Deloitte.

Web 2.0
User Interface Styles & Standards Guide

Version .01

Last modified on December 20, 2007

Table of Contents

1.0. General Information

1.1. Preface

1.1.1. Reference and Function

This document is designed to support, drive and streamline user interface (UI) design and front-end development operations in an open, transparent and readily available fashion. The UI team heartily encourages constructive feedback, recognizing that communication is a prerequisite to consensus-building, mutual understanding and the overall cohesion and effectiveness of the group.

1.1.2. Authority and Rationale

The guidelines herein are not merely recommendations; they are standards. Follow these standards in all cases, without deviation, embellishment, second-guessing or selective application. Only rarely and under specific circumstances are exceptions allowed, and then only with the express consent of UI specialists. In the absence of further clarification from authorized personnel regarding UI issues, this document is the final authority on these matters. This authority is derived directly from management.

Standards serve a vital purpose in maintaining consistency. Since learning curves are inherent to all software use, consistent UI principles condition users to expect certain predictable patterns in both presentation and functionality. Similar instructions, for example, worded in similar ways, *should* appear in the same font face, color and size in similar screen locales under similar circumstances. Furthermore, with consistency, the company's image and brand are perceived as a coherent whole. This in turn conveys stability and assures the client that their trust in the company's products is not misplaced. It fosters loyalty and cultivates an emotional bond with the brand. In short, consistently applied standards make for a better product and user experience.

1.1.3. Evolution and Availability

This is an evolving document. With refinement and improvement, Deloitte products adhering to its tenets will continually mature. The prime objective in this endeavor is, of course, to produce the best quality software possible. The fluid nature of software development necessitates that these guidelines variably act and react to new conditions. Therefore, try to consult this document on a regular basis to remain up to date on revisions, additions and clarifications. Everyone affected by changes in this document must remain current on all standards.

1.2. Background

1.2.1. Scope

Many roles attached to the project benefit from its existence. This document attempts to present standards in a straight-forward manner, with background, when possible, on the rationale and reflection behind those decisions. The reason for this is three-fold.

- First, **transparency**. It demonstrates that decisions adhere to sound principles (UI best practices) and are **NOT** arrived at arbitrarily. This in turn instills greater accountability, accuracy and confidence through team-wide checks and balances.
- Second, **erudition**. It elucidates the decision-making path, such that the underlying logic becomes second-nature and propagates through the team. This guide is **NOT** meant to be all-inclusive. It assumes that the reader is capable of extrapolating the reasoning beneath the guidelines, so that patterns can be discerned and similar solutions applied under like circumstances.
- Third, **efficiency**. It streamlines development by reducing churn. By consensus-building through stakeholder buy-in early in the design process, standards can be established and locked down sooner. As a result, the rework rate plummets dramatically and the product moves more rapidly from idea to implementation.

1.2.2. Authors

All style guide changes funnel through the User Interface team to keep the quantity and quality of changes at a manageable level. This ensures consistency and helps eliminate the dissemination of contradictory or otherwise faulty information. Please thoroughly consult this guide before you contact UI with inquiries.

The main point of contact is **Nick Zedlar** (nzedlar@deloitte.com), who is located in Atlanta. His primary contact number is (770) 649-9586; his cell number is (404) 729-0334. Other contacts include (in alphabetical order): Nicole Canty, Jennifer Gunn, Narayan Kansal, Hania Kutcher and Jacqui Miller.

1.2.3. Frequently Asked Questions

1.2.3.1. Updates and Maintenance

- **What constitutes an update?**
  An update is considered brand new content or a significant change in meaning or way of presenting something (for example, adding screen captures). Minor formatting, wording or spelling corrections are **NOT** marked as updates.

- **How do I know what's been updated?**
  It's easy. Check the Table of Contents (TOC) to see at a glance what sections have been updated since the previous version; they are marked with the ⭐ symbol. Parent headings are not update-marked unless every child section under it has been updated. In this case, the parent and all children are shown as updated.

1.2.3.2. Submitting Feedback

- **What types of things are appropriate to communicate to the UI team?**
  Virtually anything that pertains to the form and function of the application user interface *on the front end* and this Guide that documents them. Welcome are praise or constructive criticism of spelling, word choice, control types, colors, fonts, images, layout—anything you can think of that would make the product itself or the User Guide better and easier to use.

- **Who do I contact regarding the Style Guide?**
  See Authors above.

1.2.3.3. Using the Guide

- **This is a big document. How do I find what I'm looking for, get there fast and get out?**
  Your best bet is the Table of Contents. Navigate via the first- (chapter), second- (section) or third-level (subsection) heading in the TOC to jump to a topical part of this document.

- **What are the hyperlinks in this document for?**
  In almost all cases, these are bookmark links that jump within the Guide itself, usually to provide more information about a particular related or relevant topic. Occasionally, links to external content can be found, which always explicitly spell out the URL.

- **What if I don't know what a particular acronym stands for?**
  Check the Acronyms section of the Glossary (if it's not there, search the document for it). Usually there are multiple instances of the acronym spelled out and sprinkled throughout the Guide.

- **In the event of conflicting information, which takes precedence: the requirements documentation, the Guide or mockups?**
  The Guide catalogs generally applicable, predominant system-wide behaviors, properties and appearances. As such, it overrides requirements, but does **NOT** trump mockups. Therefore, consider the following list in descending order of precedence:
  - mockups/wireframes
  - the Guide
  - requirements documentation

1.2.3.4. Baselining the Guide

- **If this is a "living document" that changes fairly often, how can anyone ever keep up to date?**
  The answer is baselining. Sections of the Guide get "locked" after an intensive stakeholder group review process. This ensures that those sections receive the official "stamp of approval" and should, theoretically, experience relatively few changes afterwards. As section after section is baselined, the Guide as a whole becomes a more stable and orderly reference document.

- **How does one know by looking at the Guide what has or hasn't been baselined?**
  Just as updated material is denoted by the ⭐ icon, recognize baselined material by the 🔒 symbol attached to the applicable first-, second- or third-level heading. The symbol automatically appears in the Table of Contents, so approved or "locked" material is easy to spot at a glance. The 📝 icon shows line items that require baseline review. Items specifically excluded from baseline review receive the 🚫 icon.

2.0. Development Specifications

2.1. Main Application Window

2.1.1. Title Bar

⬆ Figure 1: Browser Titlebar (Windows Vista)

The main browser window features the standard left-aligned IE icon, the online location and the application title. It also indicates the current section in the following form: <IE Icon> + <Web Location> + <Hyphen (Unicode 002D)> + <Application Name> + <Hyphen (Unicode 002D)> + <Hyphen (Unicode 002D)> + "Windows Internet Explorer." Text in the title bar is also contingent upon the user's role, since the title reflects the active tab or selected section. If a tab/section does not, for some reason apply, the title bar reverts to its default state, as described above. The title bar shows to only the tab/section level.

Buttons in the title bar are the standard Minimize, Restore/Maximize and Close, as applicable (Windows XP).

Present in the Title Bar
(Minimize, Restore or Maximize and Close)

2.1.2. Window Properties

2.1.2.1. Window Dimensions

The main browser window at launch is always maximized (full screen). It is designed for display resolutions of **1024 pixels x 768 pixels** and higher.

2.1.2.2. Color Scheme

The browser interface itself takes its look and feel from Microsoft Windows (in other words, themes are user-configurable at the operating system level, and the browser inherits those settings). Discussion of these themes is beyond the scope of this document.

However, browser *content* that comprises the Deloitte application have predefined characteristics.

2.1.2.3. Status Bar

The main application *does* have a status bar (unlike dialog boxes). <mark>Any behavior?</mark>

2.1.3. Resizing Behavior

The resizing behavior described here is intended to minimize implementation time for Release 1. Expect this behavior to change in future versions, so that users with higher-resolution displays will get greater value from resizing the application.

2.1.3.1. Toolbar

When the user resizes the width of the display, anchor the buttons on the tool bar to the left-hand side of the application window. When the width of the line of tool bar buttons exceeds the width of the display, hide the tool bar buttons that extend beyond the width of the display on the right side. Restore these buttons when the user expands the application window wide enough to include the previous hidden buttons.

2.1.3.2. Font Attributes

ACTIVE TOP LEVEL TEXT
Arial Normal 11pt #FFFFFF

INACTIVE TOP LEVEL TEXT
Arial Normal 11pt #000000

Workflow   Viewer

Manage Worklists ▶
Manage Tasks ▶
Manage Content

INACTIVE SELECTION TEXT
Arial Normal 11pt #000000

ACTIVE SELECTION TEXT
Arial Normal 11pt #FFFFFF
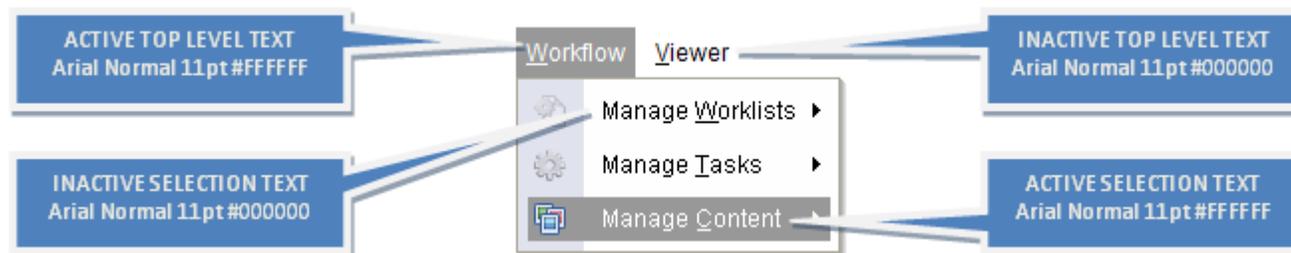
⬆ Figure 2: Cascading Menu Font Attributes

2.1.4. Button Bar/Toolbar

The button bar occurs only in the main application menu, **NOT** in dialog boxes. Detailed button bar documentation occurs in individual User Interface Specification documents, while the images and a description of their properties occur in this document in the Button Bar/Toolbar Icons section.

2.1.5. Breadcrumb Trail (BCT)

The breadcrumb trail (BCT) behavior and presentation depend upon where the user is in the application (i.e., either a content- or operation-based tab/module). It occurs only in the main application menu, **NOT** in dialog boxes. The BCT at DPM launch shows the domain name only. If the user has access to multiple domains, the dropdown domain selector is present, showing by default the first domain in the list (it sorts alphabetically ascending by default). See the Operation-based Behavior description of the BCT for an example of each.

When the mouse pointer hovers over any portion of the path, a tooltip displays the entire path. If the tooltip itself should be too long to display in one line, it wraps.

2.1.5.1. Font Attributes

↑ Figure 3: BCT Font Attributes

2.1.6. Module Tabs

The main navigation tabs in serve as single-click vectors to the primary modules of the application. For details of the proposed visibility feature, see Tab Row Persistence. At launch, the user's role dictates the initial active tab and what options exist within it. A user's access rights limit certain options, as some controls are either disabled or absent. Senior personnel enjoy a wider range of controls and permissions.

Note: the following are presented in the order they appear in the interface from left to right, not alphabetically.

2.1.6.1. Admin

- **12** -
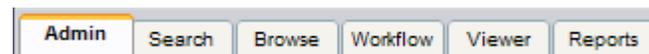
↑ Figure 4: Admin Tab

Upon logon, if the user's role is that of administrator, the default landing area is the Admin tab. For the system administrator, this tab has a dual role: it provides quick access to the system administrative module, where one can manage users, groups, ACLs, roles, worklists, reports, metadata, domains, folders, mail server configuration, content types and more; it also gives the means for the Administrator to manage personal user settings, such as auto logoff configuration and logs. For the standard user, this tab's singular function is to give the means for personal user settings management, such as auto logoff and logs. Whereas system administrators land on the Admin tab upon logon, other users do NOT; they land on Search.

2.1.6.2. Search
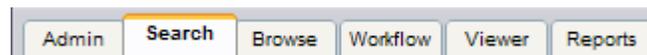
↑ Figure 5: Search Tab

The function of the Search tab is to jump immediately to the Search form, so that the user can personalize a search to locate and retrieve specific content. The module complements Browse, while it offers more powerful means to sift (i.e., proactively "filter" expected results via specific criteria, such as name, creation date, version, type and other metadata and properties), locate and sort files. Search is the default landing tab at logon for users *without* system administrator privileges. Also, see an example of the Search form.

2.1.6.3. Browse

↑ Figure 6: Browse Tab

The Browse tab allows the user to navigate through a familiar hierarchical directory tree to manage content. Considerably more basic in functionality than its Search counterpart, Browse works well for manipulating content routinely accessed from specific folders. It is well-suited for users that rely upon a formally defined folder structure rather than metadata to store and retrieve content. Also, see an example of the Browse screen.

2.1.6.4. Workflow
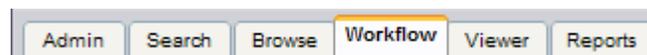
↑ Figure 7: Workflow Tab

- **13** -

The Workflow tab delivers the user to a sophisticated "to do" list of tasks and their various assignments, statuses, deadlines and other tracking criteria. The module affords the ability to manage (e.g., add, delete, assign and reassign) worklists and tasks according to a logical and prescribed process.

2.1.6.5. Viewer

| Admin | Search | Browse | Workflow | Viewer | Reports |
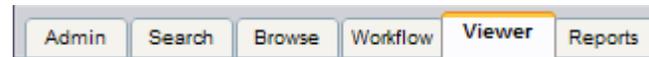
⬆ Figure 8: Viewer Tab

The Viewer tab/module allows the user to view content in a read-only environment. In this mode, the user can **NOT** permanently alter a system-resident file in any way, though zoom, rotate, flip, book view and other presentation options exist. The application natively supports a large number of popular formats. In the event that DPM can **NOT** internally handle rendering of a file, it hands off the task to another application. Content is **NOT** directly accessible in the Viewer tab; the user must first choose a piece of content from elsewhere in the system, such as in the Search or the Browse modules or in cases where content is linked to task.

2.1.6.6. Reports

| Admin | Search | Browse | Workflow | Viewer | Reports |
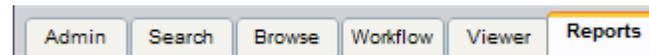
⬆ Figure 9: Reports Tab

For the *system administrator*, this tab has a dual role: it provides quick access to system audit, account maintenance and matrix reports; it also gives the means for the Administrator to manage personal user reports regarding content manager, content loader and workflow. For the *standard user* (i.e., one lacking system administrative privileges), this tab's singular function is to give the means for the same personal user reports mentioned above.

2.2. Dialog Boxes

2.2.1. Relationship with Parent

Dialog boxes appear horizontally and vertically centered atop the triggering shell display (that is, the DPM main window, not necessarily the center of the monitor, unless DPM is maximized). That is to say, the parent window is never blank; instead, it contains the fully rendered display the user had just seen prior to the pop-up launch. The only exception to this rule regards auto logoff, in which case the user sees a "stripped" version of the main application window.

Users often use buttons to launch dialog boxes to complete a specific operation, such as looking up and selecting users, groups, content, etc. Such button labels are verbs that describe the operation followed by an ellipse (…). The ellipse cues the user that the operation continues in a popup window, not the main application window. An example would be "Reassign…," which launches a dialog box, whereas "Own" occurs in the parent window.

Likewise, if the operation is known to consistently launch an external editor/viewer application or leverages some inherent Windows functionality—such as a "Browse Folder" dialog—it is appropriate to use the ellipse in such cases as well. However, in the event that an action switches the interface to another section within the same window—for instance, a different tab in the ODP—do NOT use the ellipses after the verb.

2.2.2. Pop-up Anatomy

2.2.2.1. Window Dimensions
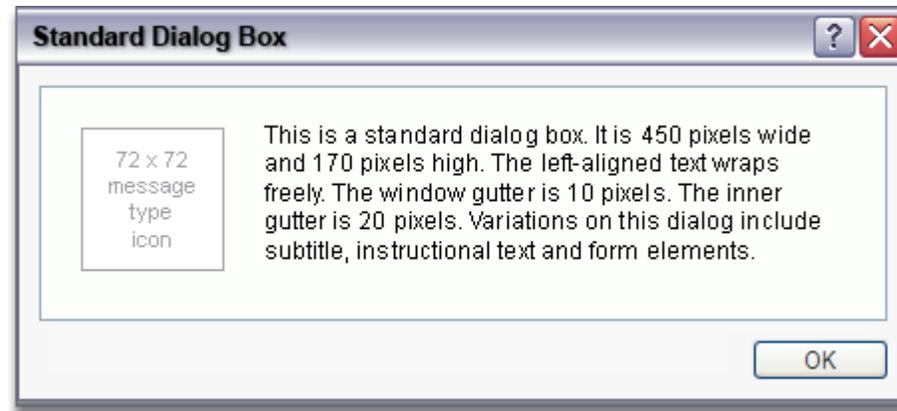
↑ Figure 10: Standard Dialog Box

Maximum typical dimensions are 500 pixels wide x 600 pixels high, though there are a few exceptions to this rule. Use scrollbars, but only if absolutely necessary. See actual-sized example image below. The standard dialog box's dimensions are fixed at 450 pixels wide x 170 pixels high.

Add Content

Window Title
≡
L

Screen-Level Help (F1)
≡
R

Close Window Button
≡
R

Add File/URL

Select format, enter metadata and upload.

Format

URL

Name

Content Type

Description

Comments

Source

Editor Select...

Artist

Developer

Analyst Select...

Manager

✱ = required

Add   Cancel

Window Sub-title
≡
L

Instructional Text
≡
L

Metadata Scroll Bar
≡
R

Form Field Labels
≡
R

Editable Field
≡
L

Required Field Icons
≡
R

Non-Editable Field
≡
L

Required Field Tip
≡
L

Primary Task Buttons
≡
R

⬆ Figure 11: Jumbo Dialog Box with Form

2.2.2.2. Title Bar

For the sake of consistency in informing the user of where they are and what the consequences are of their actions, dialog window titles take on the verbiage of the trigger that spawned them; thus, a button that reads "Add Content" results in a dialog box with an

"Add Content" title. A "Reset Password" dialog will have assumed the namesake of its corresponding cascading menu selection. Both the icon and the "Horizon Enterprise Content Manager" text are absent from pop-ups.

Note that title bar specifications for dialog boxes differ from those of the main window. See also status bar.

Present in Title Bar (help and close)	**Absent from Title Bar** (minimize, maximize and restore)	**Button(s)** (usually just "OK")
[?] [X]	[_] [□] [⧉]	[OK]
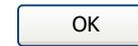
2.2.2.3. Modality and Resizing

All dialog boxes in DPM are modal, *except for Help*. Help is not modal by design, so that the user can continue to simultaneously work in DPM and reference the Help window at will. Modality is application-level, not system. Dialogs do **NOT** resize in any fashion.

2.2.2.4. Status Bar

Dialog boxes do **NOT** have status bars.

2.2.2.5. Font Properties
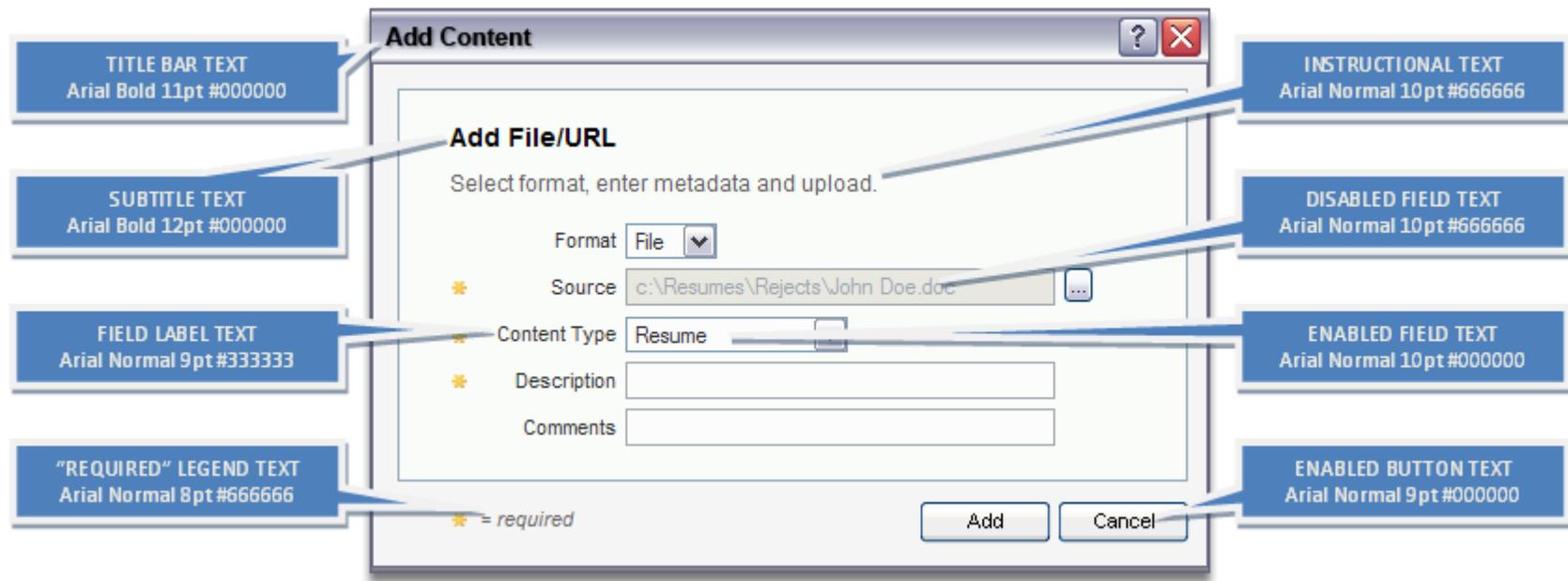
⬆ Figure 12: Dialog Box Font Attributes

2.3. Forms

2.3.1. Layout

In general, form elements are stacked vertically in a form. Form elements may display in two columns. Think of a form element as comprised of four parts:

- required field indicator, when applicable
- form element text label
- the (enabled or disabled) form element itself
- post-validation tooltip-equipped error icon, when applicable

Often data or operations logically group together on the same screen and in close proximity. For example, add, delete, view and

rename operations can fall under the common label "Manage." Likewise, data like street, city, state and zip code can be said to fall under the category "Address Information." Be constantly alert to these logical conglomerations, as they make the user interface more intuitive, easier to use and orderly.

2.3.2. Common Elements

DPM uses a number of form elements users recognize from the Windows operating system and browser-based applications. See also Field Labels. **Note:** appearance will vary between operating systems; examples are from Windows XP Service Pack 2.

2.3.2.1. Labels

All labels sit right-aligned to the left of form elements. Labels next to (radio buttons and checkboxes) are also clickable. A colon (:) does **NOT** follow any label. Labels do **NOT** wrap, except in extreme circumstances when severe space limitations so dictate.

2.3.2.2. Sortable Columns

Tables are always sortable by column header, unless otherwise specified. A subsequent mouse-click on a column header reverses the previous sort. Tables are default sorted by the first column in ascending order, unless otherwise specified.

Clicking a different column to sort triggers an ascending order sort. Click the same column a second time to descend-sort. Sorting is case-insensitive (i.e., a=A). Ascending sorts have an upward-pointing triangle, while descending have a downward-pointing triangle.

Null or blank values are treated as the first items in an array sorted in ascending order. See sort order for column values, below.

Sort Type	Alphabetical	Numerical	Alphanumerical	Dates	Directories	Checkboxes
▲ Ascending	Low → High (e.g., null, a, b, c…)	Low → High (e.g., null, 1, 2, 3…)	Low → High (e.g., null, 1, 2, 3, a, b, c…)	Earlier → Later (e.g., null, Jan. 1, Feb. 1, Mar. 1…)	Folders → Files Low → High	☑ → ☐
Descending ▼	Low ← High (e.g., c, b, a, null…)	Low ← High (e.g., 3, 2, 1, null…)	Low ← High (e.g., c, b, a, 3, 2, 1, null…)	Earlier ← Later (e.g., Mar. 1, Feb. 1, Jan. 1, null…)	Folders ← Files Low ← High	☑ ← ☐

Note: Checkboxes take primary sort when their headers are clicked, then the next column (e.g., "Name") takes secondary sort.

The following example shows an alphabetical, ascending sort of some fictitious files.

⬆ Figure 13: Column Header Sorting

2.3.2.3. Text Fields

Standard Size

- **Width:** 200 px fixed
- **Height:** one 20-px-high row (including 1-px border)
- **Max Characters:** 80
- **Wraps:** No

Intermediate Size

- **Width:** 500 px fixed
- **Height:** one 20-px-high row (including 1-px border)
- **Max Characters:** 250
- **Wraps:** Yes

Jumbo Size

- **Width:** 500 px fixed
- **Height:** three 20-px-high rows (including 1-px border)
- **Max Characters:** 500

- **Wraps:** Yes

Hybrid Size
- **Width:** 200 px fixed
- **Height:** three 20-px-high rows (including 1-px border)
- **Max Characters:** 250
- **Wraps:** Yes
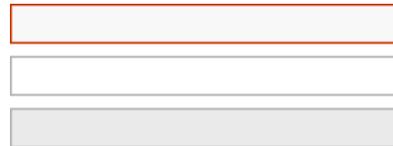
⬆ Figure 14: Standard Text Fields (Error, Active and Inactive)

Use text fields to collect free-form user text input and secondarily to display pre-populated text. They are blank by default, unless otherwise specified. Wrapping applies to jumbo-sized fields only. Employ such fields sparingly. For example, "Comment" and "Description" fields implicitly call for longer text strings than, say, "First Name." Where two columns of elements are common, span long fields across the columns (so that they do not dictate column width). Examples are for comments or descriptions, which require more space). For more information on colors and fonts attributes associated with text fields, see Messaging and Fonts.

2.3.2.4. Dropdowns

Standard Size

- **Width:** variable, dictated by content (no fixed size)
- **Height:** one 20-px-high row (including 1-px border)
- **Max Characters:** N/A
- **Wraps:** No

⬆ Figure 15: Dropdowns

Primarily used for selecting a single value from two or more that must be explicitly spelled out (for simple toggles—a simple binary choice—single checkboxes, checked or unchecked, are usually more useful and real estate-efficient). The default initial value is

"Select…" or "Select <item>…", unless otherwise specified. In the absence of other instructions, values should be shown in alphabetical order. Length of the element is only as long as the contained text (i.e., no fixed width).

2.3.2.5. Checkboxes

Standard Size: .NET default

☑ Enabled and Selected
☐ Enabled and Unselected
☑ Disabled and Selected
☐ Disabled and Unselected

⬆ Figure 16: Checkboxes

Checkboxes are binary switches: they toggle a single item on and off. By default, checkboxes are blank (unchecked), unless otherwise specified. Dimensions are Windows standard. Checkboxes are center-aligned in columns, when used. However, DPM does not use Checkboxes to select tabular data. Clicking the box itself or the label toggles the checkbox if it is enabled. For checkbox sorting details in grids, refer to the table in the [Sortable Columns](#) section.

2.3.2.6. Radio Buttons

Standard Size: .NET default

◉ Enabled and Selected
◯ Enabled and Unselected
◉ Disabled and Selected
◯ Disabled and Unselected
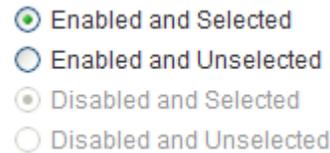
⬆ Figure 17: Radio Buttons

Radio buttons are also binary switches when used in pairs. When three or more are used in conjunction, they present the user with mutually exclusive selection options (i.e., only one can be chosen at a time). By default, radio buttons are blank, unless a default selection is otherwise specified. Radio buttons align left with other form elements, but generally labels do **NOT** accompany them to their left. A group of radio buttons extends out horizontally on the same line, as they are **NOT** stacked vertically (unless there is need to wrap, which should be a rare occurrence). For an example of a horizontal group and its affect on tab order, see [Tab Key and Tab Order](#).

2.3.2.7. Spinner Controls

Standard Size

- **Width:** variable, dictated by content (no fixed size)
- **Height:** one 20-px-high row (including 1-px border)
- **Max Characters:** N/A
- **Wraps:** No
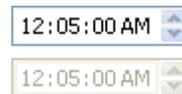
⬆ Figure 18: Spinner Controls

Spinners are a convenient way to increment or decrement a selection. The user may left-click on the arrow buttons or use the vertical arrow keys (Up is next, Down is previous). The horizontal arrow keys transfer selection left and right to jump from item to item (in the example above, between hours, minutes and seconds. As of this writing, the only example of a spinner control in DPM is the Time Picker.

2.3.2.8. Mover Boxes

Standard Size: Variable, depending upon pane and case-by-case use (no fixed size)

⬆ Figure 19: Mover Boxes

Mover boxes are used to transfer list items back and forth between two text areas. By default, usually one is blank and the other is

populated. DPM mover boxes have grids to clarify the line item through descriptions, comments and/or other cues.

2.3.2.9. Buttons

Standard (Minimum) Size: 69 pixels wide and 21 pixels high; variable to accommodate content, as required*

69 x 21 px	All buttons must comply with the minimal size directive, even those with very short text on them, such as "OK." Exceptions are specialty buttons, such as title and toolbar, look-up, mover buttons and the like.

*If button text exceeds the standard 69 pixels wide, the text plus its six-pixel buffer dictate button size. See below for illustration.

Some Long Text in Here
= six pixels wide
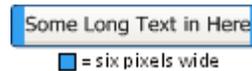
⬆ Figure 20: How to Determine Long Button Size

Buttons that are not (yet) usable at a certain point in an operation are visible, but disabled. If, based on a user selection, a button becomes relevant to the operation (for instance, the user enters all required information into a given form), the button becomes enabled. The user may then proceed. For more information, see Enabled, Disabled or Absent?

Text on buttons use verbs in the imperative and often a noun object for clarification. For example, "Apply Settings" or "Add" is more immediately descriptive and useful at a glance than a vague "OK," which, by itself, doesn't tell the user what the button explicitly does.

Also, word choice for a given button/action must remain constant throughout the application. For instance, although "Add," "New" or "Create" may all be equally valid choices for a certain operations, one must pick one and consistently use it. For this reason, the following table is maintained to keep track of whether the same button with the same function elsewhere in the application reads, say, "Quit," "End" or "Finish" instead of "Close."

Button	Description / Purpose	Status
...	Launches a look-up dialog.	
< <<	Transfer Left (all items and single item, respectively); used in mover boxes.	
> >>	Transfer Right (single item and all items, respectively); used in mover boxes.	
Apply Settings	Saves and immediately applies any changes made to a (usually admin) form.	

Cancel	Quits the current operation without saving changes (if applicable), but does not exit application.	
Change Password	Provides functionality to change a password.	
Close	Closes and cancels the operation. It returns to a "default" operation specifically defined for that operation area.	
Continue	Used for intermediate steps in a multi-screen operation.	
Create	Triggers the operation that enables the *non-admin* user to add a new item.	
Delete	Deletes the active record shown in the displayed form.	
Exit	Closes all open windows and exits the application to the OS desktop.	
Generate	Instructs the application to generate some value, for example, a temporary password.	
Log Off	Logs user off of application, but does not quit the application.	
New	Triggers the operation that enables the *admin* user to add a new item.	
OK	Used in dialogs when the user has nothing to do but acknowledge the box to dismiss it.	
Refresh	Refreshes the view (field values, grid data, etc.) of a form by asking the server for the most recent data. Used primarily in operations where a list needs to be refreshed from the server more frequently than upon form submission, or when the user wishes to verify visually the addition of new data in a record or a file to the system.	
Reset	Clears all user-entered data and resets a form to its default state.	
Resubmit	See Submit.	
Retry	Closes a dialog, so the user can return to a given task and try again.	
Save	Generic button for general use across the application to save changes and finish a particular operation.	
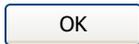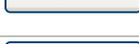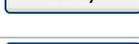
Save Settings	Saves any changes made to a (usually admin) form.	
Search	Clicked after a user enters parameters to run a search.	
Select	Indicates selection of a line item in a grid.	
Skip	Allows the user to decline a system prompt to undertake some optional operation.	
Submit	Submits and saves data, but does **NOT** close an operation. Form remains populated and the record remains active in the form until the user clicks [Close] or [New], or selects a different item to edit.	
Unlock Account	Provides functionality to unlock a locked account.	
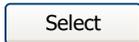
2.3.2.10. Grids

If the length of the data in a row cell exceeds the amount of space for the column the name will be truncated and will end in "…". If a user hovers over the data, the tooltip will display the data in its entirety. Neither the "…" nor the tooltip apply to a grid column header.

2.3.3. Spacing, Placement and Alignment

2.3.3.1. Horizontal Alignment

Alignment	Description / Details
Left	Numerical grid data
Left	All form elements (text fields, dropdowns, checkboxes, radio buttons, etc.)
Left	Most non-numeric text in tables, whether in fields or in cells
Left	Breadcrumb trail
Left	Text in the LSP, including banners
Left	Menu items

≡ L Left	Buttons on the button bar
≡ C Center	Dates and date ranges in tables
≡ C Center	Tab labels (both active and inactive)
≡ C Center	Text and icons on buttons
≡ R Right	Field labels
≡ R Right	Control buttons themselves
≡ R Right	Object count display

2.3.3.2. In Dialog Boxes
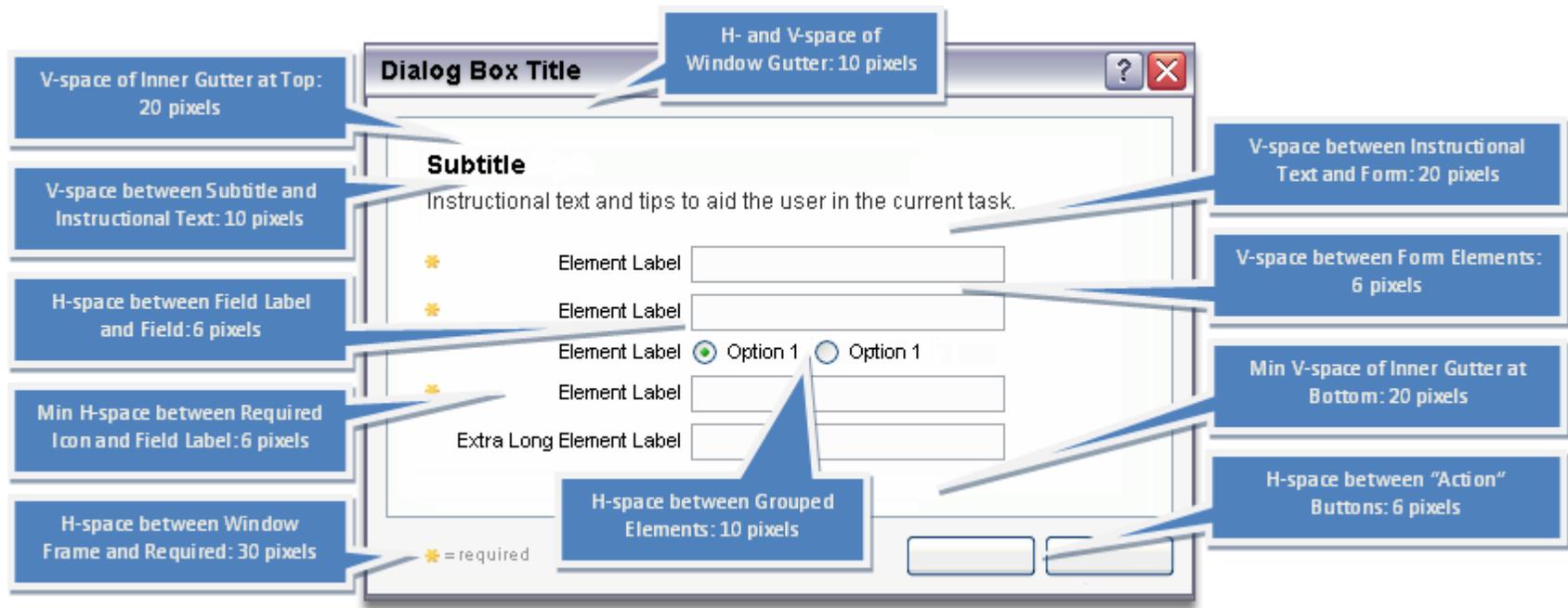
↑ Figure 21: Form Element Spacing in Dialogs

2.3.4. Special Considerations

2.3.4.1. Date Picker
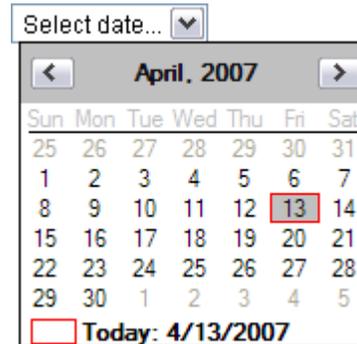
⬆ Figure 22: Date Picker

DPM uses a calendar control developed in-house. It offers additional customization options over the default .NET one. Specifically, it initially displays no date and reads "Select date…," as required by, for example, Search. It accepts manual input or use of a dropdown-navigable calendar to choose a specific date.

Manual Input

The user can **NOT** directly type in a date. Instead, the user must open the control and choose

2.3.4.2. Time Picker
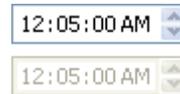
⬆ Figure 23: Time Picker

DPM uses the standard Microsoft .NET clock control. It uses a spinner control to increment or decrement hours, minutes and seconds. The user may optionally jump between the three using horizontal arrow keys, and increase or decrease numbers using the vertical arrow keys. The time picker inherits its format (e.g., a 12- or 24- hour display) from the operating system's regional and language settings.

2.3.4.3. Required Fields

Occasionally, the user must address certain form elements in order to proceed with a given DPM operation. For example, user name and password fields are mandatory in order for the user to log on to the system. The "Required Field" marker appears in any pane that houses mandatory form elements. It is not uncommon for all panes to simultaneously display the marker. The marker image has the "required" text included in the image itself; no additional text is necessary.

⬆ Figure 24: Required Field Indicator Key

The icon in the marker aligns vertically with the required icons (✳) next to each field in the first column (see the vertical arrow shown in the image below). The marker simultaneously aligns horizontally with the center of the main buttons in the pane (see the horizontal arrow shown in the image below).

⬆ Figure 25: Required Field Marker Alignment

Note: In addition to any other existing validation rules, "Space" and "Tab" key entry in required fields is invalid and will generate an error.

2.3.4.4. Enabled, Disabled or Absent?

What manner of alchemy or voodoo explains when a given control—be it a button, menu option, tab or other element—is visible or invisible, enabled or disabled? The answers to a few questions make it clear:

- **Absent** – Does the user possess sufficient security rights to view it? For example, a regular user will never see a strictly Admin-level control, because it is absent from the application altogether. As a general rule, if the user will **NEVER** have an opportunity under any circumstances to use a given element, *remove it from the interface*.
- **Disabled** – Have all logical preconditions for a certain action been fulfilled? For instance, assuming a user has the appropriate access, a "Save" option will activate only when in an editable module (**NOT** read-only) *and* the user modifies some aspect of the contained data. Until these criteria are met, the "Save" command remains disabled (i.e., visible, but inactive). Even if a user **RARELY** has an opportunity (even under a single set of circumstances) to use a given element, *include it in the interface*.
- **Enabled** – Whereas security rights and preconditions dictate the appearance and functionality of many elements, others are perpetually enabled, because they **ALWAYS** meet all of the aforementioned criteria. For example, the options in Help are available to any user at any time, without regard to contingencies or security levels. Users can elect to cancel an operation, jump between tabs or exit the system at any time.

2.3.4.5. File Size

Content file size will be displayed to the user following the below rules:

- Show and round values out to no more than two decimal places, as appropriate.
- If value is a whole number (all values after the decimal are zero), no decimal is necessary.
- If hundredths digit (the second one after the decimal) is zero, show only to tenths (just the first one after the decimal).
- If the value of the tenths digit is zero, but that of the rounded hundredths is 1 – 9, show to hundredths (both digits after the decimal).
- **RIGHT:** 1 MB, 2.5 KB, 5.05 TB
- **WRONG:** 1.00 MB, 2.50 KB or 5.0 TB

Range	Display Format	Abbreviated?	Notes
0 bytes – 999.99 bytes	bytes	No	Use entire lowercase word
1 KB – 999.99 KB	KB	Yes	Stands for "kilobytes"; use uppercase
1 MB – 999.99 MB	MB	Yes	Stands for "megabytes"; use uppercase
1 GB – 999.99 GB	GB	Yes	Stands for "gigabytes"; use uppercase
1 TB – 999.9 TB	TB	Yes	Stands for "terabytes"; use uppercase

2.4. Mouse and Keyboard Input

2.4.1. Mouse

2.4.1.1. Left Click

Left mouse clicks select rows in grids, activate controls and place focus on fields, as applicable. Sustained left mouse clicks provide drag-and-drop capability. Shift+ click allows selection of multiple rows in lists of content and/or folders. Are these true?

2.4.1.2. Right Click

Right mouse clicks give rise to contextual menus for specific relevant operations to a given module in accordance with rules regarding content manipulation, the user's permissions, etc. Is this true?

2.4.1.3. Double (Left) Click

Double mouse clicks do what?

2.4.1.4. Pointers

OS settings dictate the appearance and behavior of mouse pointers. There is no contravening customization at this time.

Arrow Pointer   Horizontal Resizer   Text Selector   Vertical Resizer

⬆ Figure 26: Mouse Pointer Reference

2.4.2. Keyboard

Physical impairment, environmental factors, hardware availability and maintenance and a host of other variables affect the user's input device options. As a matter of principle, the UI team recognizes the importance of keyboard-only access to every facet of the application. Although initial release does **NOT** attain an analogous level of utility and navigability to mouse input, future versions will.

Please note that the following is **NOT** a comprehensive account of all supported Windows standard hotkeys and key combinations, just those deemed most relevant for the majority of key DPM operations.

2.4.2.1. Menu Hotkeys

In accordance with standard Windows behavior, each cascading menu patriarch accepts keyboard input to receive focus and highlight the first selection, as denoted by the underlined letter.

Second- and third-level cascading menu selections also accept keyboard navigation. Once the user chooses a menu, a subsequent letter key press—corresponding to one of the underlined letters in the menu options—shifts selection to that item. Note that the letter assignments *per menu* must be unique, but *the same letter may be used in any or all menus without adverse consequences*, since no more than one cascading menu can ever be activated and in use at the same time.

2.4.2.2. Function Keys

⬆ Figure 27: F1 Key

Windows Default:	Displays help.
Application:	Launches universal application help.
States Toggle:	No.

⬆ Figure 28: F5 Key

Windows Default:	Updates (refreshes) the active window.
Application:	Refreshes view.
States Toggle:	No.

⬆ Figure 29: F8 Key

Windows Default:	None.
Application:	Exits the application to desktop.
States Toggle:	No.
Notes:	Equivalent to Windows' ALT+F4, which is also supported.

⬆ Figure 30: F10 Key

Windows Default:	Activates the menu bar in the active program.
Application:	Same as Windows default.
States Toggle:	Yes.

↑ Figure 31: F12 Key

Windows Default:	Shows/hides the toolbar in Internet Explorer.
Application:	Same as Windows default.
States Toggle:	Yes.

2.4.2.3. Tab Key and Tab Order

Specifying a tab order allows users to access items in a logical order. Otherwise, applications generally present the sequence in the order it appears in the source code, which may appear haphazard and difficult to understand. Unless otherwise specified, tab order logically progresses line by line from top to bottom, left to right.

↑ Figure 32: Tab Key

Menus: Cycles vertically through menu selections or horizontally to a child menu.
Fields: Jumps between *editable elements* in a predetermined tab index.
Trees: Shifts focus from tree to next *editable element*, but retains selected directory.
Grids: Moves focus from grid to next *editable element* outside the grid.
Notes: The Shift+Tab key combination reverses the direction of the above actions.

A few notable rules regarding tab order in forms follow:

- Error conditions do **NOT** alter tab order (in other words, tab order remains the same whether field data is erroneous or not).
- Tabbing entirely skips field labels; titles, subtitles and other text; plus "Required Field," error and other images and disabled form elements, such as read-only fields.
- In two-column scenarios (such as that shown in the image below), the left column completes first row-by-row, then jumps to the top of the right column and proceeds in like manner to the end of the form (consider jumbo fields part of the left column, though they span).
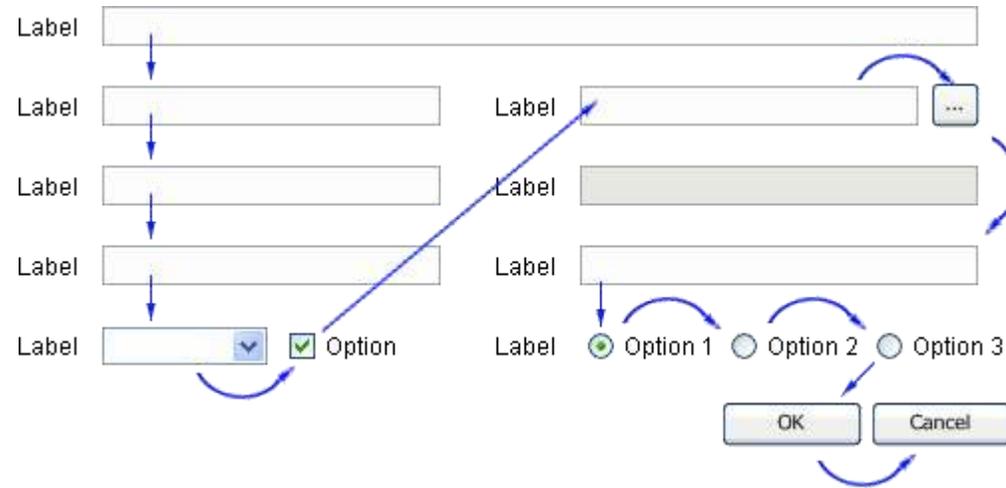
↑ Figure 33: Form Tab Order Example

Note: spacing is not to scale in the above image to more easily illustrate tab order.

2.4.2.4. Other Keys

↑ Figure 34: U/D Arrow Keys

Menus: Cycle vertically through menu selections in the active menu.
Fields: Expands dropdowns; shifts a row up/down in list boxes, text areas, dropdowns, etc.
Trees: Up selects previous shown item (above); Down selects next shown item (below).
Grids: Jumps between *editable elements* in a column; moves selection one row.
Notes: In a radio button group, Up selects previous, Down selects next

↑ Figure 35: L/R Arrow Keys

Menus: Cycle between top-level menus; expands menus and selects top-most option in each.
Fields: Moves the cursor through text one character forward or backward at a time.
Trees: Left collapses folders; Right expands folders.
Grids: Jumps between *editable elements* in a row.
Notes: In a radio button group, Left selects previous, Right selects next

↑ Figure 36: Spacebar

Menus: No effect.
Fields: Selects/deselects checkboxes.
Trees: No effect.
Grids: No effect.

↑ Figure 37: Delete Key

Menus: No effect.
Fields: Deletes selected contents of a field, but **NOT** the field itself.
Trees: Deletes objects, if the user has sufficient rights, and no system dependencies exist.
Grids: Deletes objects, if the user has sufficient rights, and no system dependencies exist.

↑ Figure 38: Shift+Click

Menus: No effect.
Fields: No effect.
Trees: Select multiple adjacent objects, if the user has sufficient rights and module supports.
Grids: Select multiple adjacent objects, if the user has sufficient rights and module supports.

↑ Figure 39: Control+Click

Menus: No effect.
Fields: No effect.
Trees: Select multiple objects, if the user has sufficient rights and module supports.
Grids: Select multiple objects, if the user has sufficient rights and module supports.

2.5. Images

2.5.1. Library

All images share a naming convention and remain consistent with the file names listed in this document.

Below is the current list of images used. Icons with active and inactive states follow the naming convention that active and inactive names are the same except for their suffixes, "-a" and "-i," respectively. For example, the active state for "open" is "open-a" and the inactive state is "open-i."

Review the following on a line item-by-line item basis, not by section.

2.5.1.1. Cascading Menu and LSP Banner Icons

Active	Inactive	File Name	Dimensions	Format	Description	In Library	Status

2.5.1.2. Other Application Icons

The following icons are listed in no particular order.

Active	Inactive	File Name	Dimensions	Format	Reference	Description	In Library	Status

2.5.2. Special Cases

2.5.2.1. Animation

There is no custom animation used in the application at this time. In general, if a situation arises that warrants the use of animation, limit animation to two instances per screen, preferably one.

2.5.2.2. Logos

There is presently no contravening direction on logo use. If a Deloitte corporate or product logo is employed, please refer to global company standards regarding their appropriate and approved use.

2.6. System Messaging

2.6.1. Types

Note: Example images below show only the "message type indicator" that appears superimposed over the module. Various application components have their own distinct icons. The icons apply only to messaging in dialog boxes, though the descriptions of message types hold for messaging within the main application window as well.

2.6.1.1. Alert

The Alert is used when the system warns of some serious or negative consequences of the user's or system's (in)action, *excluding errors.* It is employed when the user must know and *mandatorily* act upon a serious *non-error* condition. The modal dialog box often requires confirmation before proceeding and is phrased in the interrogatory. An example would be a dialog asking the user "are you sure you want to delete this very important item that will affect all users?"

2.6.1.2. Default

No corner icon is used.

The Default icon/message combination indicates an "at rest" or an *a priori* state, in which none of the other message categories apply. For example, a logon dialog in the absence of any errors, alerts or informational dialogs would be considered "default," because no adverse effects have yet been encountered.

2.6.1.3. Error

DPM uses a somewhat configurable, "plug-and-play" Microsoft .NET error messaging solution for errors system-wide: in dialog boxes and all the panes. The Error message is self-evident: the system warns of some negative consequences of the user's or system's (in)action or other problems, such as transfer failures, unsupported types, invalid data or incorrect credentials. These messages often suggest specific solution(s). Like an Alert, the user must *mandatorily* act upon the information, because some chain of events has been interrupted. The user must directly intervene and comply fully

before continuing.

Note: Disable the blinking icon feature. Customize field and label colors as shown. Tooltip includes specific message.
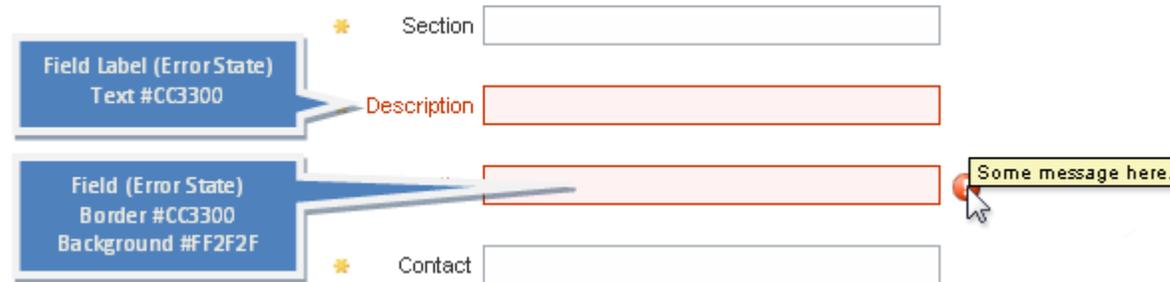
⬆ Figure 40: DPM Error Handling

2.6.1.4. Info

The Info message is usually positive or neutral feedback, statements, tips, hints or directions regarding an operation the user simply acknowledges (such as clicking "OK") or can *optionally* undertake ("Your password expires in *x* days. Do you want to change your password now?" In this case, the user may choose "Change Password" or "Skip"). The system reports a status (or change of status) in a declaratory "FYI" fashion and may phrase voluntary operations as a question.

2.6.1.5. Success

The Success message can be considered a special type of Info message. It provides the user positive feedback regarding an event, especially one that immediately impacts system or personal settings or some aspect of how the user interacts with the program. The user simply acknowledges (such as clicking "OK" after a "Something successfully happened").

2.6.2. Alerts

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.1. Default

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.2. Errors

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.3. Tooltips

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.4. Headings

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.5. Information

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.6. Instructions

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.7. Success

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.8. Other

Bid(s)	BID Name	Code	Message	Type / Notes	Status

2.6.2.9. Help

Both static Help (also known as "Global Help" or "Universal Help") and contextual Help are available for initial release. Universal Help (defined as non-specific help that presents all topics at once in sort of an "owner's manual" fashion) is available principally from the

main cascading menu. See also [hotkeys](). Screen-level Help is always available in both the [main window]() and in [dialog boxes](), unless otherwise specified, by the following access points:

Main Window	**Dialog Boxes**
(Contextual Help icon)	(Contextual Help button)

2.7. Tooltips

Use tooltips throughout the application for two purposes:

- to elaborate on the function of various controls
- to provide specific feedback messages (e.g., error details) and helpful tips

Only use tooltips when they give additional information; never have a tooltip read the same as the label/button text. Toolbar tips are readable sentences, punctuated and capitalized as such. They are short, concise and relevant. They do **NO**T wrap unless so long that they would otherwise extend beyond the display's viewable area. See the Error-handling section for details on tooltip use in conjunction with errors.

Note: column headers do **NOT** have tooltips.

2.8. Canned Reports

2.8.1. Standard Layouts

More information regarding standard headers, footers, tabular data formats and summary information will occupy this space.

2.8.2. Exceptions and Special Cases

More information regarding special reports and their properties will occupy this space.

2.9. Table of Figures

3.0. Glossary

3.1. Terms

3.1.1. A – F

Arial: a sans serif font common to most major computer operating systems

Baseline: status of an artifact denoting completion of the formal review process and subsequent approval

Best practices: methodical, repeatable processes that most efficiently and effectively produce a desired outcome

Boolean: a complete system for binary operations and algebraic logic named for George Boole, an English mathematician at University College Cork in the mid 19th century

Cascading menu: a collapsing navigation scheme that "unfolds" options from general to specific

Color(s):

Analogous – any adjacent colors on the 12-value color wheel

Complementary – any two colors directly opposite each other on the 12-value color wheel

Primary – red, yellow and blue, colors impossible to obtain from the mixture of any other colors

Secondary – green, orange and purple, colors formed by mixing primary colors

Theory – the idea that color choices affect mood, cognition and perception in positive and negative ways

Extension: the (typically three) letters/numbers after the dot in a file name, indicating the type of file to both the user and the OS

Grouping: the logical orienting of similar operations in proximity to enhance usability and the overall user experience

Gutter: the print term, now also common in computer UI parlance, for the space between columns of copy

3.1.2. G – L

Heuristics: best practice "rules of thumb" guidelines learned through experience to produce better quality product

3.1.3. M – R

Sans Serif: a font type lacking serifs

Pixel: short for "pixel element," one of the millions of "dots" on your display that comprise the computer's user interface

Section 508: portion of the Rehabilitation Act of 1973 – legislation that requires federal agencies to provide disabled employees with access to IT resources, such that their experience approximates that of able-bodied employees as much as possible

Serif: the flairs on the ends of some strokes that make up letters and symbols

Trebuchet MS: a sans serif font common to most major computer operating systems

Usability: the measure of how efficient, effective, engaging, error-tolerant and easy to use a product is for meeting users' needs

Verdana: a sans serif font common to most major computer operating systems

3.1.4. S – Z

No entries.

3.2. Acronyms

3.2.1. A – F

CL: Access Command List – the list of system permissions and rights assigned to a user role

BCT: Breadcrumb Trail – the navigable display that indicates the user's current location

BID: Behavioral and Interactive Design (Document) – *See* **UIS**.

BMP: Bitmap – a 1- to 24-bit lossless image type that supports neither animation nor transparency

DIB: Device Independent Bitmap – *See* **BMP**.

3.2.2. G – L

GIF: Graphics Interchange Format – an 8-bit lossless image type that supports animation and simple transparency

JPG (or JPEG): Joint Photographic Experts Group – a 16-bit lossy image type that supports neither transparency nor animation

3.2.3. M – R

OOS: Out of Scope – any feature, concept or functionality explicitly excluded from a project for a particular release

OS: Operating System – a computer program that controls a system's hardware components and file system, thereby facilitating the operation of other specialized programs; examples include Microsoft Windows, MacOS and Linux

PNG: Portable Graphics Format – a capable 8- to 24-bit image type that supports animation and sophisticated transparency

PX: Pixel – *See* **Pixel**.

3.2.4. S – Z

TBD: To Be Determined – something pending further consideration before a final decision is made

UCD: User-Centered Design – user interface design that focuses on the user's goals, mental models, operations and requirements

UI: User Interface – the system by which a computer user interacts with software and/or hardware to accomplish goals

URL: Universal Resource Locator – the Web address of, or link to, specific content on the Internet

4.0. Document History

4.1. Baseline Approval

4.2. Changes

Date	Description / Details