

```
/*
"(c) Copyright 1993-1997 Pad++ Consortium {University of New Mexico
(UNM),
and New York University (NYU)}, All Rights Reserved."
Licensee can not remove or obscure any of the
copyright or trademark notices in this software.
```

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

See the file "License" for general information on usage and redistribution, and the file "LicenseTerms" for the specific license agreement on usage and redistribution of this file, and the Pad++ software in general.

```
*/
```

```
#include "defs.h"
#include "object.h"
#include "pad.h"
#include "text.h"
#include "html.h"
#include "image.h"
#include "object.h"
#include "pad-tcl.h"
#include "pad-string.h"
#include "line.h"
#include "win.h"
#include "restorer.h"
#include "group.h"
#include "callback.h"
#include "transform.h"
#include "global.h"
#include "imagedata.h"
```

```
#include <string.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/stat.h>
```

```
#ifdef PAD_UNIX
# include <unistd.h>
#endif
```

```
#define HTML_DEFAULT_FILL          "gray90"
```

```

#define HTML_DEFAULT_BORDER          "black"
#define HTML_DEFAULT_BORDERWIDTH    1
#define HTML_DEFAULT_DONESCRIPT     ""
#define HTML_DEFAULT_ERRORSCRIPT    ""
#define HTML_DEFAULT_FONT           "Line-1.2"
#define HTML_DEFAULT_URL             ""
#define HTML_DEFAULT_UPDATESCRIPT   ""
#define HTML_DEFAULT_STATE          HTML_ANCHOR_UNVISITED

#define HTML_MIN_SIZE                100

// The html object itself is really just a wrapper for loading in
// all kinds of sub-Objects (like text, images, etc.) It gets
// rendered as the background of the html file.
//
// The html object and all of its components are created with pixel
// coordinates. Text is created one pixel high. Images are
// scaled down by a factor of 10 (yielding text that is effectively
// 10 pixels high).
//
// The current Position (x_curr_pos, y_curr_pos) indicates the nw corner
// of each object. x_left_marg represents the left margin.

static float IMAGE_SIZE = 0.1,
            DUMMY_IMAGE_SIZE = 5.0;

static char *StartTagArray[] = {
    "html",
    "body",
    "address",
    "h1",
    "h2",
    "h3",
    "h4",
    "h5",
    "h6",
    "p",
    "a",
    "img",
    "ul",
    "li",
    "dl",
    "dt",
    "de",
    "dd",
    "hr",
    "br",
    "title",
    "head",
    "em",
    "i",
    "b",
    "center",
    "menu",

```

```

    "cite",
    "font",
    "pre",
    "kbd",
    "strong",
    "base",
    "code",
    "ol",
    "!--",
    "form",
    "input",
    ""
};

static HTML_TagOrKey tagArray[] = {
    {"html", TAG_HTML},
    {"body", TAG_BODY},
    {"address", TAG_ADDRESS},
    {"h1", TAG_H1},
    {"h2", TAG_H2},
    {"h3", TAG_H3},
    {"h4", TAG_H4},
    {"h5", TAG_H5},
    {"h6", TAG_H6},
    {"p", TAG_P},
    {"a", TAG_A},
    {"img", TAG_IMG},
    {"ul", TAG_UL},
    {"li", TAG_LI},
    {"dl", TAG_DL},
    {"dt", TAG_DT},
    {"de", TAG_DE},
    {"dd", TAG_DD},
    {"hr", TAG_HR},
    {"br", TAG_BR},
    {"title", TAG_TITLE},
    {"head", TAG_HEAD},
    {"em", TAG_EM},
    {"i", TAG_I},
    {"b", TAG_B},
    {"center", TAG_CENTER},
    {"menu", TAG_MENU},
    {"cite", TAG_CITE},
    {"font", TAG_FONT},
    {"pre", TAG_PRE},
    {"kbd", TAG_KBD},
    {"strong", TAG_STRONG},
    {"base", TAG_BASE},
    {"code", TAG_CODE},
    {"ol", TAG_OL},
    {"!--", TAG_COMMENT1},
    {"!---", TAG_COMMENT2},
    {"meta", TAG_META},
    {"textarea", TAG_TEXTAREA},

```

```

    {"form", TAG_FORM},
    {"input", TAG_INPUT},
    {"table", TAG_TABLE},
    {"tr", TAG_TR},
    {"td", TAG_TD},
    {"th", TAG_TH},
    {"blockquote", TAG_BLOCKQUOTE},
    {"blink", TAG_BLINK},
    {"link", TAG_LINK},
    {"caption", TAG_CAPTION},
    {"header", TAG_HEADER},
    {"", TAG_UNKNOWN},
    {NULL, 0}
};

static HTML_TagOrKey keyArray[] = {
    {"href", KEY_HREF},
    {"src", KEY_SRC},
    {"alt", KEY_ALT},
    {"name", KEY_NAME},
    {"align", KEY_ALIGN},
    {"border", KEY_BORDER},
    {"width", KEY_WIDTH},
    {"height", KEY_HEIGHT},
    {"ismap", KEY_ISMAP},
    {"bgcolor", KEY_BGCOLOR},
    {"text", KEY_TEXT},
    {"link", KEY_LINK},
    {"alink", KEY_ALINK},
    {"vlink", KEY_VLINK},
    {"pad_scale", KEY_PAD_SCALE_BEGIN},
    {"pad_scale", KEY_PAD_SCALE_END},
    {"pad_tcl", KEY_PAD_TCL},
    {"", KEY_UNKNOWN},
    {NULL, 0}
};

static Pad_String Token;

float x_curr_pos,
      y_curr_pos,
      x_left_marg;

static Pad_Bool Anchor_active = FALSE; // True if creating an anchor
static HTML_Anchor *Anchor_first; // First anchor component
static HTML_Anchor *Anchor_current; // Current anchor component
static Pad_String Anchor_href; // Current anchor href
static Pad_String Anchor_name; // Current anchor name
static int NewLine; // Number of recent newlines
static Pad_Bool Title; // True if inside Title tag
static Pad_Bool H1; // True if inside H1 tag
static Pad_Bool H2; // True if inside H2 tag
static Pad_Bool H3; // True if inside H3 tag
static Pad_Bool H4; // True if inside H4 tag

```

```

static Pad_Bool H5;           // True if inside H5 tag
static Pad_Bool H6;           // True if inside H6 tag
static Pad_Bool HTML_DD;     // True if inside DD tag
static Pad_Bool PreFormatted; // True if inside PRE tag

static Pad_List LineObjs;     // List of objects on the current
line
static float DefaultLineSize; // Default space to next line
static float LineSize;        // Height of text line
static float MaxLineSize;     // Keeps track of how much room to add
for a newline

static HTML_EntryTagTcl *Pad_tcl_entry;
static Pad_Bool LayingOut;    // Keeps track of whether we are in
// Create_display_list() or not. Used for
// adding Pad_Objects to HTML after
creation

static Pad_Bool Initializing; // Keeps track of whether we are in
// Set_url() or not. Used for
// adding Pad_Objects to HTML after
creation

static Pad_List MSML_Scale_list; // The stack of scale items for MSML
static Pad_List MSML_Left_margin; // The stack of left margins for MSML
static float MSML_Scale;         // What the current MSML scale is
static float HTML_VERTICAL_IMAGE_OFFSET; // Space added to images for
vertical padding

////////////////////////////////////
//
// Support Routines
//
////////////////////////////////////

//
// Maintain stack of scale tags for MSML extension.
//
static void
MSML_Reset_scale(void)
{
    float *f;

    while ((f = (float *)MSML_Scale_list.Pop())) {
        delete f;
    }
    MSML_Scale = 1.0;
}

static float
MSML_Push_scale(float newScale)
{
    float *f = new float(newScale);

```

# 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.