

"Express Mail" mailing label number:

EL803198675US

**LOGICAL AND CONSTRAINT BASED BROWSE
HIERARCHY WITH PROPAGATION FEATURES**

Scott Bonneau
Michael Nonemacher
Jeremy Weinrib

5

CROSS REFERENCE TO RELATED APPLICATIONS

This application relates to application serial no. _____ (attorney docket M-10956 US), filed on same day herewith, entitled "Rules Based Provision of Custom Pricing for Multiple Entities" and naming Scott Bonneau, Michael Nonemacher and Jeremy Weinrib as inventors, the application being incorporated herein by reference in its entirety.

This application relates to application serial no. _____ (attorney docket M-10957 US), filed on same day herewith, entitled "Rules Based Custom Catalogs Generated from a Central Catalog Database for Multiple Entities" and naming Scott Bonneau, Michael Nonemacher and Jeremy Weinrib as inventors, the application being incorporated herein by reference in its entirety.

This application relates to application serial no. _____ (attorney docket M-11857 US), filed on same day herewith, entitled "Browse Hierarchies Customized for Rules Based Custom Catalogs" and naming Scott Bonneau, Michael Nonemacher and Jeremy Weinrib as inventors, the application being incorporated herein by reference in its entirety.

20 This application relates to application serial no. _____ (attorney docket M-10959 US), filed on same day herewith, entitled "A Method For Building Digital Databases Optimized For Maintenance, Descriptiveness, And Fast Search" and naming Scott Bonneau and Michael Nonemacher as inventors, the application being incorporated herein by reference in its entirety.

090430 091000
10
15

Versata Exh. 2005
Volusion v. Versata
CBM2013-00017

BACKGROUND OF THE INVENTION

Field of the Invention

5 The present invention relates to browsing on-line catalogs and web sites, and more specifically to a flexible and arbitrarily expressive rules-based browsing hierarchy for on-line catalogs and web sites.

Description of the Related Art

10 With the advent of Internet based commerce, organizations on both the buy and sell side of business-to-business (B2B) procurement relationships have sought to harness computer networks as a means for automating the procurement process between them. To facilitate e-commerce, and particularly e-procurement, suppliers of goods and services have developed electronic catalogs by which potential buyers can receive and display information regarding the goods and services offered by the seller, including descriptive information, pictures and prices.

15
20
25
30
FOR THE "BEST COPY" OF THIS DOCUMENT

One issue confronted by sellers offering goods and services for sale over the Internet, whether through electronic catalogs or a web site, is how to present their product information to a buyer. One simple approach is to mimic a user's interaction with a paper catalog. This approach involves presenting the buyer with a sequence of catalog pages displayed on the buyer's computer screen, each page consisting of descriptive data in the form of text and images that cover some number of items being offered by the seller. Using this technique for catalogs containing a large number of items will likely require the buyer to browse too many pages to find the items in which the user is interested. For a catalog or web site with even a moderately expansive offering of items, this solution is not practicable. The buyer will probably lose interest before finding the item being sought, and the seller will lose a sale.

One way to make the browsing process more manageable is to organize the catalog items in the database into some form of hierarchy. The presentation created by the hierarchy should be expressive, and should guide the buyer through the catalog of product offerings (as stored in the database of the seller) to specific items of interest to the buyer with reasonable ease and flexibility. A hierarchy typically attempts to classify and/or categorize catalog items starting with relatively general levels of specificity, and gradually becomes more specific based on values of particular attributes associated with the items. Such a hierarchy

can be thought of as a simple tree structure, with higher-order nodes representing more general classifications for the items, and lower-order nodes (i.e. the children of the more general nodes) representing a narrowing of the scope of items that occupy the lower levels of the hierarchy.

5 One way sellers have been known to hierarchically organize their catalog data for browsing is in accordance with a “classification - category - vendor” hierarchy. A simple representation of this hierarchy is illustrated in Fig. 1a. Imposing this hierarchy requires that products be stored in the database along with values for each of these three attributes. At the classification level 100, catalog items are split into some predetermined number of classes
10 each represented by a unique class label. Each class node is then split at the category level 120 into some number of category nodes equal to the number of predetermined categories established under each class. The category nodes are then split at the vendor level 140 to create some number of vendor nodes under each category equal to the precise number of vendors supplying products in the category.

15 Figure 1b is illustrates one possible example of a hierarchy that has been created in accordance with the model of Fig 1a. In the example, the items in the catalog database are divided into two very general classes: hardware and software. Thus, the hierarchy of Fig. 1b reflects these classes with two nodes at the class level 100, one labeled as “hardware” 180 and one labeled as “software” 160. The class nodes are then split at the category level 120
20 into categories represented by the nodes “O/S (operating systems)” 162, “applications” 164 and “video games” 166. Under the “hardware” node 180, the categories might be “PCs (personal computers)” 182, “peripherals” 186, and “storage” 184. Finally, the nodes at the category level 120 are split at the vendor level 140 into a number of nodes representing the particular vendors supplying items from each category as illustrated.

25 Fig. 1c illustrates the manner in which a seller might display this hierarchy in the form of hyperlinks on a web page displayed by a browser program. Typically, the top level is displayed first. If a user-buyer selects one of the classes, the categories are then displayed under the selected class in a manner that indicates they are children falling under the class (e.g. indented). Those of skill in the art will recognize that there a number of ways in which a
30 user can select hyperlinks, including activating the link by clicking a button of a computer mouse. Selection of a category then causes the vendor choices to be displayed under the

selected category. In Fig. 1c, the class node “Software” was selected, which led to the display of category nodes “Applications,” “O/S” and “Video Games.” The “Applications” node was then selected, leading to the display of vendor nodes “Adobe,” “Microsoft” and “Intuit.” Selecting the “Adobe” node will then typically retrieve all Adobe applications software products, and descriptive marketing text and images associated with those products.

The database consisting of the data for each item can be flat (i.e. the database has no physical hierarchy such that all of the items are at the same level). In this case, items fall into a particular one of the terminal nodes of the hierarchy (i.e. a node having no children of its own) based on the attribute values that are ascribed to that node of the hierarchy. In the alternative, the database itself may be physically ordered hierarchically, in which case the location into which an item is stored in the database implicitly dictates its position in the hierarchy, rather than expressly using the values of attributes stored with the items. While a hierarchically arranged database can be more economical in size because attribute values are implied rather than explicitly stored, sellers have been moving away from this approach. Requiring the ordering of the database in a certain manner removes even more flexibility from the seller with respect to how the hierarchy is employed. For example, a seller may wish to organize the hierarchy as “vendor-class-category” but would be restricted from doing so if the database was hierarchically ordered in the manner illustrated by Fig. 1a.

Either way, the foregoing approach is extremely inflexible because the expressiveness of the hierarchy is limited to that which has been preordained. Even if the depth of the hierarchy is expanded to include more levels of specificity, the navigational path by which each item is ultimately browsed is fixed, and so are the rules by which the hierarchy is organized. Put another way, each item can be browsed through exactly one path through the hierarchy because the manner in which the hierarchy is organized is predetermined.

Thus, it would be desirable to provide a more flexible and expressive browse hierarchy that permits the seller the freedom to organize the browsing hierarchy in an arbitrary manner. Such a hierarchy would even permit a user to browse the catalog data through multiple navigational paths to arrive at the same item because multiple instances of an item would be permitted to reside in the hierarchy. It would also be preferable for the hierarchy to be flexible enough to accommodate arbitrary logical groupings of catalog items

or classifications of items that aren't necessarily tied together by some common set of attribute values stored in association with the items in the database.

SUMMARY OF THE INVENTION

5 The invention is a hierarchy for representing a plurality of catalog items stored in a catalog database. In one embodiment, the hierarchy includes a plurality of nodes each representative of some predefined subset of the items. Each of the nodes is a child of one other node, except for a root node, which is a child of no other node and is an ancestor of all of the nodes. Some of the nodes each specify one or more constraints defining a scope of the subset of items represented by each of the nodes. A second group of the nodes specifies no constraints, and each of the second portion of nodes establishes a logical grouping of items that logically defines a scope of the subset of the items represented by each of the second portion relative to the scope of items represented by its parent.

10 The constraints specified by the first portion of nodes are based on attributes associated with the items and a permissible range of values for those attributes. Each node in the hierarchy inherits all of the constraints of its ancestors along with any of its own constraints, and the aggregation of these constraints through a logical ANDing completely defines the scope of the items represented by the node.

15 Leaf nodes comprise a third portion of the nodes, and leaf nodes have no child nodes under them. The hierarchy can facilitate computer browsing of the database when the hierarchy is made available for display on a computer terminal. In one embodiment, the nodes are operable to be activated when selected by a computer mouse. In one embodiment, selecting a node other than a leaf node causes optional text associated with the selected node to be displayed. Further, activating a node also causes the display of the activated node's child nodes and renders them available for selection. Selecting a leaf node causes an aggregation of all constraints specified by the leaf node and its ancestors, and the formulation of a search rule that includes all items in the database that meets the aggregation of constraints. The search rule is then transmitted to a database server in the former of a search query that returns a list of the items from the database that belong to the subset represented by the leaf node. The list of items, and other collateral information is then returned to the computer terminal used to select the leaf node.

20

25

30

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.