Master Page Objects Dialog Box will appear. All existing Master Page Objects will appear in the List on the right, and any Suppressed Objects will appear in the List on the left. The Suppressed Objects can be restored by selecting the Object in the list and selecting the "Restore Object" button.

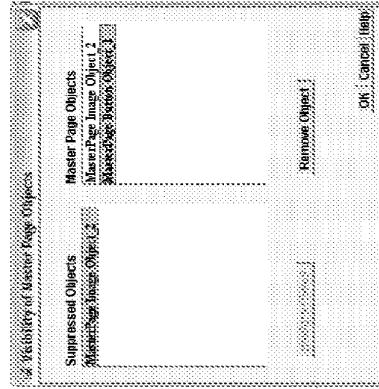


Fig. 2-1

Fig 2-1 shows what happens when a Master Page Object was selected and the "Remove Object" button was pressed.

e. Current Master Page Restrictions.

Soft Key Commands are disabled for Master Pages.

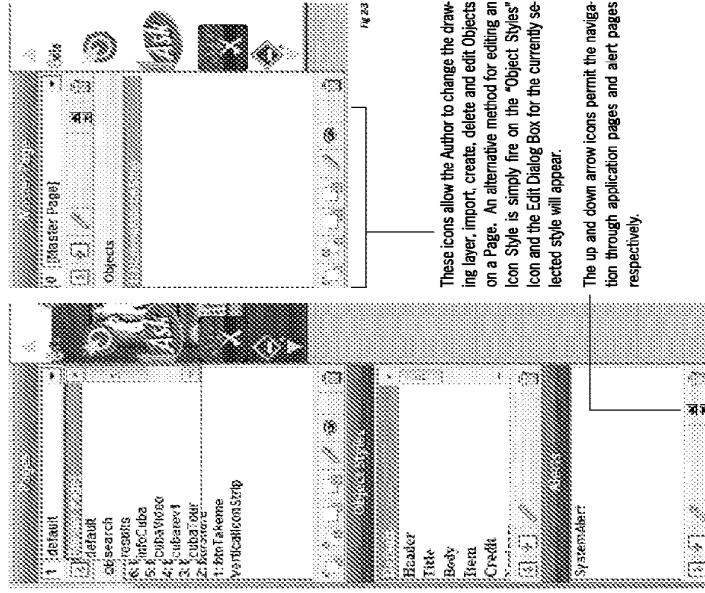
4. Effect on Launch Strips.

Launch Strips are part of the Master Page. The only difference between Launch Strips and other repeating objects are that the Uber-application's Launch Strips will be inherited as repeating objects for all sub-applications. Any Launch Strips defined in a sub-application will inherit all the icons from the uber-application, with its own icons appended. These sub-application specific icons will only be available while the sub-application is the current application for the player. While a sub-application is running, its Launch Strip attributes will take precedence over the uber-applications Launch Strip attributes, if both sub-application and uber-application define the same-named launch strip attribute.

B: UI Implementation

The Master Page can be selected in the Master Page entry in the Page List or by selecting the "Master Page" command under the file menu. The result will be the graphic below.

To return to an Application Page or an Alert Page either Select the "Master Page" or "Alerts" icons or by Selecting a page or alert from their respective dropdown menus.



These icons allow the Author to change the drawing layer, import, create, delete and edit Objects on a Page. An alternative method for editing an Icon Style is simply fire on the "Object Styles" icon and the Edit Dialog Box for the currently selected style will appear.

The up and down arrow icons permit the navigation through application pages and alert pages respectively.

Selectively hiding an Application Page Object works as follows:

- Hiding Objects**

Select the object and then Select the "Eye" icon. A non-alpha character similar to a lowercase character will appear to the left of the object name. When a hidden object is selected the "Eye" icon appears in a grayed out state. (See Fig 2-4)
- Restoring Hidden Objects**

Select the grayed out "Eye" icon as shown in Fig 2-4.
- Effect of Hiding Objects**

A hidden object will only have effect during the current editing session. To hide a Master Object for a page during execution refer to "A.3.d" above.

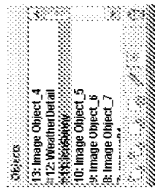


Fig 24

### C: Application Pages

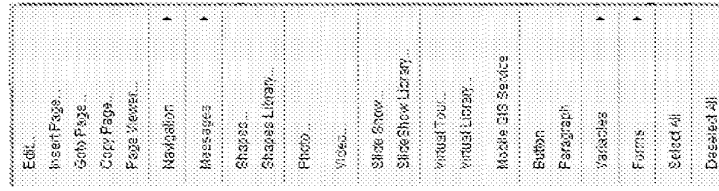
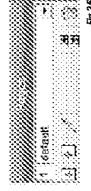


Fig 25



There are four possible controls for working with pages.

- The PopUp Menu to the left which appears by using the right mouse button when over the Canvas.
- The Page Control from the "Pages/Objects Panel" above.
- The Page oriented commands from the File Menu to the right.
- The Pages menu below.

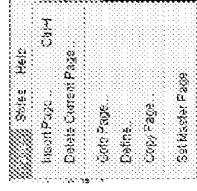


Fig 26

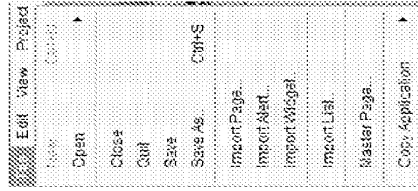


Fig 27

1: Page Resizing

Any page can be set to a custom size. If not, then the size of the master page will be inherited. Automatic generation of QVGA and QCIF applications is supported.

2: UI Implementation

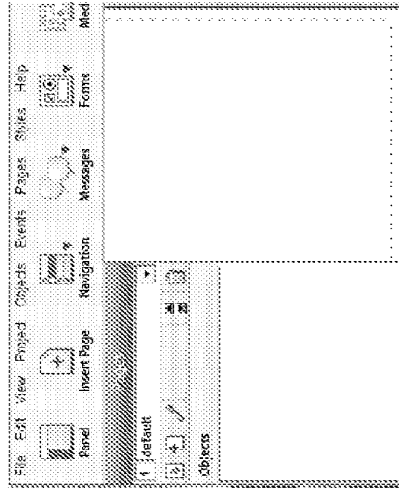


Fig 2.9

All pages have a hot zone (or a safe area) that is represented by two gray dotted lines. As long as the Author does not have a selectable object intersect these boundaries then a maximum of one scroll bar will be drawn assuming that one of the pages dimensions are equal to, or less than that of the device's canvas.

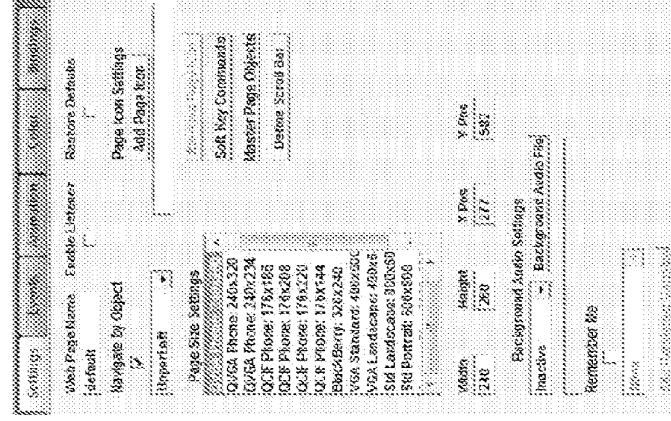


Fig 2.10

Page Size Settings represent the size for the currently selected page. This size is initially inherited from the Master Page, but can be changed by editing the width and/or height fields. If the resulting size does not match a standard form factor then "Custom Size" will be selected.

An icon can be added so that if the Page is saved as a Widget then this icon can become part of a Widget Selection list.

Objects that were defined in the Master Page are automatically placed in each application page. However, touching the "Master Page Objects" button will open up a dialog box where Master Page inherited objects can be selectively suppressed for a given application page.

An Audio File can be attached to the Page, so that when the Page is active that Audio will play or loop, depending upon the setting. Persistent variables can be added to a page. They are used so that the background audio could be defined dynamically through a Web Component.

In addition, there is a button "Define Scroll Bar" which gives the author the ability to custom define the scroll bars for the currently selected page, if any. These values are also initially inherited from the Master Page.

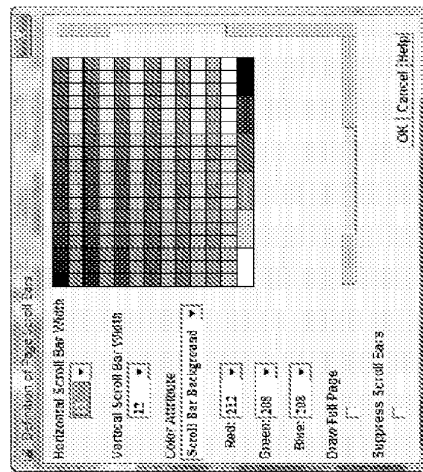


Fig 2.11

The above Dialog Box gives the author the following choices.

- Horizontal Scroll Bar width (from 6 to 20 pixels)
- Vertical Scroll Bar width (from 6 to 20 pixels)
- Scroll Bar Background Color
- Elevator Background Color
- Scroll Bar Frame 1 Color
- Scroll Bar Frame 2 Color
- Elevator Frame 1 Color
- Elevator Frame 2 Color
- Draw Full Page
- Suppress Scroll Bars

The six colors permit a scroll bar definition that mimics a Windows or Java J2SE scroll bar. Any of these color attributes can be changed and the feedback will immediately reflect that color.

The Scroll Bar widths will change the width of the respective hot zone, and will establish the width of the scroll bar for that particular page.

If Draw Full Page is selected, then the player will switch the phone to its full page draw mode, thus increasing the size of the canvas. This would minimize the need for scroll bars as well as show more of the page in a single view. However, usually this means that the labels for the soft key commands will not be drawn, and though the commands themselves will remain available.

If Suppress Scroll Bars is selected, then no scroll bars will draw for this page

xpressmo.com

### 3: Player Implementation

The drawing model scrolls the page so that the newly selected object is fully visible.

A vertical scroll bar is displayed if the height size of the page exceeds the current canvas height. A horizontal scroll bar is displayed if the width of the page exceeds the current canvas width.

### 4: Format Factor Considerations

#### a: When to Use "Draw Full Page"

Although drawing the Full Page increases the amount of content that will be displayed, on most phones it will also suppress the drawing of the Soft Key Labels. The Soft Keys are still available, but the user will have no visual reference.

This is a trade off in terms of usability which could vary page by page.

#### b: Conditional "Draw Full Page"

There is a range of form factor heights within both the QVGA and QCIF form factor classes. "Draw Scroll Bars" will deal with all cases, but there is, depending on the page design, phones that will have many pages that will scroll.

An optimal design would be that the page exactly matches the default form factor of the phone, this giving the user the best possible navigation experience. However, a compromise that would reduce the number of required form versions would be to use "Conditional Full Page Draw". If this is employed, then for phones whose default canvas height is less than the reference page height for the application, the view is switched automatically to "Draw Full Page" so that the entire page is visible. Otherwise, the optimal visual presentation will occur.

To implement this conditional capability set the page to:

- suppress scroll bars
- draw full page is turned off

When the Player, under these conditions, detects that the default form factor is not sufficient to draw the entire page it will automatically switch to "Draw Full Page".

### 5: Soft Key Commands

Any number of Soft Key commands can be assigned to a page.

Authoring Soft Key commands will have a similar UI as authoring Lists.

The designer will be able to add, edit and remove a soft key from the Edit Page Dialog Box in a manner similar to that of working with list items in the List Edit dialog box.

The default Back Command will appear and cannot be erased. However, it can be reset to "Goto Page" if desired.

When adding or Editing Soft Key commands the following attributes will be assignable.

a. A drop down for selection of the attached command. The 8 choices are:

1. Goto Page
2. Back (this is defined by default)
3. Goto Page & Report
4. Back 1 View & Report
5. Goto Start Page
6. Goto Start Page & Report
7. Login
8. Logout
9. Execute a Web Component

If a command with "Report" is selected, then the normal `getParentObjectSelected` interface method will be called, with a boolean identifying this as a soft key command. The command name is used as the object name and the command number is used as the object number.

If "Execute a Web Component" is selected, then any Web Component that has been previously assigned to the Page or any object on the Page will become available as a Command Parameter.

b. A text field to define the string that will appear as the soft key list (which, on many phones, may appear as a scrolling pane after a certain number of commands are defined).

## D: Alerts

### 1: Types

There are 3 classes of Alert Popup Pages, all which will have the same coordinates as regular pages but with a transparent background.

- 1: Application Specific
- 2: Author Specific
- 3: System Default

Alert Pages create a multi-layered drawing model for pages.

The author-defined exit events will display an Alert Popup Page. It is permissible for Alert Pages to call and display another Alert Page.

The Player will first look in the PDL Image Table for the Alert Popup page. If not found it will look at the Author Specific Alert Popup Page Library (much like an Xpressmo Server Page).

System Alerts, there is a default System Alert Template Page available for all applications. This Page has defined text objects for the actual test message that will be displayed based on the real-time condition that triggered the System Alert. The author is able to add other objects as well as control the look of the Text Objects.

There are three types of Popup Alerts.

#### 1: Non-modal

The Alert will draw. All functions for underlying page are active.

A non-modal alert can be discarded by:

- a: A timeout assigned to the Alert Page
  - b: An Exit Event
  - c: A backend API
- 2: Modal (User dismissible)

The Alert will draw. All functions for underlying page will be disabled except for Exit application. A dimmed Page mask will be drawn over the underlying page.

A User Dismissible Modal alert will be discarded on the first "Fire" Key Event.

3. Modal (Event Driven)

The Alert will draw. All functions for underlying page will be disabled except for Exit application. A dimmed Page mask will be drawn over the underlying page.

An Event Driven Modal alert can either be discarded by:

- a: An Exit Event
- b A backend API

"Fire" and Author -defined Soft Key Commands are the only interactive events supported. They will apply to the entire Popup Alert object, and it is effective for Popup Alert Type 2. The Events and Animation Tabs of the Resource Director are not available. Navigation within the Alert is not supported. Export and Import of Alerts from the Author and/or System Libraries are supported.

Alert Pages of all three types can be displayed or disposed by backend APIs.

2. Authoring UI

When an Alert is created or selected the background will draw as transparent page.

When an Alert is created, there will be a test to make sure that its name is unique, both in terms of other application specific alerts and/or application pages.

Alerts appear as a choice under "Messages", under the Objects-Messages... menu bar, and in the Layer Inspector as shown to the right.

On the Layer Inspector there is a dropdown showing the current alerts defined for the application. It works like the Pages dropdown.

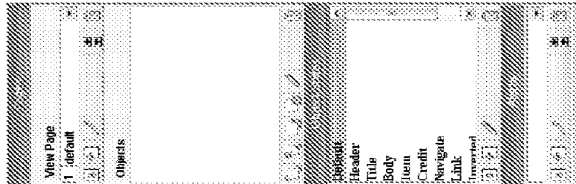


Fig 2.12

When an Alert is selected, there will be a list with 3 possible selections.

1: Nonmodal

If "Stay" is selected, A non-modal alert can only be dismissed by:

- a: A backend API
  - b: A "hide alert" Exit Event.
- Otherwise, if not yet dismissed, the timeout will disable the Alert.

2: Modal (User dismissible)

3: Modal (Event Driven)

These are the same choices that a Page has in terms of defining Soft Key Commands.

The Alert Object Name can be renamed, and, for convenience, the x,y location of the cursor, is displayed as well as the size of the underlying page.

The Objects that can be defined for an Alert are Forms (non-interactive), Media, Shapes and/or Text. Child Objects are available. Overlapping Objects are also supported.

3. How to make an Alert Page appear

- 1: By API
- 2: By Error Condition
- 3: By Exit Event
  - Slide
  - Listitem
  - Object

E: Libraries

a: Publisher Level (publicly available libraries)

- Pages
  - Page Icons
- b: Author Level
- Pages (Page Library)
  - Page Icons (all page icons that were uploaded by author)

c: Application Level

- Background (Page Icons)
- Icons (application icons)
- pages (Express Mobile Server Pages)

## F: Saving Files

The Save As File Menu can be reached either from the File Menu or from the Command Palette.

There are a minimum of six file types that can be selected, and, depending upon the file type, various options will become available. For the case of saving an application, the following is available:

- 1: Generate QCIF Form Factor

This will be available if the application size is QVGA. (Note that if the application size is VGA then scaling to QVGA will also be available.)

- 2: Embed Application, Images/Slides or Audio Video.

Usually not necessary unless there is interest to make the application fully available in an area that does not have an Internet connection.

- 3: Reset Web Components.

If the author has manually transferred the application or applications to a different server, then selecting this choice will reset all the Web Components to point to the current server.

File Type can also include the Currently Selected:

- Slide Show
- Complex Vector
- List

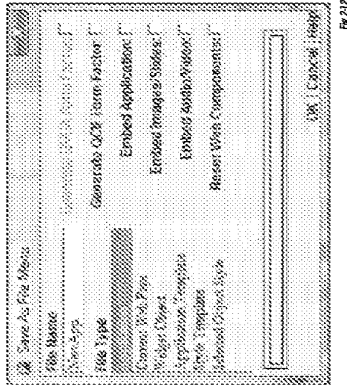


Fig 212

## G: Enhanced Navigation Specification

1. Navigation Controls

"Navigation Controls" give the author the following capabilities.

- Select navigation order identical to that which "Edge Detector" produced.
- Set any navigation sequence for vertical navigation.
- Set any navigation sequence for horizontal navigation.
- For Lists and Slide Shows have controls to have these objects cycle when reaching either end of their elements, or have the focus switch to the next object in the ordered lists.
- Under certain circumstances, when there are no logical conflicts for any navigation event, set the page so that it maintains the focus for all selectable objects simultaneously.

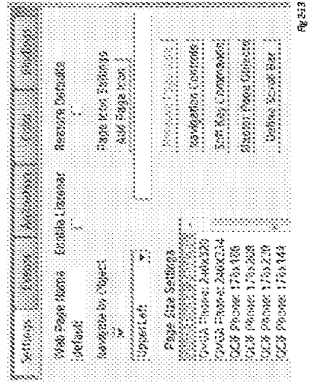


Fig 213



2. Authoring UI  
 a. Working with Navigation

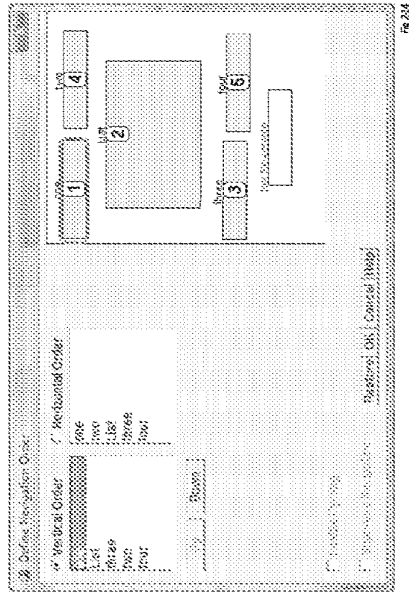


Fig. 214

When touching the "Navigation Controls" button, the above dialog box will appear with the current page layout available as feedback.

Only selectable objects will be assigned a navigation sequence. Any non-selectable objects will be drawn but not numbered. The initial state has the "Vertical Order" selected with "Up" and "Down" buttons available. The ordering sequence is both displayed as labels in the feedback and by the objects' position in the Vertical List.

The "Restore" button, if touched, will set the navigation order back to the original order as defined by the prior "edge Detection" algorithm.

The feedback is interactive so that objects can be selected for setting various forms of navigation either by:

1. Touching an object in the feedback.
2. Touching any item in either the Horizontal or Vertical lists.

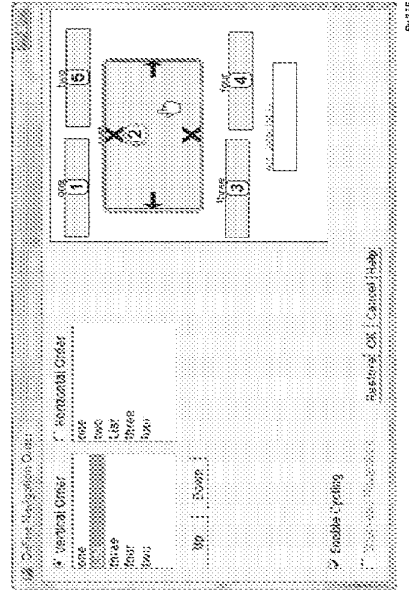


Fig. 215

The graphic above show what would happen if the List Object was touched.

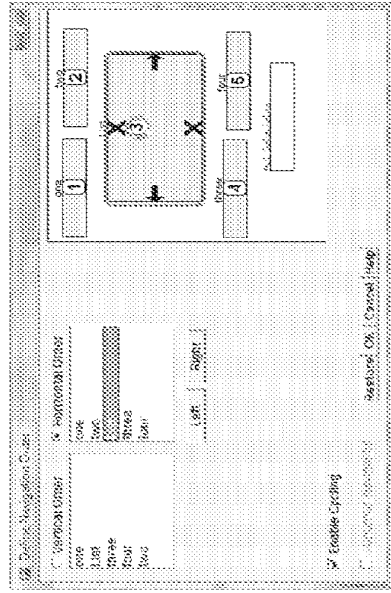


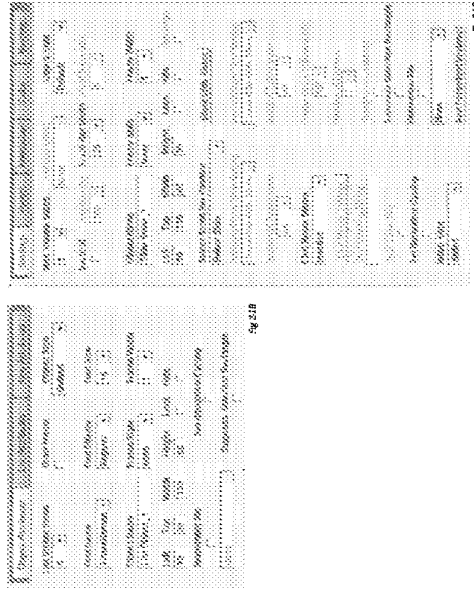
Fig. 216

The graphic above shows what would happen if the List item was touched in the Horizontal Order List.

Note:

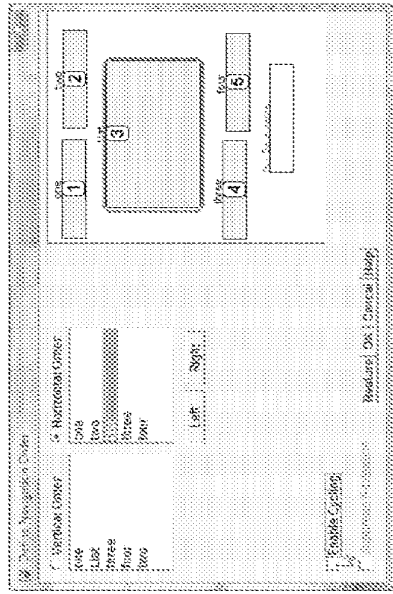
- The "Horizontal Order" radio button was implicitly selected.
- Navigation Control buttons are now labels "Left" and "Right".
- The Feedback now has labels representing the horizontal navigation order.

Using the "Enable Cycling" check box the Cycling navigation state can be turned on or off. Enable Cycling will also be a check box for both Slide Shows and Lists

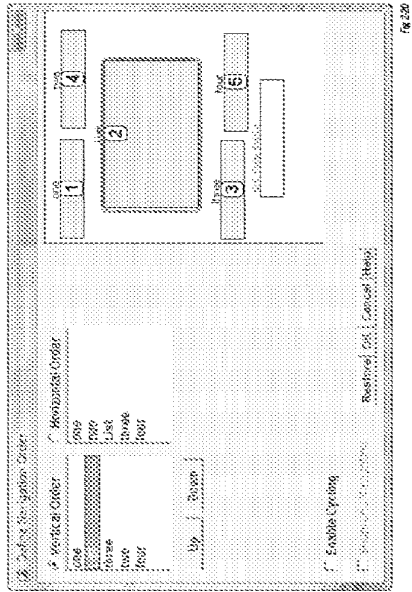


b. Working with Cycling

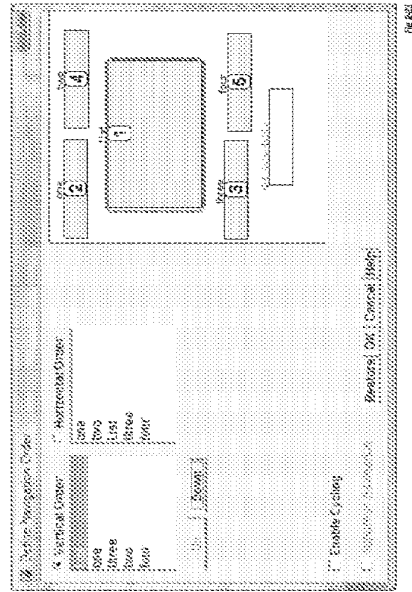
The default state for both Lists and Slide Shows is that they cycle. This is shown in the graphics above both by a Cycling icon and the "Enable Cycling" check box selected.



c. Changing the Navigation Order



If the List is selected (see graphic on the top) and the "Up" button is touched the result is shown in the graphic below



d. Navigation Rules with Changing the Page Layout

If a page is in the "default navigation" mode, then all operations will proceed and at the time of the page being saved, the current "edge detection" order will be applied.

Otherwise, the following rules will be applied.

If an object is added, the navigation order will be set for that object so that it is at the bottom of both the horizontal and vertical ordered lists.

Note: This would be equivalent to placing the object in the extreme lower right corner of the page if the "edge detection" algorithm was applied.

If an object is deleted it will be removed from the ordered lists. If there is an undo operation, then the object will resume its previously defined place in the ordered lists.

e. "All Objects in Focus" Page

The following page layout conditions must exist for an "All Objects in Focus" mode to be available.

- There are exactly two Selectable Objects in the page. (See appendix D of the User's Guide).
- One object is either a Text Field or a Horizontal Slide Show.
- The other object is either a List or a Vertical Slide Show.

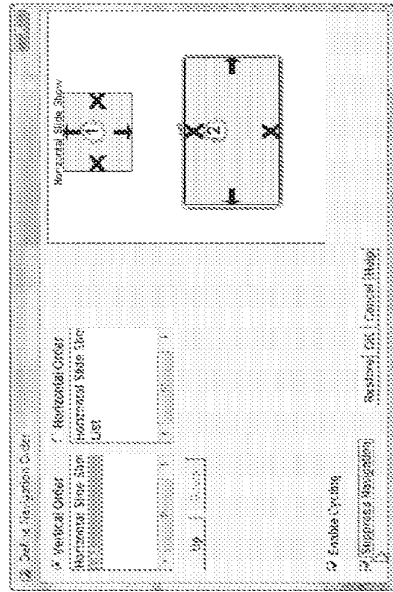


Fig. 2.22

Under the page layout above, the Slide show will accept the horizontal navigation keys and the List will accept the vertical keys. Cycling will automatically be turned on for any Slide Show or List.

FFRE will be honored by the following defaults.

If, as above, there is a Slide Show and List, the List will accept the FFRE event, unless there are no exit events assigned. Otherwise, the Slide Show will accept the FFRE event.

If there is a Vertical Slide Show and a Text Field, the Text Field will accept the FFRE event, unless there are no exit events assigned. Otherwise, the Slide Show will accept the FFRE event.

In the event that the Author selects "Suppress Navigation" for a page, and then, in a subsequent editing event, attempts to either delete, add, or redefine an Object that would cause the necessary conditions for this page to be violated, a warning message will appear. The author will have the choice to continue, and have the "All Objects in Focus" attribute set to false, or cancel the operation.

3. PDI Specification for Enhanced Navigation

a. Page Definition

BooleanAttribute[8] = Default Navigation Overwritten

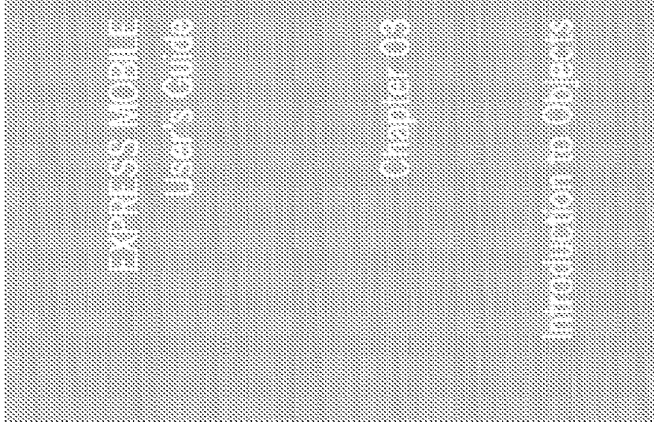
BooleanAttribute[9] = Suppress Navigation

b. Primitive Object Definitions

ObjectBooleanAttribute[28]= SlideShow & List: Suppress Cycling

ObjectIntegerAttribute[44] = Horizontal Navigation Order (1=not selectable, 0 = first, 1..n-1 = additional order, where n is number of selectable objects on page)

ObjectIntegerAttribute[45] = Vertical Navigation Order (1=not selectable, 0 = first, 1..n = additional order, where n is number of selectable objects on page)



### Chapter 3: Introduction to Objects

- A: How to Create 03.05
- B: Keyboard Operations with UI Objects 03.05
- C: Persistent Variables 03.06
- D: Layout by Pixel vs Layout by Font 03.15
- E: Suppression of Selection Rectangle 03.17
- F: Using Object Libraries 03.19

### Chapter 3: Glossary

- Objects**  
An intelligent entity that has both a defined appearance and a set of defined behaviors. These behaviors can be activated based on:  
1: User interaction  
2: Time  
3: Location when in an animation.
- Suppression**  
The selective disabling of a certain function of behavior. Examples include the suppression of the Selection Rectangle for a given UI Parent Object or the suppression of one or more objects that are defined in the Master Page for a given Application Page.
- Layout by Pixel**  
Overrides the changes to the width and height of a Text Object, that would normally occur based on the Font Metrics of the mobile device. This guarantees that the size and position of all text objects remain unchanged across device types, although the actual size of the text itself will now vary to a greater degree.

## Introduction to Objects

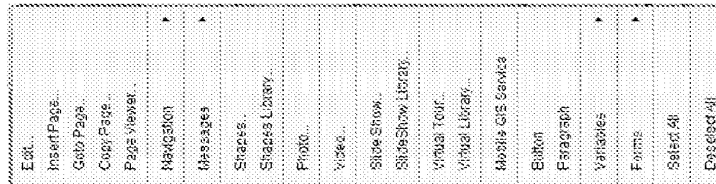


Fig 31



Fig 32

- The commands for working with objects come from the following four sources:
1. The Object Oriented Commands from the toolbar above.
  2. The Popup Menu to the left which appears by using the right mouse button when over the Canvas.
  3. The Objects Menu to the right.
  4. The Object Icons from the "Pages/Objects Panel" below.

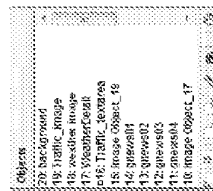


Fig 34

## A: How to Create

There are three different ways to create UI Objects.

1. The Object Oriented Commands from the Toolbar.
2. The Popup Menu , using the right mouse button when over the Canvas, and no object is selected.
3. The Objects Menu in the Menu Bar.

## B: Keyboard Operations with UI Objects

### 1: Keyboard Shortcuts

The following shortcuts are available with the Publisher:

- a. New\_ (CTRL N) (COMMAND N) for Macintosh
- b. Open\_ (CTRL O) (COMMAND O)
- c. Save\_ (CTRL S) (COMMAND S)
- d. Cut\_ (CTRL X) (COMMAND X)
- e. Copy\_ (CTRL C) (COMMAND C)
- f. Paste\_ (CTRL V) (COMMAND V)
- g. Delete\_ (DEL)
- h. Deselect\_ (CTRL D) (COMMAND X) Deselects all objects on the page.
- i. Undo\_ (CTRL Z) (COMMAND Z) Currently this is an undelete only function.
- j. Edit\_ (CTRL E) (COMMAND E) Opens up the Edit Dialog Box
- k. Insert\_ (CTRL I) (COMMAND I) Inserts a new Page
- l. Restore\_ (CTRL R) (COMMAND R) For Images only. Will restore the image to its actual size.

### 2: Resizing with the Keyboard.

For images and videos, with the Shift Key pressed, touching the arrow keys will cause the Object's width or height to increase (right or down arrow keys) or decrease (left or top arrow keys).

### 3: Dragging with the Keyboard.

With the CTRL key pressed, touching the arrow keys will move the selected object in the direction of the arrow keys.

### 4: Maintaining the Aspect Ratio.

The aspect ratio of the selected object can be maintained by selecting an attachment point with the mouse and then, with the Shift key pressed, dragging the mouse.

## C. Player Data Persistence Specification

### 1. Using Escape Characters

In the player code, there is an existing implementation that allows storage and retrieval of data objects required by an application. The data is stored as a key-value pair, much like in a properties file. The data persists across sessions, and is only removed when the user deletes the application/player from the device. The current limitation is that all keys and values must be a String object. This is suitable for storing text and numbers but is not suitable for storing complex objects, such as all the attributes of a list item.

Data persistence is accessible in the player, and can be designated by the author of the application using a special notation in text fields. `%variableName%` is the format, where the variable escape `%` characters act to enclose the variable name. A typical use of this is to ask the user for a login name and have this login name persist within the application, and across sessions.

The `%variableName%` can be placed in any text field where normal text is placed. The player will automatically substitute the key, `"variableName"` with the value associated with that variable within the persistent storage module. If `variableName` is not found in persistent storage, then a null value is returned.

The population of persistent storage was either accomplished with an Exit Event ("Set Starting Page") or implicitly with Chat for the "UserName" Handle.

### 2. Supported Objects

#### a. Scalar Variables

The following objects will be supported for Player Data Persistence for Scalar type Variables:

- Text Fields
- Text Areas
- Choice Objects
- List Objects
- Check Boxes
- Video Objects
- Launch Strips

Object Type	Value that is saved
Checkbox	State of the checkbox (true = "checked", false = "unchecked")
Dropdown List	Last selected item's display text
Complex List	Last selected item's display text(0) (top left text)
Text Field	String value of text field
Text Area	String value of text area.
Video	The video URL. If DefineByURL then it's the absolute URL
Launch Strip	The currently selected channel

The following objects will be supported for Player Data Persistence based on a comparison to a hidden String Attribute:

- List Objects
- Slide Show Objects
- Text Button
- Vector
- Image
- 

All Scalar type Variables will support a replace only function.



b. Non-Scalar Variables

The following objects will be supported for adding new entries to Non-Scalar Variables:

- Text Fields
- Text Areas
- Check Boxes

Object Type	Value that is saved
Checkbox	State of the checkbox (true = "checked", false = "unchecked")
Text Field	String value of text field
Text Area	String value of text area.

The following objects will be supported for selecting the default value of a Non-Scalar Variable:

- Choices Objects
- List Objects

3. Persistent Variable Types

a. Scalar:

Used when the variable will only have a single value. Adding a value to a scalar variable is equivalent to replacing the existing value with a new value.

b. Non-Scalar:

- Stack (LIFO):

As each value is placed in the persistent variable record, it is placed on the top so that, when retrieved, they appear in the order based on the newest item first. The list of values must be unique

- Vector (FIFO):

As each value is placed in the persistent variable record, it is appended so that, when retrieved, they appear in the order based on the oldest item first. The list of values must be unique

- History:

For adding values to the persistent variable record, History will act similarly to a Stack, but will not enforce uniqueness.

All persistent variable types will be supported for all objects as defined in section 3 Supported Objects in terms of adding values to the persistent variable record. However, only the following objects will display the full Stack, Queue or History and permit the override of the default selection.

- a. Choice Objects
- b. List Objects

If no selection has yet occurred, then all other objects will convert the multi-item record to a scalar value based on the selection as follows:

- a. Stack: Top Item
- b. Queue: Last Item
- c. History: Top Item

4. User Interface

When a supported Object is selected, the initial state will be the available checkboxes:

- \* "Remember Me"
- \* "Suppress Selection Rectangle"

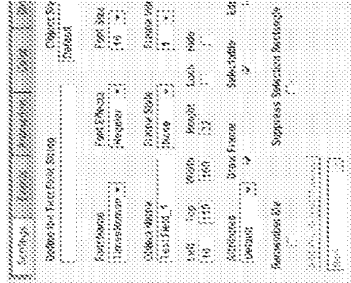


Fig. 35

When the "Remember Me" attribute is selected, then two additional UI objects become enabled.

2. Add Persistent Variable

The default variable name will be "PageName" - "ObjectName". However the author can rename it to any name they wish. However, if the name, as entered, has already been defined the text will turn red. If OK is touched before correction then an Error Message will appear.

If an object is selected, and it is a then there are four types of data structures that can be assigned as described above.

Upon a successful completion the Variable name will be added to the drop down menu and the current object will have that variable selected.

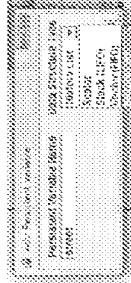


Fig. 36

b. Select a Persistent Variable

At any time the author can choose a different persistent variable to assign to the selected object. This means that multiple objects on the same page and/or on different pages can share the same persistent variable.

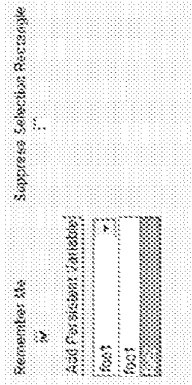


Fig. 37

c. Working with History Lists

To create a History List, first create a Dropdown Menu and then select the "Dropdown History List" in the List Selector type choice list. A "Select History Variables" button will appear for assigning non-scalar variable lists to up to three different elements for each item in the History List.

Touching this button will display the following Dialog Box.

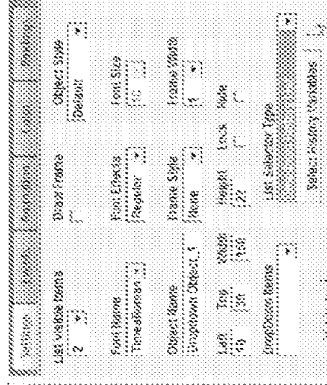


Fig. 38

d. Removing a Persistent Variable.  
If "None" is selected for an object that had a previously assigned persistent variable, then its reference is removed. If there are no remaining references to that persistent variable, then the persistent variable is also removed.

e. Removing all Persistent Variables in an Application.  
If the author wishes to clear out all persistent variables defined in an application, then Select "Remove all Remember Me" from the Edit Menu.

- f. Behaviors
- Objects with the scalar persistent variable attribute can serve a dual purpose:
    - Setting a Persistent Storage Value.
    - Retrieving and Displaying a previously stored Persistent Value.

2. If there is no interest in having that object reset the current stored value for that persistent variable then give that object either:

- Noteditable attribute
  - Not selectable attribute
  - Text Buttons, Vectors and Images only retrieve a stored Persistent Value
3. The memory for persistent variables is application specific. Even if there are two applications with the same name, they will be differentiated based on the UserName attribute.

4. FIRE will set the persistent variable's value as described in "Section D.3.b" on the previous page. In the event that there are any objects on the same page that share the same persistent variable, then they will be updated immediately. If one of these objects is a check box, then the checkbox will become selected if, and only if, the string of the object that is fired on equals "true".

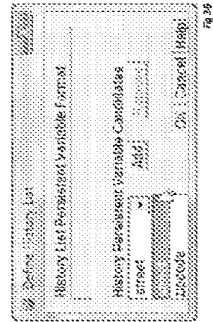
5. Any Exit Event on the page that changes the page view or the application will implicitly set the persistent variable's value for all objects on that page that have the "Remember Me" attribute as described in "Section C.3.a" on the previous page.

6. The rules for the current %variableName% notation for Text Fields will be augmented so that, as long as there are two, and only two variable escape "%" characters in the String, then:

- Embedded spaces will be permitted.
- The %variableName% substring will be replaced by its value. Any text outside the %variableName% substring will be preserved.

7. If the persistent Variable is other than Scalar, and the rules as defined in Section 4 Persistent Variable Types above apply.

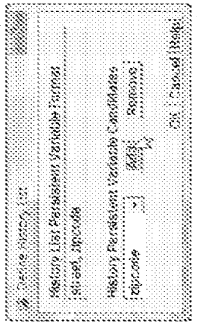
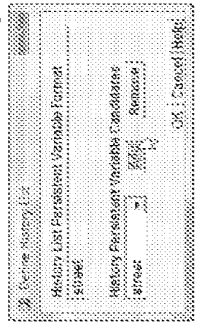
8. Non-Scalar History variables will be editable only on the page that has a History List object defined. When there is an exit from such a page, all history variables that are referred to by the History List will be simultaneously updated with their current value, only if there is uniqueness as a group of values.



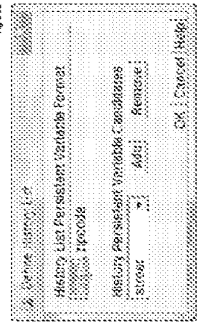
All History Persistent Variables that have been defined on the current page will be displayed as candidates. Any other references to History Variables from objects that are not on the same page that the "History List" resides will be display only, and will be set as non-editable.

In the example to the right there were two text fields that had been assigned "history" persistent variables "street" and "zipcode" respectively.

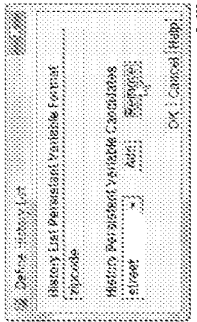
The graphic to the right shows the result when selecting the "street" candidate and touching the "Add" button. Note that the "Remove" button is now enabled.



The graphic to the right shows the result when selecting the "zipcode" candidate and touching the "Add" button. Note that a comma and space has been added as delimiters.



The Text Field is interactive so that touching "street" will select the "street" string as well as the "street" item in the candidate list. Alternatively the "street" item could be selected directly by touching the "street" item in the Candidates List.



Finally the graphic to the right shows the result when touching the "Remove" button when the "zipcode" candidate is selected.

6. Persistent Variable Naming Conventions

Persistent variables are all key-value pairs. The key name identifies the variable and the value is the stored information associated with the key in String form. The VariableName follows the convention:

Application\Author\Account.ApplicationName.VariableName.

D: Layout by Pixel vs. Layout by Font

1. Background

All Form objects currently are by, default, sized in terms of height, by the line height as generated by the selected font size. Submit, and Clear buttons have their width defined by the width of the character string, as also determined by the selected font metrics.

Although there is a provision to "don't scale" in many cases, it still means that the Author must control the actual layout through the font metric settings.

The difficulty is that MIDP does not have a font generator, and the fonts that are supported on phones can have very different font metrics, for the same logical font size.

Since all other platforms do have compatible font metrics, the current method will remain, but there will be a new selection under the "Project" menu, "Define Form Objects by Pixels". This will become a user preference and will be remembered between editing sessions. It can be changed at any time, even while editing an application.

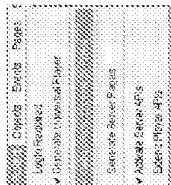


Fig 3.16

2. UI Implementation

Font Size will be a text field showing the calculated value and will not be selectable

Both the width and height fields will always be selectable. The height field will still represent the total height of the Object. For Text Areas and Lists the Line Height will always be the Object Height divided by the Number of Visible Lines.

In addition, all Form objects, other than a check box, will now have 8 drag points so that the object can be resized with the mouse. Upon releasing the mouse a new font, on a best fit basis, will be selected and the text will then be reformatted based on this new font. Only a change in the height will affect the font, a width change will only change the line lengths that will be drawn.

For Submit and Clear buttons there will remain a "Don't Scale" Check Box. If scaling is enabled then the width of the button will be dynamically calculated on the phone based on the actual length of the Text String after a best-fit font is selected.

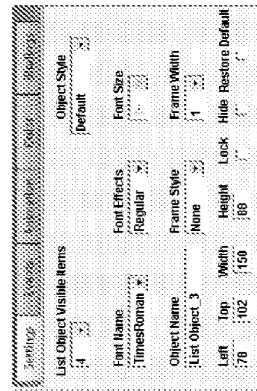


Fig 3.17



Fig 3.18

3: Effect on Vertical Positioning

All text objects, including text buttons and paragraphs will be affected.

The only exceptions are Lists as they were previously vertically aligned by the top of the bounding box of the first item in the list.

When in "Define Form Objects by Pixels" mode the following objects will be vertically aligned based on the top of the bounding box of the first text string.

- Paragraph
- Text Button
- Submit Button
- Clear button
- Text Field
- Text Area
- Drop Down (Choice) List.

When in the normal "Font Typographic" mode these objects will be vertically aligned on the baseline of the first text string.

E: Suppression of Selection Rectangle

Normally a selection rectangle will be drawn around any "handable" object as defined in Appendix D. The primary color can be set under the Color Tab when the "Master Page" is selected. The player will use this color to draw a 2 pixel wide 3d rectangle with the color variations selected as if the sun was coming from the upper left. (The algorithm is identical to that of Java AWT draw3DRectangle() api.

If the "Suppress Selection rectangle" checkbox is selected, then the object will still be selected but no selection rectangle will be drawn.

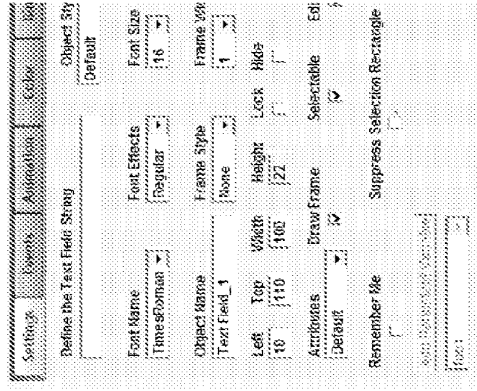


Fig 3.19

## F: Using Object Libraries

### 1: Saving

The following objects can be placed into a private library for use with any application:

- a. Slide Shows
- b. Lists
- c. Vectors
- d. Images
- e. Video

Images and Video, if they were originally resident on your local system, will automatically be placed in their respective libraries when an application is saved.

For Slide Shows, Lists or Vectors, be sure that the object is selected and then choose "Save As..." from the File Menu or touch the Save icon from the Command Floating Palette. Under File Type a choice will appear for saving the selected object.

All of these five object libraries will also have public versions. See your System Administrator for adding any of your private content into these public libraries if you want them available to all other Authors.

### 2: Retrieving

A standard way to retrieve object libraries is from the Object Libraries menu under the File menu

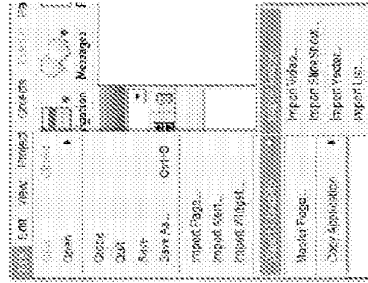


Fig. 3-40

There are also contextual ways to work with object libraries as described below.

### a. Slide Shows

Slide Shows can be imported into an application in the following ways:

- 1) Right Click Mouse and Select Slide Show Libraries;
- 2) Work with Slides and "Enter Slide Library" from the Import Icon.

### b. Lists

"Import List..." from the File Menu.

### c. Vectors

Vectors can be imported into an application in the following ways:

- 1) Right Click Mouse and Select Shape Libraries;
- 2) Shapes Library from the Shapes Icon from the Icon bar;
- 3) Touch "Enter Shapes Library" from Shapes0 from Objects in the Menu Bar.

### d. Images

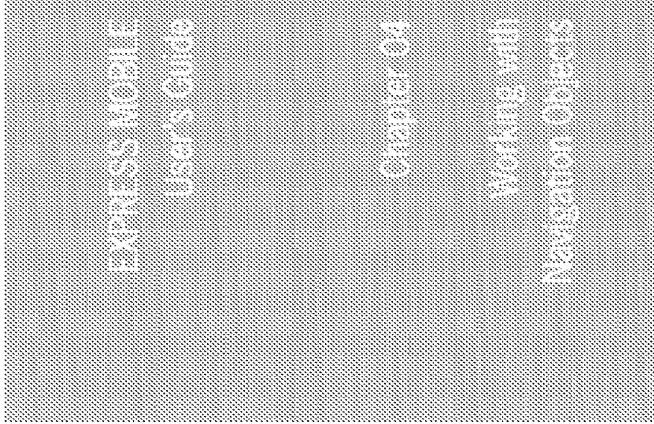
Images can be imported into an application in the following ways:

- 1) Picture from the Media Icon from the Icon bar
- 2) Touch the "Replace Photo File" button in the Resource Inspector.

### e. Video

Video can be imported into an application in the following ways:

- 3) Video from the Media Icon from the Icon bar
- 4) Touch the "Replace Video" button in the Resource Inspector.



### Chapter 4: Working with Navigation Objects

- A: Creating a Launch Strip 04.04
- B: Working with Launch Strips 04.04
- C: Working with Complex Lists 04.07
- D: Inter-Application Navigation 04.08
- E: Bubble Help Color 04.08

### Chapter 4: Glossary

Uber Launch Strip	This is the Launch Strip assigned to the Initial or Uber-Application. It will be available for any application page for the Uber-Application or any Sub-Application.
Explode Effect	The visual behavior associated with a launch strip as set by the author, when there is a change in focus to a icon.
Bubble Help	The optional text string that can be defined for each icon in a launch strip.



## A: Creating an Launch Strip

There are 2 starting points for Launch Strips as shown to the right. Once created, there are no restrictions in terms of working with these attributes.

The only differences in these two choices are:

- 1: Orientation
  - 2: Type of Launch Strip
- You can also add a slide transition and auto sound when navigating between icons.
- It is possible to have both a vertical and horizontal Launch strip defined simultaneously.

## B: Working with Launch Strips

Use the Resource Inspector to define the Launch Strip..

- 1: Maximum Number of Visible Slides

This setting will display, in the orientation selected, the number of actual slides, or the Visible Slides setting, whichever is smaller.

- 2: Explode Effect

Determines the visual effect of a slide being selected.

Explode Arc and Explode Icon are supported

- 3: Vertical

Switches orientation.

- 4: Explode Percent.

This defines the relationship between the Selected Icon Size and that of the smallest Icon. For Exploding Arc, the intermediate icons are the average between the two.

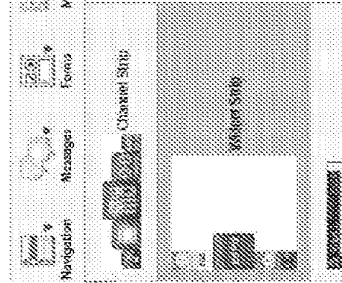


Fig 41

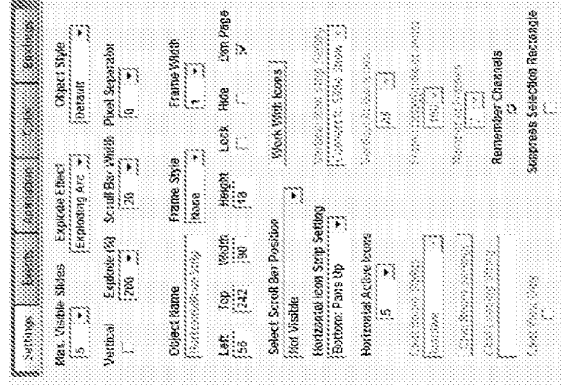


Fig 42

## 5: Scroll Bar Width and Select Scroll Bar Position

These settings control the presence, width, and placement for a Scroll Bar.

## 6: Pixel Separator

This is the number of pixels, from 0 to 15, that will define the separation between slides.

## 7: ObjectName

This name is fixed and is not editable.

## 8: Frame Style and Width:

Not used with Launch Strip.

## 9: Left, Top, Width and Height.

These values should generally not be changed, as the positioning and size is being calculated automatically by the number of visible slides, the size of the original images, the Explode Percentage Setting, the Exploding Effect, the Page Size, and the Orientation and placement selection. However, to adjust the value that controls centering from "centered within the page" to something else based on aesthetics is available.

## 10: Lock and Hide.

Not used with Launch Strip.

## 11: Dim Page

This determines whether the page should be dimmed with the Launch Strip is visible and active.

## 12: Work With Icons

This permits the insertion, deletion, and setting attributes for specific icons.

## 13: Setting Navigation for Starting an Application on a Specific Page

If the Execute Application Exit Event is chosen, then both a dropdown list for the available applications in your workspace will become available for selection, as well as a text field that lets you determine the page name that you want to be the initial page.

## 14: Horizontal Launch Strip Setting (or Vertical Launch Strip Setting)

This determines where the Launch strip will be placed based on its orientation, whether you want it always visible or enter from a page edge, or whether you want this Launch strip to become a slide show.

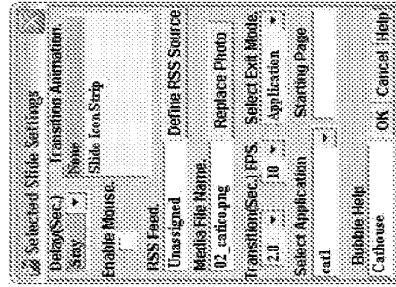


Fig 43



### D: Inter-Application Navigation

Similar to B.1.3 above, it is now possible to assign a particular starting page when utilizing the EditEvents for all XpressMo Objects and from a particular Choice Object Item or a Complex List Item.

There are up to 2 Uber Launch Strips that can be assigned during a Player Session, a Vertical Uber Launch Strip and/or a Horizontal Uber Launch Strip.

The first Uber Launch Strip-enabled application that is visited by the Player during a given session will be the owner of its Uber Launch Strip. This could be the UberPortal, or, assuming that the Uber Portal only has one Uber Launch Strip defined, then the first application that defines the other type of Uber Launch Strip will be the Owner.

Ownership as defined as having a set of icons that remains persistent throughout the session. These icons may be visible or invisible, but they will be persistent.

When a subordinate application is visited that has an Uber Launch Strip whose type already has an owner, then the subordinate applications icons will be appended to the Uber Launch Strip. This process can continue for any number of Application Layers.

When the owner of an Uber Launch Strip is visited, all subordinate icons are immediately removed.

### E: Bubble Help Color

There are 3 color controls for bubble help.

- 1: Bubble help background  
This will share the same color as the slide background color.
- 2: Bubble Help Text Color:  
This will be the same for all bubble help
- 3: Bubble Help Frame color  
This will be the same for all bubble help

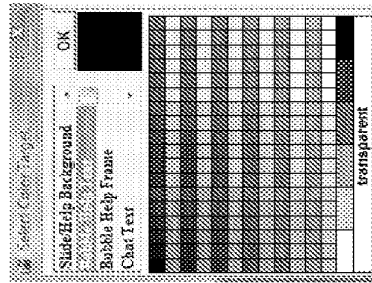
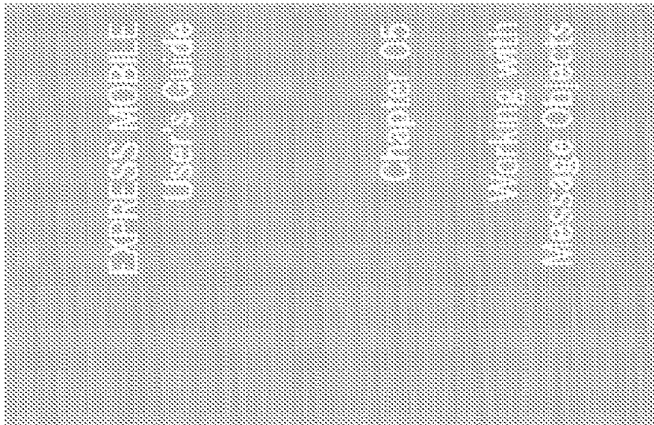


Fig. 47



## Chapter 5: Working with Message Objects

- A: Working with Chat 5.04
- B: Working with Phone Calls 5.07
- C: Working with SMS Messages 5.09
- D: Working with Rich Media Alerts 5.11

## Chapter 5: Glossary

PIM Data

The data available on the mobile device such as Contact Lists, TODO Lists, and Event Lists.

Rich Media Alerts

Alert Pages that can contain both static and dynamic content. These pages can be sent to one or more users that have the Express Mobile Player installed.

## A: Working with Chat

Setting Colors:

Chats add two additional elements to a Slide Show. The first is the Chat Trail itself and the second is a Text Edit Field where anyone you authorize can add time and date stamped messages to your Chat Trail. Once added, anyone who visits this Chat Application, and navigates to the particular slide that the message was added to, will be able to read that message, and through manual and/or automatic scrolling, view all messages that were added to this Chat Trail, in chronological or reverse chronological order.

To help add your own look and feel to your Chat you can set the text and background colors for both the scrollable Chat Trail as well as for the Editable Text Field.

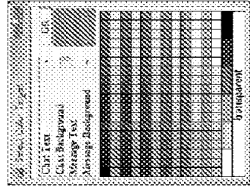


Fig 5.1

Setting the Chat:

a: Chat Status: This control gives you the ability to determine where you want the Chat to be placed relative to the slide show.

b: Chat Database: Publisher comes with a "Demo" Chat database installed, but you can add as many additional ones as you wish. Simply touch the "New" icon and then enter the name.

**Note:** A given slide show Chat can only have one Chat Database assigned to it, but each slide will point a different Chat Message Trail record in that database. However, you can have as many Chats as you want in your application, usually on a different page.

c: Date and Time Stamping: Depending upon your selections, a date and/or time stamp will appear just above each Chat message in the Chat Trail.

d: Visible Lines: A Chat trail can have up to 65,535 characters, so the full Chat trail could have many text lines defined. However, you can control the number of visible lines that will appear on your phone at one time. This selection will be determined by the design choices that you make for your Chat. For most phones it is recommended that you have no more than 4 lines visible due to the screen space that the Photo and/or Video will take, as well as the Navigation Icons, if selected.

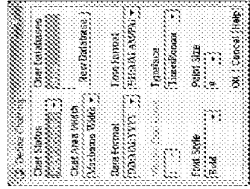


Fig 5.2

e: Maximum Chat Width: Based on the choice, the Chat will be centered as best as possible relative to the Slide Show. There are 4 choices:

- Slide Show Width
- Twice the Slide Show Width
- Three Times the Slide Show Width
- Maximum Width

f: Font Name, Style and Size: This sets the way the text will look, both in the Chat Trail and the Editable Text Field. Phones, however, usually only have a very limited number of Fonts and Font Sizes available. Express Mobile does its best to map your choices to what the phone can support. If you choose a Font Size of 9 or less, then for these limited Phones the "small" text size will be chosen. A size from 10 to 13 will map to the "medium" text size and over 13 will map to the "large" size.

Chat Operation on Wireless Devices

Chats adopt the characteristics of the device that they are being viewed on, so that the user experience is optimized.

The Chat above was designed and was running on a standard 3G phone, which usually has a screen size of 176 by 208 pixels, 2 soft keys, a joy stick, and 12 numeric buttons. Currently the Chat Trail for a particular application is being shown.

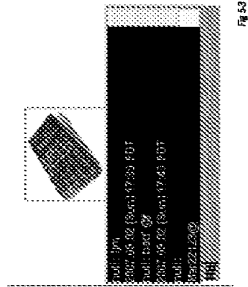


Fig. 53

1: MIDP Form Behaviors:

Assuming that the default Text Entry Mode is MIDP Form, or the user has used the "Toggle MIDP Text Entry Form" to enable T9, then the behaviors, when that Text Field is selected, upon touching any numeric keys or FIRE are:

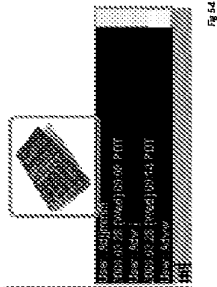


Fig. 54

Initial State:

a: User touches the "space" key.

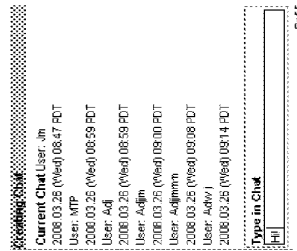


Fig. 55

b: User enters "this is a test.", Touches Soft Key

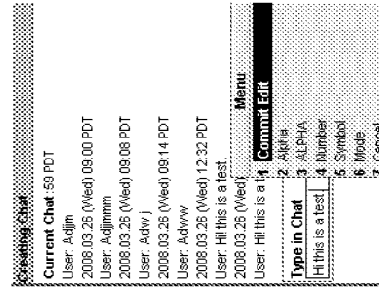


Fig. 56

c: Touches "Commit Edit" Result

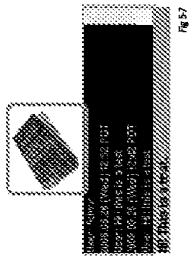


Fig. 57

d: Touches FIRE Chat has been sent and Text Field Reset

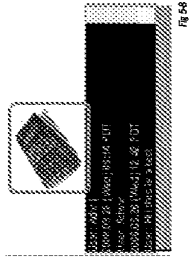


Fig. 58

2: Chat with Direct Text Entry Behaviors

Text will be entered directly into the message text field using the phones numeric key pad or keyboard.

a: Enter "Hi This is a Test."

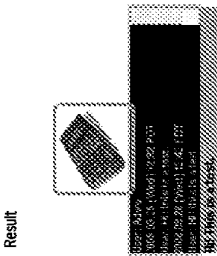


Fig. 59

b: Touch FIRE Chat has been sent and Text Field Reset

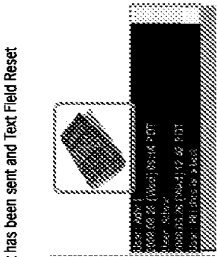


Fig. 60

The Soft Keys available are similar to that of MIDP Form text entry, except that there is no Commit command.

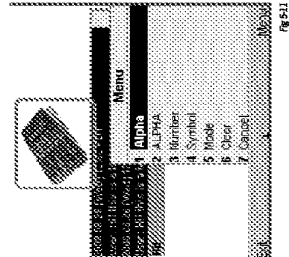


Fig. 61

## B: Working with Phone Calls.

### 1: Manual Entry

Phone Number Text fields are automatically defined as numeric only data entry fields. They can be defined as:

- N. Am Phone #

Under the condition that 10 numeric characters have been entered, when leaving the Text Field the numbers will be formatted as:

(xxx) xxxxxxx.

- Other Phone #

No formatting is applied.

When either of these two types of fields are fired on that phone call is placed. On some phones it will be required to exit the application before the phone call can be placed, on others it will be placed immediately.

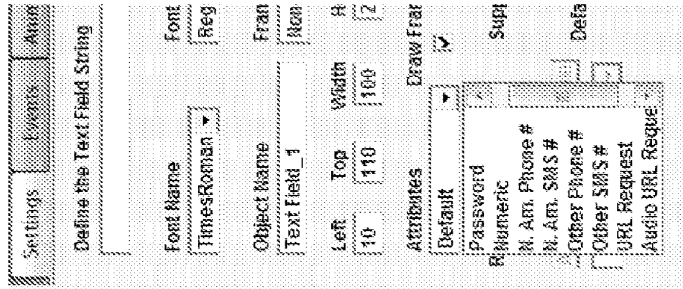


Fig 5.12

### 2: Using Web Components

Various Web Components automatically generate Phone Lists.

Two examples are:

#### MapQuest PointsOfInterest:

When using the MapQuest PointsOfInterest feature, a list is returned with the names of those points-of-interest that were returned on the map. Firing on any location in that list will place a phone call to that location.

#### Face Book:

One feature for Face Book is to present a Buddy List Phone Directory. When firing on any name in that list a phone call will be placed to that person.

### 3: From Phone Contact List (PIM Data)

If the Phone List Message object is selected, and the phone supports access to the Phone's Contact List, then this list will automatically be populated with the names on that list. Firing on any name will place a phone call to that person.

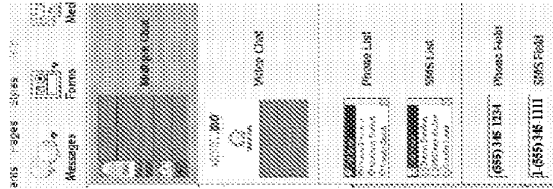


Fig 5.12

### C: Working with SMS Messages

1: Manual Entry

SMS Number Text fields are automatically defined as numeric only data entry fields. They can be defined as:

- N. Am SMS #: Under the condition that 10 or 11 numeric characters have been entered, when leaving the Text Field the numbers will be formatted as: 1-(xxx) xxx-xxxx.
- In the event only 10 numeric characters were entered then the number "1" will be prepended.

- Other SMS #: No formatting is applied.

It is necessary to apply an Exit Event to this field "Send SMS". At that time the author must select either a Text Field or Text Area that will contain the SMS message. When either of these two types of fields are fired or that SMS message will be sent.

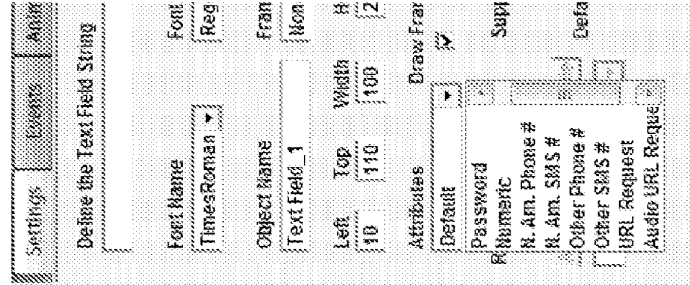


Fig 512

2: From Phone Contact List (PM Data)

When inserting this "SMS List" object into an application, the author must go to the Item Attributes Tab and select a Text Field or Text Area for linking that field's contents for the SMS message. If the SMS List Message object is selected, and the phone supports access to the Phone's Contact List, then this list will automatically be populated with the names on that list. Firing on any name will send the SMS message to that person.

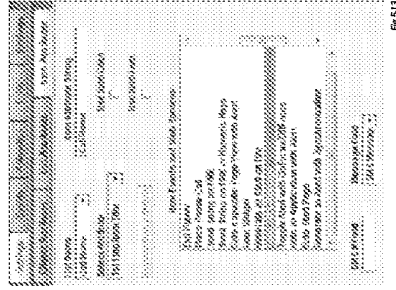


Fig 513

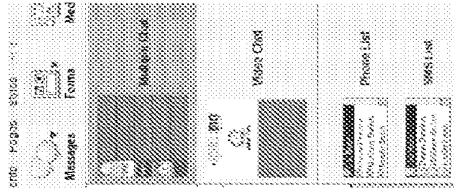


Fig 514



## D: Working with Rich Media Alerts.

1: Using Chat for Instant Messaging.

Section A of this chapter describes the various ways of chatting with another person that is connected to the Internet.

2: Using Web Components

One feature for the Face Book Web Component is to present a Buddy List. Firing on any name in that list will send an IM to that person's Face Book account. They will be notified of this pending message either through their Xpressmo Face Book Web Component Application and/or their Desktop Face Book Page.

3: Sending Rich Media Alert Messages.

The author can define Alert Pages for encapsulating rich media content as well as a user generated Instant Message. Included in this Alert Page could be dynamically defined Images or Video for sending real-time content from one person to others.

The following must be defined by the author:

An Alert Page:

This is where dynamic content such as Live Video, RSS Feeds, etc. could be linked.

A Message Field:

A Text Field or Text Area where the User generated message will be entered.

A Phone Number Field:

This is where the user will enter the phone number. The author must place a "Send Instant Alert Message" Exit Event on the Phone Number text Field and select which Popup Alert and which Message Field they want to associate this operation with.

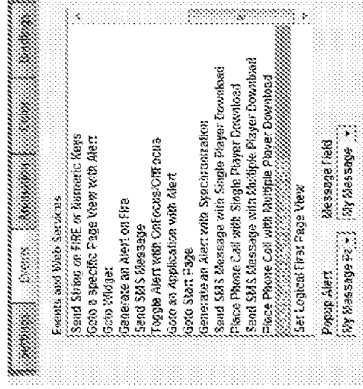
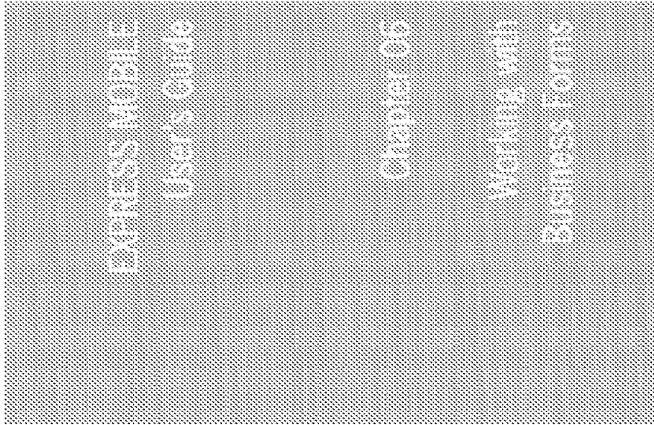


FIG 515



## Chapter 6: Working with Business Forms

A:	Introduction	06.04
B:	Text Fields	06.05
C:	Always On Text Fields	06.06
D:	Lists and Dropdowns	06.09
E:	Text Entry	06.23

## Chapter 6: Glossary

List Layout Style

An extension of Object styles that apply to the default settings available for complex lists.

RSS List

One of many types of Lists that have available templates in the Publisher that can be created and then subsequently be changed. The RSS List Template specifically defines a convenient way for assigning RSS feeds to a List and then displaying its fields in a user-friendly way.

Fire

The act of touching the FIRE key, touching the screen with a pointing device, clicking the mouse, or pressing the ENTER key on a keyboard.

## A: Introduction

Business forms are a set of very powerful UI objects that can be used in many ways for, among other things:

1. Transactions
2. Sophisticated Navigation Objects
3. Enhancing Web Components.

All business forms can have the same set of Events, Colors, Animations and Bindings that are available to any other UI Object.

Text Fields, Dropdown Menus, and Selection Lists are further discussed later in this chapter.

Scrolling Text Areas behave as a browser-based Text Area would behave. The number of visible lines can be set, and a scroll bar will automatically be drawn if the number of text lines exceed that of the number of visible lines.

Check Boxes behave as a browser-based Check Box would behave.

Submit Buttons will, by default, send the current values for all the other business form objects up to the Express Mobile Transaction Server for processing.

Clear Buttons will reset the values of all the business forms on the page to their initial values when the page was first displayed.

Submit and Clear buttons have a "Don't Scale" state. If enabled the MIDP player will maintain the original width of the button object. This is useful if you are superimposing these type of buttons over rasterized text or images.

## B: Text Fields

The following types of text fields are supported:

1. Default (normal alpha numeric field)
2. Password
3. Numeric
4. North American Formatted Phone Number
5. North American Formatted SMS Number
6. Unformatted Phone Number
7. Unformatted SMS Number
8. URL Request (See Web Components)
9. Audio URL Request (See Web Components)
10. Purchase URL Request (See Web Components)

If Text Fields are of the types:

- 1: Numeric
- 2: Phone Number
- 3: SMS Number

Then the text entry model is always direct. Only numeric character can be entered. Cursor movement, editing and deleting within the string are also supported. When a direct entry text field is selected the "back" key, if available on the device, will similarly delete one character to the right of the insertion point.

In the event that the Text Field has the attribute Phone Number, then, under the condition that 10 numeric characters have been entered, when leaving the Text Field the numbers will be formatted as:

(xxx) xxx-xxxx.

In the event that the Text Field has the attribute SMS Number, then:

- 1: under the condition that 10 numeric characters have been entered, when leaving the Text Field the number "1" will be prepended.

In either case, the number will be formatted as:

1-(xxx) xxx-xxxx.

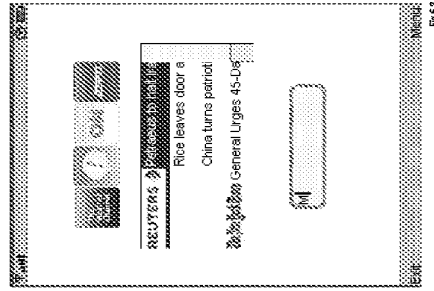
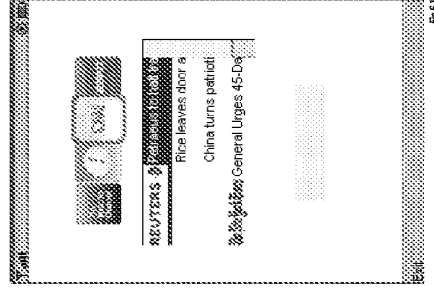
When entering these text fields the formatting will be removed and just the numeric characters will display.

## C: Always On Text Fields

- 1: Background

Certain applications may have pages where the text entry function should be always available, even as the user can navigate and operate on other objects on that same page.

For example, the page could have a Slide Show and a List. If while either is selected and the '6' key is touched, the focus and selection will switch immediately to the Always On Text Field and the letter 'M' will display and a blinking insertion point will appear. From that point on, the Always On Text Field operates as a normal data entry field, as specified in the Publisher.



2: UI Implementation

Using the example above the Application, with the Always On Text Field Selected is as shown below.

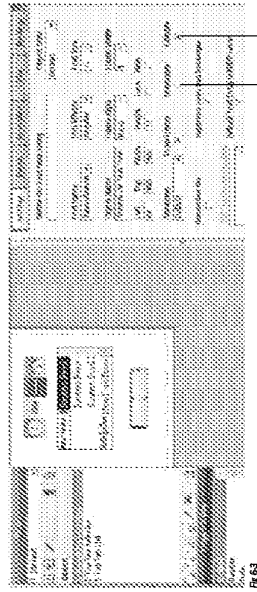


Fig 63

The following conditions must be met for a Text Field to support the 'Always On' attribute.

- There is a total of one Text Field, either in the Master Page or the Application Page.
- There are no Text Areas in either the Master Page or the Application Page
- The Text Field is set to:
  - Selectable=false
  - Editable=true

3: MDP Player Implementation

As shown in the Section 1, when the Text Field has been selected through a data entry event, the soft key 'Menu' will appear. Touching that Soft Key will offer the same choices as expected.

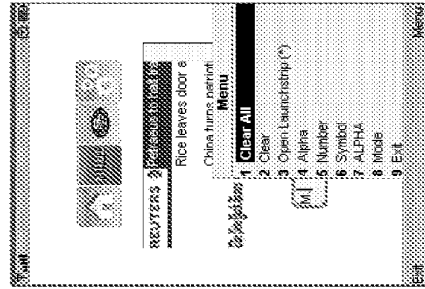


Fig 64

Any navigation event that is valid for a Text Field, such as an UP event, will cause the Selection Rectangle and Focus to move to the appropriate object on that page and the blinking insertion point to disappear, as shown in the graphic to the right.

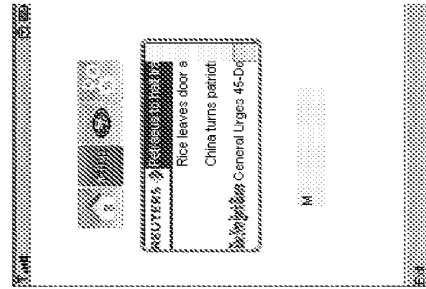


Fig 65

## D: Lists and Dropdowns

### 1: Complex List Types.

Lists can be of the following types:

- a. Dropdown Choice
  - Default
  - Active Channel Personalization
  - Inactive Channel Personalization
  - Search From List
  - Widget Selection List from User's Library
  - Phone List
  - SMS List
  - Widget Selection list from Runtime Widget Library
  - Phone List from PIM Data
  - SMS List from PIM Data
  - Search Response list
  - RSS List
  - Control List
  - History List (See Chapter 3: Persistent Variables)
  - Web Component List

### b. Selection List

- Default
- Multi-Select
- Category List
- Channel Personalization
- Widget Selection List From User's Library
- Phone List
- SMS List
- Widget Selection list from Runtime Widget Library
- Phone List from PIM Data
- SMS List from PIM Data
- Search Response list
- RSS List
- Control List
- Web Component List

Services are supplied to support the following:

- a. Personalization of the Channel Launch Strip
 

Dropdown lists have list types for both adding and removing channels (icons and their related intelligence) from the Channel Launch Strip.

If the Channel Launch Strip has the "Remember Channels" attribute selected, then in all cases the current state of the Active Channels will be reflected in these Lists, and the memory of the selected active channels will be saved across sessions.
- b. Widget Selection from User's Library
 

The list object will automatically populate with the icons and the widget names that exist in the author's private widget library. No new items can be added or existing widgets deleted, but other attributes for each widget list item can be added.
- c. Widget Selection from Runtime Widget Library

Both Lists have Widget Selection Lists that are automatically populated with an entry for each Widget that is available to that player. Each Widget, at authoring time, can be assigned a unique icon.

If an item in a Widget Selection list has a "Goto Widget Object" Exit Event then a Widget Icon Will be drawn. If the Widget Page has already had a Page Icon defined then it will be drawn. Otherwise the default Widget Icon will be drawn.

At runtime when firing on an item in a Widget Selection List that Widget will automatically be loaded and executed on the device.



Fig 66

### d. Xpressmo Server Page Selection

For both Lists that are set to their default type, if there are any of the following exit events associated with a List or Choice Item:

- Goto a specific Internal Web Page
- Goto the next Internal Web Page
- Goto a specific Internal Web Page with setting starting slide
- Goto a specific Internal Web Page with Alert
- Goto Logical First Page

and that internal WebPage has a Page Icon defined then it will be drawn. Otherwise, no icon will be drawn.

### e. Automatic Population and Execution of Phone and SMS Lists:

Both Lists can be set to be automatically populated with the PIM data defined in the phone's contact list.

If the device supports and permits access to the PIM content list data, then when firing on a Phone List item the following will occur. If the phone supports simultaneous use of the phone with an application then the Xpressmo application will continue, otherwise the user will be prompted whether they wish to exit the Xpressmo application and then place the phone call

If the device supports and permits access to the PIM content list data, then when firing on an SMS List item the SMS message will immediately be sent to that contact.

2. List Layout Styles.

All object styles have an associated List Layout Style. The author can create as many List Layout Styles as desired by selecting the "Create List Button". If so, then that new List Layout Style will become the associated List layout Style for that Object Style. The author can always change which is the associated List layout Style for a given Object Style by changing the selected List object Style.

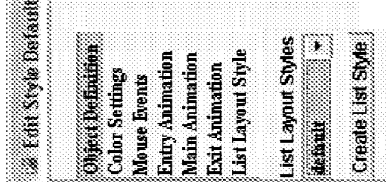


Fig 67

If the author wants to change the definition for a given List Layout Style then Select the "List Layout Style" in the Object style's Settings list will display the dialog box below.

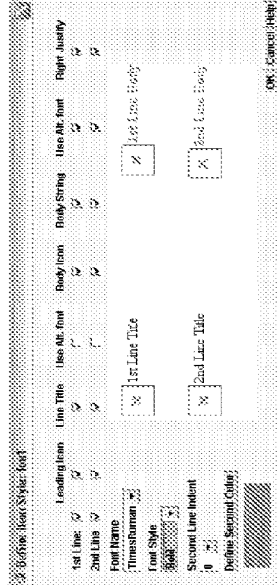


Fig 69

The List Layout Style has the following controls:

- a. First Line:
  - Title Icon (optional)
  - Title String (mandatory)
  - Body Icon (Optional)
  - Body Text (Optional)

The body icon and text can be left or right justified.

The Title Text and body text can have a selection of either 2 different fonts, including color. The font size will always be derived by the Object Style Font Size for both fonts.

b. Second line (Optional)

- An indent can be defined relative to the first line.
- All other settings available for the first line are available and optional for the second line.

3. Working with Selection Lists.

Selection Lists have "Settings" attributes that are defined at three distinct levels:

- Object Level
- List Level
- List Item Level

You can switch between attributes levels by firing on the desired tab.

Object Attributes

Object Attributes deal with the basic Selection list object, including the Object's font Style, number of List Items that are visible (which either directly or indirectly define the height of the List object).

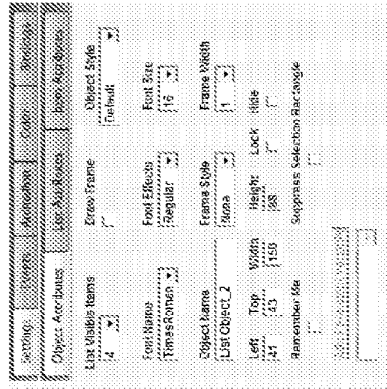


Fig 610

4; List Attributes

List items can be added by selecting the "Add Item" button. The initial item name, and its corresponding "Title String" will be set to a default value of "ItemNumber#" where # will be the number in the list that the new item was placed in, and immediately appear in the List Items Dropdown menu.

The settings for Define List Item Layout are initially inherited from the List Layout Style that is assigned to the current Object Style when the Selection List Object was created.

Any other List Style can now be selected if desired by selecting a list Style from the List Style Dropdown menu.

Any changes made to the "List Item Layout" will override the inherited settings. Any changes in terms of the availability of icons and/or Strings at the list item level will be reflected in the "Item Attributes" screen when a new item is created. At the List Attributes level the following controls are available

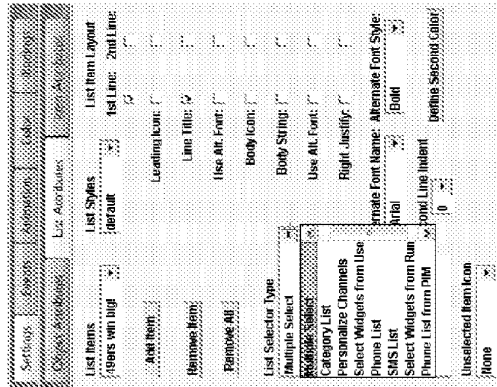


Fig 6.11

1. For Selection Lists of types Default, Search Response, RSS Lists, Multiple Select, Category, Select Channels and Select Widgets, the following elements can be defined..

- a. First Line:
  - i. Title icon (optional)
  - ii. Title String (mandatory)
  - iii. Body Icon (Optional)
  - iv. Body Text (Optional)
  - v. The body icon and text can be left or right justified.
  - vi. The Title text and body text can have a selection of either 2 different fonts and/or colors.
- b. Second line (Optional)
  - i. An indent can be defined relative to the first line.
  - ii. All other settings available for the first line are available and optional for the second line.

- 2. For Multiple Select Lists a palette of both selection and unselected vectors and icons are available.
- 3. For Category Lists a palette of vectors and icons are available.
- 4. For Search Response Lists:

The author can create a fully defined Search Response List from the Text Dropdown Menu.

- Four objects will be created:
- Search Response List
  - Text Field
  - Text Area
  - Slide Show

They will be logically linked together. The author will then be presented with three dropdown lists for possible defining the logical relationship with any Search Response List item and other related objects on the page that could display metadata that is associated with that response. The maximum number of possible Search Response List items can be set.

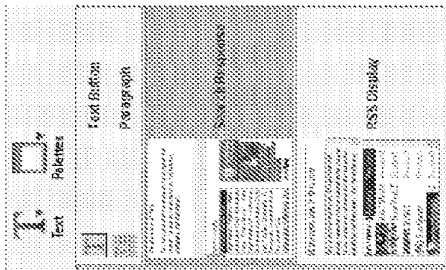


Fig 6.12



The Three Logically Linked Objects are:

- **Title Object**  
This could be either a text button or a text field, the difference being whether the width is fixed with possible scrolling or the width adapts itself to the length of the title string.  
If a text field, then the "Title Marquee Enabled" check box becomes enabled. If selected, then, if the string for the title is too long for the width of the text field, a marquee animation will begin when this search result is selected.
- **Message Object**  
This could be either a paragraph or a text area, the difference being whether the number of visible lines is fixed with possible scrolling or the height adapts itself to the number of lines after the text string has been formatted to the width of the text area
- **Media Object**  
This must be a Slide Show. Any image or video metadata will be attached to slide that is in the same relative position as the list item.

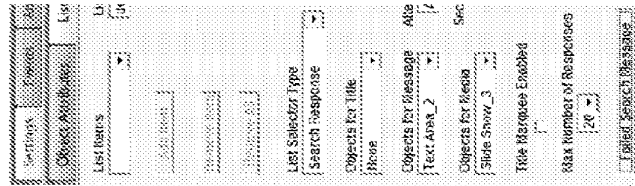


Fig 613

During runtime, the object names for the Title, Message and Media object will be used to dynamically update their content as each item in the Response List is visited. As with all list items, any exit event can be assigned.

The author can also override the default error alert that will be displayed by the player if the search returns no responses.

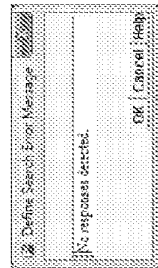


Fig 614

5: For RSS Display Lists:

- The author can create an RSS Display List in a manner very similar to that of a Search Response list.  
In the Template available to the right, The Title and Message Abstract Objects are defined and logically connected to the RSS Display Lists.
- This template also places, if available, the Channel's Image icon as the title icon. The icon is automatically sized so that its height will be equal to the item's height, and the width is then calculated to maintain the image's aspect ratio.
- This is the second example of a Web Component utilizing the same technology as that of a Search Response List. See Chapter 11 for a full description.

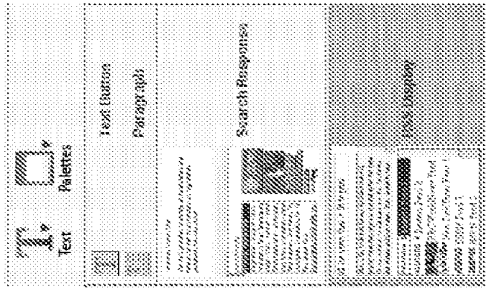


Fig 615

The author adds channels by selecting the "Add Item" button under the "List Attributes" tab of the Resource Inspector. In addition to selecting the RSS Channel, the author can set the maximum number of feeds that will be extracted and displayed on the mobile device.

The author can remove any of the channels or remove all of the channels as desired. The Title Object String is dynamically bound to the 1st line title string of List. The Abstract Object String is dynamically bound to the 2nd line body string of the List.

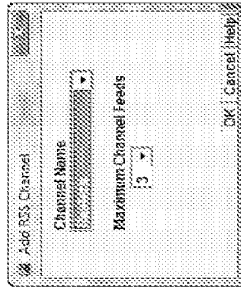


Fig 616

Item Attributes

The graphic to the right shows the choices if a String item element is selected. In that case there are choices on whether to apply the Alternate Item Color and/or the Alternate Item font. Any choices will override the defaults that were inherited from the List Layout. The "Item Attribute String" can be edited and will immediately appear in the Workspace's Selected List Object. If the "Selected Attribute" is the 1st Line Item Title then this will become the List Item name as displayed in the "List Item" Dropdown menu under the List Attribute Tab.

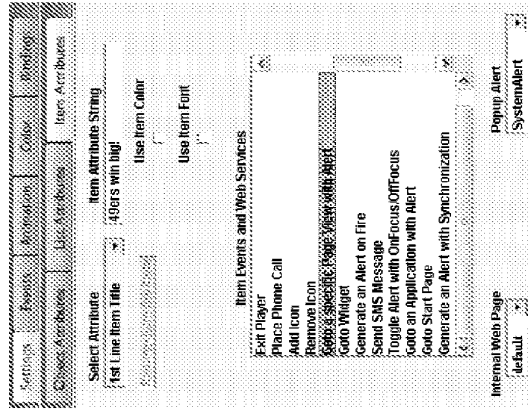


Fig 6-17

The graphic to the right shows the choices if an icon element is selected. If "Select an Image Icon" is touched then the "Icon Management" dialog box is displayed. Note that the 1st Line Title attribute can never be removed.

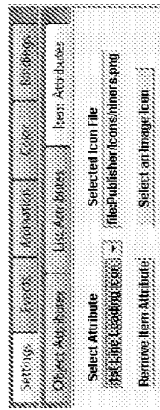


Fig 6-18

5: Working with Dropdown Choice Lists.

Touching the "Add Item" button will display a dialog box where the item's attributes can be fully defined. If these attributes are to be changed, then select that item in the "Dropdown Items" Dropdown menu, and Select the "Edit Item" button.

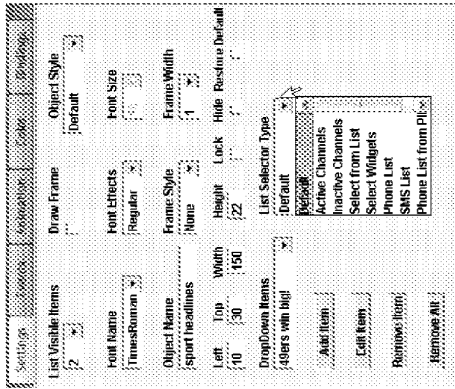


Fig 6-19

The differences in List Selector Type from a Selection list are as follows:

- Multiple Select and Category Selector Types are not available.
- Personalization of the Channel Launch Strip. The "Select Channels" selector type is replaced by "Active Channels" and "Inactive Channels".
- Search From List

The "Search From List" Object can be populated in the same ways as other lists.

When unselected, the "Search From List" Object will draw in a manner similar to that of a Text Field. When selected the following actions will occur:

For the indirect Text Entry Mode:

- On pressing Fire, or a text or numeric key the page view will switch as in a manner similar to that of a Text Field with the text field on the top of the screen representing the last successful search, if any. The search list items, starting from the first valid match (in alphabetical order) will be drawn below the Text Field.
- Characters can be deleted and/or entered.
- As soon as the first character is entered the Selector List will open.
- If the entered character is valid (the current search string has at least one partial match to a search list item) then:

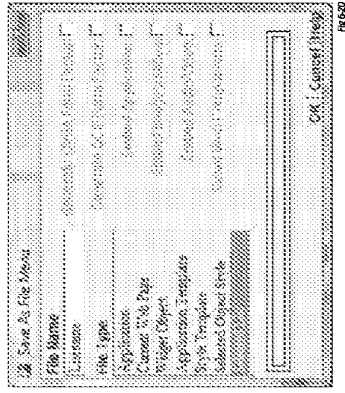
- If there is only one unique match then:
  - The page view will return to the original page view
  - The unique search list item will be selected.
- If there is more than one list item partial match then the search list items drawn below the Text Field will be redrawn, starting now from the first valid partial match of the current search string..
- If the character entered defines a string that does not match any search list items, then that illegal character will be removed.
- There will be two soft key commands.
  - Cancel:
  - Commit

The following behaviors will be supported:

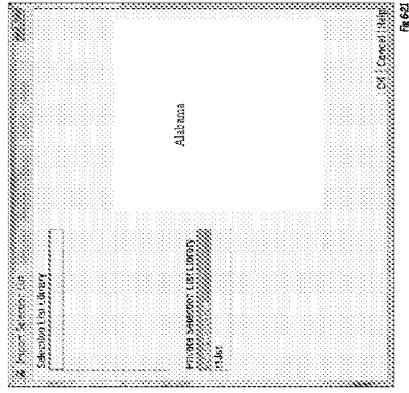
- If the string entered is completely valid and matches, at least in part, a minimum of two items in the list, then:
  - The first item that has a successful partial match will be selected.
  - The dropdown will be open for navigation to other possible search list items.
  - The search list items in the dropdown will start from the currently selected item.
- Search is not case sensitive.

6: Working with Selection List Libraries.

If a dropdown list or selection list is selected and a "Save As" operation is requested, then there will be a new Choice under File Types.



These Saved Lists, including all their icons and dynamic binding, can now be imported into an application, either from a public Xpressmo library or from the User's List Library. This process begins by touching "Import List" under the "File" Menu.



7: Icon management.

Icons are similar to images in that they have both private and public libraries and can be seamlessly uploaded from your PC to your private library.

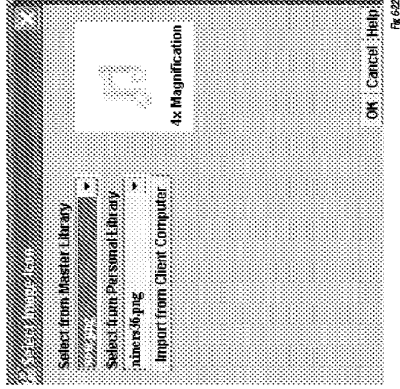


Fig 6.22

8: Suppress Frame

Only applies to List Objects.

If this checkbox is Selected and the number of items in the list are less than or equal to the number that are visible, then the Frame drawing will be suppressed.

9: Non-scaling Lists

By choosing the 'Define Form Objects by Pixels' under the 'Project Menu' the List will now be defined similar to that of an image. The location and size are specifically defined, and they can be manipulated by dragging an attachment point.

The font size, when in 'Define Form Objects by Pixels' mode, will be calculated based on the number of items that are visible and the height of the List object.

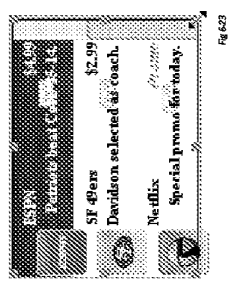


Fig 6.23

10: Working with the Edit Dialog Box.

You can work with a Selection List utilizing the Edit Dialog Box which is available from the Layer Inspector or by right clicking over the Selection list Object.

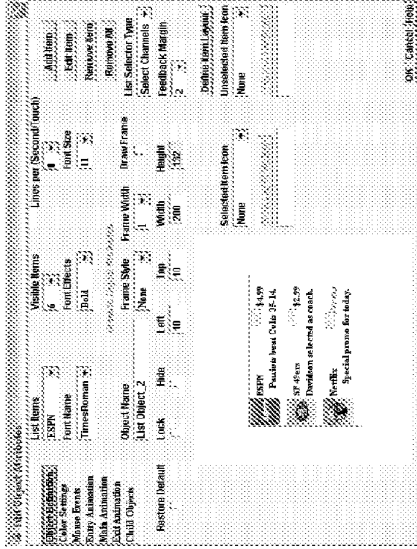


Fig 6.24

If the 'Define Item Layout' button is touched then, in a manner identical to the 'List Attributes' screen in the resource inspector, the Item Layout settings, which were inherited from the Object Style, can be changed.

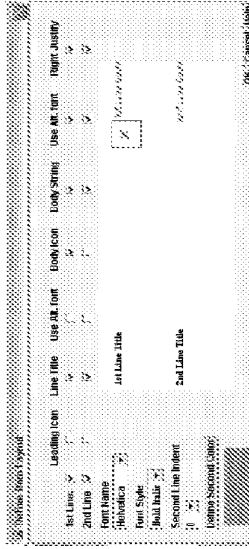


Fig 6.25

When touching either the 'Add Item' or 'Edit Item' buttons, the dialog box will have an 'Extended Item Attributes' button. This will display the 'Extended Item Attributes' dialog box which will work in a manner similar to that of the 'Item Attributes' screen in the resource inspector.

### E: Text Entry

1: Background

For languages that can be sufficiently supported using Unicode characters 33 through 126 (this excludes, among others, Chinese, Japanese and Korean) it is possible to support direct text entry as long as the device returns to Java the keycode that was touched. Currently all phones with numeric key pads will be supported. MIDP 2.0 devices with soft keyboards currently will have to continue to use the indirect "MIDP Form" text entry model. Also, all CDC and J2SE devices will also be supported.

For MIDP 2.0 devices with soft keyboards we probably will need to rely on the Response Director to inform the player that it must use the "MIDP Form" text entry model. The current implementation will assume a numeric keypad so it will not work on MIDP 2.0 devices with soft keyboards.

2: Supported Objects for Direct Text Entry

The following objects will be supported in build 2.1.46:

- Text Fields
- Chat Messages
- Selection Lists
- Used for Forms to minimize touches
- Used for prompted Search to assist User in selecting a useful search term

3: UI

In addition to all the other attributes that can be assigned to Text Fields, it is possible for the author to set the default text entry mode.

The user can subsequently change the input mode (See "4: Behaviors" on the following page).

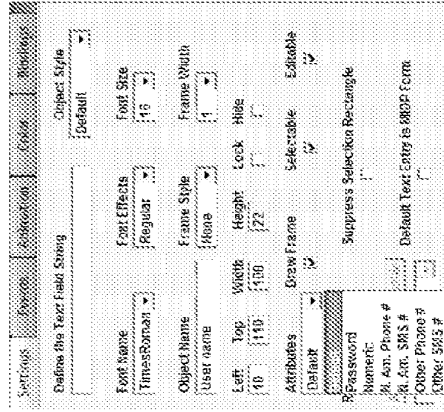


Fig 6.26

4; Behaviors

- a. Direct Text Entry Behaviors:

When a supported object is selected it will appear as a text field with a blinking cursor.

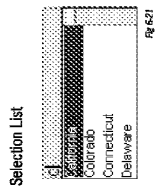


Fig 6-27

Standard Text Field



Fig 6-27

A text insertion point will be active and will blink 2.5 times per second.

There are seven possible text entry states for Standard Text Fields.

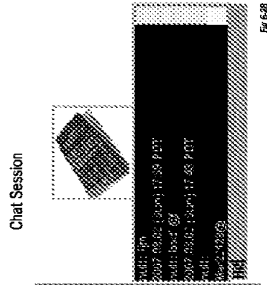


Fig 6-28

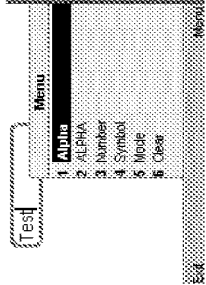


Fig 6-29

There are eight possible text entry states for Selection Lists and Chat Text Fields which can be selected through soft key commands.

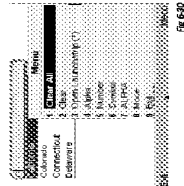


Fig 6-30

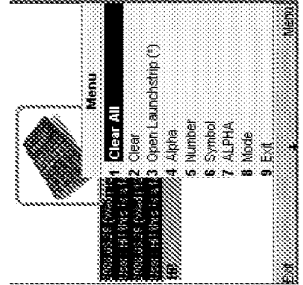


Fig 6-31

Alpha

- In this state touching a numeric key will cause an immediate echo of the first character assigned to that key. As long as the delay before touching that same key is less than 1.6 seconds the insertion point will not move and the next character assigned to that key will be echoed. If the number of touches exceed the number of characters assigned to that key the choices will be repeated. If either a different key is touched or the same key is touched after a 1.6 second delay, then the insertion point moves one character to the right, previous character is shown, and the new character, as appropriate, is echoed at the insertion point.
- Capitalization will occur if the character is an alpha character and it's the first character in the string, or the character is immediately preceded by a space character.

The current implementation is identical to that of the Sprint implementation for Samsung A900 phones. (See appendix A)

- ALPHA
- Same behaviors as with "Alpha" except that all characters entered will be in uppercase.
- Numbers

In this state touching a numeric key will cause an immediate echo of the numeric character assigned to that key and the insertion point will move one character to the right.

- Symbols
- When in this state a palette of symbols will be drawn on the top half of the screen, with a selection rectangle drawn either around the symbol in the upper left, or the last symbol selected. Navigation will work in both dimensions, with recycling when hitting a symbol palette boundary. Once the desired symbol is selected FIRE will place the symbol in the text field at the insertion point and the symbol palette will be erased. A soft key command is available to escape from the symbol palette. See Appendix A for list of symbols available from the Symbol palette.

- Clear
- Clear will delete the character to the left of the insertion point but will only be available when not on the leftmost character.

- Clear All
- All the content in the text field will be removed
- Mode

The user can switch between Direct Text Entry and MDP Form text entry. This gives the user access to 19 support.

The following additional soft key Command will be available for Chat Messages and Selection Lists:

- "Cancel"
- In both cases the field will be cleared of all characters.

The following Navigation Keys will be supported.

- LEFT

This will move the insertion point left one character. If already at the left most character then this will cause the focus to leave this object and the next appropriate object will be selected.

- RIGHT

This will move the insertion point right one character. If already at the right most character then this will cause the focus to leave this object.

- FIRE

For Chat Messages and Selection Lists FIRE will complete the operation. For example:

- With Chat the chat message will be sent to the user's audience as well as become visible, with the defined time stamp and user handle, preceding the message.
- For Selection Lists that are inputs to a Search Function, the partial or complete search term will be sent to the search engine for responses.
- For Selection Lists used for minimizing touches when selecting a string from a list, the Selection List will close and display the current value as defined below.
- For Text fields the Exit Event, if any, will be executed.

- BACK

If available, it will delete the character to the left of the insertion point. However, when not on the leftmost character, the "Back One View" operation will be executed.

- UP and DOWN

These will remain as object selection keys when on a text field. If on a Chat Message or List then their meaning has already been in the Chat and Complex List chapters of the User's Guide.

Special Behaviors for Selection Lists when in Direct Text Entry Mode

When unselected, the Selection List Object will draw in a manner similar to that of a Text Field. When selected the following actions will occur:

- As soon as the first character is entered the Selector List will open.
- Characters can be deleted and/or inserted.
- If the entered character is valid (the current search string has at least one partial match to a search list item) then:
  - If there is only one unique match then the unique search list item will be selected and the Selection List will close.
  - If there is more than one list item partial match then the search list items drawn below the Text Field will be redrawn, starting now from the first valid partial match of the current search string.
- If the character entered defines a string that does not match any search list items the character will be inserted but the selection list items will not be redrawn.
- While the selection list is open the UP and DOWN operations will be supported.
- If FIRE is touched:
  - If there is a first valid partial match with the selected item, or the item selected after navigating via the UP and DOWN operations, that item is selected and the list will close.
  - If there is no match then the list will close with the current non-matching entry will remain visible.



Fig 6-32

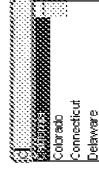


Fig 6-33

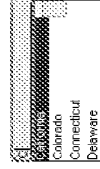


Fig 6-34

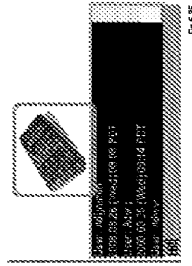
b. MDP Form Behaviors:

Assuming that the default Text Entry Mode is MDP Form, or the user has used the "Toggle MDP Text Entry Form" to enable T9, then the behaviors, when that Text Field is selected, upon touching any numeric keys or FIRE are:

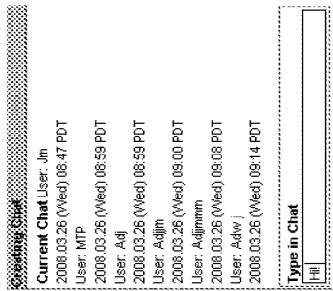
Chat Text Fields

Initial State:

- a: User touches the "space" key.
- b: User enters "this is a test".



Touches Soft Key



c: Touches "Commit Edit" Result

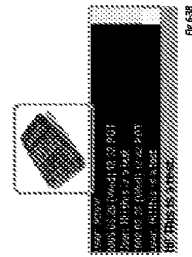


Fig 6.38

d: Touches FIRE Chat has been sent and Text Field Reset

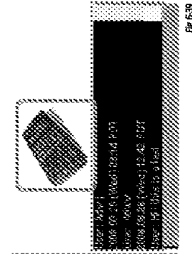
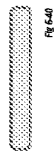


Fig 6.37

Selection Lists



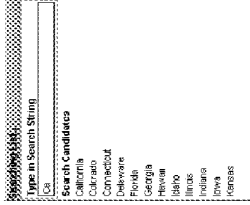
Initial State

When the first character is entered a Form displays in which the search string is on top, and below it are the search candidates, sorted in alphabetical order, that are equal or higher in the collating sequence of the search string.

As with Direct Text Entry, characters can be deleted and/or inserted.

All other Direct Text Entry behaviors, with the exception of navigating with UP and DOWN events through the search candidate list, are also supported.

When a unique search string (as shown below/left) Occurs, then the Form will close and the final result will appear as shown below/right.

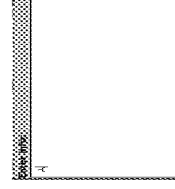


Standard Text Fields

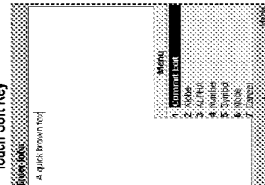
a: Initial State



b: Touch the "Z" numeric key.

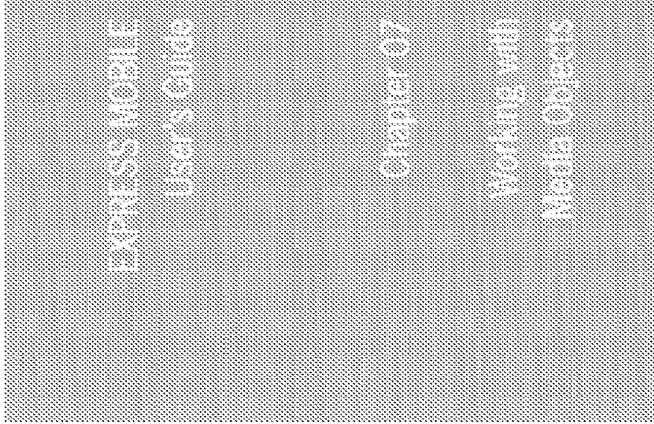


c: Enter "quick brown fox".



d: Final State





### Chapter 7: Working with Media Objects

- A: Pictures (Bitmap Images) 07.04
- B: How to Edit Using Photo Enhance 07.05
- C: Slide Shows 07.09
- D: Video 07.16
- E: LBS Maps 07.19

### Chapter 7: Glossary

- Library  
All media objects can have both private and public libraries, where frequently used media content can be placed for subsequent use in future editing sessions and with any application.
- Slide Show  
A very powerful collection of images and or Videos, organized in various ways, visualized in various ways, and behave in various ways. The author has full control over these options.

### A: Pictures (Bitmap Images)

Pictures may be imported from:

- The designer's local computer
- The designer's private library
- The public library

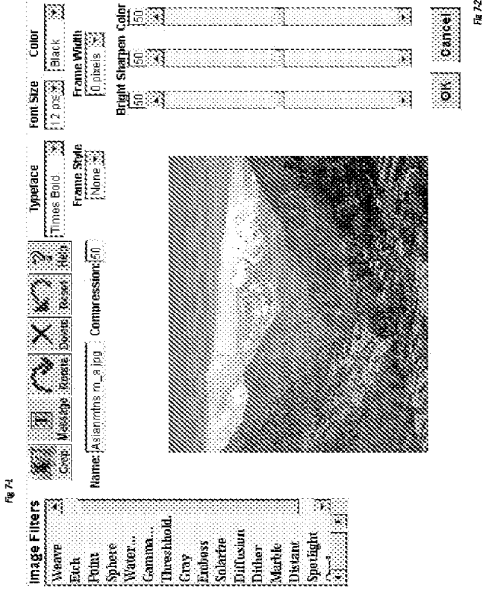
Any picture that is imported from the designer's local computer will, when the application is saved, automatically be uploaded and placed into the designer's private library for subsequent use with other applications and/or pages in the current application.

If the picture is resized, a scaled variant of that picture will automatically be created, and placed with the original in the application's image folder. For performance and bandwidth efficiency, it is the scaled version of the picture that will be downloaded to the network connected device when the application is being executed. This automatic scaling also occurs for all pictures that have been placed in Slide Shows.

If the picture was in the GIF format, then when the application is saved a PNG version of the picture is automatically created. This means that for phones that only support PNG there will be valid picture for downloading. This process also happens automatically.

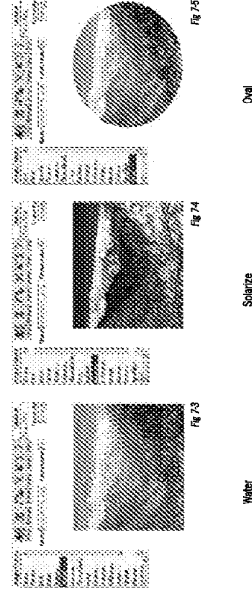
### B: How to Edit using Photo Enhance

The "Photo Enhance" feature allows you to add many different effects to your images.



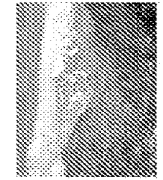
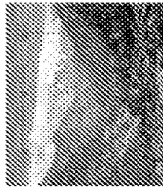
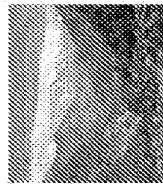
#### 1. Image Filters:

Use this feature to add different filtering effects. Some examples are below.



2. Crop:  This will eliminate unwanted portions of your image and enlarge the area selected. Click on any of the drag points with your mouse and drag until you reach the desired size. Note that the "Crop" icon has changed into "Apply Crop". Click "Apply Crop".


This will eliminate unwanted portions of your image and enlarge the area selected. Click on any of the drag points with your mouse and drag until you reach the desired size. Note that the "Crop" icon has changed into "Apply Crop". Click "Apply Crop".



Before Crop

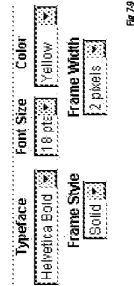
Area selected for Crop

After Crop

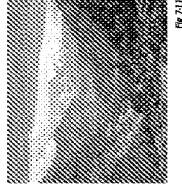
3. Text Message: 

Add a text message to your image by clicking this icon. A box will appear in the image; click in the box and type your message.

By selecting the dropdown boxes, you can change the font type, size and color. You can place the message anywhere on the image by dragging while selected. You can also add a frame around the text message by choosing from Frame Style and establish its width by choosing from Frame Width.




4. Rotate:  Rotate your image by increments of 90°.




Original


Rotated 90°

5. Delete: 

Used to delete text that was added to an image. Make sure the text area is selected first.

6. Reset: 

To return the image to its original state, click the "Reset" icon. Any and all changes you made will be eliminated.

7. Typeface, Font Size, Color: 


Change the font type, size and colors by using these dropdown menus.



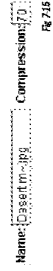
8. Frame: 

Choose a frame style and frame width for the text message by using these dropdown boxes.



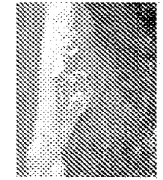
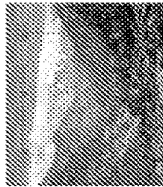
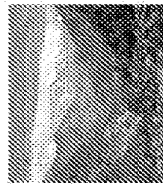
9. Name and Compression: 

Shows the file name of the selected image and how much it is compressed (similar to reducing resolution in other photo edit programs).



2. Crop: 


This will eliminate unwanted portions of your image and enlarge the area selected. Click on any of the drag points with your mouse and drag until you reach the desired size. Note that the "Crop" icon has changed into "Apply Crop". Click "Apply Crop".



Before Crop

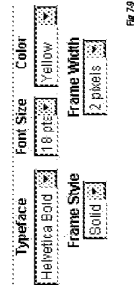
Area selected for Crop

After Crop

3. Text Message: 

Add a text message to your image by clicking this icon. A box will appear in the image; click in the box and type your message.

By selecting the dropdown boxes, you can change the font type, size and color. You can place the message anywhere on the image by dragging while selected. You can also add a frame around the text message by choosing from Frame Style and establish its width by choosing from Frame Width.



### 10. Bright, Sharpness, Color

These three sliders allow you to change the brightness, contrast (sharpness) and color of your image. You can also type a number in directly.



Fig 716

Brightness:

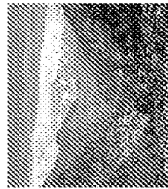


Fig 717



Fig 718

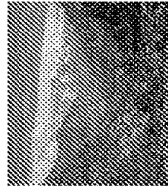


Fig 720

Original

80

30

Sharpness:

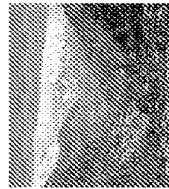


Fig 721

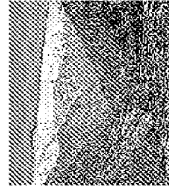


Fig 722

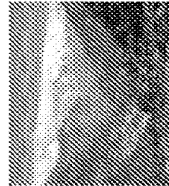


Fig 723

Original

80

30

Color:

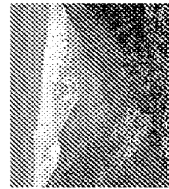


Fig 724

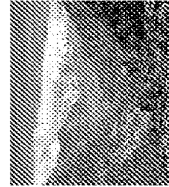


Fig 725

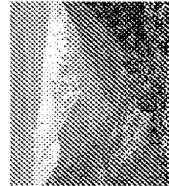


Fig 726

Original

80

30

### C: Slide Shows

Slide shows are very powerful objects, which can include photos, audio and video. When extended to a Chat Session (see "Working with Chat" in the "Working with Messages" Chapter) a slide show can become a real time multimedia community communication facility.

#### How to Define a Slide Show

For a simple slide show there are a number of attributes under the "Settings" tab that are unique.

- Max. Visible Slides:

The number of slides that would be visible at the same time. If the total number of slides is less than this number then only the actual number will be drawn.

- Vertical:

Whether the Slide show should be considered to be in a vertical or horizontal alignment. This will affect the following:

- How Multiple Visible Slides would be drawn.
- How the navigation keys would work.
- Where the Scroll Bar would be drawn (if any)

- Chat Room Status: (See Messaging)

- Initial Slide:

This will be the slide that will be first drawn when the page is initially entered. The default is the first slide.

- Add Persistent Variable (See Persistent Variables)

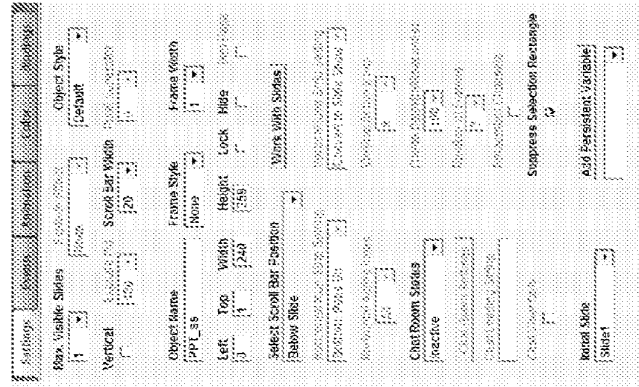


Fig 727

- Work with Slides

This choice opens up the Slide show Dialog Box as shown below.

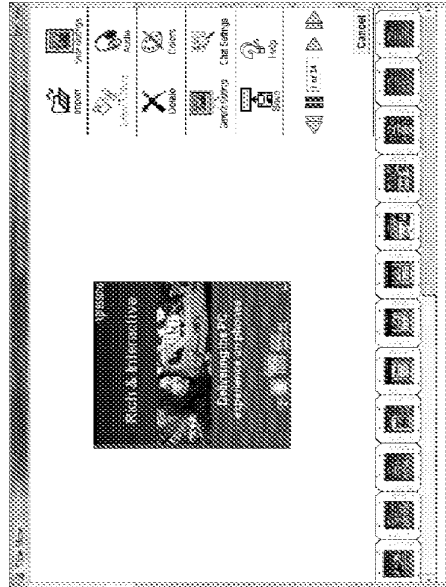


Fig. 7.28

1: Import:

By clicking this icon, you can import a video clip, single image, a folder of images or an existing image album.

NOTE: the maximum image size for a slide show to be used on a computer is 380 (h) by 440 (w) pixels. For phones and PDAs, the maximum image size will be defined by your screen dimensions

Import Video:

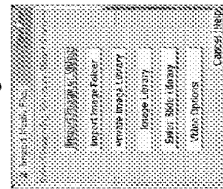


Fig. 7.29

Import Video Library

Express Mobile Video Library

Fig. 7.30

To add a personal clip, touch "Import Image or Video" button. This allows you to choose which clip to add from your files. Once the clip is selected, touch the "Slide Settings" button and under "Delay," select "Stay" or a time sufficient for the video to complete. To play, select the "Play" control.

Import Photographs:

- To import a photo from the Express Mobile library, touch the "Image Library" button.
- To import a photo from your private library, touch the "Private Image Library" button.
- To import a single photo touch "Import Image or Video" button.
- To import a folder of photos touch "Import Image Folder", double click on any of the photos in the folder or touch "Open". All of the photos in that folder will be imported.

Import Slide Show Library Objects:

In addition to the above, there is a library in which you can save or access finished slide shows. To get to the library, touch "Enter Slide Library". This will bring you to the library and you can then choose which show you want to import, either from your own private library or from the Express Mobile Library.

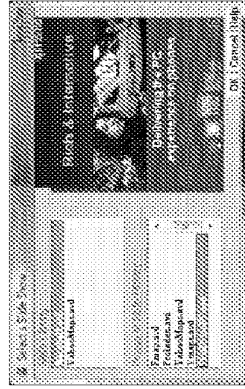


Fig. 7.31

2: Slide Settings:

This feature lets you define, animate and link your images.

- Animate

This feature allows you to change the timing and look of an individual photo. Note that you can have different attributes for some of the photos in your gallery, while others can be the same. This gives you some interesting options for how your photos will look when played.

- Slide Pause Time: this is the time each photo is on the screen.
  - Animated Transition: determines how the next photo will appear on the screen. "Zoom In" and "Zoom Out" create a 2 times zoom of 1 slide to the next with the zoom centered at the center of the slide. These options work well if you have a series of images (such as GIS images) that zoom into an area or object. By using the slideshow, it will appear that there is an almost seamless zoom in or out. "Expand Over" zooms in once to the center of the image. "Shrink Out" zooms out once from the center if the image. Random is a compilation of all the animations.
  - Transition Time: the amount of time it takes to transition from one photo to the next.
  - Frames/Second: how smoothly the photo transitions. This is a function of how many frames of the photo display in 1 second. The more frames per second means a smoother transition.
- Replace Media

To replace an existing video clip or photo touch the "Replace Media" button.

- c: Select an Exit Mode for this Slide:

By selecting this feature, you can link a video clip or slide to the next slide, an external web page or a specific slide. When the slide show is playing, a new web page will open when the show reaches a video clip or slide that's been linked. You can have the video clip or slide automatically go to the Web page, or have it be activated by a mouse click by selecting the "Mouse Enabled" checkbox.

If the video clip or slide is exiting to an external Web page or external page in a new window, you must enter the URL of that site. To go to a specific slide, select the number.

To go to a specific internal Page Number the Mouse Enabled function must be selected. With this choice the slide will time out after the Pause Time has elapsed and the next slide will be shown unless the user selects the slide while it is in its paused state. See Chapter 14 for more details.

- d: Bubble Help

Bubble help can be defined, so that when that slide is defined this string will appear, centered directly above the Selected Slide.

- e: Fire Enabled

If selected, visiting a slide through a normal navigation event (up, down, left or right) will execute the exit event of the newly visited slide.

- 3: Enhance Photo:

This takes you to the Express Mobile Photo tool where you can add special effects to a photo, such as adding

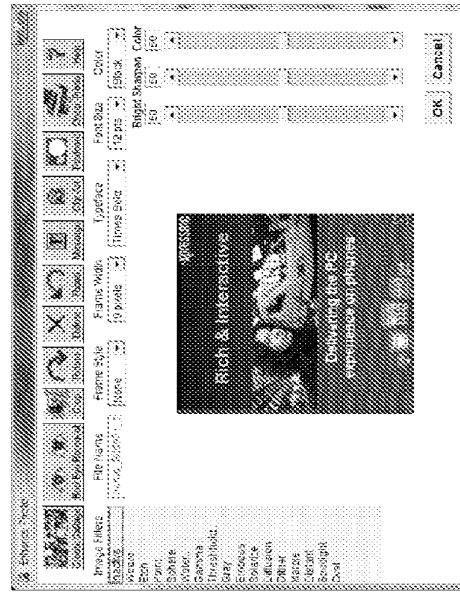


Fig 732

filters, text messages, and cropping. After you've made the changes, click "OK" and you'll be returned to the Slide show Dialog Box. For detailed information, see Pictures in the "Working With Media Objects" Chapter.

- 4: Audio:

You can add an audio track or audio stream to your photo and/or the transition between photos. You can also have a continuous background track with the ability to superimpose a separate track for an individual photo or transition.

- a. To add a track for the time that the photo is on the screen, either type in the file pathname of your own clip or select from the audio library and click "open". You can have the track play once, in a continuous loop or stream.
- b. To add a track to the transition between photos or the gallery background, follow this same procedure.

- 5: Delete:

This deletes the currently selected photo.

- 6: Colors:

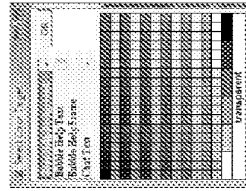


Fig 734

7: Generic Settings:

Choose how you want your photos to play and transition. These choices determine how the entire gallery or album will look and play. To change an individual photo only, select the "Slide Settings" button.

- Fire Enabled:
- If selected it makes all Slides mouse sensitive, and when touched, will execute the defined exit event associated with that slide, if any.
- Slide Show Transition:
  - Determines how each photo will enter the screen Random is a compilation of all the transitions.
- Delay:
- How much time the photo is on the screen.
- Transition Time:
- How long the transition takes from one photo to the next.
- FPS (Frames per Second):
- How many frames of a photo play per second. The higher the number, the smoother the transition will look.
- Scroll Bar Position:

The scroll bar can be hidden or placed in 6 different positions relative to the Slide Show. If the Slide Show has been set as Vertical, then the Scroll Bar will draw vertically and its placement will be relative to its horizontal position.

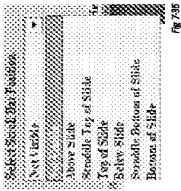


Fig 7.36

8: Numbering:

This shows you the currently selected image in numerical order.



9: Controls:

Similar to controls on a VCR.

From Left to Right:

- a. Reverse: go back one frame at a time
- b. Pause: stops the slide show at the current image
- c. Play: begins or resumes the slide show
- d. Forward: advances one frame at a time

10: Image Index: To go to a specific image, click on it in the index.

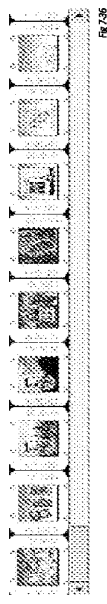


Fig 7.36

You can change the order of the images by clicking and dragging. In the example below, the first image is being moved to the second place. The current #2 image will become #1. You can move any image to any location.

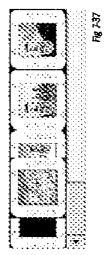


Fig 7.37

Moving #1 to #2

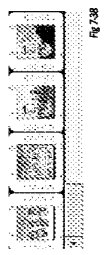


Fig 7.38

Result





An attempt to connect to the resource pointed to by this URL is made. If connection cannot be established a Warning popup will appear, displaying the error condition reported by QuickTime. There may be valid reasons why a resource is not available at authoring time (such as if the resource has not been created yet) so this is popup is only cautionary. Once the popup is dismissed, the UI will return to the main layout of Publisher.

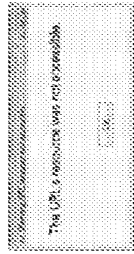


Fig. 7.43

Once validated, the Publisher will be set to store the value from this field in `ObjectString(1)` in the Video Object's parameters within the PDL file, when the application is saved. Similarly, the Video Object's `ObjectInteger(Attribute)2` is set to (-1). This indicates that this video object's source is from a URL rather than from a file that originated on the Publisher client file system or from one of the Publisher server "libraries".

The UI will return to the main layout of Publisher, showing a placeholder for the video on the layout page. The size of the video will be initially determined by the size of the encoded video stream. The author may edit the size of this video object using the Resource Inspector or by manipulating the object directly on the page layout.

The video URL field is not editable in this tab. Any change must be made through a dialog box that will be displayed when button labeled "Change Video" is selected. This will display the same dialog as "Select Video by URL", with the current URL value in the text field, if a Select Video by URL was previously used. Otherwise the Create Video Object dialog will appear.

The Video, as with all videos, will be viewable in "Preview".

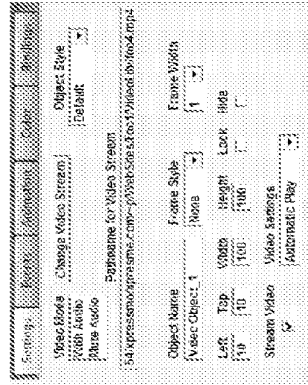


Fig. 7.44

#### Video in Slide Shows

A slide show could always have multiple video slides, but they never worked very well in terms of:

- Streaming
- Slide Transitions
- Player Support

New Slide Show based video is at the same level of support as Video Objects. In the MIDP Player there remains a restriction that only one video can play at a time, but in the generic player multiple video slides could be visible at the same time.

## E: LBS MAP

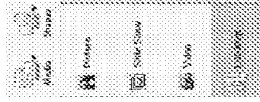


Fig. 7.45

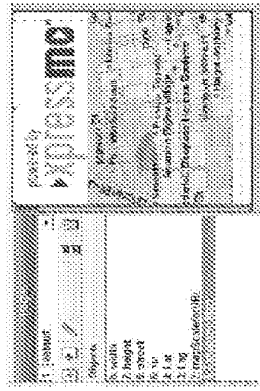


Fig. 7.46

### 1: UI

Selecting the LMS Map Icon from the Tool Bar will generate a fully functional LBS Map which is another example of a Web Component. This LBS Map object will inherently support panning through the Left, Top, Right and Down navigation keys. Zooming will have a zoom-in default but that can be changed through auto-generated Sort Key Commands or automatically when the map has reached its maximum zoom-in or zoom-out resolution.

A scroll bar with an elevator is drawn on the left to indicate the current zoom level.

In addition, hidden text fields are generated and initialized to default values, which can be changed by the author. They include:

- width: Set to the width of the view port plus 160 pixels, which, by default, is the width of the current page plus 160 pixels less the width of the scroll bar.
- height: Set to the height of the view port plus 160 pixels, which, by default is the height of the current page plus 160 pixels.
- street: An optional input that, by default, is set to the street address for Express Mobile.
- zip: A required input field that, by default, is set to the zip code for Express mobile.
- Lat: This is the geospatial latitude for the current location. It will either be set by the phone if it has a supported GPS receiver, or by the Web Component when the address is transmitted.
- Lng: This is the geospatial longitude for the current location. It will either be set by the phone if it has a supported GPS receiver, or by the Web Component when the address is transmitted.
- mapScaleForURL: This is the default zoom level, which, by default, is at a resolution of 48000:1.

All of these hidden text fields can be edited by the author, and all have been assigned persistent variables so that other pages in the application can either define or receive these values.

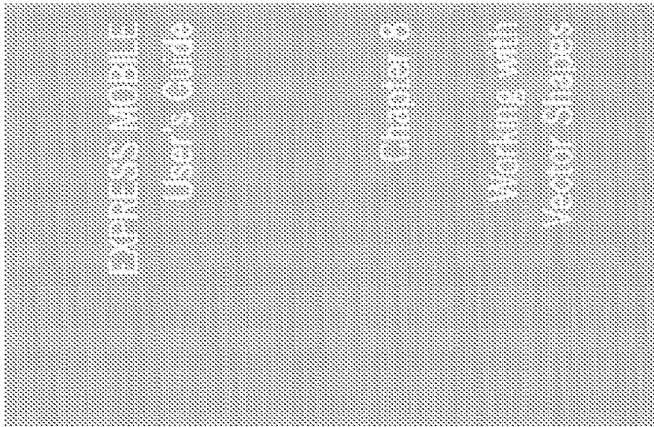
Chapter 11 will go into greater detail in how Web components can be used for a wide range of LBS applications.

## 2: How it Works:

The all Map Images that are initially loaded are centered relative to the view port, one image map for each zoom level. If a pan operation is selected, then the existing image map will pan until it reaches the boundary of the view port. Simultaneously, the Web Component Manager requests for a new set of map images, whose geospatial center will conform to the geospatial point after the 80 pixel offset is reached.

An image map is brought down to the device for each zoom level, utilizing Xpressmo's multithreaded Network Communication Manager, with a priority given to the image map for the current zoom level.

This means that, in general, there is no perceived delay by the user when panning or zooming.



## Chapter 8: Working with Vector Shapes

- A: How to Create 08.04
- B: How to Create a Group 08.06
- C: How to Edit 08.07

## Chapter 8: Glossary

Group

The act of selecting two or more UI objects and then logically connecting them together so that a single operation can be applied to all these objects at the same time.

Arc

One of several vector graphic primitives that can be created with the Publisher. Arcs are circle segments that can be used for creating curved surfaces.

Fill

The color that will be drawn inside a vector object. If set to transparent then no color will be drawn.

Gradient

A technique for creating the sense of depth for a two dimensional vector object. This is done by using two different colors in various ways to create effects such as sunlight reflecting off a 3 dimensional object.

## A: How to Create:

There are three convenient ways to create vector shapes.

- Touch the "Shapes" icon from the Toolbar. The menu below on the left will appear.
- Select "Shapes..." from the Object Menu from the Menubar. The menu below on the right will appear
- Right Click on the Canvas with no object selected and select "Shapes...". The menu below on the right will appear

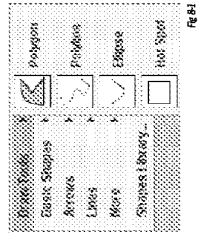


Fig 8.1

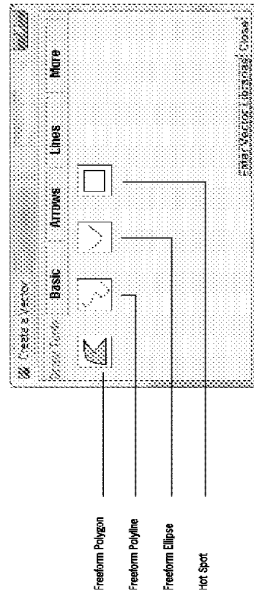


Fig 8.2

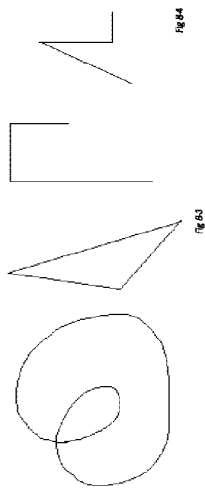
In all three cases you can also access the vector shapes libraries, which contains a number of shapes already created, either by the designer in the private library or by others in the public library. Click on your choice and the shape will appear on the Web page with a red selection frame surrounding it, with the exception of Free Polygon, Polyline or Ellipse drawing tools. You can now edit the object.

To create a freeform polygon or polyline, click on the first icon as above and a crosshair will display. To begin, click where you want the first part of the shape to begin. You can then drag the mouse to design the shape. Alternatively you can click the mouse wherever you want the next point to be defined. At any time, touching and releasing the shift key will cause the next line segment to be horizontal or vertical. Hold and move your mouse to define the shape desired.

To draw a straight line other than horizontal or vertical, move the mouse (without clicking) to the point where you want the line to end and then double click the left or right click your mouse.

To end the drawing, either double click (left) or right click your mouse. The shape is now ready for editing.

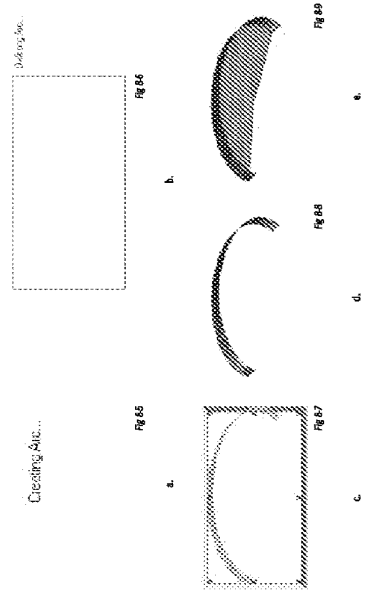
A freeform polyline is similar to a freeform polygon, except the lines don't need to form a closed-in object. Follow the instructions as for polygons to create



Freeform Polygons

Freeform Polyline

To create an ellipse, click on Ellipse drawing tool. A message will display prompting you to create the arc (a, below). Click where you want the center point of the arc to be and click again where you want the other side of the arc to be. A message prompting you to define the arc will display (b, below). Draw where you want the arc to be (c, below). You can use the drag points on the frame to make the arc into the shape desired. (d) and (e) below show edited versions of the arc.



The starting point of the angle and/or the angle sweep can be changed by editing (see Section C).

### B: How to Create a Group:

If you have a group of shapes that you want to stay together for placement purposes, you can choose to group them. To do so, hold down the "Shift" key and then with the mouse, choose the objects to be grouped (a red selection frame will be around each object).

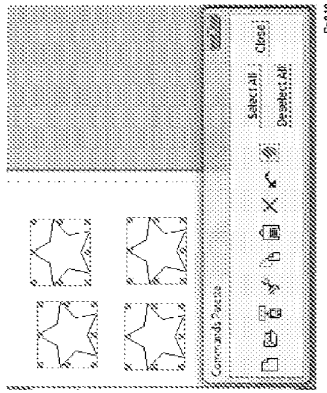


Fig 8.10

To group, touch the "Group" icon in the Commands or right click and select "Group".

The group will have a boundary box around it and will be listed as a group in the Object Status Box as in the example below. Groups are designated with the letter "G".

If you wish to group all objects on the page, you can go to "Edit" dropdown list and click on "Select All" or right click the page and click "Select All" at the bottom of the list.

To ungroup, click the "Ungroup" icon or select "Ungroup" from the Edit menu or do a double left mouse click and select "Ungroup".

The group can be sized by dragging any of the red drag points.

Polygon groups can be saved in the Polygon library for future use. Make sure the group is selected and click "Save As", choose "Selected Shape Group" and name it.

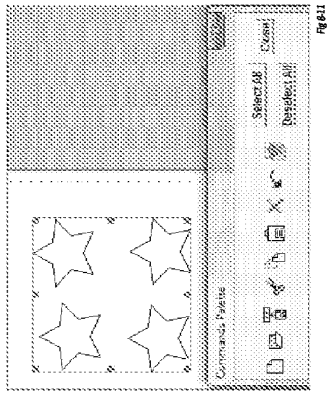


Fig 8.11

### C: How to Edit:

To access the Edit mode, make sure the shape is selected and either:

- Select the Edit icon on the Layer Inspector.
- Select "Edit" on the Edit menu from the menu bar.
- Right click and select "Edit" ...
- Work with the Resource inspector.

#### 1: Define:

##### Editing Shapes Other Than Arcs:

Although all editing operations are available with the Resource Inspector the discussion below assumes that editing is occurring through the Edit dialog box.

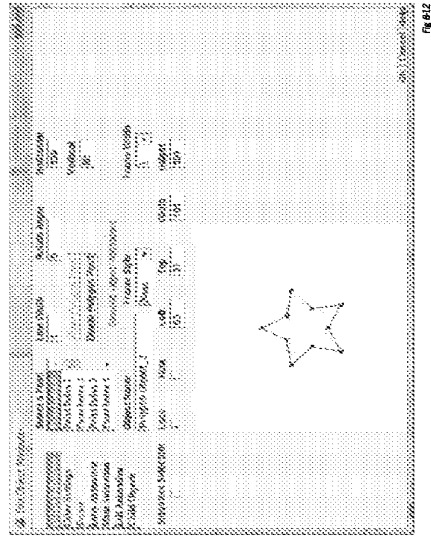


Fig 8.12

Select a Point: each shape is made up of a number of points, which are represented by the blue squares. To choose a particular point, you can click in the dropdown list and select. The point selected turns into a red square and then change the value in its horizontal or vertical position. The "Move Polygon Point" button will become active.

The Example below shows the result of changing the Vertical Position to 40 and Selecting the "Move Polygon Point" button.

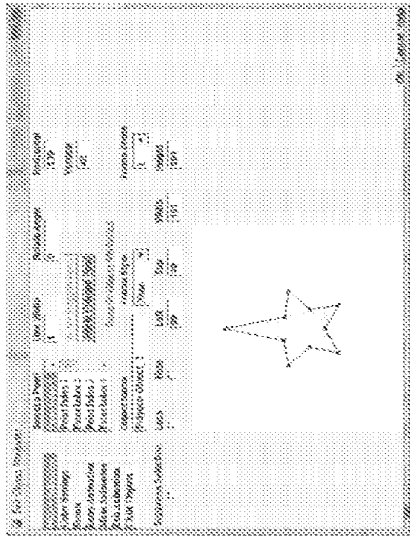


Fig 8.13

- **Line Width:** changes the width of the shape lines. The following example has a line width of 10.

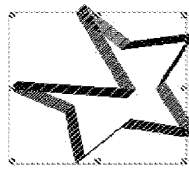


Fig 8.14

- **Rotate Angle:** rotate a shape by a specific selected angle. Type in the angle degree desired.

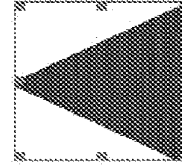


Fig 8.15

Triangle

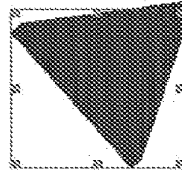


Fig 8.16

Triangle rotated 30°

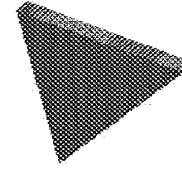


Fig 8.17

Rotated 50°

Note that you can also do a free form rotate by selecting the shape and doing a double left mouse click or a single right click. A message saying "Rotate Mode" appears; begin to move the object with the mouse to the desired rotate angle.

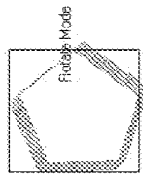
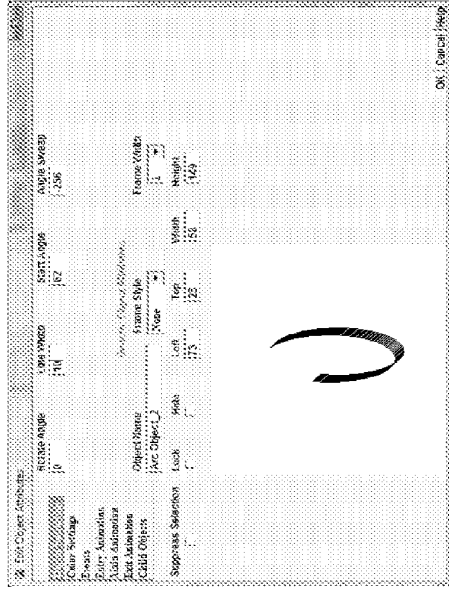


Fig 8.28

Free Form Rotate

- **Object Name:** the default name is "Polygon Object". Change the name to something more descriptive by typing in a new name.
- **Frame Style:** this adds a frame around the shape. The choices are raised, pressed or solid.
- **Frame Width:** determines how wide the frame around the shape is.
- **Lock:** prevents the shape from being dragged or resized by using the mouse. Other editing functions are still available.
- **Left/Top:** this changes the placement of the shape on the web page.
- **Width/Height:** changes the size of the shape.

**Editing Ellipses:**



- **Line Width:** changes the width of the shape lines. The following examples show an initial line width of 1 and then of 20.

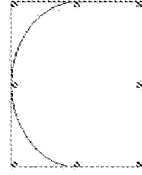


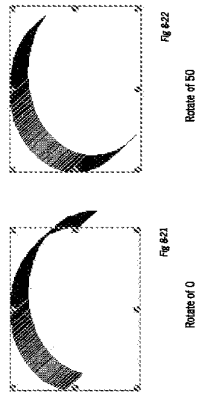
Fig 8.19



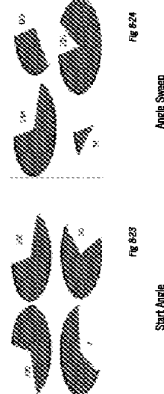
Fig 8.20

- **Rotate Angle:** with an arc you can also rotate the shape by a selected angle. In addition, the start angle and angle sweep can be adjusted.
- You can perform a free form rotate by right clicking your mouse and selecting "Rotate" from the menu. A message saying "Rotate Mode" appears and when you click, drag the mouse to define the desired effect.





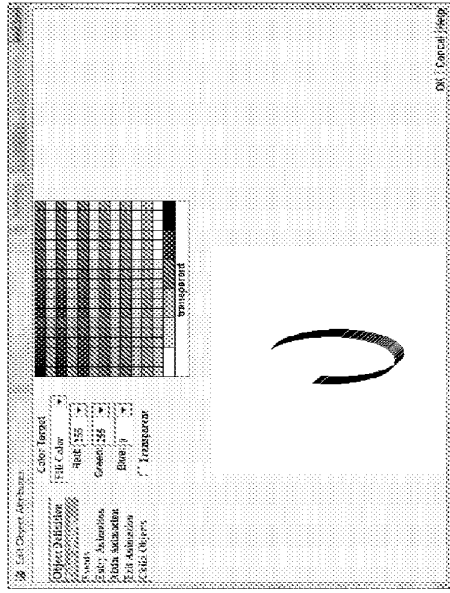
Below are different examples of how the start angle and angle sweep look at various settings for ellipses.



- **Object Name:** the default name is "Arc Object". Change the name to something more descriptive by typing in a new name.
- **Frame Style:** this adds a frame around the arc. The choices are raised, pressed or solid.
- **Frame Width:** determines how wide the frame around the arc is.
- **Lock:** prevents the arc from being dragged or resized by using the mouse. Other editing functions are still available.
- **Left/Top:** this changes the placement of the arc on the web page.
- **Width/Height:** changes the size of the arc.

2. Color and Fill:

- a. For Arcs (not polygons):



Color Target allows you to set the line, fill and frame color. You can use either the color palette or create a custom color by using the red, green and blue dropdown boxes. The feedback box gives you a preview of what your choices will look like.

- b. For Polygons:

You have the choice of a Single Color fill, a Linear Gradient fill or a Radial fill.

- i. Single Color Fill: See above for ellipses.
- ii. Linear Gradient Fill:

This divides the shape object into 4 quadrants. You have a choice of 2 fill colors. The percentage of each color can be adjusted by changing the gradient mid point. The origin of the color changes, depending on the gradient angle.

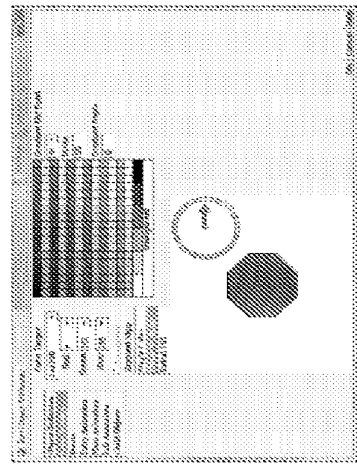


Fig. 8-25 Gradient midpoint of 50 and 0° Gradient Angle

Figure 8-25 shows the editing of the colors for a polygon using the Edit dialog Box. Working this way, the gradient angle arrow can be dragged or adjusted by changing the value in the Gradient Angle text field.

In Figure 8-25 above, the 1st color fill is red and the 2nd color is blue. The mid point is set at 50, which means there is an even amount of the 2 colors and the gradient angle is at 0.

Gradient midpoints change the percentage of the 1st fill color. Utilizing the Resource Inspector, with the Color Tab selected, the same functions can be performed. As seen below, at 20, the percentage of the red (1st fill color) is decreased.

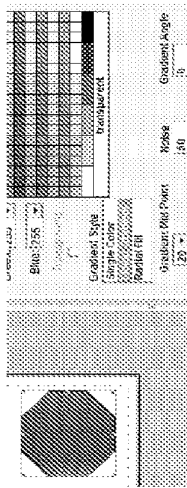


Fig. 8-26 Gradient midpoint of 20 and 0° Gradient Angle

The result at 70% is shown below.

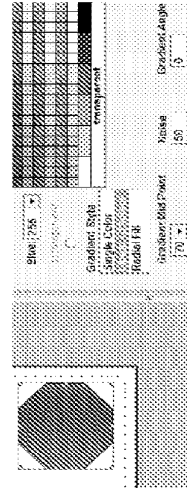


Fig. 8-27 Gradient midpoint of 70 and 0° Gradient Angle

Below are the results when the gradient angle is changed to 80, 180 and 270 degrees.

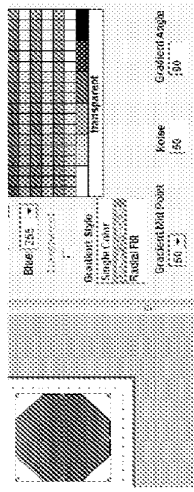


Fig. 8-28 Gradient midpoint of 50 and 80° Gradient Angle

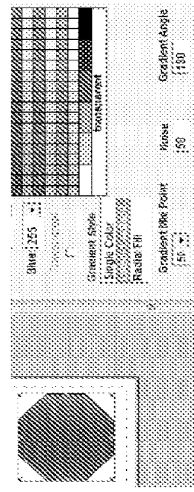


Fig. 8-29 Gradient midpoint of 50 and 180° Gradient Angle

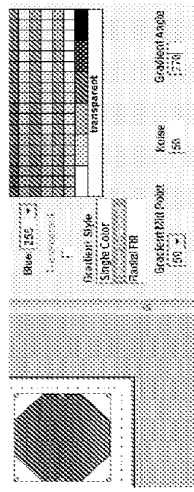


Fig. 8-30 Gradient midpoint of 50 and 270° Gradient Angle

iii. Radial Fill:

Radial Fill has a center point from where the color radiates out. Using the same colors as for the Linear Gradient, Figure 8-30 below shows the fill radiating from the center of the object and a gradient midpoint of 50. The percentage of the 1st fill color increases or decreases when this number is changed

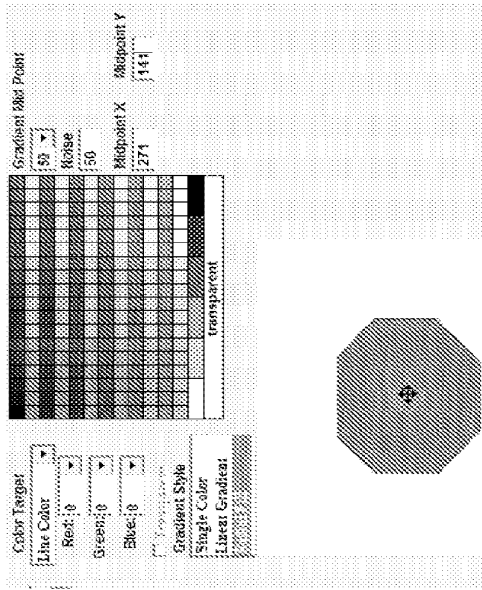
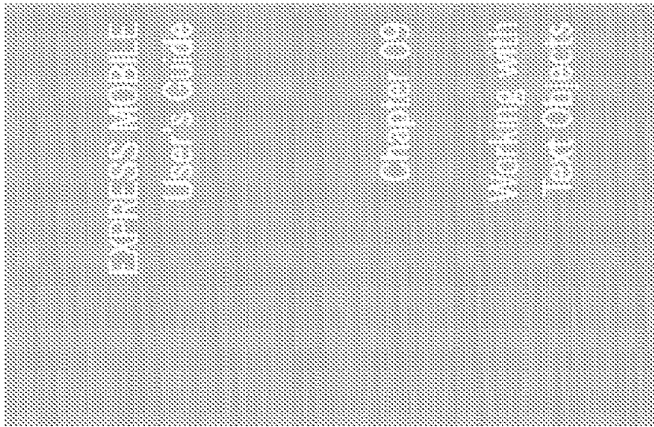


Fig 8-30 Gradient Midpoint of 50

The X & Y midpoints can also be changed, so that the center point changes location. The X coordinate can be from 173 to 369 and the Y from 52 to 231.

## Chapter 9: Working with Text Objects

- A: How to Create 09.04
- B: How to Edit 09.05

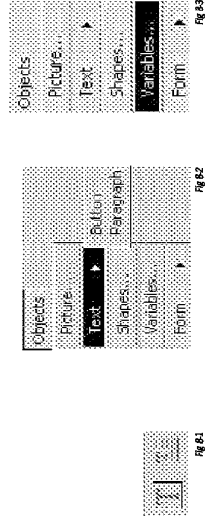


### Chapter 9: Glossary

Lines per (Sec/Touch)

This is the vertical scrolling speed for a multiline text object such as Text Areas and Paragraphs.

### A: How to Create



- a.
- b.
- c.

#### 1: Button:

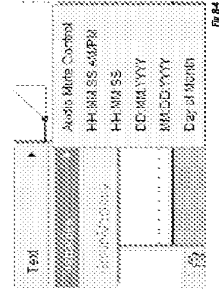
Click on either the "Button" icon (a) or "Text" and then "Button" under the "Objects" menu (b) or right click and choose "Button". The button will be placed onto the Web page with the words "Button Text" and a red selection rectangle around the button; type in the desired text. Editing is now enabled, as are many commands and icons on the status lines. To move the button to another location, click on the button text (but don't release the mouse button) and drag to the desired location. Note you can move the button before entering text if you wish.

#### 2: Paragraph:

Click on the "Paragraph" icon (a) or "Text" and then "Paragraph" under the "Objects" menu (b) or right click and choose "Paragraph". The paragraph will be placed onto the Web page with the words "Paragraph Text" and a red selection rectangle around the paragraph; type in the desired text. Editing is now enabled, as are many commands and icons on the status lines. The procedure for moving the paragraph is the same as for text above.

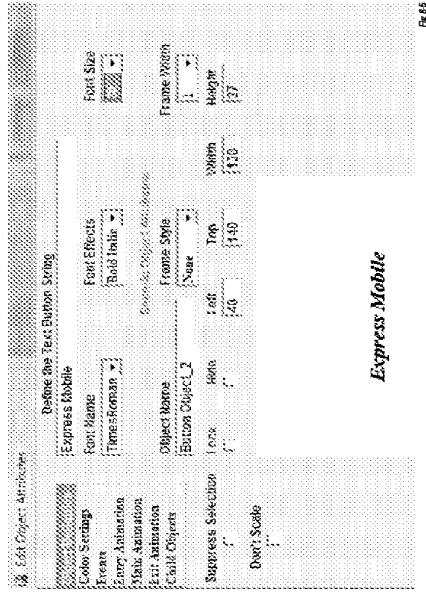
#### 3: Variables:

Click on "Variables" under the "Objects" menu (c) above. Your choices are listed in the graphic to the right. Audio Mute Control will mute any audio track that may be present. The other choices will be placed on the screen and you can then edit and drag as for buttons and paragraphs.



### B: How to Edit:

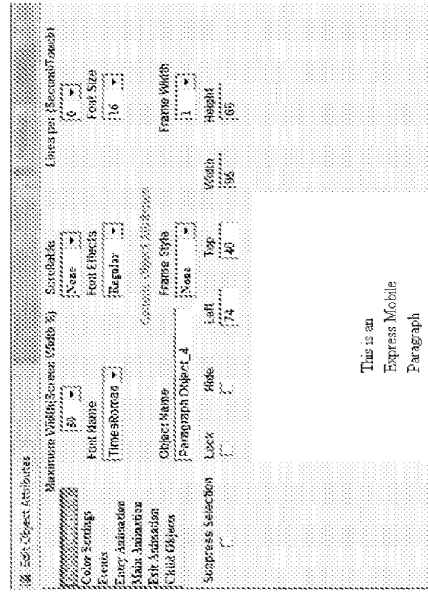
The Edit function works the same for buttons and paragraphs with the exception of the "Define" tab. Editing a variable is identical to editing a button, except the text cannot be changed.



Define for Buttons

Define for Buttons:

"Edit the Text Button" allows you to edit the text that was entered when the button was first created.



Define for Paragraphs

Define for Paragraphs:

"Maximum Paragraph Width as a % of Screen Width" lets you decide how wide a paragraph should be in relation to the screen width.

"Scroll" and "Lines per (Sec/Touch)" apply to paragraphs that have been bound to a real-time text feed such as RSS (See appendix D).

The remainder of the "Define" tab is identical for all. You can change the font, font effects (regular, bold, italic or bold italic), and font size (from 4 points to 5000 points). The object name can be replaced by a name that is more descriptive, such as "Button1". The frame style choices are: none, raised, pressed and solid.

If you have chosen a frame to outline your text object, you can select the frame width from 1 to 200. The frames above are set at 6.

"Lock" allows you to lock the object in to prevent it from being dragged or resized by using the mouse. All other editing functions are available.

The "Left" and "Top" selections are for the position of the button, paragraph or variable in number of pixels from the top or left of the screen.

2: Color:

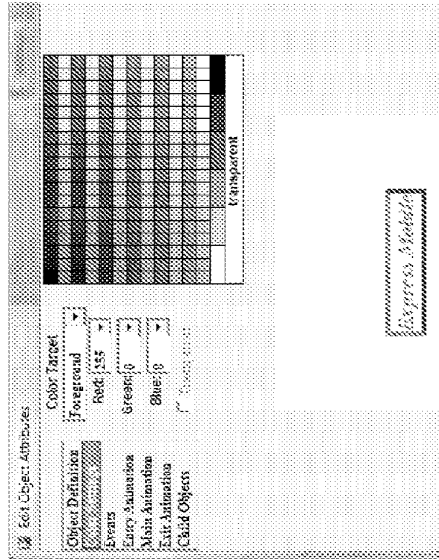
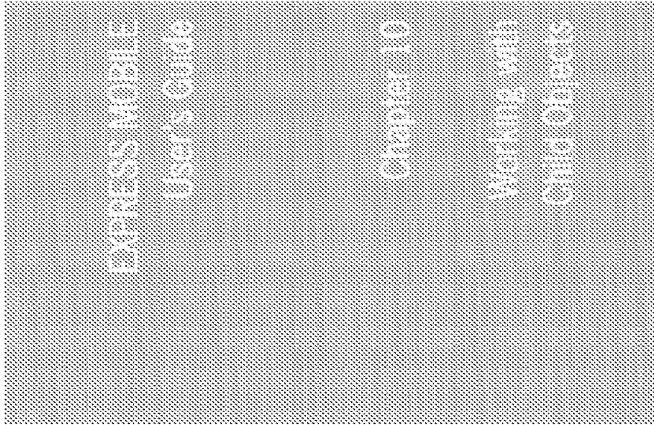


Fig 66

Color Target allows you to set the color of the text, button or paragraph background and frame. You can use either the color palette or create a custom color by using the red, green and blue dropdown boxes. There is a feedback box; in this example the text color (Foreground color) is red, text background (background color) is yellow and frame (frame color) is blue.

## Chapter 10: Working with Child Objects

A: How to Create	10.04
B: How to Select	10.06
C: Behaviors	10.07





## Chapter 10: Glossary

Behaviors	Describes the intelligence of an Object as it responds to User interaction, time, or location.
Child Objects	Used for creating dynamic visual effects or progressive disclosure of information

A child object may be:

- Vector
- Video
- Image
- Slide show
- Paragraph
- Text button

There are no limits to the number of child objects that can be attached to any particular parent object. Similarly, there is no limit to the number of parent objects for any given Web page. Therefore, this can be an extremely powerful feature.

There are 3 general functions:

1. Selection: when a parent object is selected, all child objects appear.
2. Fire: the child object is not visible until the parent object is fired on. To enable this feature, see Section D later in this chapter.
3. Timelines: a) a regular timeline that is activated at the time the parent object is activated and b) a timeline event attached to a child object. See Chapters 13 and 14 for a detailed description of timelines.

### A: How to Create:

To create a child object, make sure the parent object is selected.

Then either:

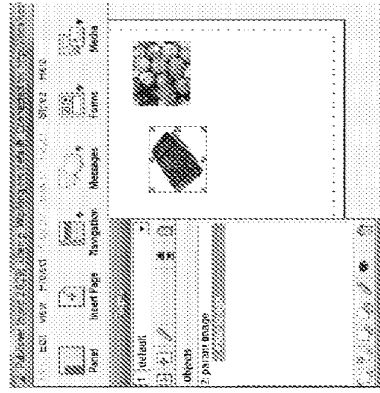
- go to the Menu Bar, click on "Events" and then on "Child Objects".
- Go to the Events Tab of the Resource Inspector and touch the "Work with Child Objects and Mouse Overs" button.
- right-click on the selected parent object and select "Child Object".

### B: How to Select:

When Child Objects are created they will appear in the Layer Inspector as seen above. Selecting a Child Object from the Layer Inspector will:

1. Cause an implicit switch into Child Mode.
2. That Child Object will be selected as shown above in green while its associated parent object will be selected in red.
3. All the relevant controls in the Resource Inspector that were available to a Parent Object will be available to a Child Object.
4. The Layer commands are available.
5. The Delete command is available.
6. Undo is available.
7. The child object name can be edited in the resource inspector.
8. All other child objects associated with the selected parent can be selected at this time for editing.
9. To implicitly leave child mode:
  - a. Touch any parent object in the Layer Inspector.
  - b. Touch any Parent Object in the Workspace.
  - c. Deselect all objects by either touching a free space in the workspace or by selecting "deselect all" from "Edit" off the Menu Bar.

Fig 105



### A "Child Object Mode" floating palette will appear.

The author can now, without limit, place child objects on the page from any of the choices as shown on the right. As each child object is placed on the page, it becomes selected as shown below. The author can change from one selected child object to another through mouse selection. Each child object can then be further defined as described below.

Selecting the "Close" button will exit the Child Object mode

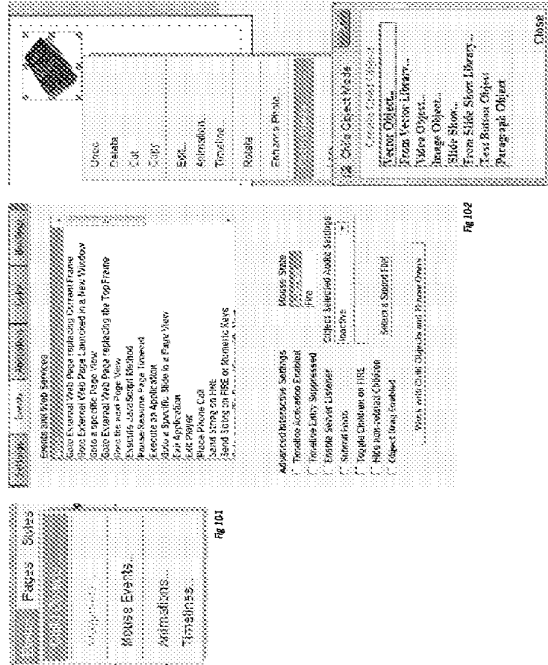


Fig 102

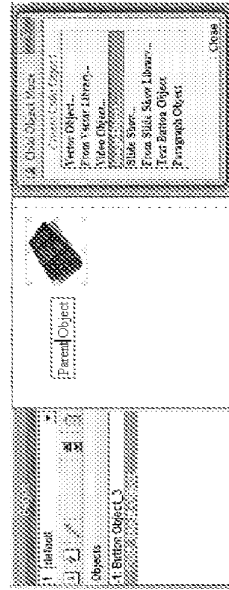


Fig 104

### Fig 101



Fig 101

### C: Behaviors

a: Visibility

There are three possible techniques for causing child objects to change their visibility.

- Selection of the Parent Object

Upon selection all child objects will appear (subject to their timeline definitions, if any) and will disappear when the parent object is deselected.

- Toggle Children on FIRE

When this attribute is selected for the parent object then selection or deselection itself will have no effect. The first FIRE on the parent object, the child objects will become visible and remain that way until another FIRE event occurs on that parent object, or when another Parent Object is FIRED on that has the "Hide non-related Children" attribute selected.

- b: Child Slide Shows, Timelines and Animations

Child objects can be assigned all the timeline and animation attributes that a parent object can be assigned, and a Child Slide Show can be assigned all the slide transition animations that a Parent Slide Show can be assigned.

The activation of these visual and audio effects begin when the child object becomes visible and ends whenever the child object loses visibility or the timeline is completed, whichever occurs first.

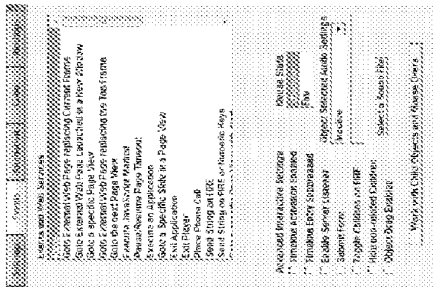
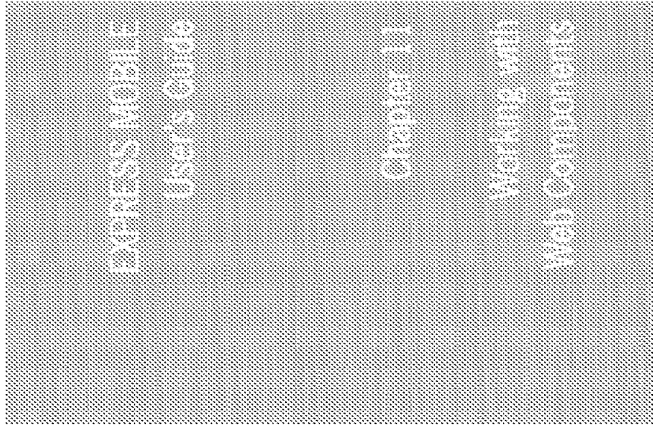


Fig. 10.6

Fig. 10.6  
The activation of these visual and audio effects begin when the child object becomes visible and ends whenever the child object loses visibility or the timeline is completed, whichever occurs first.



Chapter 11: Working with Web Components

A:	Supported Objects	11.05
B:	How it Works	11.07
C:	Behaviors and UI	11.10
D:	Adding a Web Component	11.11
E:	Completing a Web Component with the "Generate UI" Option	11.13
F:	Completing a Web Component with the Existing UI Objects	11.19
G:	Persistent Variables	11.23
H:	Enhancing Web Components with Auto-Generated Persistent Variables	11.25
I:	Restrictions for Editing UI Objects related to Web Components	11.29
J:	Working with Control Lists	11.31
K:	Working with Collection Lists	11.38
L:	Collection List Keys and Complex List Indexes	11.40
M:	Sharing a Web Component	11.42
N:	How to create a Web Component Model	11.47

## Chapter 11: Glossary

### Web Component Widgets

Unlike the widgets a discussed in Chapter 2, Web Component Widgets are a series of one or more related pages that employ Web Components. Often a Web component Widget will execute a Web component immediately upon being activated.

### Data Types

Data Types represent a specification for each input (Input Data Type) to a Web Service or output (Output Data Type) from a Web Service. A data type will describe the format of the data that the Web Service provides and how that data may be organized. Data Types become helpful hints so that:

- The Publisher can make an intelligent decision on what type of UI Object to generate when auto-generating.
- Present the author with a concise list of possible and valid UI Object candidates with creating a Web Component manually or when Editing a Web Component and changing the binding from one UI Object to another.

### Symbolic Field Name:

This is the unique symbolic pointer to a specific input or output that is provided by a Web Service.

### Binding:

Binding is the mechanism which ties a Symbolic Field Name as known to the Web Service to a UI Object as known to the Publisher through its Object Name.

### Control List:

A Control List is an Output Data Type that has the following characteristics: It has a related Symbolic Field Name that points to a Web Service Output. Its Symbolic Field Name is prepended with "Control List" for easy identification and is drawn in red.

It aggregates a set of Output Symbolic Fields that are logically related to the Control List's Output Symbolic Field. In data structure terms the Control List's Symbolic Field is an index to a data record that contains logically related data fields, each of which has an Output Symbolic Field reference.

Some examples of Control List implementations are:

1. Search Response List Control List

The short Title for each of the returned matches could be the Control List Symbolic Field.

Related Symbolic Fields:

- a. A detailed description
- b. A related image banner Ad

xpressmo.com

- c. The URL to the web site would represent related.

### 2. RSS Display List

The Title for each RSS Feed could be the Control List Symbolic Field.

Related Symbolic Fields:

- a. The Summary
- b. A related icon Image
- c. The URL to the full story web page..

### 3. Store Front Product List

The name of the product could be the Control List Symbolic Field.

Related Symbolic Fields:

- a. Product Description
- b. Product Preview (Image, Video, specification,, Audio, etc.)
- c. The URL to the WAP Store Front

### Child Output Objects:

The UI Objects that have been bound to a set of Output Symbolic Fields that are logically related to the Control List's Output Symbolic Field.

### Collection Lists:

Collection Lists are Lists that contain a set of hidden attributes that point to a set of Control List definitions. In data structure terms the Collection List's set of hidden attributes are keys to a table of Control List Definitions.

## A: Supported Objects

The following objects will be supported in this first implementation:

- 1: Input UI Objects:
  - Text Fields
    - Alpha
    - Numeric
    - Phone Number
    - SMS Number
  - Text Areas
  - Choice Objects
    - Returning the Selected Visible String
    - Returning a Numeric Hidden Attribute.
  - Single Item Selection Lists
    - Returning the Selected Visible String
    - Returning a Numeric Hidden Attribute
  - Multi Item Selection Lists
    - Returning all selected items
      - Visible Text String
      - Hidden Attribute
    - Cluster Item Selection Lists
      - Returning the hidden attributes for all items
  - Check Boxes
  - Slide Show
    - Returning a Numeric Hidden Attribute
    - Returning a String Hidden Attribute
    - Returning the hidden attributes for all slides
  - Submit Function (Can be assigned to any object including Submit Buttons, Vectors, etc.)

## 2: Response UI Objects:

1. Single Line Text Object
  - a. Text Field
    - i. URL
    - ii. Audio URL
    - iii. Purchase URL
    - iv. Other
  - b. Text Button
  - c. Submit Button
  - d. Clear Button
2. Multiple Line Text Object
  - a. Text Area
  - b. Paragraph
3. Check Box
4. Image
5. Video
6. Slide Show (with either Video or Image slides, or both)
7. Choice Objects
8. List Objects
9. Control Lists (they control all the subordinate Output UI objects for that Web Component.
  - a. List Type:
    - i. Search Response List
    - ii. RSS Display List
    - iii. Other
  - b. Choice Type:
    - i. Search Response List
    - ii. RSS Display List
    - iii. Other

## B: How it Works

Both Xpresso customers and third parties can register their web components directly into the Xpresso Authoring Platform.

### 1: Xpresso XML Web Component Registry

Customers and third parties can use XML to register their Web Components into the Xpresso XML Web Component Registry. The Registry may contain references to:

- All Consumer Inputs related to the Web Component.
- Any Environmental data such as PIM, time or location values.
- Any persistent variable data.
- All outputs related to the Web Component.

Appendix A shows the ten current Web Components currently registered into the Xpresso Web Component Registry.

- RSS: USA Today Top Stories
- Yahoo Search
- Google Search
- MapQuest
- Yahoo Maps
- Yahoo Weather
- Accuweather
- Movies
- ThumbPlay Store Front
- Send SMS Message
- Stock Quotes
- Face Book

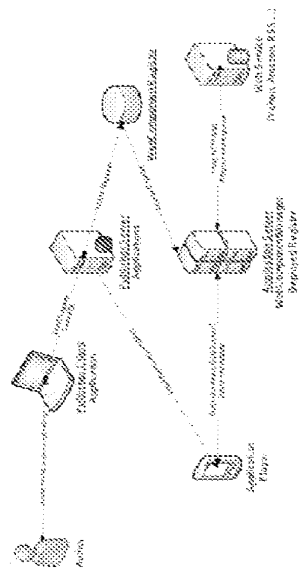


Fig 11.2

### 2: Binding to the Author's Application

The XML Web Component Registry for all registered Web Components is loaded into the Publisher Client tool at login time. The author can then assign any of these Web Components to an application without any need to write code. Each Input and/or Output related to the Web Component may be hinted by the Web Component Registry Administrator. The more hinting the more information is passed to the Application Designer in terms of presenting a better focused set of possible UI Object candidates for binding. As a Web Component is selected the authoring platform presents the author with dialog boxes that enable the binding of all the inputs and outputs of the Web Component to GUI components of the application. Multiple Web Components can be assigned to any UI Object or Event in order to facilitate mashups. These UI Object and/or Event bindings, for each instance of a Web Component, are stored in the PDL. The war file containing a WebComponentManager to handle data from player to web component automatically is deployed as a web application archive to any application server.

Again, looking at Appendix A, some of the key Fields are:

- Parameter Name:  
This is the Symbolic Field Name that is displayed to the Author.
- DataType:  
This is where the Web Component Registry Administrator can add hinting to help focus which types of UI Objects can be selected by the designer.
- ParmType:  
If defined, it is set to "Request" in order to inform the Web Component Manager how to process this information.
- Use:  
This can be set either to "Required" or "Optional". If set to Required, then the Author will be required to assign a UI object to this Field, and the Consumer will be required to have data entered if the Required UI Object is an Input. In either case, warnings will be issued if the Required Fields are not handled correctly.

• ServiceMapping:

This is the Symbolic Field Name that is sent to the Web Component Manager for communicating with the Web Component. It uses XPath to map the output result to the result XML Element or Attribute.

• Comment:

This is information that the author will see displayed in the Comments field for each Symbolic Field, to assist the author in understanding the meaning of the Web Component.

- In the event that the Web Component returns lists of results then there is an additional Field named ControlParam:
- This is the Symbolic Field Name that will point to an Output UI Control List object. When used, the Control List manages the visible presentation for all other Output UI Objects associated with a particular Implementation for a Web Component in a given application.

3: Execution of Web Components

The Xpressmo Player, upon detecting an event in which a Web Component has been defined, assembles and sends all related inputs to the WebComponentManager URL which proxies the request to the 3rd party, customer, or Xpressmo web service and returns outputs to the Xpressmo Player. The Xpressmo Player then takes the outputs and binds the data to the UI components in the application.

For example, if the author wanted to use an ATOM feed in their application, they would look at the UI Components available in the Publisher Client, pick the feed they want to use, and bind the output of the feed summary to a textbox. The bindings would be saved into the PDL and processed by the player at runtime. If the ATOM feed does not exist as a new one can be added to the Registry that contains all the configuration data required, such as the actual feed URL, the WebComponentManager URL, and what output fields are available for binding.

C: Behaviors and UI

Under the Bindings Tab there are new choices related to binding Web components with any page or any object on a page. The drop down will list all these Web Components that have already been bound to this Object. These Web Component choices are available at all times, regardless of whether an object is selected or not.

When one or more Web Components have been bound then the "Edit" and "Remove" buttons will become active.

Dynamic Binding is still available, and can be used simultaneously with any Web Component.

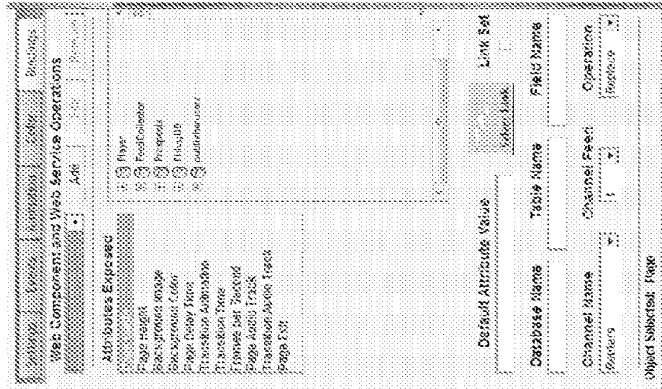


Fig 11.2



### D: Adding a Web Component

When initially touching the "Add" button on the screen in the below appears, with the first Web Component that has been registered selected. In this case it is the Web Component that still relies on our Feed Collector for presenting any of dozens of RSS feeds to the application. In this case, since this is a Web Component implementation of the RSS Display object, "Generate UI Objects" will be selected.

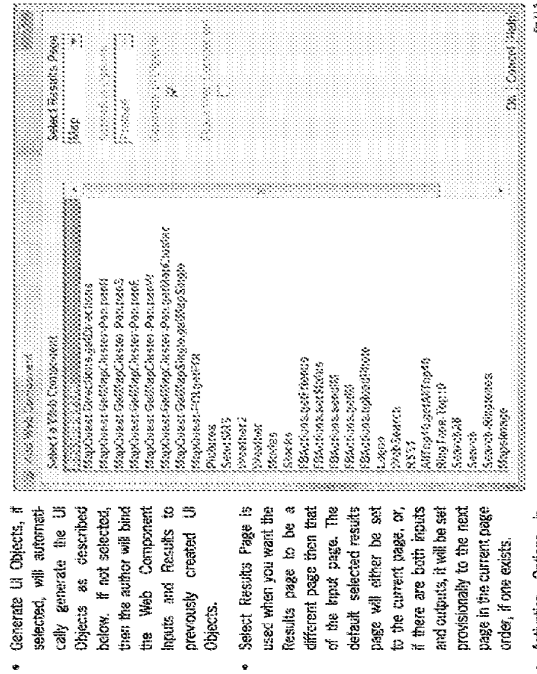


Fig. 11.9

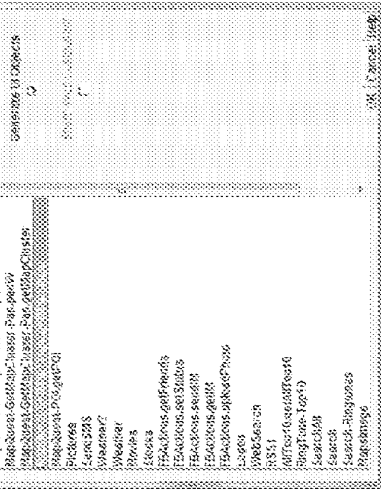


Fig. 11.4

The example below shows how "Generate UI Objects" will be selected for a Mapping application.

- Generate UI Objects, if selected, will automatically generate the UI Objects as described below. If not selected, then the author will bind the Web Component inputs and Results to previously created UI Objects.
- Select Results Page is used when you want the Results page to be a different page than that of the input page. The default selected results page will either be set to the current page, or, if there are both inputs and outputs, it will be set provisionally to the next page in the current page order, if one exists.
- Activation Options include, if there are no input UI Objects, a choice to either "Preload" the Web Component (similar to how Dynamic Binding currently works), or have the Web Component executed when the "Results" page is viewed by the consumer.
- Share Web Component becomes available and will become selected under the following conditions:
  - A Web Component is Selected which already has been used by the current application.
  - The current input page is also a "Result" page for that Web component.

This permits the consumer, after viewing the results, to extend the Web Component (See Section 4) allow the consumer to make additional queries against the same Web Component. Some examples are interactive paging and zooming for a Mapping application, or additional and/or refined searches for a Search application.

### E: Completing a Web Component with the "Generate UI" Option.

When the Generate UI Object's Dialog Box first appears any required Generated Inputs and Outputs are pre-registered in the "Generated" inputs and Outputs Lists. All optional candidates, if any, are placed in the "Candidates" inputs and Outputs Lists. All inputs and Outputs are defined by their "Symbolic Field Names", which will, during this process, finally be bound to auto-generated Xpressrmo UI Objects.

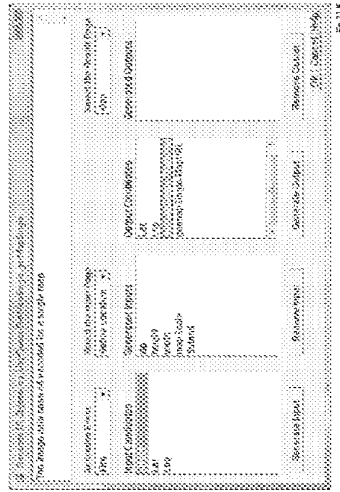


Fig. 11.14

#### e. Comments:

As an Input or Output Object is selected from any of the four lists below, the Comment, retrieved from the Web Component, is displayed in the Comments field on the top for informing the Author about the meaning of that Object.

#### b. Select Input Page

This is set initially to the current page of the application. It can then be reset to any other Page. In Phase 1 there is a restriction of only one Input Page.

#### c. Select Results Page

This is set initially to the "Select Results Page" from the Add Web Component Dialog Box. It can then be reset to any other Page. In Phase 1 there is a restriction of only one Output Page.

#### d. Input Candidates

Any Optional Input Candidate can be selected and, when the "Generate Input" Button is touched, placed into the Generated Inputs List. Note that as each Candidate is selected the Comment related to it is displayed.

#### e. Generated Inputs

Any Generated Input can be selected and, when the "Remove Input" Button is touched, returned to the Optional Inputs List, unless it was a required Input... Note that as each Generated Input is selected the Comment related to it is displayed.

#### f. Output Candidates

Any Optional Output Candidate can be selected and, when the "Generate Output" Button is touched, placed into the Generated Outputs List. Note that as each Candidate is selected the Comment related to it is displayed.

#### g. Generated Outputs

Any Optional Output Candidate can be selected and, when the "Generate Output" Button is touched, placed into the Generated Outputs List. Note that as each Candidate is selected the Comment related to it is displayed.

#### h. OK

When the OK Button is touched the Dialog Box will close and all the Items listed in the Generated Inputs will appear on the "Input" Page and all the Items listed in the Generated Outputs Lists will appear on the "Results" Page.

Using the example of MapQuest (Get Single Map), the graphic below shows the state after the Optional Input for Street and the optional Outputs for Latitude, Longitude and a Single MapImage have been selected.

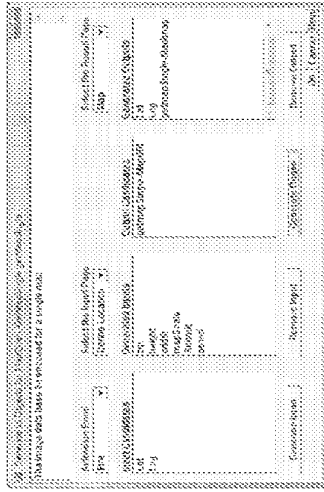


Fig. 11.15

1. The Input Page is shown below. Note that:
  - There are 5 Text Fields lined up 2 per line.
2. There is a Submit Button; that when selected, shows under the "Bindings" Tab the Web Component instance 1: Input\_Maps has been assigned.
  - a. The Web component instance Naming Convention includes three components:
    - i. The Number of the instance (This permits multiple unique assignments of the Same Web Component to the same input page)
    - ii. The Input Page Name (This permits multiple unique assignments of the Same Web Component to multiple Input Pages,
    - iii. The Name of the Web Component.
  - b. The "Edit" and "Remove" Buttons are now active.

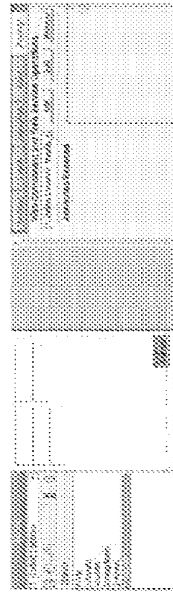


Fig. 11.7

After the UI is generated the author can make any changes to the UI objects that is desired. There are no known restrictions. The generated results page is shown to the right. The Latitude and Longitude fields, and how they are utilized, will be discussed later.

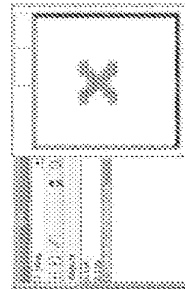


Fig. 11.8

If the "Edit" Button is touched under the Bindings Tab then the Dialog Box below will appear. All the bindings of the Data Fields defined by the Web Component are now bound to the generated UI Objects. Any required inputs or Outputs are identified with a light red background. From here the author could choose to select other UI Objects to replace the ones that had been automatically generated, or to add other optional UI Objects. If any generated required UI Object had been deleted a warning message will appear when the dialog box appears. The designer can then select any other UI Object that had been defined that meet the conditions of the binding as defined by the Web Component Registry Administrator.

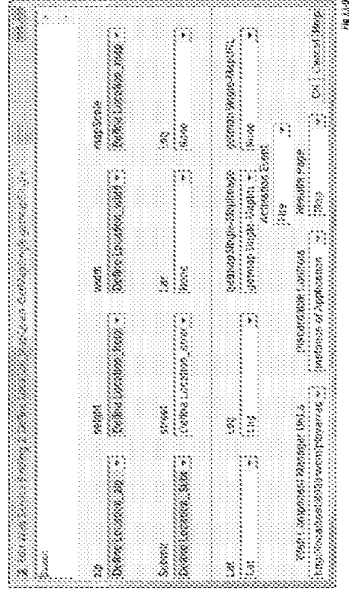


Fig. 11.9

Required inputs can be assigned to Persistent Variables either manually or automatically (see Section F).

The activation event can be mapped to a Page Entry Event or any of 5 Navigation Actions.

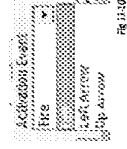


Fig. 11.10

The designer can also change the Server that will be used for the Web Component Manager. The default will be the Server that is currently selected for the Publishing Platform.

For Express Mobile designers the current available server choices are:

- The Express Mobile QA Server: "http://qa.xpressrmo.com/wcm/playernq"
- The Express Mobile Dev Server: "http://dev.xpressrmo.com/wcm/playernq"
- The Staging Server: "http://stg.xpressrmo.com/wcm/playernq"
- The Public Server: "http://st.xpressrmo.com/wcm/playernq"
- Local Host: "http://localhost:8080/wcm/playernq"

The currently selected Server that the Publishing Tool is logged into will be the default.

Discardable Controls are not implemented in Phase One but are defined in the PDL (See Appendix E).

There are a series of rules that the Publishing Client follows for this automatic generation.

Input Object Generation:

1. Text Fields (Single Line Alpha or Numeric Fields)
  - Text fields are generated first, with the rules:
    - a. If there are more than 2 then there will be 2 per line, otherwise they will span the width of the page, with left and right margins of 10 pixels.
    - b. The height will be calculated based on the selected Font from the currently selected Object Style.
2. Text Areas (Multiple Line Alpha Fields)
  - Text areas are generated second, with the rules:
    - a. Two Lines will be visible.
    - b. The Object will span the width of the page, with left and right margins of 10 pixels.
    - c. The height will be calculated based the number of visible lines and on the selected Font from the currently selected Object Style.
3. Check Boxes will be generated next with the following rules:
  - a. All Check Boxes will appear on the same line, with a size of 20 by 20 pixels.
  - b. There will be a left margin of 10 pixels.
  - c. Each Check Box will be separated by 20 pixels.
4. Lists will be generated next with the rules:
  - a. Three items will be visible.
  - b. The Object will span the width of the page, with left and right margins of 10 pixels.
  - c. The height will be calculated based the number of items and on the selected Font from the currently selected Object Style.
5. Last will be the Submit Button, if the Submit Function has not been assigned to another input.
  - The rules will be:
    - a. The String " Submit " will be the default string.
    - b. The Standard HTML Form Colors will be applied.
    - c. Submit Button will be placed in the lower right hand corner of the visible page with a right margin and bottom margin of 10 pixels.
6. All Lines will be separated by 5 pixels.
7. The Object Names will be the same as the Symbolic Field Name as defined by the Web Component.

xpressmo.com

Output Object Generation:

1. If the "Results" page is the same as the "Input Page", then the Output UI Objects will appear directly below the last Input UI Object Line, otherwise it will start with a top margin of 20 pixels.
2. Text Fields (Single Line Alpha or Numeric Fields)
  - Text fields are generated first, with the rules:
    - a. The Object will span the width of the page, with left and right margins of 10 pixels.
    - b. The height will be calculated based on the selected Font from the currently selected Object Style.
3. Text Areas (Multiple Line Alpha Fields)
  - Text areas are generated second, with the rules:
    - a. Two Lines will be visible.
    - b. The Object will span the width of the page, with left and right margins of 10 pixels.
    - c. The height will be calculated based the number of visible lines and on the selected Font from the currently selected Object Style.
4. Check Boxes will be generated next with the following rules:
  - a. All Check Boxes will appear on the same line, with a size of 20 by 20 pixels.
  - b. There will be a left margin of 10 pixels.
  - c. Each Check Box will be separated by 20 pixels.
5. Images and Slide Shows will be generated next, with the rules:
  - a. The Object will span the width of the page, with left and right margins of 10 pixels.
  - b. The Height will either be 60 pixels if there are multiple UI Objects defined for the Result Page, or the Height will span to the bottom of the page.
6. Videos will be generated next, with the rules:
  - a. The Object will span the width of the page, with left and right margins of 10 pixels.
  - b. The Height will either be 60 pixels if there are multiple UI Objects defined for the Result Page, or the Height will span to the bottom of the page.
7. Lists will be generated next with the rules:
  - a. Three items will be visible.
  - b. The Object will span the width of the page, with left and right margins of 10 pixels.
  - c. The height will be calculated based the number of items and on the selected Font from the currently selected Object Style.
8. All Lines will be separated by 5 pixels.
9. The Object Names will be the same as the Symbolic Field Name as defined by the Web Component.

xpressmo.com

### F: Completing a Web Component with Existing UI Objects.

The example below is adding a Yahoo Search Web Component with bindings to previously created UI Objects.

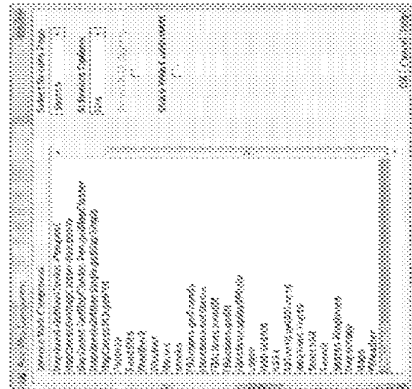


Fig 11.21

Upon touching the "OK" Button the graphic below will appear. Note that "In1", "Submit" and the "Control List: Title" are all populated with an existing previously created UI Object, while all the other fields have "None" selected.

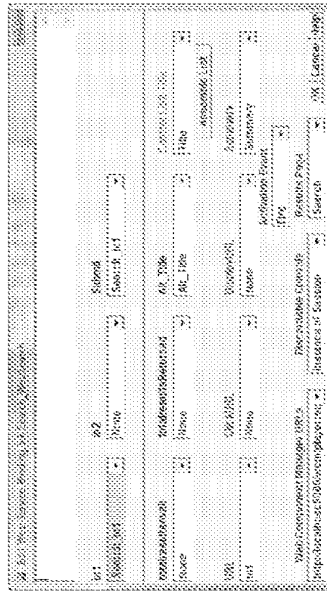


Fig 11.22

As with the MapQuest Web Component example, there are both required and optional input and/or output symbolic fields. In this case there is a required Symbolic Objects as follows:

- An Input object for entering in the Search String.
- An Output Control List which will display the Search Result Titles (See Section I).

Color Coding in the graphic above has the following meanings:

**Required Input or Output**

Control List

Control Object Types

All other symbolic fields are optional.

The rules for selecting which existing UI Objects should be made available to a given Symbolic Field is based on the web Component Definition and the hinting that was applied by the Web Component Registry Administrator.

Section B shows all of the currently supported UI Objects for Xpressrmo's Web Component. These will be mapped to a Symbolic Field Name based on the Supported Data Types shown below.

1. "Boolean"
2. "Int"
3. "String"
4. "MultilineString"
5. "ImageURL"
6. "VideoURL"
7. "List" (Contains a parameter that is converted to an Index)
8. "ComplexList"
9. "Slideshow"
10. "SearchResponseList"
11. "RSSList"
12. "SingleSelectionList"
13. "MultiSelectionList"
14. "ServiceActivation"
15. "ChannelImageURL"
16. "ChannelDescription"
17. "ChannelTitle"
18. "URL"
19. "Audio URL"
20. "Purchase URL"
21. "ImageData"
22. "ImageListData"
23. "PersistentVariable"
24. "PipelineMultipleSelect"
25. "PhoneNumber"
26. "HiddenAttribute"
27. "CollectionList"

The Table below shows the mapping of Xpressmo UI Objects to the Supported Data Types.

Data Types	Preferred Input	Input Candidates	Preferred Output	Output Candidates
boolean	Check Box	Check Box	Check Box	Check Box
int	Text Field (Integer)	Text Field (Integer) Text Field (Phone #) Text Field (SMS #)	Text Field (Integer)	Text Field (Integer) Text Field (Phone #) Text Field (SMS #)
String	Text Field (Alpha)	Choice List (single select)	Text Field (Alpha)	Choice List (single select) Text Button
multilineString	Text Area	Any Text Area	Text Area	Any Text Area Paragraph
ImageURL	N/A	N/A	Image	Image Slide Show
VideoURL	N/A	N/A	Video	Video Slide Show
List	Single Item List	Single Item List Multi-Select List Complex List Choice Slide Show	Single Item List	Any List Type Any Choice Type (see Complex List Specification) List Specification
ComplexList	Complex List	Single Item List Multi-Select List Complex List	Single Item List	Any List Type (see Complex List Specification)
SlideShow	Slide Show	Slide Show	Slide Show	Slide Show
SearchResponseList	N/A	N/A	Search Response List	Search Response List Control List Complex List Choice
RSSList	N/A	N/A	RSS Display List	RSS Display List Control List Complex List Choice
SingleSelectionList	Choice	Choice Complex List	Choice	Choice Complex List
MultiSelectionList	Multi-Selection List	Multi-Selection List Multi-Selection List	Multi-Selection List	Multi-Selection List
ServiceAction	Submit Button	Any	N/A	N/A
ChannelImageURL	N/A	N/A	Image	Image Video Slide Show
ChannelDescription	N/A	N/A	Text Area	Text Area Paragraph Text Field Text Button List Choice

ChannelTitle	N/A	N/A	Text Field	Text Field Text Button Paragraph Text Area List Choice
URL			Text Field (URL request) Text Field (Audio URL request)	Text Field (URL request) Text Field (Audio URL request)
Audio URL			Text Field (Purchase URL request)	Text Field (Purchase URL request)
Purchase URL			Image	Image Slide Show
Image Data			Image	Image Slide Show
Image List Data			Slide Show	Slide Show
Persistent Variable	N/A	N/A	N/A	Image N/A
Popline Multiple Select	Multi-select List	Multi-select List Complex List Slide Show	N/A	N/A
Phone Number	Text Field (numeric type) Complex List	Text Field (numeric type) Complex List	Text Field (numeric type) Complex List	Text Field Text Button Complex List
Hidden Attribute	Complex List	Complex List	Complex List	Complex List
Collection List	N/A	N/A	Slide Show	Slide Show Complex List Complex List Slide Show

## G: Persistent Variables.

When editing a WebComponent in the publisher, any persistent variable that has been defined will appear as a choice for any input. The variable name will be preceded by a lozenge character to differentiate it from a UI Object Name.

If selected, then the UI Object that had previously been assigned to that Symbolic Field will be released and can now be deleted...

In the player when this page is entered, and if the persistent variable has been defined, the WebComponent will be executed using the persistent variable value as an input.

Persistent variables can be used in many ways to make the UI of the Web Component more efficient and intuitive.

### 1. Utilizing login or PIM data:

Often the application was, when first activated, a login page so that the user could register themselves for various web component services or communication functions. This data, if placed in a persistent variable, could then be an input to any web component that, as an input, uses location, phone number, name, password, etc. automatically.

### 2. Improving the UI:

Very often the output from a web component is a URL that will execute a particular web service, such as previewing a product to be purchased, or initiating the purchase transaction itself.

An Image, that clearly labels the web service that could be initiated, could share the same persistent variable as the Text UI object that was assigned to the Web Component output that received the URL. The Text UI object could be hidden so that the UI is more attractive and intuitive.

### 3. Integrating Web Components.

A persistent variable can be utilized in three different ways.

- As a direct input to a WebComponent.
- Assigned to a UI Object that is selected as a Web Component Input
- Assigned a UI Object that is selected as a Web Component Output

Each of these uses of Web Components can come in handy for many use cases.

A few examples are:

### 1. Have Web Component Output Values become inputs to a different Web Component.

In this case assign a persistent variable to the output UI Object. Upon completion of this Web Component the persistent variable will be updated to the new value returned. This persistent variable can now be referred to as an input to any other Web Component, no matter what pag it originates from.

### 2. Set Defaults at Authoring time for a Web Component.

The author could set a default input value to a hidden UI object, and then assign a persistent variable to it. When the Web Component is first invoked it will use this value.

### 3. Mashing Up Outputs from Multiple Web Components

Since any output from any web component can be assigned a persistent variable, the author can now link these persistent variables, as desired, to any UI Object, any relevant Web Component input, or any combination of the two.

## H: Enhancing Web Components with Auto-generated Persistent Variables.

Let's work with the Web Component MapQuest-getMapCluster. It is a more powerful Web Component than the MapQuest-getSingleMap used in Section D.

- 1: First, auto-generate the UI objects.

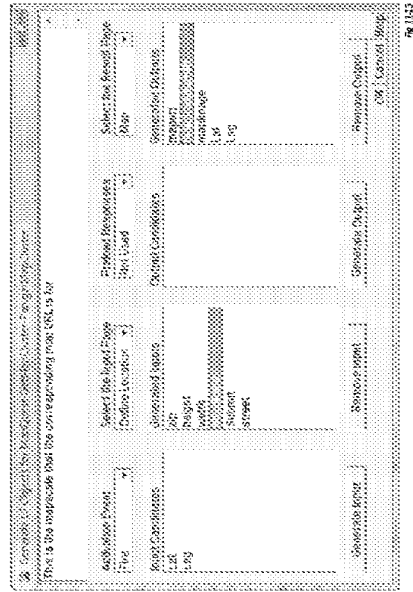


Fig 11.2

- 2: Then Edit the Web Component.

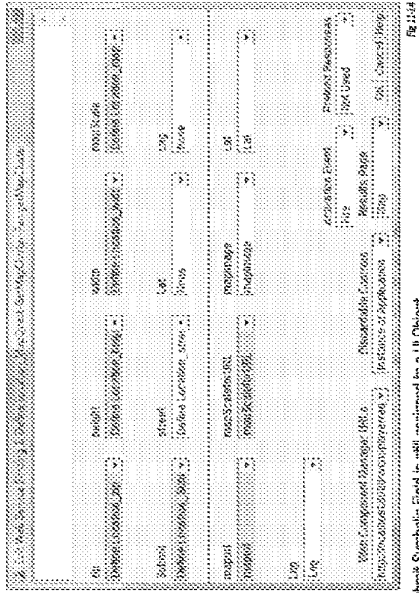


Fig 11.3

Note: The Submit Symbolic Field is still assigned to a UI Object.

- 3: Now set the Submit Symbolic Field to 'None'

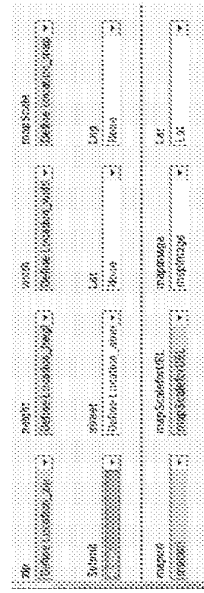


Fig 11.4





## I: Restrictions for Editing UI Objects related to Web Components:

There are three general cases in which the editing of a UI object will be restricted.

1. Delete a UI Object that has a Web Component Attached.

In this case the following message will appear and deletion will be suppressed.

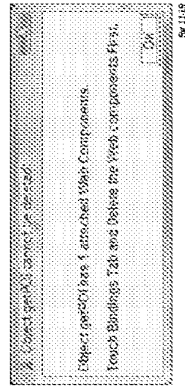


Fig. 11.28

If the author's intent is to reassign the Submit function to another UI Object then first create that UI Object, then Edit the Web Component and reassign the Submit Symbolic Function to the new UI Object. The original UI Object can now be deleted.

2. Delete a UI Object that is bound to a Web Component input or Output or delete a UI Object that is attached to a persistent variable that is bound to a Web Component Input.

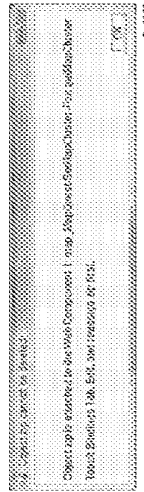


Fig. 11.29

If the author's intent is to reassign the Web Component Symbolic Field to another UI Object then first create that UI Object. Then touch the "Bindings" tab related to the UI Object for Page that has the identified Web component attached. Touch "Edit" and then either reassign the Symbolic Field Function to the new UI Object or set that function to "None". The original UI Object can now be deleted.

3. Remove a Persistent Variable from a UI Object in which that Persistent Variable is bound to a Web Component.

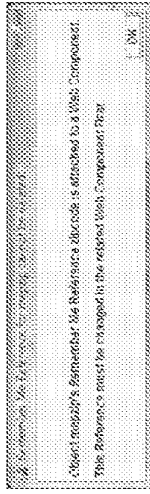


Fig. 11.31

If the author's intent is to reassign the Web Component Symbolic Field to another UI Object then first create that UI Object. Then touch the "Bindings" tab related to the UI Object for Page that has the identified Web component attached. Touch "Edit" and then either reassign the Symbolic Field Function to the new UI Object, reassign it to a different Persistent Variable or set that function to "None". The original UI Object's Persistent Variable binding can now be removed.

## J: Working with Control Lists:

The following Object Types are variations of Control Lists:

- Complex Lists
- RSS Display List
- Search Response List

The Control List Model definition includes any number of subordinate (Child Output Objects) whose contents are known at all times to the Control List, and the Control List manages the appropriate display of their contents in two very different ways.

1. Display as independent UI Objects:

Each Child Output Object is visible, and may or may not be selectable. As the user navigates down through the Control Lists, the values for each of the child lists, related to that item, is displayed. An example is a RSS Display List where the Summary is displayed as a child Object as the Title for the RSS Feed, is selected within the Control Lists.

2. Display as a Mesh up within the Control List:

Instead of displaying the attributes of each child object in a separate UI object, that attribute is displayed as part of the Item of the Control List. Usually the Child Output Object is set to "hidden".

It is possible to employ both techniques within a Control List implementation.

## A: UI Implementation

If a Web component has a Central List, it will be a required Output field. The example to the right is the Thumbnailer AllOpen Score Front Web Component. The automated feature "Generate UI Objects" was selected.

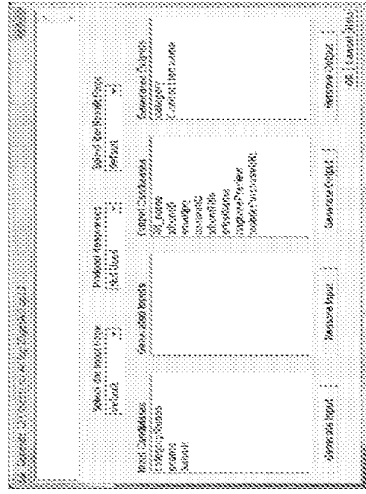


Fig. 11.32

Note: Control List will be identified by the prefix "Control List".

Note: In the example below the Child Output Objects:

- `smallimg`
  - `albumTitle`
  - `artistName`
  - `ringtonePreview`
  - `mobilePurchaseURL`
- have been selected for generation.

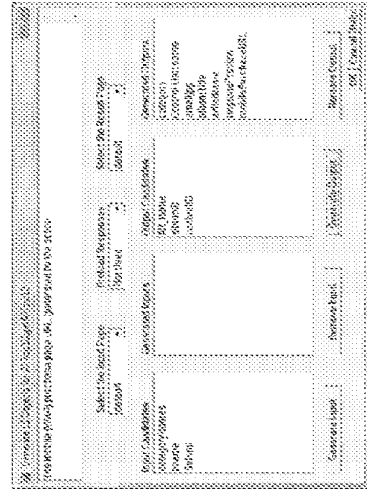


Fig. 11.33

The page view to the right shows the results of the generated outputs after a bit of editing. The Output UI Objects that were generated or modified with persistent variables are:

Output Object	Default UI Object
Control List: name:	Complex List
Smalling	Image
albumTitle	Text Field
artistName	Text Field
ringtonePreview	Text Field
mobilePurchaseURL	Text Field

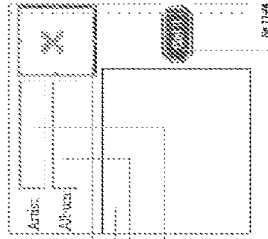


Fig 11.24

The first four objects continue to use the default generated UI objects. The mobilePurchaseURL was initially generated as a text field, but an image icon "Gall" was placed on the page and then a subsequent editing of the Web Component switched mobilePurchaseURL to the image.

Note: This could also have been accomplished by assigning the same "Remember Me" persistent variable to both the generated Text field and the icon image. Hiding the Text field would then effectively switch the Purchase function to the image icon.

Control Lists have a special property that is any of the following object types are assigned to any of its Child Output Objects, and the Events for the Control List are not overridden, then the function assigned to these child objects will be executed when the item selected in the List is fired on.

These Object Types (in precedence order) are:

- HiddenAttribute (Web component Specific Function)
- \* PhoneNumber (To place a phone call)
- \* AudioURL (to stream, if supported, and play an Audio File)
- URL (to WAP or Web Browser Page)

Since the ringtonePreview is of the object type AudioURL, then the current URL assigned to that control List items ringtonePreview Field will automatically be played on a fire event.

If the author wishes for a particular FIRE event to override any of the auto-generated events as described above, and for that choice to take precedence, then follow the following steps:

1. First touch the Events Tab when the Control List bounded Complex List is selected.

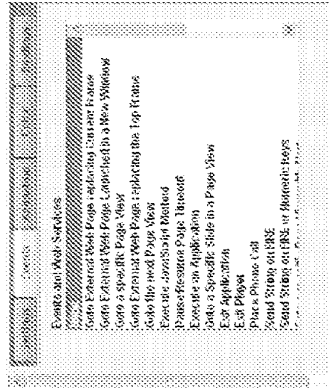


Fig 11.25

2. Then Select an Events or Web Services action. When selecting the following warning will appear.

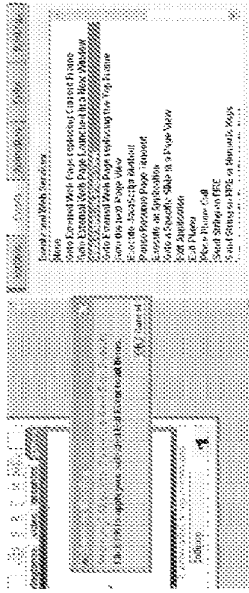


Fig 11.26

3. Touch OK and complete the Selection Process. This setting will now be executed when firing on any of the Control Lists items.



B: Behaviors

The graphic to the right shows how the Control List will be drawn based on the setting as shown in the example below. When the item is selected the Object and List Item fonts will display but the Text Color will be white and the background blue.

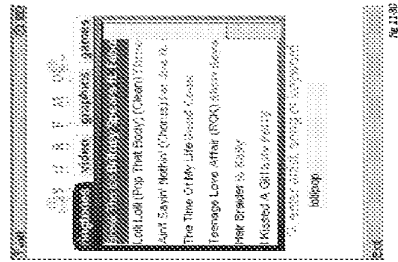


Fig 11.3B

K: Working with Collection Lists:

Collection Lists will be identified by having their symbolic name drawn in **Control**. Collection Lists are only available if the Web Component defines it and there is a Control List associated with it.

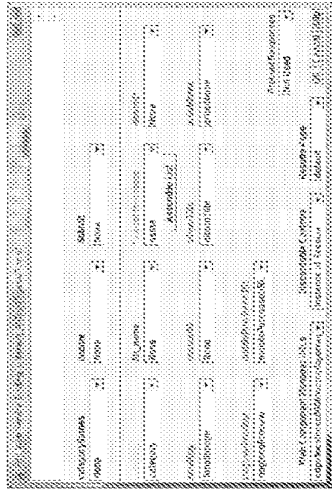


Fig 11.3C

A Collection List will have one and one one Child UI Object, and that Object must be a Control List.

Either Slide Shows or Complex Lists can be assigned to a Collection List although the default will be a Slide Show. There will be a symbolic reference to each element of the Slide Show or Complex List to a Control List definition, which will also include and the attributes for each Control List set of Child UI Objects.

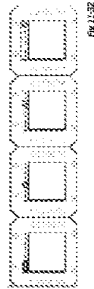


Fig 11.3D

In the example below, the Tabs are actually a Slide Show, as shown in the "Working With Slides" Dialog Box to the right.

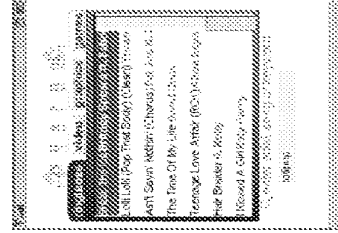


Fig 11.3E

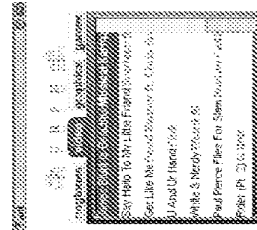


Fig 11.3F

Touching the Right Arrow Key will cause the Slide Show to advance one slide. As each slide contains a symbolic reference to a different Control List definition the Control List is repopulated with that definition.



### M: Sharing a Web Component.

Section H, "Enhancing Web Components with Persistent Variables" described a powerful way to combine, extend and mash up various Web Components to create great Services. However, sometimes the requirement is just to use a single Web Component, but extend it through sharing to make it much more interactive and usable. For example:

Using the example of Yahoo Maps, the graphic below shows the state after the Optional Inputs for Street, City, State, Zip, Image\_Height, Image\_Width and Zoom have been selected.

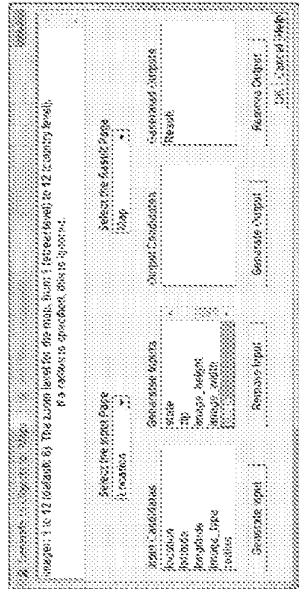


Fig. 11-39

The generated "Location" page is shown below.

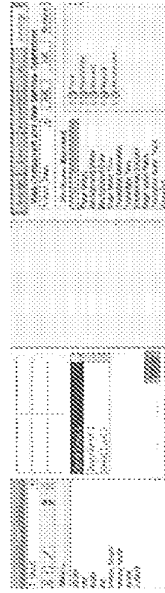


Fig. 11-39

The generated "Map" page is shown to the right.

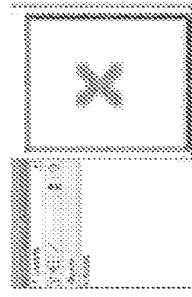


Fig. 11-40

### 1: Complex Lists.

Control Lists may also have hidden attributes which to define one or more values for requesting to the Web Service (as an Input List Data Type) or as an index for visualization as an Output Data Type of various field values as part of a set of data records.

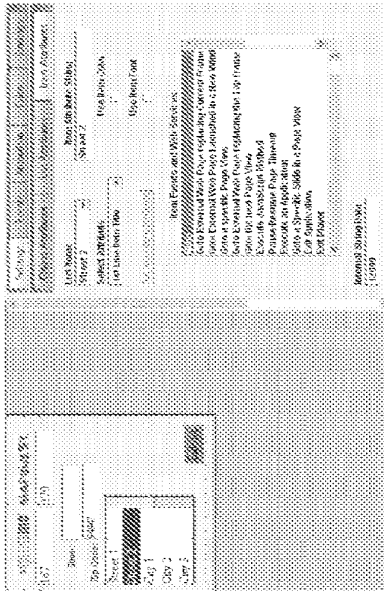


Fig. 11-38

In the case above, the MapQuest Web Component generated a Complex List for each zoom level that would be supported. The Internal String Data for each item on the list was retrieved from the Model. And defined below and placed appropriately for each list item.

```
<ListItem value="6000"><Street 1</ListItem>
<ListItem value="12000"><Street 2</ListItem>
<ListItem value="24000"><City 1</ListItem>
<ListItem value="48000"><City 2</ListItem>
<ListItem value="96000"><City 3</ListItem>
<ListItem value="192000"><County 1</ListItem>
<ListItem value="400000"><County 2</ListItem>
<ListItem value="800000"><County 3</ListItem>
<ListItem value="1600000"><State 1</ListItem>
```

This process could have also been done manually by entering each index from the Model to the Internal String Data for each List item.



1. Go to the Results Page ("Map") of this application.
2. Add appropriate UI Objects for further interactions with the Web Component. (We might add a Slide Show UI Object that would be assigned to the Zoom Symbolic Field. Hence of a mapping Web Component. In this case we will use Yahoo Maps as shown below)
3. Touch the "Bindings" Tab and Select "Maps"

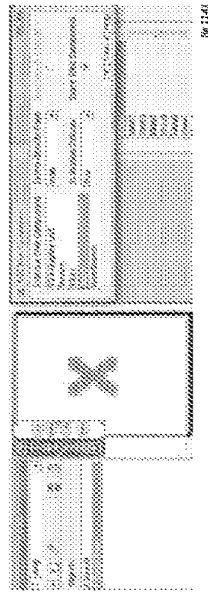


Fig. 11.43

Note that "Generate UI Objects" is not available, and "Share Web Component" is available and Selected. When you touch "OK" the Dialog Box below will appear.

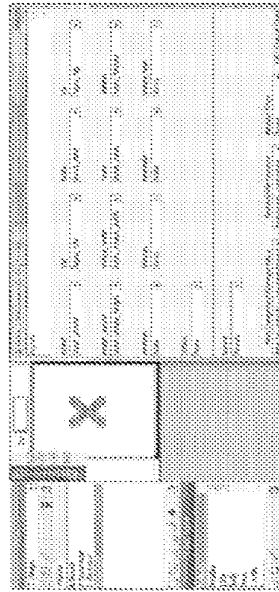


Fig. 11.44

Note:

As seen above, it is possible to assign a Slide Show as a "List" Output Object, so that you can produce graphical controls for various types of Web Components. In addition, there is now an exit event for Slides that permit you to assign "Previous Slide" as well as "Next Slide". This means that functions such as "Zoom in" or "Zoom Out" can be implemented in various interesting ways. Now assign the Submit Symbolic Field and the Zoom Symbolic Field to the Slide Show UI Object and the Location Symbolic Field to the Location Text Field as shown below. Also, since location will be used for the target map, Street, City, State and Zip can now be assigned to "None".

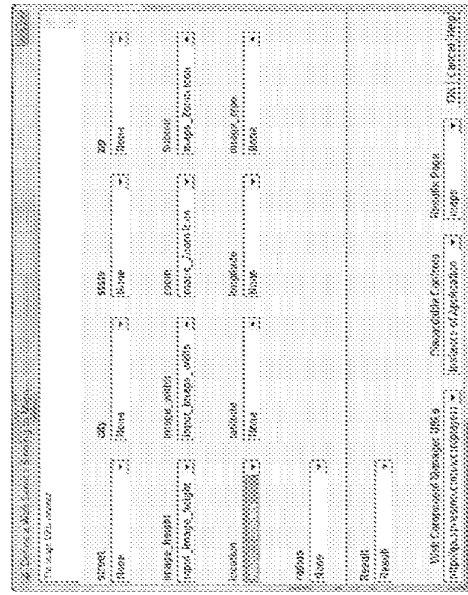


Fig. 11.49

After touching "OK" the "Map" the Extended Web component 3map\_Maps will be assigned to the Slide Show UI Object.

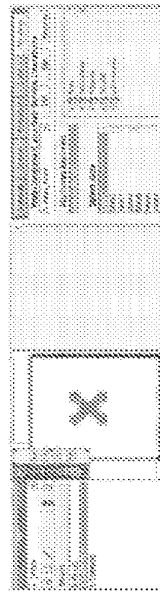


Fig. 11.44

Now save the application and view the results on a phone or in a MDCP 2.0 Emulator.

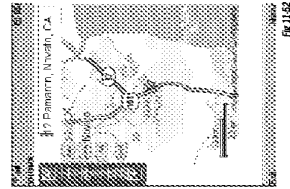


Fig 11.45

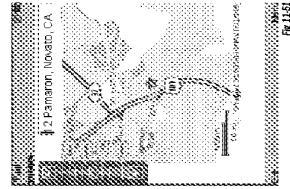


Fig 11.46

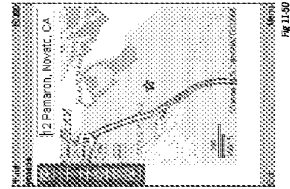


Fig 11.47

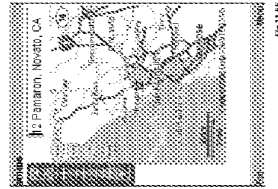


Fig 11.48

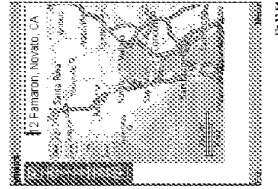


Fig 11.49



Fig 11.50



Fig 11.51

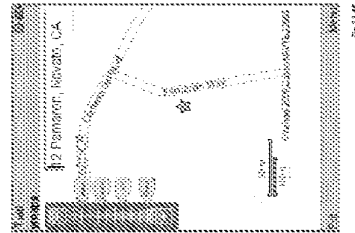


Fig 11.45

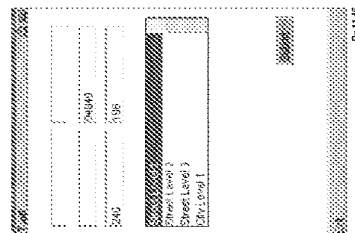


Fig 11.46

Initial View

Result

- Initial move will be to Zoom in if not on the Maximum zoom level.
- FIRE or, if Enable Fire is selected, the forward and next navigation keys, will Zoom In the Map One Level and move the Slice Indicator.
- If a navigation key is touched that reverses direction, that will switch the mode to that direction.
- FIRE or, if Enable Fire is selected, the forward and next navigation keys, will then Zoom Out or Zoom In One Level the Map but the Slice Indicator will not move.
- The next FIRE or, if Enable Fire is selected, the forward and next navigation keys, will both Zoom Out/In and move the slice indicator.
- Hitting the maximum or minimum zoom will cause an automatic direction change.

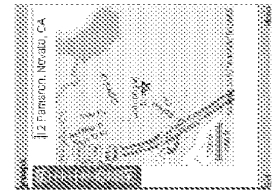


Fig 11.47



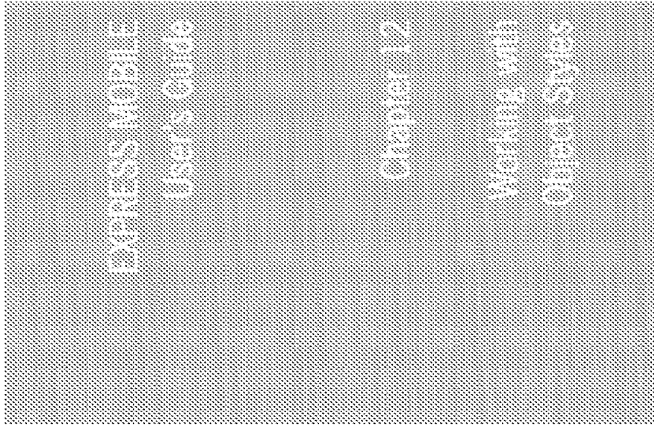
Fig 11.48



Fig 11.49

## N: How to create a Web Component Model:

See Appendix C.



## Chapter 12: Working with Object Styles

A: Introduction	12.04
B: Object Styles	12.05

## Chapter 12: Glossary

### Styles

The underlying set of default appearance attributes or behaviors that define any object that is attached to a style. A very convenient way for quickly creating complex objects, and for changing a whole collection of objects by just modifying their common style.

## A: Introduction:

Object styles are used to define a certain default set of attributes for any given object when it is created. They are extremely useful if you are building pages that have multiple objects and you want them all to have the same attributes. For example, if you are building a Web page and want identical navigation buttons, using this feature will let you click on the style you've defined – font size, color, type, effects, background color, etc. – and all future buttons you add will be identical. You can also modify the style and all existing objects with that style will be changed, even if they are on multiple Web pages. You can also assign a different style to an existing object. This will cause all the attributes of the new style to be assigned to that object.

Page styles are useful if you want the same attributes on each page, including objects. For example, a company Website may want their logo and a navigation bar in the same position as well as the same background color on each page. Using page styles means you can create a style once and each subsequent page added will contain the same objects versus having to create the same style for each individual page while working on this site.

Using these features can save an enormous amount of time – and frustration!

### B: Object Styles:

1: How to Create an Object Style:

Object styles can be defined for any object (text, picture, shape, etc.) and then applied to an object already on the page or to objects yet to be added. There are currently 10 object styles pre-defined.

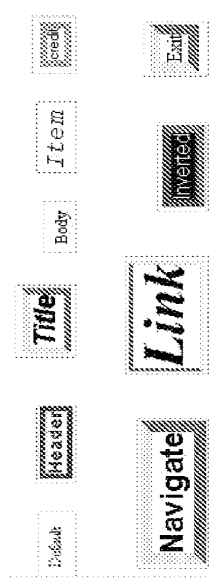


Fig 124

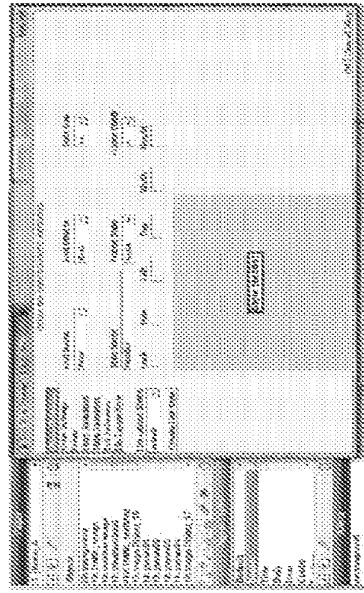


Fig 122

Change several of the attributes – font name, effects, size, frame style and width and click "ok".

Note that the feedback box shows what the changes look like.

3: How to Save an Object Style:

Touch the Save icon from the Command Floating Palette or select "Save As..." from the File Menu.

The dialog box to the right will appear. Once this object style is saved it can be imported in another editing session as described below.

4: How to Import an Object Style from the Styles Library:

To access this style in the future, select

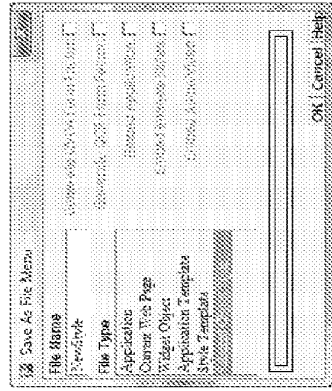


Fig 123

2: How to Modify an Object Style:

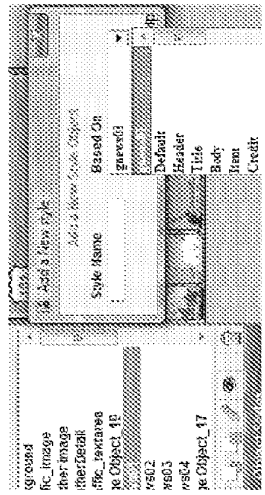
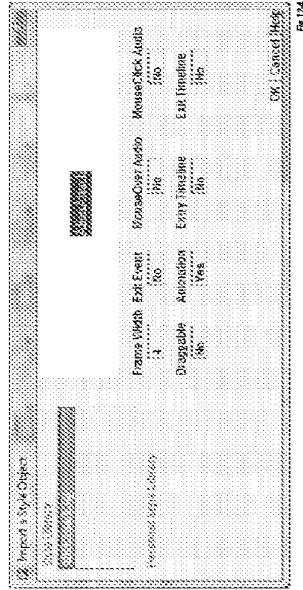


Fig 122

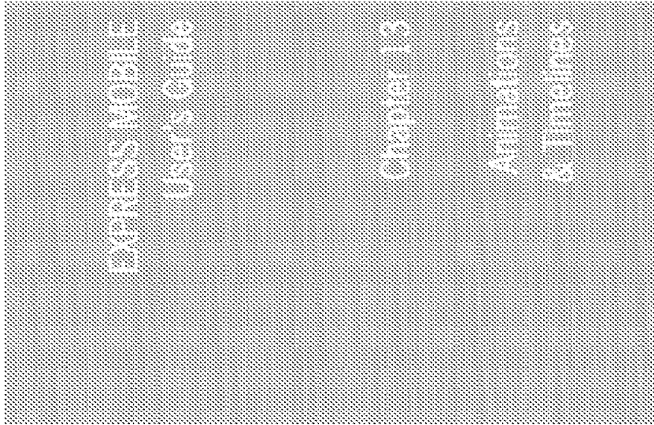
"Styles" and then click on "Import", or Select the Import icon in the Layer Inspector for Object Styles.



The "Import an Object Style" dialog box appears; select "AlienAttach.spj" and click "OK". This object style will now appear in the "Style" List in the Layout Inspector.

5: Inheritance:

Once you create an object using a style and then change some attribute of the object while onscreen without modifying the style itself – if you go back & then modify the style and change that same attribute to reflect something else, the attribute for that object won't change. For example, you created a button created using a style that had a red font. You then – onscreen (by using "Edit" or the color drop down menu) - change the font color to blue. You then modify the style to a font color of yellow. The font color of the button onscreen will remain blue.



## Chapter 13: Animations & Timelines

A:	Introduction	13.04
B:	How to Define	13.06
C:	Child Object Animations & Behaviors	13.09
D:	Defining Object Entry Timelines	13.11
E:	Defining Object Exit Timelines	13.12
F:	Tips	13.13



## Chapter 13: Glossary

### Slide Show Animations

The transition animations that can occur when changing from one slide to another.

### Waypoints

The vertices for any object that has an animation trajectory path defined. Each waypoint can have numerous attributes and behaviors assigned to it, so that the user experience can be enhanced in significant ways.

## A: Introduction

Text objects, images, videos, slide shows, shapes and form objects (with the exception of complex lists) can have both directions and animations assigned to them. In addition, directions and animations can be assigned to all of these object types that are defined as child objects. Note that Form Objects cannot be assigned as a child object. To animate the object, be sure that the object is selected and select one of the methods below:

- Touch the "Animation" tab of the resource inspector.
- Touch the Edit Icon from the Layer Inspector and select the "Animation" tab.
- Select "Animations..." from Events off the Menu Bar.

As a shortcut, animations can be defined without activating the timeline.

- **Delay:**  
This is the time, in seconds, that defines when the animation should start relative to when the page comes into view.
- **Direction:**  
This defines the location of the object over time. The choices vary depending upon the object type. Generally these types of animations will work quite well, even on fairly low end feature phones.
- **Movement:**  
This defines the animation effect, and these choices also vary by object type. It is permissible to have both a direction and/or movement assigned simultaneously to the same object.
- **Duration:**  
This represents the time that the animation will remain active from start to finish.
- **Frames:**  
These are the maximum number of frames per second that the animation will be drawn. In the event that the device does not have the processing power to perform at this frame rate then the frame rate will automatically be reduced so that the animation finishes within its allotted time.

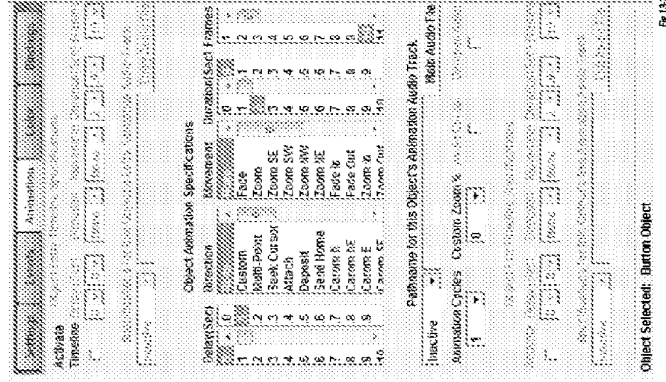


Fig.13.1

- Animation Audio:

An audio can be assigned to the animation. It is possible to set the audio to play once or to continue to loop until the animation has been completed.

- Animation Cycles:

Any number of animation cycles can be set. This would cause the animation to repeat itself based on the number of cycles chosen.

- Custom Zoom %:

Normally a zoom will zoom the object from zero up to 100 percent of its size. It is possible to increase the maximum size up to 10 times its normal size. This feature is not recommended for feature phones.

## B: How to Define

### 1: Define Form Objects, Text Button or Paragraph Animations

Any of 31 different directions and 18 different animations can be assigned (See Appendix B).

Select either the "Edit" or "Animate" icons as described above. Choose the desired delay time, direction and/or animation, the time you want the animation to last, the number of animation frames that will be drawn per second, and the number of times the animation should occur. The time for the animation can be set from 0.5 seconds to 1 minute, the number of frames per second from 1 to 50, and the number of animation cycles from 1 to 50 or to a continuous ("loop") setting.

To choose a custom zoom level for zooming animations, the range can be from zero to 1000 percent of the defined size for the button or paragraph. 1% means it will zoom to a very small level; 100% means it will remain the same size and anything above 100% will zoom larger.

**Avoid Cursor:** if the object, during its animation, comes in contact with the cursor it is repelled in such a way that it never intrudes upon the cursor's location. When the cursor is removed from the object's trajectory, the animation resumes as expected. During the time that the object is being repelled by the cursor, no animation time lapses, so that when the animation resumes, it continues as if the collision with the cursor had not occurred, with the exception of the delay.

To add an audio track, select one from the Express Mobile Library or add your own (must be in ".au" or ".wav" formats). You can have the audio track play once or loop. An audio stream can also be added.

Click on the "OK" icon to complete the animation. With the exception of the "Custom" animation, the animation will then be executed for confirmation.

### 2: Define Image or Slide Show Animations.

Any of 31 different directions and 26 different animations can be assigned (See Appendix B). The process for defining the animation is the same as for buttons and paragraphs above.

3. Define Shape Animations.

Any of 31 different directions and 28 different animations can be assigned (See Appendix B). The process for defining the animation is the same as for buttons, paragraphs and images above.

i: Define General Directions

1: Custom Animation

When the "Custom" animation is selected, a "Define Animation Path" button will appear. After clicking on this button, a message saying "Recording Custom track" will display. Drag the mouse across the Canvas to draw the desired animation. A blue line will be drawn to show the animation (see example below). To stop drawing, right click the mouse and release.

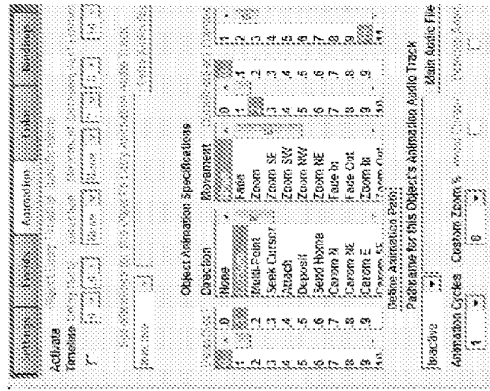


Fig. 13.7

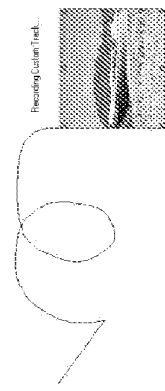


Fig. 13.8

Immediately the trajectory (all the points on the screen that the object will move through) will disappear. The number of frames per second setting is ignored.

To use waypoints, see the explanation in the "Events" chapter - Waypoints.

2: Multi-Point

The same as Custom, but with straight lines only. To create, click where you want the line to end. To stop drawing, right click your mouse and release.

To use waypoints, see the explanation in the "Events" chapter - Waypoints.

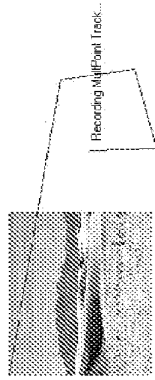


Fig. 13.4

3: Seek Cursor Movement

The object immediately moves towards the cursor. The speed of movement is calculated to be the time that the object would cross the entire Web page within the time set for the animation. When the object reaches the cursor it immediately jumps to its home location and the animation is completed. It is permissible to assign an animation that will execute simultaneously while the object is seeking the cursor.

4: Attach

The object immediately jumps to the cursor and stays attached to the cursor during the time that the animation is alive. When the time for the animation has elapsed the object will jump to its home position. It is permissible to assign an animation that will execute simultaneously while the object is attached to the cursor.

5: Deposit

The object immediately jumps to the cursor and stays attached to the cursor during the time that the animation is alive or until the user presses the mouse button, whichever occurs sooner. If the time for the animation elapses without a mouse click the object will jump to its home position. If a mouse click occurs while the animation is alive then the object is deposited at the location of the mouse click and the animation is terminated. It is permissible to assign an animation that will execute simultaneously while the object is attached to the cursor.

6: Send Home

The object immediately jumps to the cursor and stays attached to the cursor during the time that the animation is alive or until the user presses the mouse button, whichever occurs sooner. If the time for the animation elapses without a mouse click the object will jump to its home position. If a mouse click occurs while the animation is alive then the object immediately jumps to its home position and the animation is terminated. It is permissible to assign an animation that will execute simultaneously while the object is attached to the cursor.

7: Hover

This is used when you want to play an audio track for a specified period of time for a specific object. The object will hover (remain stationary) and not perform any animations.

## c. Define General Animations

Most are self-explanatory, with the exception of Spin, Rotate, Swing and Swivel. These 4 animations, around either a vertical axis or a horizontal axis, are available for shapes and images.

Spin is a clockwise or counter-clockwise movement.

Rotate is a 360-degree horizontal or vertical (west to east or north to south) rotation using the vertical or horizontal center of the image as the axis of rotation.

Swing is a 360-degree horizontal or vertical (west to east or north to south) rotation using the right or bottom edge of the image as the axis of rotation.

Swivel is a 180-degree horizontal or vertical (west to east or north to south) rotation and then a 180-degree horizontal or vertical clockwise rotation using the vertical or horizontal center of the image as the axis of rotation.

## d. Continuous ("Loop") Animation

When selected, the animation will continue to repeat itself until any of the following conditions occur.

The Web page is terminated either because its time setting has been reached or the Web page is left because of an exit event.

The object had been assigned the "Deposit" or "Send Home" movement and a mouse click had occurred.

## C: Child Object Animations and Behaviors

## 1: Application Page Initialization

When an Application Page becomes available all Parent Object Animations immediately begin to execute for that Page. If Child Animations had been defined for the Parent Object, then those animations will begin immediately after the parent object animation concludes. When the child animations conclude, the Child Objects will disappear.

## 2: Parent Object Selected

If a Parent Object is selected in which the "Timeline Activation" event (see chapter 14) has not been selected, then the following would happen. Assuming that this Object has Child Objects which have defined animations, then these animations will begin immediately. The Child Objects will disappear when the Parent Object is no longer selected, or when their animations conclude, whichever occurs first.

## 3: FIRE on Parent Object

If a FIRE event occurs on a Parent Object in which the "Timeline Activation" event (see chapter 14) has been selected, then the following would happen. Assuming that this Object has Child Objects which have defined animations, then these animations will begin immediately. The Child Objects will disappear when their animations conclude.

### D: Defining Object Entry Timelines

Then the Activate Timeline Check Box is selected, all of the Object Entry Animation choices become available.

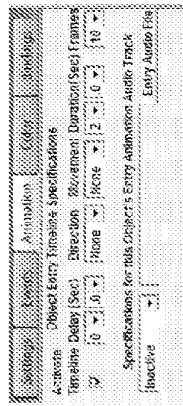


Fig. 13.5

- 1: **Delay**  
This is the time that will elapse from the moment a Page has become active until the Object appears. This delay can be set to up to 100 seconds.
- 2: **Direction**  
The way an Object enters the Page can be set. Choose "None" if no direction animation is desired. See Appendix B for the list of the direction entry direction animations available.
- 3: **Movement**  
The Object's entry to the Page can be extended with movement animations. Choose "None" if no movement animation is desired. See Appendix B for the list of the movement entry direction animations available.
- 4: **Duration**  
Time is the length of time in seconds that the entry animation will take to complete. It can be set from immediate to up to 100 seconds.
- 5: **Frames (Frames per Second)**  
The number of frames per second is the number of animations that will occur per second during the entire entry animation process. This can be set from 1 to 50. The higher the number of frames, the smoother the animation will appear.
- 6: **Audio Track**  
If an audio track has been added to the entry animation, it will start simultaneously with the beginning of the animation. The track can either play once or loop.

### E: Defining Object Exit Timelines

Selecting the "Activate" check box will immediately make available all of the Exit Timeline Settings.

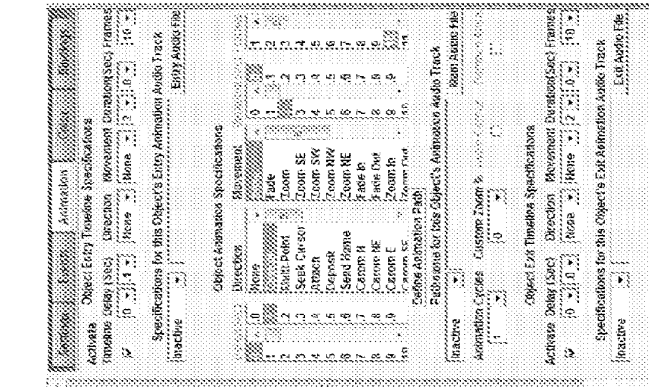


Fig. 13.6

- 1: **Delay**  
This determines whether the Object will remain on the Page until the Page is unloaded (stay), or whether it should depart after a certain period of time. This time begins from the moment that the object's animation(s) has been completed and all the child timelines, if any, have also been completed. The departure delay time can range up to 100 seconds.
- 2: **Direction**  
The depart direction is how the Object departs the Page. It can be set to depart immediately (none) or by means of an exit direction. See Appendix B for the list of the exit directions available.
- 3: **Movement**  
The Object's departure can be enhanced with a Movement animation. See Appendix B for the list of 6 exit methods/animations available.
- 4: **Duration**  
Depart time is the time in seconds that the exit animation will take to complete. It can be set from immediate to up to 100 seconds.
- 5: **Frames (Frames per Second)**  
The number of frames is the number of animations that will occur per second during the entire exit animation process. This can be set from 1 to 50.
- 6: **Audio Track**  
If an audio track has been added to the exit animation, it will start simultaneously with the beginning of the animation. The track can either play once or loop.

## F: Tips

### 1: Number of frames for button or paragraph zooming animations:

Be sure that the number of frames is less than the point size minus 5 otherwise unpredictable results could occur.

### 2: Different Animation Effects:

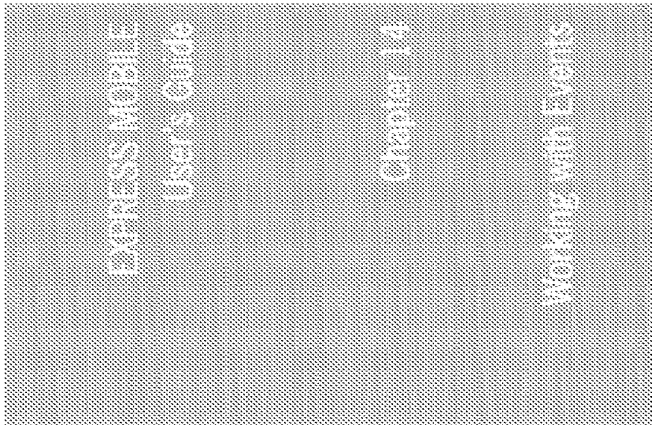
Changing the number of frames to a very small number can dramatically affect the animation effect. Changing the attributes of the object, such as changing the point size and background color for buttons, will create very different visual effects.

### 3: Animations applied to Animated GIFs:

Animations can also be applied to animated GIFs. Certain animations will work better than others, and the animation effects will be most effective if the timing of the animated GIF's transitions are well synchronized with the speed and the number of frames of the animation. The "Fade In" and "Fade Out" animations will only be effective if the animated GIF's time settings are precisely synchronized with the frames per second of the "fade" animations.

## Chapter 14: Working with Events

A: Child Object Events	14.04
B: Exit Events	14.05
C: Select and Fire Events	14.10
D: Waypoints	14.13



## Chapter 14: Glossary

### A: Child Object Events

#### I: Visibility:

There are three possible techniques for causing child objects to change their visibility.

- a. Selection of the Parent Object
  - Upon selection all child objects will appear (subject to their timeline definitions, if any) and will disappear when the parent object is deselected.
- b. Toggle Children on FIRE
  - When this attribute is selected for the parent object then selection or deselection itself will have no effect. The first FIRE on the parent object the child objects will become visible and remain that way until another FIRE event occurs on that parent object, or when another Parent Object is FIRED on that has the "Hide non-related Children" attribute selected.
- c. Child Slide Shows, Timelines and Animations
  - Child objects can be assigned all the timeline and animation attributes that a parent object can be assigned, and a Child Slide Show can be assigned all the slide transition animations that a Parent Slide Show can be assigned.

The activation of these visual and audio effects begin when the child object becomes visible and ends whenever the child object loses visibility or the timeline is completed, whichever occurs first.



## B: Exit Events

I: Discussion:

There are currently 29 possible exit events that can be assigned to an Express Mobile Object, which have made the UI very complicated and unwieldy. Although internally the exit events will be stored as per Appendix A, for the UI, the following changes are proposed.

In addition, there are now four new Events:

1: Send IM Alert (Push Rich Content Messaging)

For Sending an IM Alert Message. This can occur by:

- firing on a basic object that has a logical link to a text field or text area
- firing on a List or Dropdown Item .
- firing on a slide with an attached Chat (chat room)

An IM Alert Message will generate, from the Content Server, an IM Alert Page II may use the System Alert Object if the Alert Page cannot be found.

An IM Alert will have a Text Message defined, which will be extracted from either a designated text field or text area. Optionally, an Avatar, which could be an Image or Video, can be linked to the Alert Page.

For Chat, the Avatar will be the Slide Media Object. If the Player is open the IM Alert will immediately appear as a Modal Alert with Fire enabled for dismissing. If the player is not active, when the player later becomes active there will be an indication of the number of pending IM alerts. The method or methods for browsing the pending alerts is yet to be defined.

2: Set Starting Page

For personalizing the starting page by the consumer. This can occur by firing on a basic object or firing on a List or Dropdown Item. For any subsequent uses of the application, or any "Change View" events in the current session, if the "Change View to Starting Page" is defined, then the consumer-chosen "starting page" will appear.

3: Send String on FIRE

This exit event is available for Text Fields and Text Areas. It will execute the Submit Button Listener so that the current string will be sent to the content Server for processing.

4: Send String on FIRE or Numeric keys

This exit event is available for Text Fields and Text Areas. It will execute the Submit Button Listener so that the current string will be sent to the content Server for processing.

xpressmo.com

I: User Interface:

There will be up to six dropdown menus for defining an Event. A subset of these choices will be available, depending upon the type of Object that is selected.

1: Alerts:

- None
- Display on Fire
- Display with Synchronization on Fire
- Display/Hide OnFocus/OffFocus

2: Communication

- None
- Place Phone Call
- Send SMS Message
- Send IM alert

3: Share:

- None
- Share peer to peer
- Share multiple selection
- Share all

4: Personalization:

- Set Add Icon to Icon Strip
- Remove Icon from Icon Strip
- Logical First (Starting) Page
- Set Persistent Variable

5: Change View

- None (next slide)
- Goto External Page (WAP or HTML)

Change View options

- Replace Current View
- Launch New Window
- Replace Top View
- Goto Internal Page

xpressmo.com

- Change View Options
  - Next Internal Page
  - Specific Internal Page
  - Specific Internal Page w/Slide Number
  - Goto Widget
  - Goto Logical First (Starting) Page
- Execute an Xpresso Application
- Exit Application
- Exit Player
- Execute Scripting Method
- Pause/Resume Page Timeout

III: Object Specific Implementations:

As before, events can be assigned to three different object classes. For each class, the following events would be available.

A: All Object Classes and Page Timeout

- 0=None (or Next Slide)
- 1=Goto External Page (WAP or HTML)
- 2=Goto Internal web Page
- 3=Execute an Xpresso Application
- 27=Send IM Alert

B: The Basic Object

- 4=Place Phone Call
- 5=Send SMS Message
- 6=Exit Application
- 7=Exit Player
- 8=Execute Scripting Method
- 9=Pause/Resume Page Timeout
- 28=Set Starting Page
- 29=Set Persistent Variable

C: An Item in a List or Dropdown Object

- 4=Place Phone Call
- 5=Send SMS Message
- 10= Dropdown: Add Icon to Icon Strip List: Display/Hide Icon toggle
- 11= Dropdown: Remove Icon from Icon Strip
- 28=Set Starting Page
- 29=Set Persistent Variable

D: A Slide in a Slide Show

- 12=Goto Specific Slide

E: Web Page

- 8=Execute Scripting Method

"Goto New View Options" will only be available for the following Exit Events:

- 1=Goto External Page (WAP/HTML)
- 2=Goto Internal web Page

"Stare Options" will only be available for the following Exit Events:

- 4=Place Phone Call
- 5=Send SMS Message

IV: Mapping to Current Player Event Manager:

The Table Below describes what choices will be available by exit event for the six dropdown option menus, and how that maps to the possible list of 28 events.

UI Exit Event	Alert Options	Change View Options	Personalize	Communicate	Share	Internal Exit Event
None	None Display w/Title Display OffFocus Hide OffFocus Display w/Sync			Send IM Phone Call Phone Call Phone Call Send SMS Send SMS Send SMS	None Share All None Share All	17 19 22 27 12 24 26 23 25 13 14 28
Go to External Page		Replace Current View Launch New Window Replace Top View	Add Icon Hide Icon Set Starting Page			1 2 4
Go to Internal Page	None Display w/Sync None None	Specific/Next Page Logical 1st Page Specific w/Side #				3,5 15 21 9
Execute Application-Widget	None Display None					8 20 16
Exit						10
Application						11
Exit Player						6
Execute Script						7
Pause/Resume Page						
Go to Slide						10
Send IM Alert						27
Send String on FIRE						13
Send String on FIRE or Numeric Keys						14
Set Persistent Variable						29

C: Select and Fire Events

1: Select Events:

The following select events are available.

- Display and activate ChildItem:
  - If "Toggle Children on FIRE" is not selected, then all children will become visible as soon as the parent object is selected. If any of those children have animations or timelines assigned to them then those animations will begin immediately.
- Object Selected Audio:
  - If an object selected audio track has been assigned to this object then this audio track will immediately start to play. There are four possible settings as shown in the graphic below.

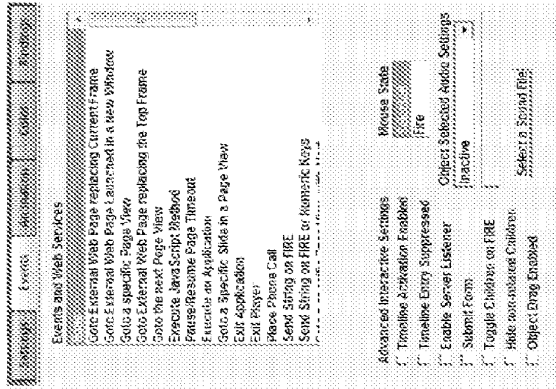


Fig 14.1

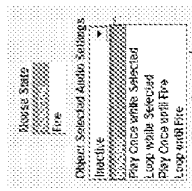


Fig 14.2

2: Fire Events:

The following fire events are available.

- Timeline Activation
  - if the "Timeline Activation Enabled" check box is selected, then the parent's animations and timelines will begin upon the detection of a FIRE event. The activation of these animations will not occur simply because the page has become available.
- Timeline Entry Suppressed.
- if the "Timeline Entry Suppressed" check box is selected, then, if there is an entry timeline defined, it will be executed when the page becomes available. It will, however, be suppressed when a FIRE event is detected.
- Display and activate Children:
- if "Toggle Children on FIRE" is selected, then all children will become visible as soon as a FIRE event is detected on the parent object. If any of those children have animations or timelines assigned to them then those animations will begin immediately. A subsequent FIRE event will cause those children to disappear.
- Hide non-related Children
- if the "Hide non-related Children" check box is selected, then any other children that were visible because of the "Display and Activate Children" event described above, then they will immediately disappear.
- Object FIRE Audio:

If an object Fire audio track has been assigned to this object then this audio track will immediately start to play. There are four possible settings as shown in the graphic below.

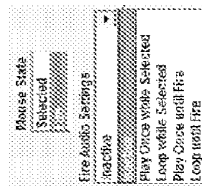


Fig 143

3: Tips

- a: Using Both Timelines and Exit Events:
  - Because the timeline events will be completed before the exit event for button and image objects, interesting farewell messages can be constructed.
- b: Advanced Event based Visual and Audio Effects:
  - In previous chapters, features and techniques have been discussed which utilized the static and timeline based overlapping of objects with the utilization of various audio and visual attributes that can be specified for these objects. With events, a new universe of user interactions can now be implemented using these dynamic visual and audio interactions between objects, and between the objects' children.
- c: Multiple Simultaneous Timeline Events:
  - You can overlay button and/or image objects on top of each other, each of which may have timeline events associated with them. When the mouse is clicked over these objects all associated timelines will be activated simultaneously.

## D: Waypoints

When an object has an animation direction of either custom or multi-point, waypoints are the starting points, points where there was a pause or direction change and the ending points of an objects course or path. They are identified by blue squares.

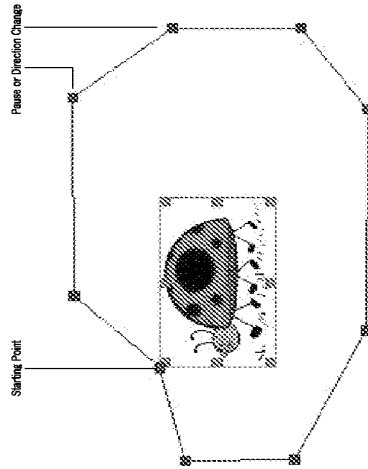


Fig 144

### 1: How to Create:

To create waypoints, first select the object to be animated and then choose either the "Custom" or "Multi-Point" animation direction and create your path.

### 2: How to Define:

To define waypoints, make sure the object is selected and then either choose the "Events" menu and click on "Waypoints" or right click on the object and choose "Waypoints".

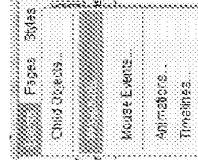


Fig 145

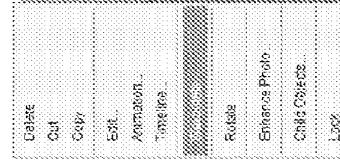


Fig 146

The "Define Waypoints" dialog box will appear. The items in this dialog box tell you what values have been assigned to a particular waypoint.

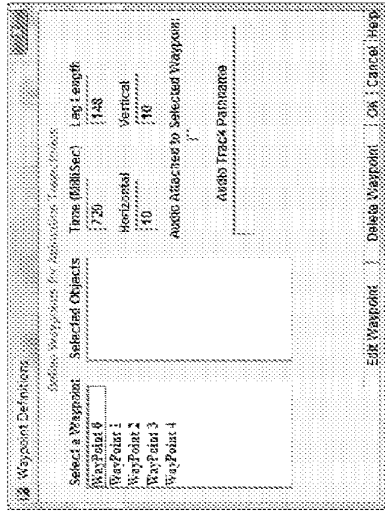


Fig 147

"Select a Waypoint" shows all the waypoints for the object.

"Selected Objects" shows which object is currently selected for defining, if there are multiple objects on the web page.

"Time" is the length of time the object takes to traverse that leg in the path.

"Horizontal" and "Vertical" shows the current position of that waypoint.

"Leg Length" is the length of the leg beginning with the selected waypoint to the next one.

The "Audio Attached" check box tells you if an audio has been assigned, and, if so, which audio track has been assigned to the selected waypoint.

3: How to Edit:

The "Edit Waypoint" button allows you to change the above positions and lengths. After selecting a waypoint, clicking this button takes you to the "Edit WayPoints" dialog box (below).

The values of the selected waypoint are displayed and can be changed by typing in a new value. An audio track for each waypoint can be changed or added; choose from the Express Mobile audio library or import your own track.

Also any number of objects can be selected for this waypoint. When that waypoint is reached while the object is moving along its trajectory, then any waypoint selected objects, if they have a timeline associated with them, will have their timeline activated.

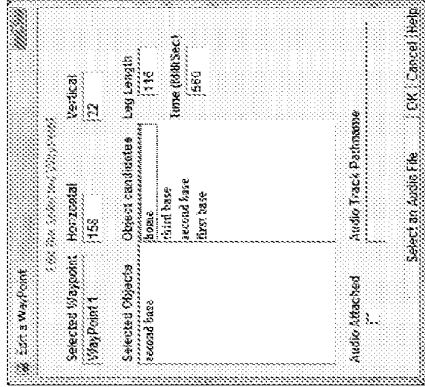


Fig. 14.8

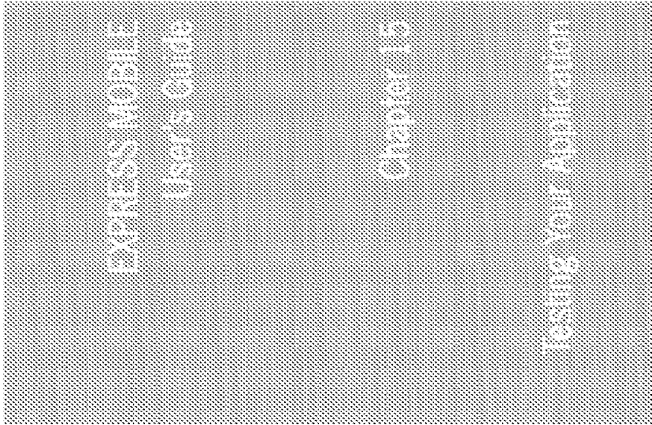
4: How to Delete a WayPoint:

Select the desired waypoint and then touch the "Delete Waypoint" button.

"Audio" tells you which audio track (if any) has been assigned to the selected waypoint.

## Chapter 15: Testing Your Application

Testing an Application  
created in Publisher 15.04



## Chapter 15 : Glossary

Emulator

Any tool that will, more or less, produce a similar result as what would occur on the Mobile Device itself.

WTK

Sun Java Wireless Toolkit for CLDC

## Testing an Application Created in Publisher

A very convenient way to observe how an application will perform is to select the "Launch" command from the View Menu. As long as one of the supported browsers is available (Opera, Internet Explorer, or Firefox) then the application will be launched in that browser. Note that the Express Mobile Java/J2SE Player will be invoked, but the operations will be quite similar, except that the mouse will become the pointing device. If the intention is to run this application on a Java CDC device (such as Set top Boxes, Google Android devices, or Java-based telematic devices, then this will demonstrate almost perfectly how the application will run in these different environments.

Once you have saved an application in Publisher you may wish to see how it would appear on a phone or on a phone emulator. To run the application on a device, the player portion of the application must be downloaded at least once. Once the player is available on the device, you can modify the application, and in most cases, will not need to download the player again. Simply re-launching the application will instantly display the changes.

This discussion assumes a development and test: environment. An application that has been edited may not be immediately available to the end user, as stated here, due to procedural requirements. However, in general, designers and developers should have immediate access to updated applications so that the applications can be tested and validated.

### Required Information

You need the following information before continuing with the player download:

- The username for Publisher (from the Log In dialog box)
- The server to which Publisher client is connected (from the Log In dialog box)
- The name given to the application (from the Save or Save As dialog box)

For this example let us use the following names:

- Username: myname
- Server: publisher.com
- Application: MyApp

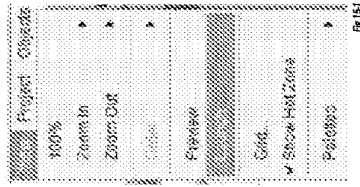


Fig 15.1



Address of Application

The web address (also called URL or Uniform Resource Locator) of the mobile application for Java enabled devices has two distinct constructions.

Accessing Applications via Response Director

Response Director provides a single URL for a given application. The Response Director will determine the proper player files to send to the device that is requesting the application.

The construct of a URL for an application is:

```
http://server/get/username/appname
```

where:

- server is the domain name or IP address of the Publisher server
- username is the author's Publisher username
- appname is the name given by the author to the mobile application.

For example:

```
http://publisher.com/get/myname/MyApp
```

This URL is case sensitive. This URL construct is the preferred method for accessing applications. However, this method may not work for all phones, nor will it work for the phone emulator. If this method does not work, use the alternate method discussed in the following section.

Accessing Applications via Direct URL

This URL construct accesses the .jad file directly. This should be used when testing via the emulator and when the Response Director method does not work for a device. Note that this method may not be available to end users for security reasons. Contact your System Administrator to see if the Direct URL method is available to you.

The Direct URL construct has several options. Initially the most common URL will be described. The format is:

```
http://server/%?username/WebSites/appname/bin/appname.jad
```

where:

- server is the domain name or IP address of the Publisher server
- username is the application author's Publisher username
- appname is the name given by the author to the mobile application.

For example:

```
http://publisher.com/%?myname/WebSites/MyApp/bin/MyApp.jad
```

This URL is case sensitive. This URL is normally for a QVGA sized application with code signing certificate.

All variations of URL are listed in the table below. Note that the QCIF variations may not be available if the author did not explicitly request auto-generation of QCIF version from QVGA version during a Save As operation.

Device Requirements	URL
QVGA Normal	http://server/%?username/WebSites/appname/bin/appname.jad
QVGA No Certificate	http://server/%?username/WebSites/appname/bin/appnameNC.jad
QVGA No File Access	http://server/%?username/WebSites/appname/bin/appnameNF.jad
QVGA No Certificate & No File Access	http://server/%?username/WebSites/appname/bin/appnameNCNF.jad
BlackBerry	http://server/%?username/WebSites/appname/bin/appnameBB.jad
PC Browser	http://server/%?username/WebSites/appname/bin/appname.html
QCIF Normal	http://server/%?username/WebSites/appnameQCIF/bin/appnameQCIF.jad
QCIF No Certificate	http://server/%?username/WebSites/appnameQCIF/bin/appnameQCIFNC.jad
QCIF No File Access	http://server/%?username/WebSites/appnameQCIF/bin/appnameQCIFNF.jad
QCIF No Certificate & No File Access	http://server/%?username/WebSites/appnameQCIF/bin/appnameQCIFNCNF.jad

Using the Emulator

To use the phone emulator, you must download and install the Sun Java Wireless Toolkit for CLDC (WTK). This is available at:

<http://java.sun.com/products/swjtoolkit/>

Your PC account must have privileges to install applications. At present, the WTK is only available for Windows and Linux environments.

Once you have installed the WTK, the recommended configuration for the emulator is as follows:

1. From Taskbar, go to Start→All Programs→Sun Java (TM) Wireless Toolkit 2.5.2 for CLDC→Preferences.
2. Go to the Security Category. For Security Policy, select JTWI; for Security Domain, select maximum.
3. Select the OK button and exit the Preferences utility.

Next select the device type that best matches the phone type you want to test. For the most general emulation the recommended setting is as follows:

1. From Taskbar, go to Start→All Programs→Sun Java (TM) Wireless Toolkit 2.5.2 for CLDC→Default Device Selection.
2. For Default Device, select DefaultColorPhone.
3. Select OK to exit the Default Device Selection utility.

To run a mobile Java application in the phone emulator, use your PC browser and type in the URL (web address) of the mobile application using the Direct URL construct. See section **Accessing Applications via Direct URL** for construction of the URL.

Downloading Over the Air (OTA) to a Mobile Device

When available, use the Response Director construct of the URL, as this is the preferred method. (As of this writing, BlackBerry devices must use the Direct URL method to access the player).

Typing in a long URL is tedious on mobile devices with a limited text entry capability. The simplest method is to use SMS to send the URL of the application to the device. For this, you will need the 10-digit NANPA (North American Numbering Plan Area) phone number, or if your SMS service allows, the country code and phone number of your mobile device with which you wish to load and test the application.

First, locate a suitable SMS service. There are several free services available from the PC to send SMS, including Yahoo! and AOL. Not all phones are reachable from some of these free services.

Once the appropriate SMS service has been located, enter in or select the phone number of the mobile device and type the URL of the application into the message area, as described in either section **Accessing Applications via Response Director** or section **Accessing Applications via Direct URL** above.

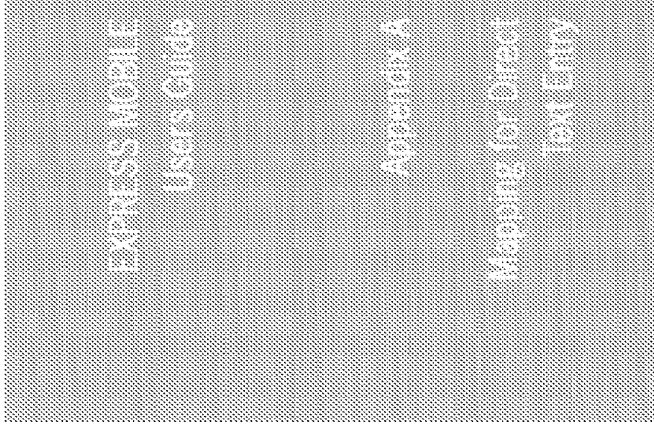
When your mobile device receives the SMS, open the message and select the URL within the message. Then select the menu option that allows you to link to the URL. These are variously labeled as "Link to" or "Go to" in the device's menu.

Along the way you may encounter some or all of these types of messages, worded differently based on device model:

Type of Message	Appropriate Answer
Confirm download of application MyApp?	Yes
Choice of connection type?	Choose the appropriate highest speed connection. The choice will depend on your operator settings.
Application MyApp is untrusted. Download OK?	Yes
Application MyApp version xxx already exists. Overwrite with version yyy?	Yes
Remove Previous Data?	If you retain the data, the state of the last run of the application will remain, such as the cache — which will speed up the access. If on the other hand you want to test the pristine state of the application, removal of the application data is recommended.

Table 1. Types of Messages during Application Page Download

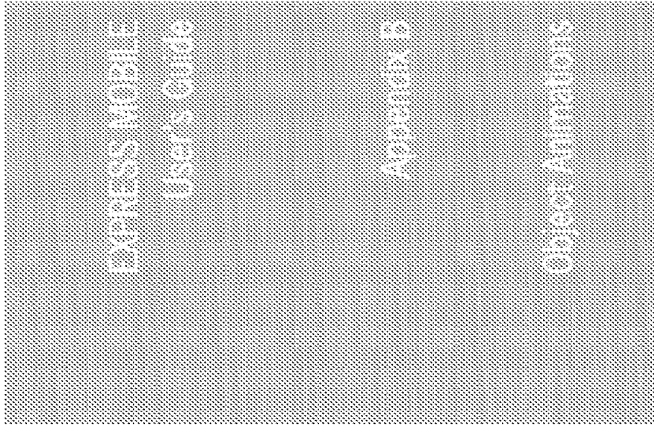
If SMS is not available, you can rely on the URL in the mobile device's browser. If you plan to download new versions frequently, you may want to use a bookmarking feature available on most browsers.



Consecutive presses on the same key without pause allow entry of the following characters.

Pressing one more than the number of defined consecutive keystrokes returns to the first character.

Number of keypresses on same key without pause→									
Key↵	1	2	3	4	5	6	7	8	9
0	space	0	+						
1	.	,	@	1	?	!	*	#	/
2	abc	a	b	c	2				
3	def	d	e	f	3				
4	ghi	g	h	i	4				
5	jkl	j	k	l	5				
6	mno	m	n	o	6				
7	pqrs	p	q	r	s	7			
8	tuvw	t	u	v	8				
9	wxyz	w	x	y	z	9			
*	(reserved for launch strip use)								
#	(reserved for launch strip use)								



General Directions:

- |                |              |
|----------------|--------------|
| 1. Custom      | 17. Enter E  |
| 2. Multi-Point | 18. Enter SE |
| 3. Seek Cursor | 19. Enter S  |
| 4. Attach      | 20. Enter SW |
| 5. Deposit     | 21. Enter W  |
| 6. Send Home   | 22. Enter NW |
| 7. Carom N     | 23. Exit N   |
| 8. Carom NE    | 24. Exit NE  |
| 9. Carom E     | 25. Exit E   |
| 10. Carom SE   | 26. Exit SE  |
| 11. Carom S    | 27. Exit S   |
| 12. Carom SW   | 28. Exit SW  |
| 13. Carom W    | 29. Exit W   |
| 14. Carom NW   | 30. Exit NW  |
| 15. Enter N    | 31. Hover    |
| 16. Enter NE   |              |

General Animations for Text Objects:

- |             |               |
|-------------|---------------|
| 1. Fade     | 10. Zoom Out  |
| 2. Zoom     | 11. Grow NW   |
| 3. Zoom SE  | 12. Grow NE   |
| 4. Zoom SW  | 13. Grow SE   |
| 5. Zoom NW  | 14. Grow SW   |
| 6. Zoom NE  | 15. Shrink SE |
| 7. Fade in  | 16. Shrink SW |
| 8. Fade Out | 17. Shrink NW |
| 9. Zoom In  | 18. Shrink NE |

General Animations for Shapes & Images:

1. Spin Left
2. Spin Right
3. Flip Left for shapes only
4. Flip Right only
5. Fade
6. Rotate WE
7. Swing WE
8. Swivel WE
9. Rotate NS
10. Swing NS
11. Swivel NS
12. Zoom
13. Zoom SE
14. Zoom SW
15. Zoom NW
16. Zoom NE
17. Fade in
18. Fade Out
19. Zoom In
20. Zoom Out
21. Grow NW
22. Grow NE
23. Grow SE
24. Grow SW
25. Shrink SE
26. Shrink SW
27. Shrink NW
28. Shrink NE

Entry Motions:

1. Enter N
2. Enter NE
3. Enter E
4. Enter SE
5. Enter S
6. Enter SW
7. Enter W
8. Enter NW

Entry Methods:

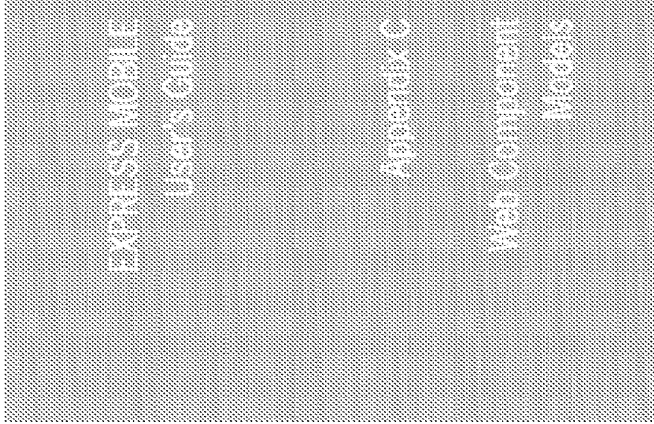
1. Fade In
2. Zoom In
3. Grow NW
4. Grow NE
5. Grow SE
6. Grow SW

Exit Motions:

1. Exit N
2. Exit NE
3. Exit E
4. Exit SE
5. Exit S
6. Exit SW
7. Exit W
8. Exit NW

Exit Methods:

1. Fade Out
2. Zoom Out
3. Shrink NW
4. Shrink NE
5. Shrink SE
6. Shrink SW



1: RSS: USA Today Top Stories

```

<?xml version="1.0" encoding="utf-8" ?>
<WebComponentModel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" name="RSS1" group="Examples" PlayerURL="http://localhost:8080/
wcm/playerreq" ServiceURL="http://rssfeeds.usatoday.com/usatoday-NewsTopStories">
<inputs ?>
</inputs>
<Outputs>
<Parameter name="channel:title" data-type="ChannelTitle" param-type="response" ServiceMap-
ping="/rss/channel/title" use="optional">
<Comment>RSS Channel Title</Comment>
</Parameter>
<Parameter name="channel:description" data-type="ChannelDescription" param-type="response"
ServiceMapping="/rss/channel/description" use="optional">
<Comment>RSS Channel Description</Comment>
</Parameter>
<Parameter name="channel:image-uri" data-type="ChannelImageURL" param-type="response" Ser-
viceMapping="/rss/channel/image/uri" use="optional">
<Comment>RSS Channel Image URL</Comment>
</Parameter>
<List ServiceMapping="/rss/channel/item" name="rssitems" controlParam="Item-title">
<Parameter name="Item-title" data-type="RSSItem" param-type="response">
ServiceMappings="title" use="required">
<Comment>RSS Feed Item title</Comment>
</Parameter>
<Parameter name="Item-link" data-type="String" param-type="response">
ServiceMappings="link" use="optional">
<Comment>RSS Feed Item links</Comment>
</Parameter>
<Parameter name="Item-description" data-type="multilineString" param-type="response">
ServiceMappings="description" use="optional">
<Comment>RSS Feed Item titles</Comment>
</Parameter>
</List>
</Outputs>
</WebComponentModel>

```

2: Yahoo Search

```

<?xml version="1.0" encoding="utf-8" ?>
<WebComponentModel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" name="WebSearch" group="Yahoo" XPathNS="urn:yahoosearch"
PlayerURL="http://localhost:8080/wcm/playerreq" ServiceURL="http://api.search.yahoo.com/Web-
SearchService/V1/webSearch?appid=ag60c_1Y4KXZp7ia_eF8r0nT0cF0J9KmjJHL4Ky/hmZ3Of.
usktTheILtB">
<inputs>
<Parameter name="in1" data-type="String" param-type="request" use="required">
ServiceMapping="query">
<Comment>the search term string</Comment>
</Parameter>
<Parameter name="in2" data-type="int" param-type="request" use="optional" ServiceMapping="results"
defaultValue="10">
<Comment>The number of results to return, defaults to 10</Comment>
</Parameter>
<Outputs>
<Parameter ServiceMapping="/ns:ResultSet@totalResultsAvailable" name="totalResultsAvail"
data-type="int" use="optional">
<Comment>The total results available</Comment>
</Parameter>
<Parameter ServiceMapping="/ns:ResultSet@totalResultsReturned" name="totalResultsReturned"
data-type="int" use="optional">
<Comment>The total results returned</Comment>
</Parameter>
<List ServiceMapping="/ns:ResultSet/ns:Result" name="searchResults" controlParam="Title">
<Parameter name="Title" data-type="ComplexList" use="optional" ServiceMapping="Title">
<Comment>The title of the search result link</Comment>
</Parameter>
<Parameter name="URL" data-type="String" ServiceMapping="UH" use="optional">
<Comment>The URL to the search result</Comment>
</Parameter>
<Parameter name="ClickURL" data-type="String" use="optional" ServiceMapping="ClickURL">
<Comment>The click thru URL to the search result</Comment>
</Parameter>
<Parameter name="DisplayURL" data-type="String" use="optional" ServiceMapping="DisplayU
rl">
<Comment>A display URL for the user to see for the search result</Comment>
</Parameter>
<Parameter name="Summary" data-type="multilineString" use="optional" ServiceMapping="S
ummary">
<Comment>Short paragraph of the search result page</Comment>
</Parameter>
</List>
</Outputs>
</WebComponentModel>

```



3: Yahoo Maps

```

<?xml version="1.0" encoding="utf-8" ?>
<WebComponentModel xmlns:sei="http://www3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressiro.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressiro.com" name="Map" group="Yahoo" PlayetURL="http://localhost:8080/
wcm/playreq?ServiceURL="http://local.yahooapis.com/MapsService/V1/mapImage?appid=dgS0s_
V94BEXZP?l_a_eF6r0tF0F2D9KtmJH4f8Y/tmZ0Of.isR:ThE1nLr?>
<Inputs>
<Parameter name="street" dataType="String" paramType="request" use="optional"
ServiceMapping="street">
<Comments>Street Address</Comment>
</Parameter>
<Parameter name="city" dataType="String" paramType="request" use="optional"
ServiceMapping="city">
<Comments>City</Comment>
</Parameter>
<Parameter name="state" dataType="String" paramType="request" use="optional"
ServiceMapping="state">
<Comments>2 Letter state code</Comment>
</Parameter>
<Parameter name="zip" dataType="Int" paramType="request" use="optional" ServiceMapping="zip">
<Comments>5 digit zip or 5-4 zip</Comment>
</Parameter>
<Parameter name="location" dataType="String" paramType="request" use="optional" ServiceMapping
="location">
<Comments>street city state zip locations</Comment>
</Parameter>
<!-- Used new data type for decimal values 1 -->
<Parameter name="latitude" dataType="String" paramType="request" use="optional" ServiceMapping=
"latitude">
<Comments>float: -90 to 90 The latitude of the starting location.</Comment>
</Parameter>
<Parameter name="longitude" dataType="String" paramType="request" use="optional" ServiceMappin
g="longitude">
<Comments>float: -180 to 180 The longitude of the starting location. If both latitude and longitude
are specified, they will take priority over all other location data. If only one of latitude or longi-
tude is specified, both will be ignored.</Comment>
</Parameter>
<Parameter name="image_type" dataType="String" paramType="request" use="optional"
ServiceMapping="image_type">
<Comments>png (default) or gif The image format for the map.</Comment>
</Parameter>
<Parameter name="image_height" dataType="Int" paramType="request" use="optional"
ServiceMapping="image_height">
<Comments>Integer: 10 to 2000 (default: 500) The height of the image being generated. In pixels.</
Comment>
</Parameter>
<Parameter name="image_width" dataType="Int" paramType="request" use="optional"
ServiceMapping="image_width">
<Comments>Integer: 10 to 2000 (default: 600) The width of the image being generated. In pixels.</
Comment>
</Parameter>

```

```

<Parameter name="zoom" dataType="List" paramType="request" use="optional"
ServiceMapping="zoom">
<Comments>Integer: 1 to 12 (default: 6) The zoom level for the map, from 1 (street level) to 12
(country level). If a radius is specified, this is ignored.</Comment>
<ListValues value="1">Street Level 1</ListValues>
<ListValues value="2">Street Level 2</ListValues>
<ListValues value="3">Street Level 3</ListValues>
<ListValues value="4">City Level 1</ListValues>
<ListValues value="5">City Level 2</ListValues>
<ListValues value="6">City Level 3</ListValues>
<ListValues value="7">County Level 1</ListValues>
<ListValues value="8">County Level 2</ListValues>
<ListValues value="9">State Level 1</ListValues>
<ListValues value="10">State Level 1</ListValues>
<ListValues value="11">State Level 2</ListValues>
<ListValues value="12">Country Level 1</ListValues>
</Parameter>
<Parameter name="radius" dataType="String" paramType="request" use="optional"
ServiceMapping="radius">
<Comments>float How far (in miles) from the specified location to display on the map. The default
radius varies according to the location given and the zoom level.</Comment>
</Parameter>
<Inputs>
<Outputs>
<Parameter ServiceMapping="/Result" name="Result" dataType="ImageURL" use="required">
<Comments>The image URL returned.</Comment>
</Parameter>
<Outputs>
</WebComponentModel>

```

4: Movies

```

<?xml version="1.0" encoding="utf-8" />
<WebComponentModel xmlns:xi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" name="Movies" group="Examples" PlayerURL="http://localhost:8080/
wcm/playerreq" ServiceURL="http://mx.xpressmo.com/movies_rss.xml">
<inputs />
<Outputs>
<Parameter name="channel-title" dataType="ChannelTitle" paramType="response" ServiceMap-
ping="/rss/channel/title" use="optional">
<Comments>RSS Channel Title </Comment>
</Parameter>
<Parameter name="channel-description" dataType="ChannelDescription" paramType="response"
ServiceMapping="/rss/channel/description" use="optional">
<Comments>RSS Channel Description </Comment>
</Parameter>
<Parameter name="channel-image-uri" dataType="ChannelImageURL" paramType="response" Ser-
viceMapping="/rss/channel/image/uri" use="optional">
<Comments>RSS Channel Image URL </Comment>
</Parameter>
<List ServiceMapping="/rss/channel/item" name="rssitems" controlId="item-title">
<Parameter name="item-title" dataType="RSSList" paramType="response"
ServiceMapping="title" use="required">
<Comments>RSS item title of video </Comment>
</Parameter>
<Parameter name="item-link" dataType="VideoURL" paramType="response"
ServiceMapping="link" use="optional">
<Comments>RSS item link - RTSP video URL </Comment>
</Parameter>
<List ServiceMapping="description" dataType="multilineString" paramType="response">
<Parameter name="description" use="optional">
<Comments>RSS item short description of video </Comment>
</Parameter>
</List>
</Outputs>
</WebComponentModel>

```

5: Accuweather

```

<?xml version="1.0" encoding="utf-8" />
<WebComponentModel xmlns:xi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" removeHTML="true" name="WeatherZ" group="Examples"
ServiceURL="http://ss.accuweather.com/rss/liveweather_rss.asp?metric=0">
<Managers>
<Manager type="qa" uri="http://qa.xpressmo.com/wcm/playerreq" />
<Manager type="dev" uri="http://dev.xpressmo.com/wcm/playerreq" />
<Manager type="local" uri="http://localhost:8080/wcm/playerreq" />
<Manager type="staging" uri="http://stg.xpressmo.com/wcm/playerreq" />
<Manager type="product" uri="http://cst.xpressmo.com/wcm/playerreq" />
<Managers>
<Parameter ServiceMapping="locCode" name="zip" use="required" dataType="int">
<Comments>The ZIP code to get weather </Comment>
</Parameter>
</Inputs>
</Outputs>
<Parameter name="channel-title" dataType="ChannelTitle" paramType="response" ServiceMap-
ping="/rss/channel/title" use="optional">
<Comments>RSS Channel Title </Comment>
</Parameter>
<Parameter name="channel-description" dataType="ChannelDescription" paramType="response"
ServiceMapping="/rss/channel/description" use="optional">
<Comments>RSS Channel Description </Comment>
</Parameter>
<Parameter name="channel-image-uri" dataType="ChannelImageURL" paramType="response" Ser-
viceMapping="/rss/channel/image/uri" use="optional">
<Comments>RSS Channel Image URL </Comment>
</Parameter>
<!-- current weather -->
<Parameter name="currentweather" dataType="String" paramType="response" ServiceMapping="/rss/
channel/item[0]/title" use="optional">
<Comments>Today's current weather </Comment>
</Parameter>
<Parameter name="currentweather-link" dataType="String" paramType="response" ServiceMapping="/
rss/channel/item[0]/link" use="optional">
<Comments>Today's current weather URL </Comment>
</Parameter>
<Parameter name="currentweather-description" dataType="multilineString" paramType="response"
ServiceMapping="/rss/channel/item[0]/description" use="optional" containsHTML="true">
<Comments>Today's weather description </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/item[1]/description" name="currentweather-img-src"
use="optional" dataType="ImageURL" paramType="HTML" EmbeddedMapping="img@src">
<Comments>Today's weather embedded image src of the forecast gif </Comment>
</Parameter>
<!-- Today's forecast -->
<Parameter name="todayweather" dataType="String" paramType="response" ServiceMapping="/rss/
channel/item[0]/title" use="optional">
<Comments>Today's forecast weather </Comment>

```

```

</Parameter>
<Parameter name="todayweather-link" dataType="String" paramType="response" ServiceMapping="/
rss/channel/item/2/link" use="optional">
<Comments>Today's forecast weather URL </Comment>
</Parameter>
<Parameter name="todayweather-description" dataType="multilineString" paramType="response"
ServiceMapping="/rss/channel/item/2/description" use="optional" containsHTML="true">
<Comments>Today's forecast weather description </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/item/2/description" name="todayweather-img-src"
use="optional" dataType="ImageURL" paramType="HTML EmbeddedMapping="img@src">
<Comments>Today's forecast weather embedded image src of the forecast gif </Comment>
</Parameter>
<!-- Tomorrow's forecast -->
<Parameter name="tomorrowweather" dataType="String" paramType="response" ServiceMapping="/
rss/channel/item/3/title" use="optional">
<Comments>Today's forecast weather </Comment>
</Parameter>
<Parameter name="tomorrowweather-link" dataType="String" paramType="response" ServiceMap-
ping="/rss/channel/item/3/link" use="optional">
<Comments>Today's forecast weather URL </Comment>
</Parameter>
<Parameter name="tomorrowweather-description" dataType="multilineString"
paramType="response" ServiceMapping="/rss/channel/item/3/description" use="optional"
containsHTML="true">
<Comments>Today's forecast weather description </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/item/3/description" name="tomorrowweather-img-src"
use="optional" dataType="ImageURL" paramType="HTML EmbeddedMapping="img@src">
<Comments>Today's forecast weather embedded image src of the forecast gif </Comment>
</Parameter>
</Outputs>
</WebComponentModel>

```

```

6. Yahoo Weather
<?xml version="1.0" encoding="utf-8" ?>
<WebComponentModel xmlns:csi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com/.webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" removeHTML="true" name="YahooWeather" group="Yahoo"
ServiceURL="http://weather.yahooapis.com/forecastrss">
<Managers>
<Manager type="qa" uri="http://qa.xpressmo.com/wcm/playerreq" />
<Manager type="dev" uri="http://dev.xpressmo.com/wcm/playerreq" />
<Manager type="local" uri="http://localhost:8080/wcm/playerreq" />
<Manager type="staging" uri="http://stg.xpressmo.com/wcm/playerreq" />
<Manager type="product" uri="http://csi.xpressmo.com/wcm/playerreq" />
</Managers>
<Namespaces>
<Namespace prefix="yweather" uri="http://xml.weather.yahoo.com/rss/1.0" />
<Namespace prefix="geo" uri="http://www.w3.org/2003/01/geo/wgs84_pos#" />
</Namespaces>
<Inputs>
<Parameter ServiceMapping="p" name="zip" use="required" dataType="Int">
<Comments>The ZIP code to get weather </Comment>
</Parameter>
</Inputs>
<Outputs>
<Parameter name="channel-title" dataType="ChannelTitle" paramType="response" ServiceMap-
ping="/rss/channel/title" use="optional">
<Comments>Channel title with Location selected </Comment>
</Parameter>
<Parameter name="channel-description" dataType="ChannelDescription" paramType="response"
ServiceMapping="/rss/channel/description" use="optional">
<Comments>Channel description with Location selected </Comment>
</Parameter>
<Parameter name="channel-image-uri" dataType="ChannelImageURL" paramType="response" Ser-
viceMapping="/rss/channel/image-uri" use="optional">
<Comments>RSS Channel Image URL </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/weather/location/@city" name="city" use="optional"
dataType="String">
<Comments>The Location City </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/weather/units/@temperature" name="units-temp"
use="optional" dataType="String">
<Comments>The units of the temperature, C or F </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/weather/atmosphere/@humidity" name="atm-humid-
ity" use="optional" dataType="String">
<Comments>The atmosphere humidity </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/weather/atmosphere/@pressure" name="pressure"
use="optional" dataType="String">
<Comments>The atmosphere pressure </Comment>
</Parameter>
<Parameter ServiceMapping="/rss/channel/weather/units/@pressure" name="units-pressure"

```

```

use="optional" dataType="String" </
<Comments>The units of pressure </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/title" name="title" use="optional"
dataType="String" >
<Comments>The text "Conditions for (Location) at (Time)" </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/geo:lat" name="geo:lat" use="optional"
dataType="String" >
<Comments>The Latitude of the Location </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/geo:long" name="geo:long" use="optional"
dataType="String" >
<Comments>The Longitude of the Location </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/link" name="link" use="optional"
dataType="String" >
<Comments>The URL to the detail page </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:condition" name="weather:condi-
tion-text" use="optional" dataType="String" >
<Comments>The short text of the weather condition </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:condition" name="weather:condi-
tion-code" use="optional" dataType="String" >
<Comments>The numeric code of the weather condition </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:condition" name="weather:condi-
tion-temp" use="optional" dataType="String" >
<Comments>The temperature </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:condition" name="weather:condi-
tion-date" use="optional" dataType="String" >
<Comments>The date </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/description" name="weather-description"
use="optional" containsHTML="true" dataType="multilineString" >
<Comments>The HTML version of the weather condition </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[1]" name="today-forecast-
day" use="optional" dataType="String" >
<Comments>Today's day of week </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[1]" name="today-forecast-
date" use="optional" dataType="String" >
<Comments>Today's date </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[1]" name="today-forecast-
low" use="optional" dataType="String" >
<Comments>Today's forecast Low </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[1]" name="today-forecast-
high" use="optional" dataType="String" >

```

```

<Comment>Today's forecast High </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[1]" name="today-forecast-
text" use="optional" dataType="String" >
<Comments>Today's forecast text </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[1]" name="today-forecast-
code" use="optional" dataType="String" >
<Comments>Today's forecast code </Comment>
</Parameter>
<!-- tomorrow -->
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[2]" name="tomorrow-fore-
cast-day" use="optional" dataType="String" >
<Comments>tomorrow's day of week </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[2]" name="tomorrow-fore-
cast-date" use="optional" dataType="String" >
<Comments>tomorrow's date </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[2]" name="tomorrow-fore-
cast-low" use="optional" dataType="String" >
<Comments>tomorrow's forecast Low </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[2]" name="tomorrow-fore-
cast-high" use="optional" dataType="String" >
<Comments>tomorrow's forecast High </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[2]" name="tomorrow-fore-
cast-text" use="optional" dataType="String" >
<Comments>tomorrow's forecast text </Comment>
</Parameter>
<Parameter ServiceMappings="/rss/channel/item/weather:forecast[2]" name="tomorrow-
forecast-code" use="optional" dataType="String" >
<Comments>tomorrow's forecast code </Comment>
</Parameter>
</Output>
</WebComponentModel>

```

7: RSS Stock Quotes

```

<?xml version="1.0" encoding="utf-8" ?>
<WebComponentModel xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" removeHTML="true" name="Stocks" group="Example"
ServiceURL="http://www.quoters.com/quote.php?fmt=0&freq=0">
<Managers>
<Manager type="qa" url="http://qa.xpressmo.com/wcm/playerreq" />
<Manager type="dev" url="http://dev.xpressmo.com/wcm/playerreq" />
<Manager type="local" url="http://localhost:8080/wcm/playerreq" />
<Manager type="staging" url="http://stg.xpressmo.com/wcm/playerreq" />
<Manager type="product" url="http://stg.xpressmo.com/wcm/playerreq" />
</Managers>
</Outputs>
<Parameter ServiceMapping="symbol" name="symbol" use="required" dataType="String">
<Comments>The stock symbol code to get quote </Comment>
</Parameter>
</Inputs>
</Outputs>
<Parameter name="channel-title" dataType="ChannelTitle" paramType="response" ServiceMap-
ping="/rss/channel/title" use="optional">
<Comments>RSS Channel Title - symbol requested </Comment>
</Parameter>
<Parameter name="channel-description" dataType="ChannelDescription" paramType="response"
ServiceMapping="/rss/channel/description" use="optional">
<Comments>RSS Channel Description : quote RSS attribution </Comment>
</Parameter>
<List ServiceMapping="/rss/channel/item" name="issitems" controlParam="item-title">
<Parameter name="item-title" dataType="RSSList" paramType="response"
ServiceMapping="title" use="required">
<Comments>Main quote string </Comment>
</Parameter>
<Parameter name="item-link" dataType="String" paramType="response" ServiceMapping="link"
use="optional">
<Comments>quote url </Comment>
</Parameter>
<Parameter name="item-description" dataType="multilineString" paramType="response"
ServiceMapping="description" use="optional" containsHTML="true">
<Comments>quote description contains HTML </Comment>
</Parameter>
<Parameter ServiceMapping="description" name="change" use="optional" dataType="String"
paramType="EHTML" EmbeddedMapping="table/body/tr[1]/td[1]/tbody/tr[1]/td[1]/font">
<Comments>The change for this stock embedded in HTML </Comment>
</Parameter>
<Parameter ServiceMapping="description" name="percentChange" use="optional"
dataType="String" paramType="EHTML" EmbeddedMapping="table/body/tr[1]/td[1]/tbody/tr[1]/td[1]/font">
<Comments>The percent change for this stock, embedded in HTML. </Comment>
</Parameter>
</List>
</Outputs>

```

8: Map Quest

```

<?xml version="1.0" encoding="utf-8" ?>
<WebComponentModel xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" ServiceURL="action" group="AOL" name="MapQuestCluster">
<Managers>
<Manager type="qa" url="http://qa.xpressmo.com/wcm/playerreq" />
<Manager type="dev" url="http://dev.xpressmo.com/wcm/playerreq" />
<Manager type="local" url="http://localhost:8080/wcm/playerreq" />
<Manager type="staging" url="http://stg.xpressmo.com/wcm/playerreq" />
<Manager type="product" url="http://stg.xpressmo.com/wcm/playerreq" />
</Managers>
</Actions>
</Outputs>
<Action name="panN" class="com.xpressmo.webcomponents.mapquest.MapClusterAction" />
<Action name="panS" class="com.xpressmo.webcomponents.mapquest.MapClusterAction" />
<Action name="panW" class="com.xpressmo.webcomponents.mapquest.MapClusterAction" />
<Action name="panE" class="com.xpressmo.webcomponents.mapquest.MapClusterAction" />
<Action name="getPOI" class="com.xpressmo.webcomponents.mapquest.MapClusterAction" />
<Action name="getMapCluster" class="com.xpressmo.webcomponents.mapquest.MapClusterAction"
/ >
</Actions>
</Outputs>
<Parameter ServiceMapping="TXID" name="TXID" dataType="String" use="optional">
<Comments>A transaction ID that the player uses when it needs to batch requests in session
(needs new data type possibly) </Comment>
</Parameter>
<Parameter ServiceMapping="zip" name="zip" dataType="String" use="required">
<Comments>zip code for map </Comment>
</Parameter>
<Parameter ServiceMapping="street" name="street" dataType="String" use="optional">
<Comments>Street address for map </Comment>
</Parameter>
<Parameter name="POI" data type="MultiSelectList" use="optional" multiSe-
lectDelimiters="|">
<Comments>Points of interest to use for the getPOI action </Comment>
<ListValues value="3014">GAS STATION </ListValues>
<ListValues value="3016">RESTAURANT </ListValues>
<ListValues value="3021">HOTEL/GUEST HOUSE </ListValues>
<ListValues value="3025">PARKING LOT </ListValues>
<ListValues value="3029">CINEMA </ListValues>
<ListValues value="3002">ATM </ListValues>
<ListValues value="3010">AIRPORT </ListValues>
<ListValues value="3005">BUS STATION </ListValues>
<ListValues value="3009">TRAIN STATION </ListValues>
</Parameter>
<Parameter ServiceMapping="height" name="height" dataType="Int" use="required">
<Comments>The returned image height </Comment>
</Parameter>
<Parameter ServiceMapping="width" name="width" dataType="Int" use="required">
<Comments>The returned image width </Comment>
</Parameter>
<Parameter ServiceMapping="mapScale" name="mapScale" dataType="PipelineMultipleSelect"

```

```

use="required" defaultvalue="46000" multiselectdelimit="|">
<Comments>Map scale is defined as the proportion of map units to real-world units. For example,
if a map's scale is expressed as 10,000, this means a scale ratio of 1:10,000, meaning one map
unit represents 10,000 realworld units. On a map where one inch equals 5 miles, the scale ratio
is 1:16,800 because there are 316,800 inches in 5 miles. A typical city-level map might have
a scale of 1:24,000, while a typical state-level map might have a scale of 1:2,000,000. </Com-
ments>
<ListValues value="6000">Street 1 </ListValues>
<ListValues value="12000">Street 2 </ListValues>
<ListValues value="24000">City 1 </ListValues>
<ListValues value="48000">City 2 </ListValues>
<ListValues value="96000">City 3 </ListValues>
<ListValues value="192000">County 1 </ListValues>
<ListValues value="400000">County 2 </ListValues>
<ListValues value="800000">County 3 </ListValues>
<ListValues value="1600000">State 1 </ListValues>
<ListValues value="3000000">State 2 </ListValues>
<ListValues value="6000000">State 3 </ListValues>
<ListValues value="12000000">Country 1 </ListValues>
<ListValues value="24000000">Country 2 </ListValues>
<ListValues value="46000000">Continent </ListValues>
</Parameter>
<Parameter ServiceMapping="Lat" name="Lat" datatype="String" use="optional">
<Comments>The latitude to use instead of address </Comment>
</Parameter>
<Parameter ServiceMapping="Lng" name="Lng" datatype="String" use="optional">
<Comments>The longitude to use instead of address </Comment>
</Parameter>
</Inputs>
</Outputs>
<Parameter ServiceMapping="Lat" name="Lat" datatype="String" use="optional">
<Comments>The latitude returned from the getmap </Comment>
</Parameter>
<Parameter ServiceMapping="Lng" name="Lng" datatype="String" use="optional">
<Comments>The longitude returned from the getmap </Comment>
</Parameter>
<Parameter ServiceMapping="POI-MapImage" name="POI-MapImage" datatype="ImageData"
use="optional">
<Comments>The image data base 64 encoded for a POI map </Comment>
</Parameter>
<Parameter ServiceMapping="POI-MapURL" name="POI-MapURL" datatype="ImageURL"
use="required">
<Comments>The output map URL for a POI map </Comment>
</Parameter>
<List ServiceMapping="mapList" name="mapList">
<!-- for clustering requests and outputs to reduce net usage -->
<Parameter ServiceMapping="mapurl" name="mapurl" datatype="ImageURL" use="required">
<Comments>The output map URL </Comment>
</Parameter>
<Parameter ServiceMapping="mapscaleforURL" name="mapscaleforURL" data type="String"
use="required">
<Comments>This is the mapscale that the corresponding map URL is for </Comment>
</Parameter>

```

```

<Parameter ServiceMapping="mapImage" name="mapImage" datatype="ImageListData"
use="optional">
<Comments>The image data base 64 encoded </Comment>
</Parameter>
</List>
<List ServiceMapping="POIdatalist" name="POIdatalist" controlParam="name">
use="optional">
<Comments>The point of interest Name </Comment>
</Parameter>
<Parameter ServiceMapping="pointKey" name="pointKey" datatype="String" use="optional">
<Comments>The point of interest Key </Comment>
</Parameter>
<Parameter ServiceMapping="pointDistance" name="pointDistance" datatype="String"
use="optional">
<Comments>The point of interest distance in miles </Comment>
</Parameter>
<Parameter ServiceMapping="pointAddress" name="pointAddress" datatype="String"
use="optional">
<Comments>The street address for the point of interest </Comment>
</Parameter>
<Parameter ServiceMapping="pointPhoneNumber" name="pointPhoneNumber"
datatype="String" use="optional">
<Comments>The phone number for the point of interest </Comment>
</Parameter>
</List>
</Outputs>
</WebComponentModel>

```

9: ThumbPlay Storefront

```

<?xml version="1.0" encoding="utf-8"?>
<WebComponentModel xmlns:sei="http://www3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressiro.com./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressiro.com" ServiceURL="http://api.hosted.thumbplay.com/api/est/search/cont
ent?name=1244&amp;pageSize=10" group="ThumbPlay" name="Search">
<Managers>
<Manager type="qa" uri="http://qa.xpressiro.com/wcm/playereq" />
<Manager type="dev" uri="http://dev.xpressiro.com/wcm/playereq" />
<Manager type="local" uri="http://localhost:8080/wcm/playereq" />
<Manager type="staging" uri="http://stg.xpressiro.com/wcm/playereq" />
<Manager type="product" uri="http://cat.xpressiro.com/wcm/playereq" />
</Managers>
<NameSpaces>
<NameSpace prefix="ns2" uri="http://search.serviceapi.thumbplay.com" />
</NameSpaces>
<Inputs>
<Parameter ServiceMapping="keywords" name="keywords" dataType="String" use="required">
<Comments>The search keywords</Comment>
</Parameter>
<Parameter ServiceMapping="contentType" name="contentType" dataType="MultiSelectList"
use="required" multiselectDelimit="|">
<Comments>The (0) separated values, One of: Music ringtone, Ringtone, Wallpaper, Animation,
Game, Video</Comment>
<ListValues value="Music ringtone">Music ringtone</ListValues>
<ListValues value="Ringtone">Ringtone</ListValues>
<ListValues value="Wallpaper">Wallpaper</ListValues>
<ListValues value="Animation">Animation</ListValues>
<ListValues value="Game">Game</ListValues>
<ListValues value="Video">Video</ListValues>
</Parameter>
</Inputs>
<Outputs>
<List ServiceMapping="//ns2:findContentResponses/contentResults/contentResult"
name="resultList" controlParam="title">
<Parameter ServiceMapping="albumName" name="albumName" dataType="String"
use="optional">
<Comments>The album name</Comment>
</Parameter>
<Parameter ServiceMapping="artist" name="artist" dataType="String" use="optional">
<Comments>The artist name</Comment>
</Parameter>
<Parameter ServiceMapping="artistId" name="artistId" dataType="String" use="optional">
<Comments>The artist ID</Comments>
</Parameter>
<Parameter ServiceMapping="contentId" name="contentId" dataType="String" use="optional">
<Comments>The content ID</Comments>
</Parameter>
<Parameter ServiceMapping="contentType" name="contentType" dataType="String"
use="optional">
<Comments>The content type</Comment>
</Parameter>

```

```

<Parameter ServiceMapping="imageSmallURL" name="imageSmallURL" dataType="ImageURL"
use="optional">
<Comments>The small image URL 75x75 jpg</Comment>
</Parameter>
<Parameter ServiceMapping="imageMediumURL" name="imageMediumURL"
dataType="ImageURL" use="optional">
<Comments>The medium image URL 150x150 jpg</Comment>
</Parameter>
<Parameter ServiceMapping="mobilePurchaseURL" name="mobilePurchaseURL"
dataType="PurchaseURL" use="optional">
<Comments>The mobile (WAP) purchase page URL</Comment>
</Parameter>
<Parameter ServiceMapping="previewURL" name="previewURL" dataType="AudioURL"
use="optional">
<Comments>The mp3 preview URL, if ringtones found</Comment>
</Parameter>
<Parameter ServiceMapping="title" name="title" dataType="ComplexList" use="optional">
<Comments>The content title</Comment>
</Parameter>
</List>
</Outputs>
</WebComponentModel>

```

10: Facebook

```
<?xml version="1.0" encoding="utf-8"?>
<WebComponentModel xmlns:sei="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://webcomponents.xpressmo.com/./webcomponentmodel.xsd" xmlns="http://
webcomponents.xpressmo.com" ServiceURL="action" group="Facebook" name="FBActions" >
<Managers>
<Manager type="qa" uri="http://qa.xpressmo.com/wcm/playerreq" />
<Manager type="dev" uri="http://dev.xpressmo.com/wcm/playerreq" />
<Manager type="local" uri="http://localhost:8080/wcm/playerreq" />
<Manager type="staging" uri="http://stg.xpressmo.com/wcm/playerreq" />
<Manager type="product" uri="http://us.xpressmo.com/wcm/playerreq" />
</Managers>
</Actions>
<Action name="getFriends" class="com.xpressmo.webcomponents.facebook.FacebookAction" />
<Action name="setStatus" class="com.xpressmo.webcomponents.facebook.FacebookAction" />
<Action name="sendIM" class="com.xpressmo.webcomponents.facebook.FacebookAction" />
</Actions>
</Inputs>
<Parameter ServiceMapping="fbSession" name="fbSession" dataType="String" use="required" />
<Comments>The infinite session key from Facebook for our application for the test user </Com-
ment>
</Parameter>
</Parameter>
<Parameter ServiceMapping="IM" name="IM" dataType="String" use="optional" />
<Comments>The instant message text to send for sendIM </Comment>
</Parameter>
<Parameter ServiceMapping="uid" name="uid" dataType="Int" use="optional" />
<Comments>The Facebook user id to send a message to for sendIM </Comment>
</Parameter>
<Parameter ServiceMapping="statusMessage" name="statusMessage" dataType="String"
use="optional" />
<Comments>The status message text to set for setStatus </Comment>
</Parameter>
</Inputs>
</Outputs>
<List ServiceMapping="friendList" name="friendList" controlId="name" uid" />
<Parameter ServiceMapping="uid" name="uid" dataType="Int" use="optional" />
<Comments>The Facebook uid for that friend </Comment>
</Parameter>
<Parameter ServiceMapping="name" name="name" dataType="String" use="optional" />
<Comments>The friend's name </Comment>
</Parameter>
<Parameter ServiceMapping="pic" name="pic" dataType="ImageURL" use="optional" />
<Comments>The URL to the friend's thumbnail JPG </Comment>
</Parameter>
<Parameter ServiceMapping="status" name="status" dataType="String" use="optional" />
<Comments>The friend's status message, may be blank </Comment>
</Parameter>
</List>
</Outputs>
</WebComponentModel>
```

How to create a Web Component Model

REST XML Example

Get the schema XSD webcomponentmodel.xsd to validate your model XML from the installation at Publishing-Server/XSD.

Load into Eclipse or Oxygen IDE

Create a new XML file and associate the schema

Add the name, group schema attributes and namespace

You should have an example of the feed XML that is returned also to construct the XPath mappings

Setup a model for the USA Today news RSS feed. Add the Feed URL as the Service URL, add the Managers and Manager Elements as the URLs to the web component manager you are using.

```
<?xml version="1.0" encoding="utf-8"?>
<WebComponentModel xmlns:sei="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://webcomponents.xpressmo.com" ServiceURL="http://www.usatoday.com/usatoday-News/TopStories"
name="Top" group="WebComponentManager" />
</Managers>
</Actions>
<Action name="getServiceURL" class="com.xpressmo.webcomponents.usatoday-News/TopStories" />
</Actions>
</Inputs>
</Outputs>
<List ServiceMapping="feed" name="feed" dataType="String" use="optional" />
<Parameter ServiceMapping="url" name="url" dataType="String" use="optional" />
<Comments>The feed URL for the feed </Comment>
</Parameter>
<Parameter ServiceMapping="product" name="product" dataType="String" use="optional" />
<Comments>The product name for the feed </Comment>
</Parameter>
</List>
</WebComponentModel>
```

How to use the Web Component Manager  
<http://rssfeeds.usatoday.com/usatoday-News/TopStories>

For this service there are no inputs, it is just a RSS feed.

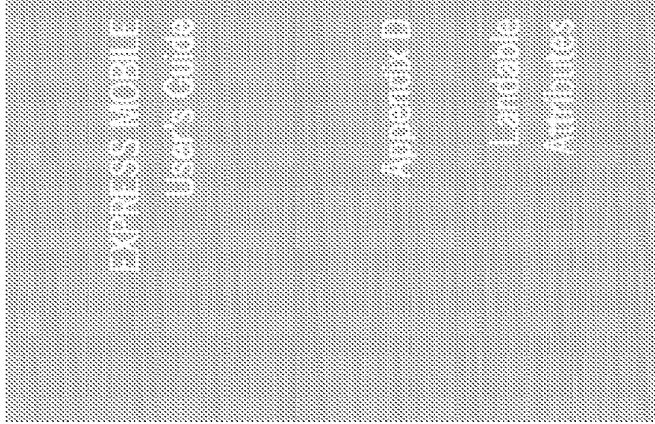
```
<?xml version="1.0" encoding="utf-8"?>
<WebComponentModel xmlns:sei="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://webcomponents.xpressmo.com" ServiceURL="http://www.usatoday.com/usatoday-News/TopStories"
name="RSS1" group="Facebook" />
</Managers>
</Actions>
<Action name="getServiceURL" class="com.xpressmo.webcomponents.usatoday-News/TopStories" />
</Actions>
</Inputs>
</Outputs>
<List ServiceMapping="feed" name="feed" dataType="String" use="optional" />
<Parameter ServiceMapping="url" name="url" dataType="String" use="optional" />
<Comments>The feed URL for the feed </Comment>
</Parameter>
<Parameter ServiceMapping="product" name="product" dataType="String" use="optional" />
<Comments>The product name for the feed </Comment>
</Parameter>
</List>
</WebComponentModel>
```











1. All Parent Video Objects
2. All Parent Slide Show Objects
3. GIS Object (N/A)
4. Virtual Tour Object (N/A)
5. All Form Objects with the following exceptions:
  - a. Text Field with non-Selectable attribute (ObjectBooleanAttribute[3]=false)
  - b. Text Area with non-Selectable attribute (ObjectBooleanAttribute[3]=false)
6. All Parent Objects that have one or more Child Objects
7. All Parent Objects with an exit Event (ObjectIntegerAttribute[12] > 0)
8. All Parent Objects with a Selection Cursor Defined (ObjectIntegerAttribute[4]>0)
9. All Parent Objects with a FRE Cursor Defined (ObjectIntegerAttribute[15]>0)
10. All Parent Objects with a Selection Audio Defined (ObjectIntegerAttribute[16]>0)
11. All Parent Objects with a FRE Audio Defined (ObjectIntegerAttribute[17]>0)
12. All Parent Objects with TimeLine Events (ObjectBooleanAttribute[6])
13. All Parent Objects with Draggable Attribute (ObjectBooleanAttribute[8])
14. All Parent Objects with Submit Form Attribute (ObjectBooleanAttribute[16])
15. All Objects that have a web component attached to them

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR2
		Application Number	
Title of Invention	Systems and Methods for Presenting Information on Mobile Devices		
The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76. This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.			

**Secrecy Order 37 CFR 5.2**

Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

**Applicant Information:**

<b>Applicant 1</b>						<a href="#">Remove</a>	
<b>Applicant Authority</b> <input checked="" type="radio"/> Inventor		<input type="radio"/> Legal Representative under 35 U.S.C. 117		<input type="radio"/> Party of Interest under 35 U.S.C. 118			
<b>Prefix</b>	<b>Given Name</b>	<b>Middle Name</b>		<b>Family Name</b>		<b>Suffix</b>	
	Steven	H.		Rempell			
<b>Residence Information (Select One)</b> <input checked="" type="radio"/> US Residency <input type="radio"/> Non US Residency <input type="radio"/> Active US Military Service							
<b>City</b>	Novato	<b>State/Province</b>	CA	<b>Country of Residence<sup>i</sup></b>		US	
<b>Citizenship under 37 CFR 1.41(b)<sup>i</sup></b>		US					
<b>Mailing Address of Applicant:</b>							
<b>Address 1</b>		38 Washington Street					
<b>Address 2</b>							
<b>City</b>	Novato	<b>State/Province</b>		CA			
<b>Postal Code</b>	94947	<b>Country<sup>i</sup></b>	US				
<b>Applicant 2</b>						<a href="#">Remove</a>	
<b>Applicant Authority</b> <input checked="" type="radio"/> Inventor		<input type="radio"/> Legal Representative under 35 U.S.C. 117		<input type="radio"/> Party of Interest under 35 U.S.C. 118			
<b>Prefix</b>	<b>Given Name</b>	<b>Middle Name</b>		<b>Family Name</b>		<b>Suffix</b>	
	David			Chrobak			
<b>Residence Information (Select One)</b> <input checked="" type="radio"/> US Residency <input type="radio"/> Non US Residency <input type="radio"/> Active US Military Service							
<b>City</b>	Pleasant Hill	<b>State/Province</b>	CA	<b>Country of Residence<sup>i</sup></b>		US	
<b>Citizenship under 37 CFR 1.41(b)<sup>i</sup></b>		US					
<b>Mailing Address of Applicant:</b>							
<b>Address 1</b>		132 Shadowood Drive					
<b>Address 2</b>							
<b>City</b>	Pleasant Hill	<b>State/Province</b>		CA			
<b>Postal Code</b>	94523	<b>Country<sup>i</sup></b>	US				
<b>Applicant 3</b>						<a href="#">Remove</a>	
<b>Applicant Authority</b> <input checked="" type="radio"/> Inventor		<input type="radio"/> Legal Representative under 35 U.S.C. 117		<input type="radio"/> Party of Interest under 35 U.S.C. 118			
<b>Prefix</b>	<b>Given Name</b>	<b>Middle Name</b>		<b>Family Name</b>		<b>Suffix</b>	
	Ken			Brown			
<b>Residence Information (Select One)</b> <input checked="" type="radio"/> US Residency <input type="radio"/> Non US Residency <input type="radio"/> Active US Military Service							
<b>City</b>	San Martin	<b>State/Province</b>	CA	<b>Country of Residence<sup>i</sup></b>		US	

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR2	
		Application Number		
Title of Invention	Systems and Methods for Presenting Information on Mobile Devices			
<b>Citizenship under 37 CFR 1.41(b) i</b>		US		
<b>Mailing Address of Applicant:</b>				
Address 1	2485 Church Avenue			
Address 2				
City	San Martin	State/Province	CA	
Postal Code	95046	Country <sup>i</sup>	US	
All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the <b>Add</b> button.				<input type="button" value="Add"/>

**Correspondence Information:**

Enter either Customer Number or complete the Correspondence Information section below. For further information see 37 CFR 1.33(a).			
<input type="checkbox"/> An Address is being provided for the correspondence Information of this application.			
Customer Number	40280		
Email Address	svosen@phdpatents.com	<input type="button" value="Add Email"/>	<input type="button" value="Remove Email"/>

**Application Information:**

Title of the Invention	Systems and Methods for Presenting Information on Mobile Devices		
Attorney Docket Number	XPR.002PR2	Small Entity Status Claimed	<input checked="" type="checkbox"/>
Application Type	Provisional		
Subject Matter	Utility		
Suggested Class (if any)		Sub Class (if any)	
Suggested Technology Center (if any)			
Total Number of Drawing Sheets (if any)		Suggested Figure for Publication (if any)	

**Publication Information:**

<input type="checkbox"/> Request Early Publication (Fee required at time of Request 37 CFR 1.219)
<input type="checkbox"/> <b>Request Not to Publish.</b> I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application <b>has not and will not</b> be the subject of an application filed in another country, or under a multilateral international agreement, that requires publication at eighteen months after filing.

**Representative Information:**

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32). Enter either Customer Number or complete the Representative Name section below. If both sections are completed the Customer Number will be used for the Representative Information during processing.			
Please Select One:	<input checked="" type="radio"/> Customer Number	<input type="radio"/> US Patent Practitioner	<input type="radio"/> Limited Recognition (37 CFR 11.9)

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR2
		Application Number	
Title of Invention	Systems and Methods for Presenting Information on Mobile Devices		
Customer Number	40280		

### Domestic Benefit/National Stage Information:

This section allows for the applicant to either claim benefit under 35 U.S.C. 119(e), 120, 121, or 365(c) or indicate National Stage entry from a PCT application. Providing this information in the application data sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78(a)(2) or CFR 1.78(a)(4), and need not otherwise be made part of the specification.			
Prior Application Status			<input type="button" value="Remove"/>
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)
Additional Domestic Benefit/National Stage Data may be generated within this form by selecting the <b>Add</b> button.			<input type="button" value="Add"/>

### Foreign Priority Information:

This section allows for the applicant to claim benefit of foreign priority and to identify any prior foreign application for which priority is not claimed. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55(a).			
			<input type="button" value="Remove"/>
Application Number	Country <sup>i</sup>	Parent Filing Date (YYYY-MM-DD)	Priority Claimed
			<input type="radio"/> Yes <input checked="" type="radio"/> No
Additional Foreign Priority Data may be generated within this form by selecting the <b>Add</b> button.			<input type="button" value="Add"/>

### Assignee Information:

Providing this information in the application data sheet does not substitute for compliance with any requirement of part 3 of Title 37 of the CFR to have an assignment recorded in the Office.				
<b>Assignee 1</b>				<input type="button" value="Remove"/>
If the Assignee is an Organization check here. <input type="checkbox"/>				
Prefix	Given Name	Middle Name	Family Name	Suffix
<b>Mailing Address Information:</b>				
Address 1				
Address 2				
City		State/Province		
Country <sup>i</sup>		Postal Code		
Phone Number		Fax Number		
Email Address				
Additional Assignee Data may be generated within this form by selecting the <b>Add</b> button.				<input type="button" value="Add"/>

### Signature:

IPR2021-01146 Page 00952

A signature of the applicant or representative is required in accordance with 37 CFR 1.33 and 10.18. Please see 37 CFR 1.4(d) for the form of the signature.



Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

<b>Application Data Sheet 37 CFR 1.76</b>		Attorney Docket Number	XPR.002PR2
		Application Number	
Title of Invention	Systems and Methods for Presenting Information on Mobile Devices		

<b>Signature</b>	/Steven R. Vosen/		<b>Date (YYYY-MM-DD)</b>	2009-04-03	
<b>First Name</b>	Steven	<b>Last Name</b>	Vosen	<b>Registration Number</b>	45186

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

## Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

## Electronic Acknowledgement Receipt

<b>EFS ID:</b>	5076389
<b>Application Number:</b>	61166651
<b>International Application Number:</b>	
<b>Confirmation Number:</b>	8595
<b>Title of Invention:</b>	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES
<b>First Named Inventor/Applicant Name:</b>	Steven H. Rempell
<b>Customer Number:</b>	40280
<b>Filer:</b>	Steven R Vosen/Tony Freitas
<b>Filer Authorized By:</b>	Steven R Vosen
<b>Attorney Docket Number:</b>	XPR.002PR2
<b>Receipt Date:</b>	03-APR-2009
<b>Filing Date:</b>	
<b>Time Stamp:</b>	19:37:22
<b>Application Type:</b>	Provisional

### Payment information:

Submitted with Payment	no
------------------------	----

### File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Specification	XPR002PR3_As_Filed.pdf	22100091 <small>2904cda4706a8aa02a1fc5837335ed14557ec0e7</small>	no	177

### Warnings:

### Information:

2	Provisional Cover Sheet (SB16)	ProvisionalCoverfsheet.pdf	999554 0630284e873ecc051b6111e12904d1bee3eef10	no	3
<b>Warnings:</b>					
<b>Information:</b>					
3	Application Data Sheet	XPR002PR2ADS.pdf	964122 9df4349a2c158b0ac449d82daaf98b50b56e04ae	no	5
<b>Warnings:</b>					
<b>Information:</b>					
4	Fee Worksheet (PTO-06)	fee-info.pdf	31397 05b46c8daa696570d0b663cac37d7ba7d690bbe83	no	2
<b>Warnings:</b>					
<b>Information:</b>					
<b>Total Files Size (in bytes):</b>			24095164		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

**New Applications Under 35 U.S.C. 111**

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

**National Stage of an International Application under 35 U.S.C. 371**

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

**New International Application Filed with the USPTO as a Receiving Office**

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

**PATENT APPLICATION FEE DETERMINATION RECORD**

Effective October 2, 2008

Application or Docket Number

12/936395

**CLAIMS AS FILED - PART I**

	(Column 1)	(Column 2)
U.S. NATIONAL STAGE FEES		
BASIC FEE	SMALL ENT. = \$165	LARGE ENT. = \$330
EXAMINATION FEE	SMALL ENT. = \$110	LARGE ENT. = \$220
SEARCH FEE	U.S. is ISA = \$50/\$100 ALL other countries = \$215/\$430	ALL other situations = \$270/\$550
FEE FOR EXTRA SPEC. PGS.	minus 100 =	/ 50 =
TOTAL CHARGEABLE CLAIMS	28 minus 20 =	8
INDEPENDENT CLAIMS	3 minus 3 =	
MULTIPLE DEPENDENT CLAIM PRESENT		

SMALL ENTITY		OR	LARGE ENTITY	
RATE	FEE		RATE	FEE
BASIC FEE	165	OR	BASIC FEE	
EXAM. FEE	110		EXAM. FEE	
SEARCH FEE	215		SEARCH FEE	
X \$ 135 =			X \$ 270 =	
X \$ 26 =	208	OR	X \$ 52 =	
X \$ 110 =		OR	X \$ 220 =	
+ \$ 195 =		OR	+ \$ 390 =	
TOTAL		OR	TOTAL	

\* If the difference in column 1 is less than zero, enter "0" in column 2

**CLAIMS AS AMENDED - PART II**

	(Column 1)	(Column 2)	(Column 3)
AMENDMENT A	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA
	Total *	Minus **	=
	Independent *	Minus ***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM <input type="checkbox"/>			

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X \$ 26 =		OR	X \$ 52 =	
X \$ 110 =		OR	X \$ 220 =	
+ \$ 195 =		OR	+ \$ 390 =	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

	(Column 1)	(Column 2)	(Column 3)
AMENDMENT B	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA
	Total *	Minus **	=
	Independent *	Minus ***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM <input type="checkbox"/>			

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X \$ 26 =		OR	X \$ 52 =	
X \$ 110 =		OR	X \$ 220 =	
+ \$ 195 =		OR	+ \$ 390 =	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

\* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.  
 \*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than '20', enter "20".  
 \*\*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than '3', enter "3".  
 The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

MULTIPLE DEPENDENT CLAIM  
FEE CALCULATION SHEET  
(FOR USE WITH FORM PTO-875)

SERIAL NO.

102 936 395

FILING DATE

APPLICANT(S)

CLAIMS

	AS FILED		AFTER 1 <sup>st</sup> AMENDMENT		AFTER 2 <sup>nd</sup> AMENDMENT	
	IND.	DEP.	IND.	DEP.	IND.	DEP.
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						
26						
27						
28						
29						
30						
31						
32						
33						
34						
35						
36						
37						
38						
39						
40						
41						
42						
43						
44						
45						
46						
47						
48						
49						
50						
TOTAL IND.		↓	3	↓		↓
TOTAL DEP.		←		←		←
TOTAL CLAIMS			28			

	AS FILED		AFTER 1 <sup>st</sup> AMENDMENT		AFTER 2 <sup>nd</sup> AMENDMENT	
	IND.	DEP.	IND.	DEP.	IND.	DEP.
51						
52						
53						
54						
55						
56						
57						
58						
59						
60						
61						
62						
63						
64						
65						
66						
67						
68						
69						
70						
71						
72						
73						
74						
75						
76						
77						
78						
79						
80						
81						
82						
83						
84						
85						
86						
87						
88						
89						
90						
91						
92						
93						
94						
95						
96						
97						
98						
99						
100						
TOTAL IND.		↓		↓		↓
TOTAL DEP.		←		←		←
TOTAL CLAIMS						

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

<b>PATENT APPLICATION FEE DETERMINATION RECORD</b> Substitute for Form PTO-875	Application or Docket Number <b>12/936,395</b>	Filing Date <b>11/03/2010</b>	<input type="checkbox"/> To be Mailed
---	---	----------------------------------	---------------------------------------

APPLICATION AS FILED – PART I			OTHER THAN SMALL ENTITY			
	(Column 1)	(Column 2)	SMALL ENTITY <input checked="" type="checkbox"/>	OR		
FOR	NUMBER FILED	NUMBER EXTRA	RATE (\$)	FEE (\$)	RATE (\$)	FEE (\$)
<input type="checkbox"/> BASIC FEE <small>(37 CFR 1.16(a), (b), or (c))</small>	N/A	N/A	N/A		N/A	
<input type="checkbox"/> SEARCH FEE <small>(37 CFR 1.16(k), (j), or (m))</small>	N/A	N/A	N/A		N/A	
<input type="checkbox"/> EXAMINATION FEE <small>(37 CFR 1.16(o), (p), or (q))</small>	N/A	N/A	N/A		N/A	
TOTAL CLAIMS <small>(37 CFR 1.16(j))</small>	minus 20 =	*	X \$ =	OR	X \$ =	
INDEPENDENT CLAIMS <small>(37 CFR 1.16(h))</small>	minus 3 =	*	X \$ =		X \$ =	
<input type="checkbox"/> APPLICATION SIZE FEE <small>(37 CFR 1.16(s))</small>	If the specification and drawings exceed 100 sheets of paper, the application size fee due is \$250 (\$125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).					
<input type="checkbox"/> MULTIPLE DEPENDENT CLAIM PRESENT <small>(37 CFR 1.16(j))</small>						
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL		TOTAL	

APPLICATION AS AMENDED – PART II					OTHER THAN SMALL ENTITY			
	(Column 1)	(Column 2)	(Column 3)					
AMENDMENT	<b>10/04/2010</b>	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE (\$)	ADDITIONAL FEE (\$)	RATE (\$)	ADDITIONAL FEE (\$)
	Total <small>(37 CFR 1.16(i))</small>	* 28	Minus ** 28	= 0	X \$26 =	0	OR	X \$ =
	Independent <small>(37 CFR 1.16(h))</small>	* 3	Minus *** 3	= 0	X \$110 =	0	OR	X \$ =
	<input type="checkbox"/> Application Size Fee <small>(37 CFR 1.16(s))</small>						OR	
	<input type="checkbox"/> FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM <small>(37 CFR 1.16(j))</small>						OR	
					TOTAL ADD'L FEE	<b>0</b>	OR	TOTAL ADD'L FEE

	(Column 1)	(Column 2)	(Column 3)					
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE (\$)	ADDITIONAL FEE (\$)	RATE (\$)	ADDITIONAL FEE (\$)
	Total <small>(37 CFR 1.16(i))</small>	*	Minus **	=	X \$ =		OR	X \$ =
	Independent <small>(37 CFR 1.16(h))</small>	*	Minus ***	=	X \$ =		OR	X \$ =
	<input type="checkbox"/> Application Size Fee <small>(37 CFR 1.16(s))</small>						OR	
	<input type="checkbox"/> FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM <small>(37 CFR 1.16(j))</small>						OR	
					TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE

\* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.  
 \*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".  
 \*\*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".  
 The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:  
/LYNN NELSON/

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Table with 3 columns: U.S. APPLICATION NUMBER NO. (12/936,395), FIRST NAMED APPLICANT (Steven H. Rempell), ATTY. DOCKET NO. (XPR.002US0)

40280
STEVEN VOSEN
1563 SOLANO AVENUE #206
BERKELEY, CA 94707

Table with 2 columns: INTERNATIONAL APPLICATION NO. (PCT/US09/39695), I.A. FILING DATE (04/06/2009), PRIORITY DATE (04/07/2008)

CONFIRMATION NO. 2369
371 FORMALITIES LETTER



Date Mailed: 10/28/2010

NOTICE OF INSUFFICIENT BASIC NATIONAL FEE REQUIRED AND/OR MISSING COPY OF INTERNATIONAL APPLICATION UNDER 35 U.S.C 371 AND 37 CFR 1.495

Indication of Small Entity Status has not been received.

The current record of this application indicates that the basic national fee (37 CFR 1.492(a)) required is \$330.

The basic national fee to enter the national stage in the United States of America under 35 U.S.C. 371 must be paid by 30 months from the priority date (37 CFR 1.495(b)(2)).

If the proper basic national fee is not paid within 30 months from the priority date, the international application will become ABANDONED as to the United States of America, and will not be accepted for national examination.

The amount required shown above reflects the correct fee as of the date of this Notice. If the amount changes prior to payment, applicant must pay the revised amount.

The current basic national fees are listed at 37 CFR 1.492(a). In addition, current PCT related fees are listed in each weekly issue of the Official Gazette of the United States Patent and Trademark Office and the amount of the fees can be obtained from the USPTO 's web site www.uspto.gov or by contacting the Public Service Center at (703) 308-4357.

Extensions of time under the provisions of 37 CFR 1.136 are not available for compliance with the requirements of 37 CFR 1.495(b)(2).

For filing the basic national fee, the first class certificate of mailing procedure of 37 CFR 1.8 is NOT available to establish the date of mailing as the date of receipt in the USPTO (see 37 CFR 1.8(a)(2)(i)(F)). The Express Mail procedure of 37 CFR 1.10 may be used.

The basic national fee may not be submitted via facsimile transmission.

Applicant is reminded that any communications to the United States Patent and Trademark Office must be mailed to the address given in the heading and include the U.S. application no. shown above (37 CFR 1.5)

Registered users of EFS-Web may alternatively submit their reply to this notice via EFS-Web. https://portal.uspto.gov/authenticate/AuthenticateUserLocalEPF.html



For more information about EFS-Web please call the USPTO Electronic Business Center at **1-866-217-9197** or visit our website at <http://www.uspto.gov/ebc>.

**If you are not using EFS-Web to submit your reply, you must include a copy of this notice.**

LAMONT M HUNTER

---

Telephone: (571) 272-0308

**PETITION FOR REVIVAL OF AN INTERNATIONAL APPLICATION FOR PATENT  
DESIGNATING THE U.S. ABANDONED UNINTENTIONALLY UNDER 37 CFR 1.137(b)**Docket Number  
(Optional)  
XPR.002US0First Named Inventor: Steven H. RempellInternational (PCT) Application No.: PCT/US09/39695 U.S. Application No.: 12/936,395  
(if known)Filed: Int'l :4/6/09; US: 10/4/10

Title:

SYSTEMS AND METHODS FOR PROGRAMMING MOBILE DEVICES

Attention: PCT Legal Staff  
Mail Stop PCT  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

The above-identified application became abandoned as to the United States because the fees and documents required by 35 U.S.C. 371(c) were not filed prior to the expiration of the time set in 37 CFR 1.495(b) or (c) as applicable. The date of abandonment is the day after the date on which the 35 U.S.C. 371(c) requirements were due. See 37 CFR 1.495(h).

**APPLICANT HEREBY PETITIONS FOR REVIVAL OF THIS APPLICATION**

NOTE: A grantable petition requires the following items:

- (1) Petition fee
- (2) Proper reply
- (3) Terminal disclaimer with disclaimer fee which is required for all international applications having an international filing date before June 8, 1995; and
- (4) Statement that the entire delay was unintentional.

## 1. Petition fee

 Small entity - fee \$ 810.00 (37 CFR 1.17(m)). Applicant claims small entity status.  
See 37 CFR 1.27.

 Other than small entity - fee \$ \_\_\_\_\_ (37 CFR 1.17(m))

## 2. Proper reply

A. The proper reply (the missing 35 U.S.C. 371(c) requirement(s)) in the form of  
the small entity basic national fee (1.495(b)(2)) (identify type of reply):

 has been filed previously on \_\_\_\_\_ .

 is enclosed herewith.

[Page 1 of 2]

This collection of information is required by 37 CFR 1.137(b). The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 1.0 hour to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Mail Stop PCT, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

## 3. Terminal disclaimer with disclaimer fee

- Since this international application has an international filing date on or after June 8, 1995, no terminal disclaimer is required.
- A terminal disclaimer (and disclaimer fee (37 CFR 1.20(d)) of \$ \_\_\_\_\_ for a small entity or \$ \_\_\_\_\_ for other than a small entity) disclaiming the required period of time is enclosed herewith (see PTO/SB/63).

4. Statement. The entire delay in filing the required reply from the due date for the required reply until the filing of a grantable petition under 37 CFR 1.137(b) was unintentional.

**WARNING:**

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

/Steven R. Vosen/

11/4/2010

Signature

Date

Steven R. Vosen

45,186

Typed or Printed Name

Registration Number, if applicable

1563 Solano Ave., #206

510-841-4711

Address

Telephone Number

Berkeley, CA 94707

Address

Enclosures:  Response Fee Payment Terminal Disclaimer Other (please identify):

## Privacy Act Statement

The **Privacy Act of 1974 (P.L. 93-579)** requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (*i.e.*, GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Under the Paper Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

<b>TRANSMITTAL FORM</b>  <i>(to be used for all correspondence after initial filing)</i>	Application Number	12/936,395	
	Filing Date	Oct 4, 2010	
	First Named Inventor	Rempell, Steven	
	Art Unit		
	Examiner Name		
Total Number of Pages in This Submission	3	Attorney Docket Number	XPR.002US0

ENCLOSURES <i>(Check all that apply)</i>				
<input checked="" type="checkbox"/> Fee Transmittal Form <input checked="" type="checkbox"/> Fee Attached <input type="checkbox"/> Amendment / Reply <input type="checkbox"/> After Final <input type="checkbox"/> Affidavits/declaration(s) <input type="checkbox"/> Extension of Time Request <input type="checkbox"/> Express Abandonment Request <input type="checkbox"/> Information Disclosure Statement <input type="checkbox"/> Certified Copy of Priority Document(s) <input type="checkbox"/> Reply to Missing Parts/ Incomplete Application <input type="checkbox"/> Reply to Missing Parts under 37 CFR 1.52 or 1.53	<input type="checkbox"/> Drawing(s) <input type="checkbox"/> Licensing-related Papers <input checked="" type="checkbox"/> Petition <input type="checkbox"/> Petition to Convert to a Provisional Application <input type="checkbox"/> Power of Attorney, Revocation Change of Correspondence Address <input type="checkbox"/> Terminal Disclaimer <input type="checkbox"/> Request for Refund <input type="checkbox"/> CD, Number of CD(s) _____ <input type="checkbox"/> Landscape Table on CD	<input type="checkbox"/> After Allowance communication to TC <input type="checkbox"/> Appeal Communication to Board of Appeals and Interferences <input type="checkbox"/> Appeal Communication to TC <b>(Appeal Notice, Brief, Reply Brief)</b> <input type="checkbox"/> Proprietary Information <input type="checkbox"/> Status Letter <input type="checkbox"/> Other Enclosure(s) (please identify below):		
<table border="1" style="width: 100%;"> <tr> <td style="width: 20%;"><b>Remarks</b></td> <td>A petition to revive an unintentionally abandoned patent application for not paying the basic national stage fee, the petition fee and the required national stage fee are included herewith.</td> </tr> </table>			<b>Remarks</b>	A petition to revive an unintentionally abandoned patent application for not paying the basic national stage fee, the petition fee and the required national stage fee are included herewith.
<b>Remarks</b>	A petition to revive an unintentionally abandoned patent application for not paying the basic national stage fee, the petition fee and the required national stage fee are included herewith.			

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT			
Firm Name			
Signature	/Steven R. Vosen/		
Printed name	Steven R. Vosen		
Date	November 3, 2010	Reg. No.	45186

CERTIFICATE OF TRANSMISSION/MAILING			
I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below.			
Signature			
Typed or printed name	Steven Vosen	Date	November 3, 2010

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. **DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

*In you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.*

## Electronic Patent Application Fee Transmittal

<b>Application Number:</b>	12936395
<b>Filing Date:</b>	
<b>Title of Invention:</b>	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES
<b>First Named Inventor/Applicant Name:</b>	Steven H. Rempell
<b>Filer:</b>	Steven R Vosen
<b>Attorney Docket Number:</b>	XPR.002U50

Filed as Small Entity

### U.S. National Stage under 35 USC 371 Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD(\$)
<b>Basic Filing:</b>				
Basic National Stage Fee	2631	1	165	165
<b>Pages:</b>				
<b>Claims:</b>				
<b>Miscellaneous-Filing:</b>				
<b>Petition:</b>				
Petition-revive unintent. abandoned appl	2453	1	810	810
<b>Patent-Appeals-and-Interference:</b>				
<b>Post-Allowance-and-Post-Issuance:</b>				

Description	Fee Code	Quantity	Amount	Sub-Total in USD(\$)
<b>Extension-of-Time:</b>				
<b>Miscellaneous:</b>				
<b>Total in USD (\$)</b>				<b>975</b>

## Electronic Acknowledgement Receipt

<b>EFS ID:</b>	8759052
<b>Application Number:</b>	12936395
<b>International Application Number:</b>	
<b>Confirmation Number:</b>	2369
<b>Title of Invention:</b>	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES
<b>First Named Inventor/Applicant Name:</b>	Steven H. Rempell
<b>Customer Number:</b>	40280
<b>Filer:</b>	Steven R Vosen
<b>Filer Authorized By:</b>	
<b>Attorney Docket Number:</b>	XPR.002US0
<b>Receipt Date:</b>	03-NOV-2010
<b>Filing Date:</b>	
<b>Time Stamp:</b>	13:35:47
<b>Application Type:</b>	U.S. National Stage under 35 USC 371

### Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	\$975
RAM confirmation Number	9370
Deposit Account	
Authorized User	

### File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
-----------------	----------------------	-----------	----------------------------------	------------------	------------------



1	Petition for review and processing by the PCT legal office.	XPR002US0_Pet_to_Revive_v2.pdf	201011 75e2ca67ac075e7e028d75d1aab413c8c01d385c	no	3
<b>Warnings:</b>					
<b>Information:</b>					
2	Transmittal Letter	XPR002US0_Pet_Revive_Trans.pdf	82987 07d6d193fe4e26149f62816cb0c274b1f9acc41	no	1
<b>Warnings:</b>					
<b>Information:</b>					
3	Fee Worksheet (PTO-875)	fee-info.pdf	31989 1b35750791e112dabc3c78f330fd3d73adee2a64	no	2
<b>Warnings:</b>					
<b>Information:</b>					
<b>Total Files Size (in bytes):</b>				315987	

**This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.**

**New Applications Under 35 U.S.C. 111**

**If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.**

**National Stage of an International Application under 35 U.S.C. 371**

**If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.**

**New International Application Filed with the USPTO as a Receiving Office**

**If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.**



40280  
Steven Vosen  
1563 Solano Avenue #206  
Berkley, CA 94707

**MAILED**

**DEC 29 2010**

**PCT LEGAL ADMINISTRATION**

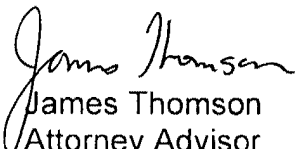
In re Application of :  
REPELL *et al* :  
U.S. Application No.: 12/936,395 :  
PCT No.: PCT/US2009/039695 :  
Int. Filing Date: 06 April 2009 :  
Priority Date: 07 April 2008 :  
Docket No.: XPR.002USO :  
For: SYSTEMS AND METHODS FOR :  
PRESENTING INFORMATION ON :  
MOBILE DEVICES :

**DECISION ON  
PETITION UNDER  
37 CFR 1.137(b)**

Applicants' petition to revive under 37 CFR 1.137(b) filed on 03 November 2010 is hereby **GRANTED** as follows:

The Basic National Stage Fee and the petition fee for a small entity have been paid. Applicants made the required statement pursuant to 37 CFR 1.137(b)(3). A terminal disclaimer is not required. Accordingly, all requirements under 37 CFR 1.137(b) have been satisfied.

This application is being forwarded to the United States Designated/Elected Office for further processing.

  
James Thomson  
Attorney Advisor  
Office of PCT Legal Administration

Tel.: (571) 272-3302



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Table with 3 columns: U.S. APPLICATION NUMBER NO. (12/936,395), FIRST NAMED APPLICANT (Steven H. Rempell), ATTY. DOCKET NO. (XPR.002US0)

40280
STEVEN VOSEN
1563 SOLANO AVENUE #206
BERKELEY, CA 94707

INTERNATIONAL APPLICATION NO.

PCT/US09/39695

Table with 2 columns: I.A. FILING DATE (04/06/2009), PRIORITY DATE (04/07/2008)

CONFIRMATION NO. 2369
371 FORMALITIES LETTER



Date Mailed: 01/25/2011

Notification of Insufficient Fees (DO/EO/US)

The following items have been submitted by the applicant or the IB to the United States Patent and Trademark Office as a Designated Office (37 CFR 1.494):

- Indication of Small Entity Status
• Priority Document
• Copy of the International Application filed on 10/04/2010
• Copy of the International Search Report filed on 10/04/2010
• Preliminary Amendments filed on 10/04/2010
• Oath or Declaration filed on 10/04/2010
• Small Entity Statement filed on 11/03/2010
• Request for Immediate Examination filed on 10/04/2010
• U.S. Basic National Fees filed on 11/03/2010
• Priority Documents filed on 10/04/2010
• Power of Attorney filed on 10/04/2010

The following items MUST be furnished within the period set forth below in order to complete the requirements for acceptance under 35 U.S.C. 371:

- Additional claim fees of \$208 as a small entity, including any required multiple dependent claim fee, are required. Applicant must submit the additional claim fees or cancel the additional claims for which fees are due.
• To avoid abandonment, a surcharge (for late submission of filing fee, search fee, examination fee or oath or declaration) as set forth in 37 CFR 1.492(h) of \$65 for a small entity in compliance with 37 CFR 1.27, must be submitted with the missing items identified in this letter.

SUMMARY OF FEES DUE:

Total additional fees required for this application is \$598 for a Small Entity:

- \$65 Surcharge.
• The application search fee has not been paid. Applicant must submit \$215 to complete the search fee. Note a surcharge will be required if submitted later than commencement of the national stage (37 CFR 1.492(h)) and the basic national fee was not paid before July 1, 2005.
• The application examination fee has not been paid. Applicant must submit \$110 to complete the examination fee for a small entity in compliance with 37 CFR 1.27. Note a surcharge will be required if submitted later than commencement of the national stage (37 CFR 1.492(h)) and the basic national fee was not paid before July 1, 2005.

Total additional claim fee(s) for this application is **\$208**  
• **\$208** for **8** total claims over 20.

**ALL OF THE ITEMS SET FORTH ABOVE MUST BE SUBMITTED WITHIN TWO (2) MONTHS FROM THE DATE OF THIS NOTICE OR BY 32 MONTHS FROM THE PRIORITY DATE FOR THE APPLICATION, WHICHEVER IS LATER. FAILURE TO PROPERLY RESPOND WILL RESULT IN ABANDONMENT.**

The time period set above may be extended by filing a petition and fee for extension of time under the provisions of 37 CFR 1.136(a).

Applicant is reminded that any communications to the United States Patent and Trademark Office must be mailed to the address given in the heading and include the U.S. application no. shown above (37 CFR 1.5)

Registered users of EFS-Web may alternatively submit their reply to this notice via EFS-Web.  
<https://portal.uspto.gov/authenticate/AuthenticateUserLocalEPF.html>

For more information about EFS-Web please call the USPTO Electronic Business Center at **1-866-217-9197** or visit our website at <http://www.uspto.gov/ebc>.

**If you are not using EFS-Web to submit your reply, you must include a copy of this notice.**

LAMONT M HUNTER

---

Telephone: (571) 272-0308

**PATENT APPLICATION FEE DETERMINATION RECORD**

Substitute for Form PTO-875

Application or Docket Number  
12/936,395

**APPLICATION AS FILED - PART I**

(Column 1) (Column 2)

FOR	NUMBER FILED	NUMBER EXTRA
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A
SEARCH FEE (37 CFR 1.16(k), (l), or (m))	N/A	N/A
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A
TOTAL CLAIMS (37 CFR 1.16(j))	28 minus 20 = *	8
INDEPENDENT CLAIMS (37 CFR 1.16(h))	3 minus 3 = *	
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is \$270 (\$135 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).	
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))		

\* If the difference in column 1 is less than zero, enter "0" in column 2.

**SMALL ENTITY**

RATE(\$)	FEE(\$)
N/A	165
N/A	215
N/A	110
x 26 =	208
x 110 =	0.00
	0.00
TOTAL	698

**OR OTHER THAN SMALL ENTITY**

RATE(\$)	FEE(\$)
N/A	
N/A	
N/A	
TOTAL	

**APPLICATION AS AMENDED - PART II**

(Column 1) (Column 2) (Column 3)

AMENDMENT A		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA
	Total (37 CFR 1.16(j))	*	Minus	**	=
	Independent (37 CFR 1.16(h))	*	Minus	***	=
	Application Size Fee (37 CFR 1.16(s))				
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					

**SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

**OR OTHER THAN SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

(Column 1) (Column 2) (Column 3)

AMENDMENT B		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA
	Total (37 CFR 1.16(j))	*	Minus	**	=
	Independent (37 CFR 1.16(h))	*	Minus	***	=
	Application Size Fee (37 CFR 1.16(s))				
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					

**SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

**OR OTHER THAN SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

\* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.

\*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".

\*\*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".

The "Highest Number Previously Paid For" (Total or Independent) is the highest found in the appropriate box in column 1.



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Table with 6 columns: APPLICATION NUMBER, FILING or 371(c) DATE, GRP ART UNIT, FIL FEE REC'D, ATTY. DOCKET NO, TOT CLAIMS, IND CLAIMS. Values: 12/936,395, 11/03/2010, 165, XPR.002US0, 28, 3

CONFIRMATION NO. 2369

40280
STEVEN VOSEN
1563 SOLANO AVENUE #206
BERKELEY, CA 94707

FILING RECEIPT



Date Mailed: 01/25/2011

Receipt is acknowledged of this non-provisional patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a Filing Receipt Correction. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections

Applicant(s)

Steven H. Rempell, Novato, CA;
David Chrobak, Clayton, CA;
Ken Brown, San Martin, CA;

Assignment For Published Patent Application

EXPRESS MOBILE INC., Novato, CA

Power of Attorney: The patent practitioners associated with Customer Number 40280

Domestic Priority data as claimed by applicant

This application is a 371 of PCT/US09/39695 04/06/2009
which claims benefit of 61/123,438 04/07/2008
and claims benefit of 61/113,471 11/11/2008
and claims benefit of 61/166,651 04/03/2009

Foreign Applications (You may be eligible to benefit from the Patent Prosecution Highway program at the USPTO. Please see http://www.uspto.gov for more information.)

If Required, Foreign Filing License Granted: 01/23/2011

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is US 12/936,395

Projected Publication Date: 05/05/2011

Non-Publication Request: No

Early Publication Request: No

\*\* SMALL ENTITY \*\*

**Title**

SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES

**Preliminary Class**

**PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES**

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at <http://www.uspto.gov/web/offices/pac/doc/general/index.html>.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, <http://www.stopfakes.gov>. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4158).

**LICENSE FOR FOREIGN FILING UNDER**

**Title 35, United States Code, Section 184**

**Title 37, Code of Federal Regulations, 5.11 & 5.15**

**GRANTED**

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier

license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign Assets Control, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

**NOT GRANTED**

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).



Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

<b>TRANSMITTAL FORM</b>  <i>(to be used for all correspondence after initial filing)</i>	Application Number	12/936,395	
	Filing Date	November 3, 2010	
	First Named Inventor	Steven H. Rempell	
	Art Unit	Not Yet Assigned	
	Examiner Name	Not Yet Assigned	
Total Number of Pages in This Submission	1	Attorney Docket Number	XPR.002US0

<b>ENCLOSURES (Check all that apply)</b>		
<input checked="" type="checkbox"/> Fee Transmittal Form <input checked="" type="checkbox"/> Fee Attached <input type="checkbox"/> Amendment/Reply <input type="checkbox"/> After Final <input type="checkbox"/> Affidavits/declaration(s) <input type="checkbox"/> Extension of Time Request <input type="checkbox"/> Express Abandonment Request <input type="checkbox"/> Information Disclosure Statement  <input type="checkbox"/> Certified Copy of Priority Document(s) <input type="checkbox"/> Reply to Missing Parts/ Incomplete Application <input type="checkbox"/> Reply to Missing Parts under 37 CFR 1.52 or 1.53	<input type="checkbox"/> Drawing(s) <input type="checkbox"/> Licensing-related Papers <input type="checkbox"/> Petition <input type="checkbox"/> Petition to Convert to a Provisional Application <input type="checkbox"/> Power of Attorney, Revocation Change of Correspondence Address <input type="checkbox"/> Terminal Disclaimer <input type="checkbox"/> Request for Refund <input type="checkbox"/> CD, Number of CD(s) _____ <input type="checkbox"/> Landscape Table on CD	<input type="checkbox"/> After Allowance Communication to TC <input type="checkbox"/> Appeal Communication to Board of Appeals and Interferences <input type="checkbox"/> Appeal Communication to TC <b>(Appeal Notice, Brief, Reply Brief)</b> <input type="checkbox"/> Proprietary Information <input type="checkbox"/> Status Letter <input type="checkbox"/> Other Enclosure(s) (please identify below):
<input type="text"/> Remarks		

**SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT**

Firm Name			
Signature	/Steven R. Vosen/		
Printed name	Steven R. Vosen		
Date	March 8, 2011	Reg. No.	45,186

**CERTIFICATE OF TRANSMISSION/MAILING**

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature			
Typed or printed name		Date	

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

## Electronic Patent Application Fee Transmittal

<b>Application Number:</b>	12936395
<b>Filing Date:</b>	03-Nov-2010
<b>Title of Invention:</b>	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES
<b>First Named Inventor/Applicant Name:</b>	Steven H. Rempell
<b>Filer:</b>	Steven R Vosen/Tony Freitas
<b>Attorney Docket Number:</b>	XPR.002U50

Filed as Small Entity

### U.S. National Stage under 35 USC 371 Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD(\$)
<b>Basic Filing:</b>				
Natl Stage Search Fee - Report provided	2642	1	215	215
Natl Stage Exam Fee - all other cases	2633	1	110	110
<b>Pages:</b>				
<b>Claims:</b>				
Claims in excess of 20	2615	8	26	208
<b>Miscellaneous-Filing:</b>				
Oath/decl > 30 mo. from priority date	2617	1	65	65

**Petition:**

Description	Fee Code	Quantity	Amount	Sub-Total in USD(\$)
<b>Patent-Appeals-and-Interference:</b>				
<b>Post-Allowance-and-Post-Issuance:</b>				
<b>Extension-of-Time:</b>				
<b>Miscellaneous:</b>				
<b>Total in USD (\$)</b>				<b>598</b>

## Electronic Acknowledgement Receipt

<b>EFS ID:</b>	9609572
<b>Application Number:</b>	12936395
<b>International Application Number:</b>	
<b>Confirmation Number:</b>	2369
<b>Title of Invention:</b>	SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES
<b>First Named Inventor/Applicant Name:</b>	Steven H. Rempell
<b>Customer Number:</b>	40280
<b>Filer:</b>	Steven R Vosen/Tony Freitas
<b>Filer Authorized By:</b>	Steven R Vosen
<b>Attorney Docket Number:</b>	XPR.002US0
<b>Receipt Date:</b>	08-MAR-2011
<b>Filing Date:</b>	03-NOV-2010
<b>Time Stamp:</b>	13:44:18
<b>Application Type:</b>	U.S. National Stage under 35 USC 371

### Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	\$598
RAM confirmation Number	46
Deposit Account	
Authorized User	

### File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
-----------------	----------------------	-----------	----------------------------------	------------------	------------------

1	Transmittal Letter	Transmittal.pdf	159088	no	1
			8fd05c8520717b6d41cf84fec43666839d8cfd1c		

**Warnings:**

**Information:**

2	Fee Worksheet (PTO-875)	fee-info.pdf	37371	no	2
			089f219d40691a51c8ab470cd84efe90e2a34497		

**Warnings:**

**Information:**

<b>Total Files Size (in bytes):</b>			196459		
-------------------------------------	--	--	--------	--	--

**This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.**

**New Applications Under 35 U.S.C. 111**

**If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.**

**National Stage of an International Application under 35 U.S.C. 371**

**If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.**

**New International Application Filed with the USPTO as a Receiving Office**

**If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.**



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Table with 3 columns: U.S. APPLICATION NUMBER NO. (12/936,395), FIRST NAMED APPLICANT (Steven H. Rempell), ATTY. DOCKET NO. (XPR.002US0)

40280
STEVEN VOSEN
1563 SOLANO AVENUE #206
BERKELEY, CA 94707

Table with 2 columns: INTERNATIONAL APPLICATION NO. (PCT/US09/39695), I.A. FILING DATE (04/06/2009), PRIORITY DATE (04/07/2008)

CONFIRMATION NO. 2369
371 ACCEPTANCE LETTER



Date Mailed: 03/21/2011

NOTICE OF ACCEPTANCE OF APPLICATION UNDER 35 U.S.C 371 AND 37 CFR 1.495

The applicant is hereby advised that the United States Patent and Trademark Office in its capacity as a Designated / Elected Office (37 CFR 1.495), has determined that the above identified international application has met the requirements of 35 U.S.C. 371, and is ACCEPTED for national patentability examination in the United States Patent and Trademark Office.

The United States Application Number assigned to the application is shown above and the relevant dates are:

Table with 2 columns: DATE OF RECEIPT OF 35 U.S.C. 371(c)(1), (c)(2) and (c)(4) REQUIREMENTS (11/03/2010), DATE OF COMPLETION OF ALL 35 U.S.C. 371 REQUIREMENTS (03/08/2011)

A Filing Receipt (PTO-103X) will be issued for the present application in due course. THE DATE APPEARING ON THE FILING RECEIPT AS THE " FILING DATE" IS THE DATE ON WHICH THE LAST OF THE 35 U.S.C. 371 (c)(1), (c)(2) and (c)(4) REQUIREMENTS HAS BEEN RECEIVED IN THE OFFICE. THIS DATE IS SHOWN ABOVE. The filing date of the above identified application is the international filing date of the international application (Article 11(3) and 35 U.S.C. 363). Once the Filing Receipt has been received, send all correspondence to the Group Art Unit designated thereon.

The following items have been received:

- Indication of Small Entity Status
• Copy of the International Application filed on 10/04/2010
• Copy of the International Search Report filed on 10/04/2010
• Preliminary Amendments filed on 10/04/2010
• Oath or Declaration filed on 10/04/2010
• Small Entity Statement filed on 11/03/2010
• Request for Immediate Examination filed on 10/04/2010
• U.S. Basic National Fees filed on 11/03/2010
• Priority Documents filed on 10/04/2010
• Power of Attorney filed on 10/04/2010

Applicant is reminded that any communications to the United States Patent and Trademark Office must be mailed to the address given in the heading and include the U.S. application no. shown above (37 CFR 1.5)

LAMONT M HUNTER

---

Telephone: (571) 272-0308



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Table with 7 columns: APPLICATION NUMBER, FILING or 371(c) DATE, GRP ART UNIT, FIL FEE REC'D, ATTY.DOCKET.NO, TOT CLAIMS, IND CLAIMS. Row 1: 12/936,395, 11/03/2010, 2172, 763, XPR.002US0, 28, 3

CONFIRMATION NO. 2369

UPDATED FILING RECEIPT

40280
STEVEN VOSEN
1563 SOLANO AVENUE #206
BERKELEY, CA 94707



Date Mailed: 03/21/2011

Receipt is acknowledged of this non-provisional patent application. The application will be taken up for examination in due course. Applicant will be notified as to the results of the examination. Any correspondence concerning the application must include the following identification information: the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. If an error is noted on this Filing Receipt, please submit a written request for a Filing Receipt Correction. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections

Applicant(s)

Steven H. Rempell, Novato, CA;
David Chrobak, Clayton, CA;
Ken Brown, San Martin, CA;

Assignment For Published Patent Application

EXPRESS MOBILE INC., Novato, CA

Power of Attorney: The patent practitioners associated with Customer Number 40280

Domestic Priority data as claimed by applicant

This application is a 371 of PCT/US09/39695 04/06/2009
which claims benefit of 61/123,438 04/07/2008
and claims benefit of 61/113,471 11/11/2008
and claims benefit of 61/166,651 04/03/2009

Foreign Applications (You may be eligible to benefit from the Patent Prosecution Highway program at the USPTO. Please see http://www.uspto.gov for more information.)

If Required, Foreign Filing License Granted: 01/23/2011

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is US 12/936,395

Projected Publication Date: 05/05/2011

Non-Publication Request: No

Early Publication Request: No

\*\* SMALL ENTITY \*\*



**Title**

SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES

**Preliminary Class**

715

**PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES**

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at <http://www.uspto.gov/web/offices/pac/doc/general/index.html>.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, <http://www.stopfakes.gov>. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4158).

**LICENSE FOR FOREIGN FILING UNDER****Title 35, United States Code, Section 184****Title 37, Code of Federal Regulations, 5.11 & 5.15****GRANTED**

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as

set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign Assets Control, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

**NOT GRANTED**

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

**PATENT APPLICATION FEE DETERMINATION RECORD**

Substitute for Form PTO-875

Application or Docket Number  
12/936,395

**APPLICATION AS FILED - PART I**

(Column 1) (Column 2)

FOR	NUMBER FILED	NUMBER EXTRA
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A
SEARCH FEE (37 CFR 1.16(k), (l), or (m))	N/A	N/A
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A
TOTAL CLAIMS (37 CFR 1.16(j))	28 minus 20 = *	8
INDEPENDENT CLAIMS (37 CFR 1.16(h))	3 minus 3 = *	
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is \$270 (\$135 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).	
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))		

\* If the difference in column 1 is less than zero, enter "0" in column 2.

**SMALL ENTITY**

RATE(\$)	FEE(\$)
N/A	165
N/A	215
N/A	110
x 26 =	208
x 110 =	0.00
	0.00
TOTAL	698

**OTHER THAN SMALL ENTITY**

RATE(\$)	FEE(\$)
N/A	
N/A	
N/A	
TOTAL	

**APPLICATION AS AMENDED - PART II**

(Column 1) (Column 2) (Column 3)

AMENDMENT A	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA
	Total (37 CFR 1.16(i))	* Minus **	=
Independent (37 CFR 1.16(h))	* Minus ***	=	
Application Size Fee (37 CFR 1.16(s))			
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))			

**SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

**OTHER THAN SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

(Column 1) (Column 2) (Column 3)

AMENDMENT B	CLAIMS REMAINING AFTER AMENDMENT	HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA
	Total (37 CFR 1.16(i))	* Minus **	=
Independent (37 CFR 1.16(h))	* Minus ***	=	
Application Size Fee (37 CFR 1.16(s))			
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))			

**SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

**OTHER THAN SMALL ENTITY**

RATE(\$)	ADDITIONAL FEE(\$)
x =	
x =	
TOTAL ADD'L FEE	

\* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.  
 \*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".  
 \*\*\* If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".  
 The "Highest Number Previously Paid For" (Total or Independent) is the highest found in the appropriate box in column 1.



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Table with 4 columns: APPLICATION NUMBER (12/936,395), FILING OR 371(C) DATE (11/03/2010), FIRST NAMED APPLICANT (Steven H. Rempell), ATTY. DOCKET NO./TITLE (XPR.002US0)

CONFIRMATION NO. 2369

PUBLICATION NOTICE

40280
STEVEN VOSEN
1563 SOLANO AVENUE #206
BERKELEY, CA 94707



Title:SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES

Publication No.US-2011-0107227-A1
Publication Date:05/05/2011

NOTICE OF PUBLICATION OF APPLICATION

The above-identified application will be electronically published as a patent application publication pursuant to 37 CFR 1.211, et seq. The patent application publication number and publication date are set forth above.

The publication may be accessed through the USPTO's publically available Searchable Databases via the Internet at www.uspto.gov. The direct link to access the publication is currently http://www.uspto.gov/patft/.

The publication process established by the Office does not provide for mailing a copy of the publication to applicant. A copy of the publication may be obtained from the Office upon payment of the appropriate fee set forth in 37 CFR 1.19(a)(1). Orders for copies of patent application publications are handled by the USPTO's Office of Public Records. The Office of Public Records can be reached by telephone at (703) 308-9726 or (800) 972-6382, by facsimile at (703) 305-8759, by mail addressed to the United States Patent and Trademark Office, Office of Public Records, Alexandria, VA 22313-1450 or via the Internet.

In addition, information on the status of the application, including the mailing date of Office actions and the dates of receipt of correspondence filed in the Office, may also be accessed via the Internet through the Patent Electronic Business Center at www.uspto.gov using the public side of the Patent Application Information and Retrieval (PAIR) system. The direct link to access this status information is currently http://pair.uspto.gov/. Prior to publication, such status information is confidential and may only be obtained by applicant using the private side of PAIR.

Further assistance in electronically accessing the publication, or about PAIR, is available by calling the Patent Electronic Business Center at 1-866-217-9197.

Office of Data Management, Application Assistance Unit (571) 272-4000, or (571) 272-4200, or 1-888-786-0101

PLUS Search Results for S/N 12936395, Searched Wed Jul 18 08:31:02 EDT 2012  
The Patent Linguistics Utility System (PLUS) is a USPTO automated search system for U.S. Patents from 1971 to the present PLUS is a query-by-example search system which produces a list of patents that are most closely related linguistically to the application searched. This search was prepared by the staff of the Scientific and Technical Information Center, SIRA.

4455935 61	5794796 53
5522148 61	
5609322 61	
5793104 61	
5911112 61	
5969845 61	
6468433 61	
7100134 61	
20020194167 61	
20050044086 61	
20050044514 61	
20120029676 61	
5351197 58	
5574843 58	
5655130 58	
5784292 58	
6101252 58	
6250930 58	
6760043 58	
6823379 58	
6961897 58	
7287093 58	
7885978 58	
7885978 58	
20010031454 58	
20020035617 58	
20020044633 58	
20020087986 58	
20040163086 58	
20060010157 58	
20090157725 58	
20100005529 58	
20110107227 58	
5537528 56	
5727156 56	
6245982 56	
6253193 56	
6363488 56	
6389402 56	
6427140 56	
5708828 54	
5392390 54	
5666553 54	
5701423 54	
5842212 54	
6189146 54	
5475601 53	
5602973 53	
5631904 53	

PLUS Search Results for S/N 12936395, Searched Wed Jul 18 08:32:40 EDT 2012  
The Patent Linguistics Utility System (PLUS) is a USPTO automated search system for U.S. Patents from 1971 to the present PLUS is a query-by-example search system which produces a list of patents that are most closely related linguistically to the application searched. This search was prepared by the staff of the Scientific and Technical Information Center, SIRA.

4455935 63	6063133 55
5522148 63	
5609322 63	
5793104 63	
5911112 63	
5969845 63	
6468433 63	
7100134 63	
20020194167 63	
20050044086 63	
20050044514 63	
20120029676 63	
5351197 60	
5784292 59	
5574843 57	
5655130 57	
6101252 57	
6250930 57	
6760043 57	
6823379 57	
6961897 57	
7287093 57	
7885978 57	
7885978 57	
20010031454 57	
20020035617 57	
20020044633 57	
20020087986 57	
20040163086 57	
20060010157 57	
20090157725 57	
20100005529 57	
20110107227 57	
5708828 56	
5392390 55	
5666553 55	
5701423 55	
5842212 55	
6189146 55	
5475601 55	
5602973 55	
5631904 55	
5794796 55	
5850550 55	
5873472 55	
5961597 55	
5966434 55	
5987452 55	
6035307 55	



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
12/936,395	11/03/2010	Steven H. Rempell	XPR.002US0	2369
40280	7590	10/11/2012	EXAMINER	
STEVEN VOSEN 1563 SOLANO AVENUE #206 BERKELEY, CA 94707			XIA, XUYANG	
			ART UNIT	PAPER NUMBER
			2178	
			MAIL DATE	DELIVERY MODE
			10/11/2012	PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.





## DETAILED ACTION

### ***Claim Rejections - 35 USC § 102***

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims **1-3,5-7,11-14,16-18,22** are rejected under 35 U.S.C. 102(b) as being **anticipated** by **McCain US Patent Application Publication No.: 2005/0273705**.

3. In regard to claim 1, McCain discloses the claimed invention (Currently Amended): a system (Paragraph [0002]) for generating code (paragraph [0081]) to provide content on a display of a device, (paragraph [0041], [0042]) said system comprising: a database of web services obtainable over a network;(Fig. 1, paragraph [0070]) an authoring tool (paragraph [0023] )configured to: define an object for presentation on the display, (paragraph [0072]) select a component of a web service included in said database,(paragraph [0073],[0071]) associate said object with said selected component, (paragraph [0080],[0081]) and produce device-specific code that, (paragraph [0084]) when executed on the device, provides said selected component on the display of the device. (paragraph [0072],[0073])

4. In regard to claim 2, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said database includes definitions of input and/or output related to said web service.(paragraph [0015])

Art Unit: 2178

5. In regard to claim 3, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.(paragraph [0072])

6. In regard to claim 5, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said object is an input field (paragraph [0015]) for a web service.(Fig.3, paragraph [0079],[0136])

7. In regard to claim 6, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said object is an input field usable to obtain said component, where said input field includes a text field, a scrolling text box, a check box, a drop down-menu, a list menu, or a submit button.(Fig. 3, paragraph [0079])

8. In regard to claim 7, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said component is an output of a web service, (paragraph [0129]) is the text provided by one or more simultaneous chat sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS display, or an advertisement.(Paragraph [0072])

9. In regard to claim 11, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said code is provided over said network.(Paragraph [0022])

10. In regard to claim 12, McCain discloses the claimed invention a method (Paragraph [0002]) for displaying content on a device (paragraph [0041], [0042])

Art Unit: 2178

utilizing a database of web services obtainable over a network, (Fig. 1, paragraph [0070]) said method comprising: defining an object for presentation on the display; (paragraph [0072]) selecting a component of a web service included in said database; (paragraph [0073],[0071]) associating said object with said selected component; (paragraph [0080],[0081]) and producing device-dependent code that, (paragraph [0084]) when executed on the device, provides said selected component on the display of the device. (paragraph [0072],[0073])

11. In regard to claim 13, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches where said database includes definitions of input and/or output related to said web service. (paragraph [0015])

12. In regard to claim 14, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches where said component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed. (paragraph [0072])

13. In regard to claim 16, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches where said object is an input field (paragraph [0015]) for a web service. (Fig.3, paragraph [0079],[0136])

14. In regard to claim 17, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches where said object is an input field usable to obtain said component, where said input field includes a text

Art Unit: 2178

field, a scrolling text box, a check box, a drop down-menu, a list menu, or a submit button. (Fig. 3, paragraph [0079])

15. In regard to claim 18, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches where said component is an output of a web service, (paragraph [0129]) is the text provided by one or more simultaneous chat sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS display, or an advertisement. (Paragraph [0072])

16. In regard to claim 22, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches further comprising: providing said code over said network. (Paragraph [0022])

### ***Claim Rejections - 35 USC § 103***

17. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Art Unit: 2178

18. Claims **4,8-9,15,19-20** are rejected under 35 U.S.C. 103(a) as being unpatentable over **McCain US Patent Application Publication No.: 2005/0273705** in view of **Delpuch et al. US Patent Application Publication No.: 2004/0055017.**

19. In regard to claim 4, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein, but fails to explicitly teach “where said object is an input field for a chat.” Delpuch et al. teach where said object is an input field for a chat. (Paragraph [0106])

It would have been obvious to one having ordinary skill in the art at the time the invention was made to incorporate Delpuch et al.’s generating and transmitting authoring data associated with distributed content into McCain’s invention as they are related to the same field endeavor of designing, implementing and deploying content. The motivation to combine these arts, as proposed above, at least because Delpuch et al.’s capability to provide input field for a chat would provide the user with real-time interactions to McCain’s creating web service application. Therefore it would have been obvious to one having ordinary skill in the art at the time the invention was made that providing an input field for chat would give the user with more ways of real-time communication to the web service application.

20. In regard to claim 8, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein, McCain also teaches where said authoring tool (paragraph [0023]), but fails to explicitly teach “is further configured to: define a phone field or list; and generate code that, when executed on the device, allows a user to supply a phone number to said phone field or list.” Delpuch et al. teach is further

Art Unit: 2178

configured to: define a phone field or list; (paragraph [0086]) and generate code that, when executed on the device, allows a user to supply a phone number to said phone field or list. (Paragraph [0081])

It would have been obvious to one having ordinary skill in the art at the time the invention was made to incorporate Delpuch et al.'s generating and transmitting authoring data associated with distributed content into McCain's invention as they are related to the same field endeavor of designing, implementing and deploying content. The motivation to combine these arts, as proposed above, at least because Delpuch et al.'s phone field would provide an address or identification associated with the content distributed to McCain's creating web service application. Therefore it would have been obvious to one having ordinary skill in the art at the time the invention was made that providing a phone field would provide the system with a way to authorize the content distributed to the web service application.

21. In regard to claim 9, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein, McCain also teaches where said authoring tool (paragraph [0023]), but fails to explicitly teach "is further configured to: define a SMS field or list; and generate code that, when executed on the device, allows a user to supply an SMS address to said SMS field or list." Delpuch et al. teach is further configured to: define a SMS field or list; (paragraph [0086]) and generate code that, when executed on the device, allows a user to supply an SMS address to said SMS field or list. (Paragraph [0069])

It would have been obvious to one having ordinary skill in the art at the time the

Art Unit: 2178

invention was made to incorporate Delpuch et al.'s generating and transmitting authoring data associated with distributed content into McCain's invention as they are related to the same field endeavor of designing, implementing and deploying content. The motivation to combine these arts, as proposed above, at least because Delpuch et al.'s SMS field would provide the user with real-time interactions to McCain's creating web service application. Therefore it would have been obvious to one having ordinary skill in the art at the time the invention was made that providing a SMS field would give the user with more ways of real-time communication to the web service application.

22. In regard to claim 15, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. But McCain fails to explicitly teach "where said object is an input field for a chat." Delpuch et al. teach where said object is an input field for a chat. (paragraph [0106])

23. In regard to claim 19, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. But McCain fails to explicitly teach "further comprising: defining a phone field or list; and generating code that, when executed on the device, allows a user to supply a phone number to said phone field or list." Delpuch et al. teach further comprising: defining a phone field or list; (paragraph [0086]) and generating code that, when executed on the device, allows a user to supply a phone number to said phone field or list. (Paragraph [0081])

24. In regard to claim 20, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. But McCain fails to explicitly teach "further comprising: defining a SMS field or list; and generating code that, when

Art Unit: 2178

executed on the device, allows a user to supply an SMS address to said SMS field or list.” Delpuch et al. teach further comprising: defining a SMS field or list; (paragraph [0086]) and generating code that, when executed on the device, allows a user to supply an SMS address to said SMS field or list. (Paragraph [0069])

25. Claims **10,21,23-28** are rejected under 35 U.S.C. 103(a) as being unpatentable over **McCain US Patent Application Publication No.: 2005/0273705** in view of **Paddon et al. US Patent Application Publication No: 2006/0063518**

26. In regard to claim 10, McCain discloses the claimed invention the system of Claim 1, the rejection is incorporated herein. McCain also teaches where said code includes two or more codes, (paragraph [0072], non-binary and binary codes) where one of said two or more codes is device specific (paragraph [0072],[0084])

But McCain fails to explicitly teach “and where one of said two or more codes in device independent.” Paddon et al. teach where one of said two or more codes in device independent. (Paragraph [0010])

It would have been obvious to one having ordinary skill in the art at the time the invention was made to incorporate Paddon et al.’s developing and deploying device independent applications into McCain’s invention as they are related to the same field endeavor of designing, implementing and deploying content. The motivation to combine these arts, as proposed above, at least because Paddon et al.’s developing device independent application would save the developers with significant time and resources to McCain’s creating web service application. Therefore it would have been obvious to one having ordinary skill in the art at the time the invention was made that developing



Art Unit: 2178

and deploying device independent applications would enable developers to efficiently develop, deploy and maintain one instance of an application that can execute on multiple mobile communication platforms.

27. In regard to claim 21, McCain discloses the claimed invention the method of Claim 12, the rejection is incorporated herein. McCain also teaches where said code includes two or more codes, (paragraph [0072], non-binary and binary codes) where one of said two or more codes is device specific, (paragraph [0072], [0084])

But McCain fails to explicitly teach “and where one of said two or more codes in device independent.” Paddon et al. teach and where one of said two or more codes in device independent. (Paragraph [0010])

28. In regard to claim 23, McCain discloses the claimed invention a method for providing information to a device on a network, (paragraph [0002] said method comprising: accepting a first code over the network, (paragraph [0022],) where said first code is device-dependent; (paragraph [0072], [0084], non-binary code is considered as a first code, binary code is considered as a second code) providing a second code over the network, (paragraph [0022] ) and executing said first code and said second code on the device to provide web components obtained over the network. (paragraph [0081])

But McCain fails to explicitly teach “where said second code is device-independent;” Paddon et al. teach where said second code is device-independent;(paragraph [0010])

29. In regard to claim 24, McCain and Paddon et al. together discloses the claimed invention the method of Claim 23, the rejection is incorporated herein. McCain also

Art Unit: 2178

teaches where said web component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed. (Paragraph [0072])

30. In regard to claim 25, McCain and Paddon et al. together discloses the claimed invention the method of Claim 23, the rejection is incorporated herein. McCain also teaches where said component is an output of a web service, (paragraph [0129]) is the text provided by one or more simultaneous chat sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS display, or an advertisement. (Paragraph [0072])

31. In regard to claim 26, McCain and Paddon et al. together discloses the claimed invention the method of Claim 23, the rejection is incorporated herein. McCain also teaches where said first code and said second code are generated using an authoring tool.(Paragraph [0071],[0079])

32. In regard to claim 27, McCain and Paddon et al. together discloses the claimed invention the method of Claim 23, the rejection is incorporated herein. McCain also teaches where said first code is a Player.(paragraph [0072], non-binary component is interpreted as a Player)

33. In regard to claim 28, McCain and Paddon et al. together discloses the claimed invention the method of Claim 23, the rejection is incorporated herein, McCain also teaches where said second code is an Application. (Paragraph [0072], binary component is interpreted as an Application)

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to XUYANG XIA whose telephone number is (571)270-3045. The examiner can normally be reached on Monday-Friday 7:30am-5pm, every other Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Stephen Hong can be reached on (571)272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/XUYANG XIA/  
Examiner, Art Unit 2178  
9/12/2012

/Stephen S. Hong/  
Supervisory Patent Examiner, Art Unit 2178

<b>Notice of References Cited</b>	Application/Control No. 12/936,395	Applicant(s)/Patent Under Reexamination REMPELL ET AL.	
	Examiner XUYANG XIA	Art Unit 2178	Page 1 of 1

**U.S. PATENT DOCUMENTS**

*	Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A US-2005/0273705	12-2005	McCain, Robert C.	715/513
*	B US-2004/0055017	03-2004	Delpuch et al.	725/110
*	C US-2006/0063518	03-2006	Paddon et al.	455/418
	D US-			
	E US-			
	F US-			
	G US-			
	H US-			
	I US-			
	J US-			
	K US-			
	L US-			
	M US-			


**FOREIGN PATENT DOCUMENTS**

*	Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N				
	O				
	P				
	Q				
	R				
	S				
	T				

**NON-PATENT DOCUMENTS**

*	Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)				
	U				
	V				
	W				
	X				

\*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)  
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

<b>Search Notes</b>  	<b>Application/Control No.</b>  12936395	<b>Applicant(s)/Patent Under Reexamination</b>  REMPELL ET AL.
	<b>Examiner</b>  XUYANG XIA	<b>Art Unit</b>  2178

<b>SEARCHED</b>			
<b>Class</b>	<b>Subclass</b>	<b>Date</b>	<b>Examiner</b>
715	738	9/11/2012	

<b>SEARCH NOTES</b>		
<b>Search Notes</b>	<b>Date</b>	<b>Examiner</b>
Inventor Name Search for Double Patenting	9/10/2012	
EAST Search	9/11/2012	

<b>INTERFERENCE SEARCH</b>			
<b>Class</b>	<b>Subclass</b>	<b>Date</b>	<b>Examiner</b>

/XUYANG XIA/ Examiner.Art Unit 2178	
--	--


**UNITED STATES PATENT AND TRADEMARK OFFICE**

UNITED STATES DEPARTMENT OF COMMERCE  
**United States Patent and Trademark Office**  
 Address: COMMISSIONER FOR PATENTS  
 P.O. Box 1450  
 Alexandria, Virginia 22313-1450  
 www.uspto.gov

**BIB DATA SHEET**
**CONFIRMATION NO. 2369**

SERIAL NUMBER	FILING or 371(c) DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.		
12/936,395	11/03/2010	715	2178	XPR.002US0		
<b>APPLICANTS</b>						
Steven H. Rempell, Novato, CA; David Chrobak, Clayton, CA; Ken Brown, San Martin, CA;						
<b>** CONTINUING DATA *****</b>						
This application is a 371 of PCT/US09/39695 04/06/2009 which claims benefit of 61/123,438 04/07/2008 and claims benefit of 61/113,471 11/11/2008 and claims benefit of 61/166,651 04/03/2009						
<b>** FOREIGN APPLICATIONS *****</b>						
<b>** IF REQUIRED, FOREIGN FILING LICENSE GRANTED ** ** SMALL ENTITY **</b> 01/23/2011						
Foreign Priority claimed 35 USC 119(a-d) conditions met Verified and Acknowledged	<input type="checkbox"/> Yes <input checked="" type="checkbox"/> No <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No /XUYANG XIA/ Examiner's Signature	<input type="checkbox"/> Met after Allowance Initials	<b>STATE OR COUNTRY</b> CA	<b>SHEETS DRAWINGS</b> 18	<b>TOTAL CLAIMS</b> 28	<b>INDEPENDENT CLAIMS</b> 3
<b>ADDRESS</b>						
STEVEN VOSEN 1563 SOLANO AVENUE #206 BERKELEY, CA 94707 UNITED STATES						
<b>TITLE</b>						
SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON MOBILE DEVICES						
<b>FILING FEE RECEIVED</b> 763	FEES: Authority has been given in Paper No. _____ to charge/credit DEPOSIT ACCOUNT No. _____ for following:				<input type="checkbox"/> All Fees <input type="checkbox"/> 1.16 Fees (Filing) <input type="checkbox"/> 1.17 Fees (Processing Ext. of time) <input type="checkbox"/> 1.18 Fees (Issue) <input type="checkbox"/> Other _____ <input type="checkbox"/> Credit	

## EAST Search History

## EAST Search History (Prior Art)


Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	1	"20050273705"	US-PGPUB; USPAT	AND	ON	2012/09/11 09:34
S2	2	("2005/0273705").URPN.	USPAT	OR	ON	2012/09/11 09:34
S3	2	("2005/0273705").URPN.	USPAT	OR	ON	2012/09/11 09:34
S4	3507	chat web service object phone SMS	US-PGPUB; USPAT	AND	ON	2012/09/11 09:38
S5	32	chat web service object phone (SMS adj address) author\$4 input output	US-PGPUB; USPAT	AND	ON	2012/09/11 09:39
S6	0	device adj indepentent adj application adj code	US-PGPUB; USPAT	AND	ON	2012/09/11 14:52
S7	0	indepentent adj application adj code	US-PGPUB; USPAT	AND	ON	2012/09/11 14:52
S8	0	(machine or device) adj indepentent adj (code or program or application)	US-PGPUB; USPAT	AND	ON	2012/09/11 15:02
S9	1052	(machine or device) adj independent adj (code or program or application)	US-PGPUB; USPAT	AND	ON	2012/09/11 15:02
S10	687	(machine or device) adj independent adj (code or program or application) object content	US-PGPUB; USPAT	AND	ON	2012/09/11 15:02
S11	428	(device) adj independent adj (code or program or application) object content	US-PGPUB; USPAT	AND	ON	2012/09/11 15:02
S12	397	(device) adj independent adj (code or program or application) object content network server	US-PGPUB; USPAT	AND	ON	2012/09/11 15:03
S13	51	(device) adj independent adj (code or program or application) object content network server video chat	US-PGPUB; USPAT	AND	ON	2012/09/11 15:03
S14	361	(device) adj independent adj (code or program or application) object content network server video	US-PGPUB; USPAT	AND	ON	2012/09/11 15:05
S15	2	(device) adj independent adj (code or program or application) same object same content network server video	US-PGPUB; USPAT	AND	ON	2012/09/11 15:06
S16	2	(device) adj independent adj (code or program or application) same object same content video	US-PGPUB; USPAT	AND	ON	2012/09/11 15:06
S17	15	(device) adj independent adj (code or	US-	AND	ON	2012/09/11

		program or application) same web same content	PGPUB; USPAT			15:08
S18	15	(machine or device) adj independent adj (code or program or application) same web same content	US- PGPUB; USPAT	AND	ON	2012/09/11 15:14
S19	23	(machine or device) adj independent adj (code or program or application) same content not S18	US- PGPUB; USPAT	AND	ON	2012/09/11 15:14

**9/ 12/ 2012 10:38:11 AM**

**C:\ Users\ xxia\ Documents\ EAST\ Workspaces\ 12936395.wsp**



<b>Index of Claims</b>  	<b>Application/Control No.</b>  12936395	<b>Applicant(s)/Patent Under Reexamination</b>  REMPELL ET AL.
	<b>Examiner</b>  XUYANG XIA	<b>Art Unit</b>  2178

✓	<b>Rejected</b>
=	<b>Allowed</b>

-	<b>Cancelled</b>
÷	<b>Restricted</b>

N	<b>Non-Elected</b>
I	<b>Interference</b>

A	<b>Appeal</b>
O	<b>Objected</b>

Claims renumbered in the same order as presented by applicant
  CPA
  T.D.
  R.1.47

CLAIM		DATE							
Final	Original	09/10/2012							
	1	✓							
	2	✓							
	3	✓							
	4	✓							
	5	✓							
	6	✓							
	7	✓							
	8	✓							
	9	✓							
	10	✓							
	11	✓							
	12	✓							
	13	✓							
	14	✓							
	15	✓							
	16	✓							
	17	✓							
	18	✓							
	19	✓							
	20	✓							
	21	✓							
	22	✓							
	23	✓							
	24	✓							
	25	✓							
	26	✓							
	27	✓							
	28	✓							

**UNITED STATES PATENT AND TRADEMARK OFFICE**

---

Applicant(s): Rempell et al.

Application No.: 12/936,395

Filed: November 3, 2010

Title: SYSTEMS AND METHODS FOR PRESENTING INFORMATION ON  
MOBILE DEVICES

Group Art Unit: 2369

Examiner: Xuyang Xia

Docket No.: XPR.002US0

---

**AMENDMENT "A" UNDER 37 C.F.R. §1.111**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Dear Commissioner:

The following amendments and remarks are made in response to the Office Action mailed October 11, 2012. A Petition for Extension of Time and the requisite fee for filing a response within the second month are enclosed. Reexamination and reconsideration of the application in light of the following Remarks are respectfully requested, where:

**Amendments to the Claims** are reflected in the **Listing of Claims** which begins on page 2 of this paper; and

**Remarks/Arguments** begin on page 7 of this paper.

### **Amendments to the Claims**

This listing of claims will replace all prior versions, and listings, of the claims in the application:

#### **Listing of Claims:**

Claim 1 (Currently Amended): A system for generating code to provide content on a display of a device, said system comprising:

a database including a registry of web services~~one or more web components related to inputs and/or outputs of a web service~~ obtainable over a network, ~~where said registry~~ each web component includes the symbolic names of inputs and/or outputs associated with the each web services;

an authoring tool configured to:

define ~~an~~ a user interface (UI) object for presentation on the display,

~~select a~~ where said UI object corresponds to a web component of a web service included in said database~~registry,~~

select a symbolic name from said registry~~web component~~ corresponding to the defined UI object,

associate said object with said~~the~~ selected symbolic name with the defined UI object,

produce an Application including the selected symbolic name of the defined UI object and symbolic name associated with the inputs and/or outputs of the web component,~~and~~

~~produce~~ associated with the defined UI object, where said Application is a device-specific~~independent code, and~~

produce a Player, where said Player is a device-dependent code,

such that, when the Application and Player are provided to the device, and when the Player is executed on the device, provides said selected~~said~~ Player interprets said Application to provide the web

component corresponding to the selected UI object and dynamically received values of the web component on the display of the device.

Claim 2 (Original): The system of claim 1, where said database includes definitions of input and/or output related to said web service.

Claim 3 (Currently Amended): The system of claim 1, where said web component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.

Claim 4 (Currently Amended): The system of claim 1, where said UI object is an input field for a chat.

Claim 5 (Currently Amended): The system of claim 1, where said UI object is an input field for a web service.

Claim 6 (Currently Amended): The system of claim 1, where said UI object is an input field usable to obtain said web component, where said input field includes a text field, a scrolling text box, a check box, a drop down-menu, a list menu, or a submit button.

Claim 7 (Currently Amended): The system of claim 1, where said web component is an output of a web service, is the text provided by one or more simultaneous chat sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS display, or an advertisement.

Claim 8 (Previously Presented): The system of claim 1, where said authoring tool is further configured to:

define a phone field or list; and

generate code that, when executed on the device, allows a user to supply a phone number to said phone field or list.

Claim 9 (Previously Presented): The system of claim 1, where said authoring tool is further configured to:

define a SMS field or list; and

generate code that, when executed on the device, allows a user to supply an SMS address to said SMS field or list.

Claim 10 (Currently Amended): The system of Claim 1, where said code includes ~~two~~three or more codes, where one of said ~~two~~three or more codes is device specific, and where ~~one~~two of said ~~two~~three or more codes

~~in~~is device independent.

Claim 11 (Original): The system of claim 1, where said code is provided over said network.

Claim 12 (Currently Amended): A method for displaying content on a device utilizing a database ~~of~~including a registry of one or more web services components related to inputs and/or outputs of a web service obtainable over a network, ~~where said registry each web component includes the symbolic names of inputs and/or outputs associated with the each web services,~~ said method comprising:

defining ~~an~~ a user interface (UI) object for presentation on the display;

~~selecting a~~, where said UI object corresponds to a web component of a web service included in said database registry;

selecting a symbolic name from said registry-web component corresponding to the defined UI object;

associating said object with said the selected component; and symbolic name with the defined UI object;

producing an Application including the selected symbolic name of the defined UI object and symbolic name associated with the inputs and/or outputs of the web component associated with the defined UI object, where said Application is a device-dependent code; and

producing a Player, where said Player is a device-dependent code;

such that, when the Application and Player are provided to the device, and when the Player is executed on the device, provides said selected said Player interprets said Application to provide the web component corresponding to the selected UI object and dynamically received values of the web component on the display of the device.

Claim 13 (Original): The method of claim 12, where said database includes definitions of input and/or output related to said web service.

Claim 14 (Currently Amended): The method of claim 12, where said web component is a text chat, a video chat, an image, a slideshow, a video, or an RSS feed.

Claim 15 (Currently Amended): The method of claim 12, where said UI object is an input field for a chat.

Claim 16 (Currently Amended): The method of claim 12, where said UI object is an input field for a web service.

Claim 17 (Currently Amended): The method of claim 12, where said UI object is an input field usable to obtain said web component, where said input field includes a text field, a scrolling text box, a check box, a drop down-menu, a list menu, or a submit button.

Claim 18 (Currently Amended): The method of claim 12, where said web component is an output of a web service, is the text provided by one or more simultaneous chat sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS display, or an advertisement.

Claim 19 (Previously Presented): The method of claim 12, further comprising:  
defining a phone field or list; and  
generating code that, when executed on the device, allows a user to supply a phone number to said phone field or list.

Claim 20 (Previously Presented): The method of claim 12, further comprising:  
defining a SMS field or list; and  
generating code that, when executed on the device, allows a user to supply an SMS address to said SMS field or list.

Claim 21 (Currently Amended): The method of Claim 12, ~~where~~ and such that said ~~code~~ includes two or more codes, where one of said two or more codes is Player interprets dynamically received, device-specific, and where one of said two or more codes in device -independent- values of the web component defined in the Application.

Claim 22 (Currently Amended): The method of claim 12, further comprising:  
providing said ~~code~~ Application and Player over said network.

Claim 23 (Currently Amended): A method for providing information from a web component of a web service to a device on a network, said method comprising:

accepting a first code over the network, where said first code is device-dependent;

~~providing~~ accepting a second code (~~PDL~~) over the network, where said second code is device-independent; and

~~executing~~ includes the symbolic names of inputs and/or outputs associated with the web service; and executing said first code ~~and said~~ on the

device, where said executing includes:

processing said symbolic names of the second code on the device,  
transmitting processed instructions to provide the web components  
obtained service, and

accepting a third code over the network-, where said third code is  
a device-independent third code including the output of the web  
component provided by the web service over the network and in response  
to the second code.

Claim 24 (Currently Amended): The method of claim 23, where said ~~web~~  
~~component~~ third code is a text chat, a video chat, an image, a slideshow, a video, or an  
RSS feed.

Claim 25 (Currently Amended): The method of claim 23, where said ~~component~~ third  
code is an output of a web service, is the text provided by one or more simultaneous chat  
sessions, is the video of a video chat session, is a video, an image, a slideshow, an RSS  
display, or an advertisement.

Claim 26 (Original): The method of claim 23, where said first code and said second  
code are generated using an authoring tool.

Claim 27 (Original): The method of claim 23, where said first code is a Player.

Claim 28 (Currently Amended): The method of claim 23, where said second code is  
an Application- which includes one or more web components.

**REMARKS**

This Amendment is submitted in response to the Office Action mailed October 11, 2012, wherein Claims 1-3, 5-7, 11-14, 16-18, and 22 were rejected as being anticipated by US Patent Application Publication No.: 2005/0273705 to McCain (“*McCain*”), Claims 4, 8-9, 15, and 19-20 were rejected under 35 U.S.C. 103(a) as being unpatentable over *McCain* in view of US Patent Application Publication No.: 2004/0055017 to Delpuch et al. (“*Delpuch*”), and Claims 10, 21, and 23-28 were rejected under 35 U.S.C. 103(a) as being unpatentable over *McCain* in view of US Patent Application Publication No: 2006/0063518 to Paddon et al. (“*Paddon*”). In response, Applicants have amended Claims 1, 3-7, 10, 12, 14-18, 21-25, and 28. Applicants respectfully submit that no new matter has been entered by any of these amendments.

**Claims 1-28 are thus pending in the application.**

For the reasons set forth below, Applicants respectfully submit that, as amended, all remaining claims in this application are patentably distinct over the prior art of record. Reconsideration and allowance of all pending claims in the application are respectfully solicited.

**AMENDMENT TO THE CLAIMS**

Claims 1, 3-7, 10, 12, 14-18, 21-25, and 28 are amended herewith.

Claims 1, 4-6, 12, and 15-17 have been amended to more clearly recite the “object” as a “user interface object.” Support for this amendment is found, for example, in paragraphs [0056] - [0057].

Claims 1, 3, 6, 7, 12, 14, 17, 18, and 23 have been amended to more clearly recite the “component” as a “web component.” Support for this amendment is found, for example, in paragraphs [0064] - [0067].

Claims 1 and 12 are amended to recite the registry and the authoring tool in more detail. Support for the database is found, for example, with reference to web component registry **220** in paragraphs [0064], [0066], [0068], and [0072].

Support for the authoring tool is found, for example, with reference to web component registry **220**, and in paragraphs [0009], [0048], [0069], and [0075].



Claims 10 and 21 are amended to recite that “where said code includes three or more codes, where one of said three or more codes is device specific, and where two of said three or more codes is device independent.” Support for this amendment is found, for example, in paragraphs [0052].

Claim 22 is amended to recite: providing said Application and Player over said network. Support for this amendment is found, for example, in FIGS.1B and 2A and the accompanying description.

Claim 23 is amended to recite the symbolic names and further detail regarding method steps. Support for this amendment is found, for example, in FIGS.1B and 2A and the accompanying description, such as paragraph [0076].

Claims 24 and 25 are amended to recite various “third codes.” Support for this amendment is found in the original claim and, for example, in paragraphs [0076] and [0080].

Claim 28 is amended to recite: where said second code is an Application which includes one or more web components. Support for this amendment is found, for example, in paragraph [0056].

No new matter has been introduced with any of these amendments.

### **REJECTION TO THE CLAIMS**

#### **Rejection under 35 U.S.C. §102**

**In Paragraph 2 of the Office Action**, Claims 1-3, 5-7, 11-14, 16-18, and 22 were rejected under 35 U.S.C. 102(b) as being anticipated by *McCain*. Claims 1, 3, 5-7, 12, 14, and 16-18 are amended herewith. Applicants respectfully submit that the claims, as amended and for the reasons presented subsequently, are patentable.

**The Present Application** includes embodiments for displaying information on a device, such as a mobile device, over a network. One embodiment is an authoring tool that generates files which together provide content over a network. The files include a **Player** (sometimes referred to herein as a “first code”) specific to each device and an **Application** (sometimes referred to herein as a “second code”) that is device-independent. The Player is activated by a web browser or other software or through a signal to device by a special telephone numbers, such as a short code. (see, for example, Paragraph [0049]).

When a device is provided with both a Player and an Application, the Player interprets the Application and dynamically provides information (sometimes referred to here as a “third code”), such as the values of the web component having a corresponding user interface (UI) object, on the display of the device. (see, for example, Paragraph [0102]).

In the present invention, the Application is a text file that includes the symbolic name of the information to be displayed, such as the symbolic name associated with the inputs and/or outputs of a web component of a web service. In certain embodiments, the information from the web component (the third code) is also device-independent, and may be interpreted by the Application.

In certain embodiments, the Application retrieves symbolic names from a database of the symbolic names corresponding to web components is accessible over a network, allowing the inventive software to determine locations on the network for accessing information.

In certain other embodiments, the Player also receives the web component output and interprets the output for display on the device.

*McCain* is concerned with automatically creating network software applications by visiting an existing web-site a network and adding new electronic content and new functionality using preexisting “solutions” available from a “solutions database.”

More specifically, the “solution” of *McCain* includes contents and software instructions to accomplish a desired functionality. The solution of *McCain* includes all necessary code to do so: **both non-binary components** (such as URLs including locations of other components) **and binary components** (such as graphics, audio, or video, executable components).

The solution of *McCain* is one code that is delivered to the device, where it is executed by a browser on the device. Thus, in paragraph [0072], *McCain* states:

A "solution" includes, but is not limited to, electronic content and software instructions to accomplish a desired functionality. A solution includes non-binary components ... [and] binary components... stored as “C2 objects. as described below.

C2 objects are further discussed in paragraph [0080], for example as including,

but not limited to, “software instructions, event controls, and components that can run within the web browser environment or access web-based services and/or communicate with back-end services.”

As discussed in paragraph [0081] the “component engine 18 processes processing instructions, **parameters**, scripts, schemas, **mark-up language tags**, binaries including images, **executables**, connection objects, etc. **and stores them in plural component definition units in extensible Markup Language (XML) format.** ... The multiple binary units are compressed and **compiled into a single unit called a C2 component.** ... The component engine 20 automatically generates all code necessary to create user interfaces, reference system components, data base connections, e-commerce connections, communication protocol connections, etc. C2 components may also be used to store solutions in the one or more solution databases 36' available via the one or more solution servers 36.

*McCain* thus clearly teaches combining all code (executable, device-dependent code as well as parameters (device-independent code) into C2 components, which are then provided to the browser to generate a display. There is no teaching or suggestion in *McCain* of separating binary, non-binary, or executable components into different codes. As noted by the Examiner in paragraph 22 of the Office Action, *McCain* fails teach a device-independent code.

In addition, *McCain* teaches that information from the web component is processed by the web browser, and there is no teaching or suggestion in *McCain* of the information from the web component being processed by the **Player** (that is, a code provided by the authoring tool).

### **Claims 1-3, 5-7 and 11**

Claims 2-3, 5-7, and 11 depend on Claim 1. Independent Claim 1 has been amended to more clearly distinguish the claimed invention over the prior art by reciting, in part:

A system for generating code to provide content on a display of a device, said system comprising ... an authoring tool configured to:  
produce an {device-independent code} Application including {a} selected symbolic name of {a} defined UI object and symbolic name associated with the inputs and/or outputs of the web component associated with the defined UI object,  
and

produce a {device-dependent code} Player,  
such that, when the Application and Player are provided to the device, and when the Player is executed on the device, said Player interprets said Application to provide the web component corresponding to the selected UI object and dynamically received values of the web component on the display of the device.

The claimed authoring tool is neither taught nor suggested by *McCain*. As discussed above, the authoring tool of *McCain* provides code (a “solution”), which is stored in a database, and is read by a browser on the device. Specifically, *McCain*'s solution includes contents and software instructions to accomplish a desired functionality, and includes **all** necessary code to do so: both *non-binary components* (such as URLs including locations of other components) and *binary components* (such as graphics, audio, or video, executable components). (see Paragraph [0072] of *McCain*).

There is no teaching or suggestion in *McCain* of an authoring tool that provides two separate codes: a device-dependent code (such as the claimed **Player**) and device-independent code (such as the claimed **Application**).

Further, the concept of delivering information over a network by providing, for example, web pages by a combination of both a device-dependent and device-independent code is nowhere found in *McCain*, who provides all of the information (binary and non-binary) for execution by a browser. There is no teaching or suggestion in *McCain* of having an Application and Player provided to the device where when the Player is executed on the device, the Player then interprets the Application to provide the web component corresponding to the selected UI object and dynamically received values of the web component on the display of the device.

Amended Claim 1 further recites a database including a registry of one or more web components related to inputs and/or outputs of a web service obtainable over a network, where ~~said registry~~ each web component includes the symbolic names of inputs and/or outputs associated with ~~the~~ each web services. As claimed, the authoring tool selects symbolic names from the registry ~~said web component~~ to produce a device-independent Application, which is then interpreted by the device-dependent Player. The combination of the claimed the database and its use are neither taught nor suggested by *McCain*.

*McCain* refers to a total of three databases: 1) database 24, 2) database 34, and 3)

database 36. Of the three databases, only database 24, shown in Fig 1, is used by the authoring tool, as claimed. Database 24, however, is not equivalent to the claimed registry, which includes an entire “solution” (both device-dependent and device-independent code). There is no teaching or suggestion in *McCain* of using such a solution database in an authoring tool to generate both a device-dependent and device-dependent codes, as claimed.

Lastly, there is no teaching or suggestion in *McCain*, or in the other references, of how a device-dependent code (the Player) may utilize symbolic references in a device-independent code (the *Application*), both of which are generated by an authoring tool, requests and then receives the output of web services and then populates associated UI Objects with this content.

For any of these reasons, *McCain* does not anticipate Claim 1, as amended. Specifically, neither the registry containing symbolic names, the use of such a registry in by an authoring tool to generate both device-dependent and device-independent code, or the use of the device-dependent and device-independent code together to return real time information to the device, are taught or suggested by *McCain*. Applicants respectfully request that the rejection of under §102 of Claim 1, as amended, and of amended Claims 2-3, 5-7, and 11, which depend on Claim 1, be withdrawn.

**Claims 12-14, 16-18, and 22**

Claims 13-14, 16-18, and 22 depend on Claim 12, which has been amended to include the same language discussed above regarding amended Claim 1.

For at least the reasons presented above for Claim 1, *McCain* does not anticipate Claim 12, as amended.

Thus, for any of the reasons presented above, Applicants respectfully request that the rejection of under §102 of Claim 12, as amended, and of amended Claims 13-14, 16-18 and 22, which depend on Claim 12, be withdrawn.

**Rejection under 35 U.S.C. §103**

**Paragraph 18 of the Office Action**, Claims 4, 8-9, 15, and 19-20 were rejected under 35 U.S.C. 103(a) as being unpatentable over *McCain* in view of *Delpuch*. Claims 4 and 15 are amended herewith, as are independent Claims 1 and 12, on which the claims