HTML5 Design with the Universal Theme in Oracle Application Express 5.0

Marc Sewtz Senior
Senior Software Development Manager
Oracle America, Inc.
New York, NY, USA

Keywords
Oracle Application Express, HTML5, CSS3, Responsive Design, Theme Roller

Introduction

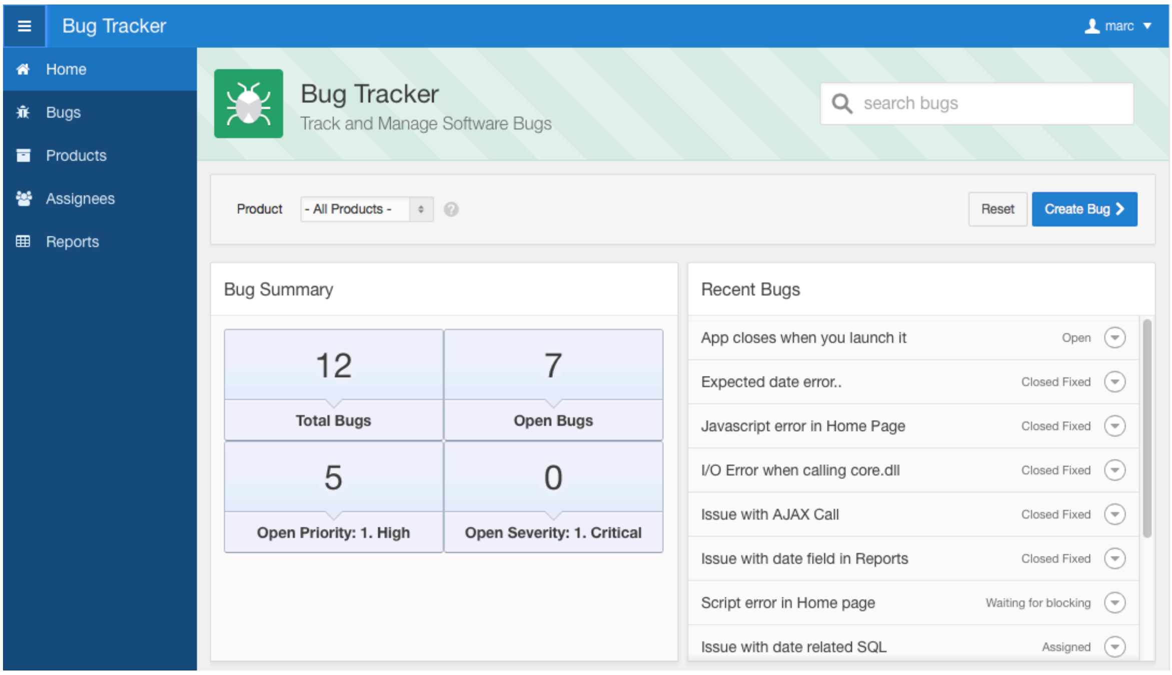
Oracle Application Express 5.0 (APEX 5.0) introduces a new approach to building modern HTML5 user interfaces for business applications. Since its inception, APEX has provided a flexible and open template architecture, which allows developers to keep the content of their application separate from the presentation layer, while providing UI designers and developers with easy access to the HTML and CSS code used in the templates that comprise the user interface of their applications. Recent trends in web design put an ever-stronger emphasis on the use of CSS for all styling elements of a web site and with the advent of responsive design, even the structure, location and size of the components presented on a web page are now often controlled through CSS. As a result of these shifts, the template architecture in Oracle Application Express needed to adopt a more declarative approach to applying CSS to the various components that make up a page. It’s no longer sufficient to simply choose between a page template that has sidebars vs. one without sidebars, select a form region template vs. report region template, or render a standard button or hot button. There’s now an infinite number of combinations of margins, padding, border styles, large buttons, small buttons, icon buttons, components that render in different sizes and different positions depending on available space, etc. To address these challenges, Oracle Application Express 5.0 is introducing three new features that enable any developer to build great looking web applications, regardless of UI skills or in-depth CSS knowledge: The Universal Theme, Template Options and Theme Styles.

Universal Theme

The Universal Theme is a new theme introduced with Oracle Application Express 5.0. In the past, APEX developers would choose a theme from a gallery of themes included in the product and select the one that most closely resembled their desired look & feel. With APEX 5.0 applications are generated using the Universal Theme by default. While older themes are still available as legacy themes, the Universal Theme is designed to meet most business needs by providing a large variety of customization options, including a selection of pre-built theme styles, template options and support for the APEX Theme Roller. Thanks to its built-in, declarative responsive features, the Universal Theme is well suited to work on a variety of different screen sizes and devices.

The Universal Theme also uses a new type of navigation. While applications in the past typically used tabs to allow navigation to different sections of an application, the Universal Theme uses list-based navigation. This allows for multi-level hierarchical menus, which can be implemented in a number of styles. The Universal Theme supports a horizontal navigation control, implemented as drop-down menus, as well as a vertical navigation control in the form of a collapsible sidebar menu.

Applications using older themes and tab-based navigation can be upgraded to the Universal Theme by switching the theme from the old theme over to the new theme. During the process, tabs will be migrated to list-based navigation, provided only one level of tabs was used. For applications using two levels of tabs, it is recommended to first manually change the tab structure to one level of tabs. The positioning of the new list based navigation can be controlled when editing the currently active user interface under shared components and selecting top navigation vs. sidebar navigation.



Product: - All Products -

Bug Summary

12 Total Bugs	7 Open Bugs
5 Open Priority: 1. High	0 Open Severity: 1. Critical

Recent Bugs

App closes when you launch it	Open
Expected date error..	Closed Fixed
Javascript error in Home Page	Closed Fixed
I/O Error when calling core.dll	Closed Fixed
Issue with AJAX Call	Closed Fixed
Issue with date field in Reports	Closed Fixed
Script error in Home page	Waiting for blocking
Issue with date related SQL	Assigned

Figure 1 - Universal Theme

The Universal Theme is used as the new standard theme for all packaged applications that ship with Oracle Application Express. There are several new UI concepts and best practices being introduced with this theme, which are showcased in these applications. This includes a standard dashboard homepage, featuring a prominent header region that shows the application name and search as well as a main content area showing high level reports and HTML5 charts – when applicable. The Universal Theme supports reports with report headers that remain frozen at the top of the page as the user scrolls down in the report, a new sidebar navigation, that can be collapsed and expanded based on available space and an application navigation bar that is rendered as a drop-down menu.

When building applications with the create application wizard, the Universal Theme will be selected as the default theme and the generated application will have the theme subscribed to the theme repository. This allows for pulling in future updates to the Universal Theme without having to switch to a newer version of that theme. The same theme-level subscription feature is available within workspaces as well, so developers will be able to build applications using custom themes that are subscribed to a workspace master theme. Any subscribed theme is read-only, meaning that none of the theme attributes can be modified and all templates and template options that are defined in the master theme are read-only. Developers who wish to modify templates in a subscribed theme have the option to create a local copy a subscribed template, i.e. the copy will no longer be tied to its master template and thus will be editable. Changes to the master theme or any of the templates in the master theme can be pushed out to all subscribing themes. This will update all subscribed templates and template options, remove any templates that were removed from the master and create any new templates in the subscribing theme. Templates that are defined locally in the subscribing theme remain untouched.

Template Options

Template Options provide a declarative way for developers to apply different styles to components on an APEX page. Template options for example allow for applying different colors or accents, different border styles, different spacing and padding, render buttons in different styles, with and without icons, form fields with different alignments, etc. All of these variations previously would have required a large number of nearly identical templates or in-depth knowledge of CSS. Template Options solve this problem by dynamically applying CSS classes to a component at runtime, based on the template options selected by the developer. The APEX Builder interface presents the developer with a choice of template options, which are available for most components. Template options are shown in the Builder using descriptive names and optionally include help text. Components that support template options are:

- Pages
- Regions
- Classic Reports
- Breadcrumbs
- Lists
- Items / Labels
- Buttons

Template options are defined on the theme and template level and applied when editing a component. Theme level template options - also known as global template options - can be edited on the theme edit page and are organized by component type, i.e. developers can define global page template options, global region template options, etc. A global template option is available for all components of a given type. Template level template options can be edited on the template edit pages and are only available to those components that reference the specific template.

Template Option Groups and Presets

A template option is defined by its display name, an identifier string, one or more CSS classes, which are referencing the theme's underlying CSS file, an optional help text and an optional template option group. Template option groups are global and thus are defined on the theme level. A template option group defines the purpose of a number of related template options. Examples for template option groups are “Button size”, “Button style”, “List style”, “Form Label Position”, “Padding” etc. So a “Button size” template option group for example would group together all button template options that allow for applying different button sizes, such as “Small”, “Large” etc.

Grouped template options allow for selecting a template option preset on the template level. A template option preset is applied to a component upon creation. For example, when creating buttons using a template that has the template options “Large” and “Small” defined as part of a template option group “Button Size”, and “Small” was defined as the preset, then any new button referencing this template would automatically have the template option “Small” applied.

Template option groups also allow for the definition of a NULL text. The purpose of the NULL text is to document the behavior of a component when none of the template options within the group are set. In the above example, if the “Button Size” has a NULL text defined as “Default Button Size”, then the developer could choose to either use the “Large” button template option, or the “Small” button template option, or not select a template option for this group, meaning the button would be rendered with its default button size.

Template options that are part of a group are optional unless a template option preset is defined on the template level. So when editing a component that is referencing a template with a grouped template option set, which does not have any template option selected as the preset, then the template option on the component is NULL-able. If the grouped template option set does have a preset defined, then this preset is used when creating the component, and when editing the component, the template option select list does not show a NULL option.

The “Advanced” flag of a template option group allows for defining secondary template option groups and template options, i.e. when editing the component in either the component view or page designer, the “advanced” template options would be included in a separate “advanced” section, which is hidden by default.

Default Template Options

Template options that are not part of a template option group can be selected as the default for a template. Defaults are automatically applied when a component referencing the template gets rendered. Default template options are not written to the components itself, instead only a #DEFAULT# substitution string is defined on the component, thus allowing to centrally modify the template option defaults when editing the component’s template. Developers can choose not to use the template option defaults, which removes the #DEFAULT# substitution string from the component. It will then be possible to selectively apply available template options to the component – even those that were previously part of the default. All template options that are selected directly when editing the component are written to the component’s meta data, represented by their corresponding CSS class string. At runtime, the selected template options as well as the template option that are part of the default – if enabled – are combined with the component’s CSS classes string and applied to the component. This means developers are able to augment template options with their own custom CSS classes.

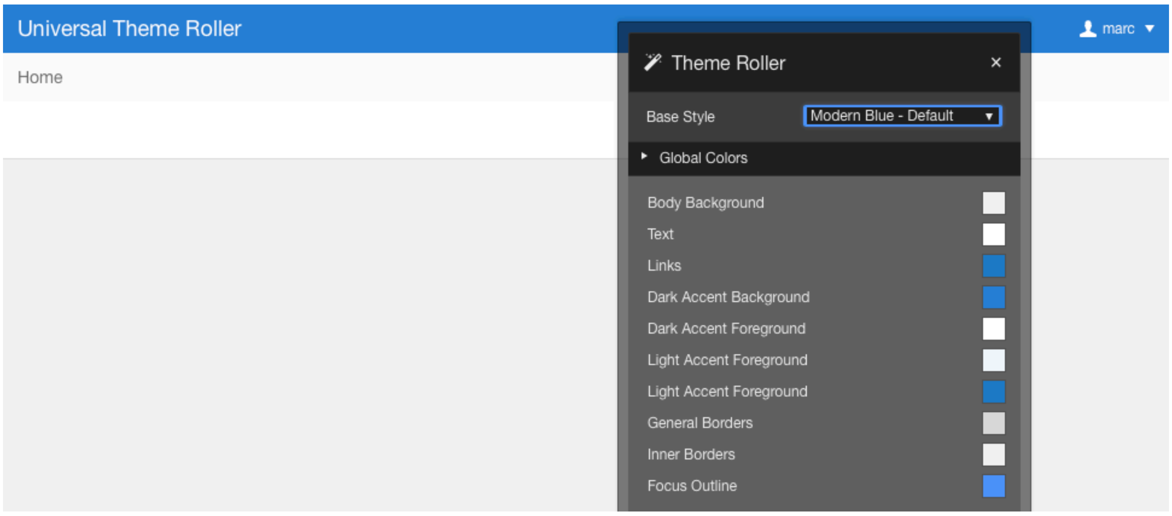
Editing Template Options

Developers can modify existing template options and add new template options to templates. When existing template options are modified, some changes need to be propagated to the components that reference the respective templates. Components reference template options by their class names, so if the CSS class name of a template option changes, all components that reference this template option are updated to change the CSS class string to the new string defined in the template option. Other template option attributes, such as the template option name, display sequence, the group and the help text can be safely modified without requiring any updates to the components that reference them. The template option identifier cannot be modified, as it is needed for mapping purposes. When deleting a template option on the template edit page, any component that references this particular template option is updated to have this string be removed. When adding new template options to a template, then these new template options are going to be available when editing the components that reference the template. If the new template option is set to be a default template option, then all components referencing the template will automatically include the new template option. Adding a new template option as a preset does not have any effect on existing components, as presets are only used upon creation of a component.

Template Options provide an easy way for developers to apply a variety of different styles to their pages, regions, and other components without having to understand CSS. Using Template Options ensures that only those CSS classes are actually used that are defined for the current theme, rather than having developers define their own CSS classes and styles on the fly in the page templates or directly on individual elements. This in turn allows theme developers to easily make changes to the available template options and have them propagated throughout all the components that make use of them.

Theme Styles

Oracle Application Express 5.0 uses the new Universal Theme for all new applications being created, while older themes are still available as legacy themes. In the past, developers would choose the theme that best met their requirements from a gallery of available themes. With APEX 5.0 and the new Universal Theme, developers will instead pick a theme style. A theme style is a collection of CSS files that provide a different look & feel based on the same underlying set of HTML5 templates. This makes it possible to have only a single theme, using the same underlying HTML markup, provide a variety of different looking styles. APEX 5.0 ships with a collection of pre-defined theme styles for the Universal Theme. The theme style can be selected when creating an application and can be modified to a different style by editing the applications user interface settings. The theme style attributes also allow developer to upload their own CSS files as a new theme style. This makes it possible for theme designers to create their own custom theme styles.



Base Style: Modern Blue - Default

Figure 2 - Theme Roller

While theme styles are defined using standard CSS files, and thus can be edited with any CSS editor, APEX 5.0 also includes the APEX Theme Roller, which can be enabled when running an APEX application using the developer toolbar. The APEX Theme Roller allows for applying styles to any element or component on the current page and to see the effect of these changes in real time. This can include changes to foreground and background colors, fonts, borders, border radius, etc. As the developer applies his UI adjustments to the current page, all his changes are stored locally in the Theme Roller widget. Considering that there's typically not any single page in an application that includes all available components, the Theme Roller allows developers to navigate to different pages in their application while retaining all current UI changes and applying them to other pages on the fly. This way developers can cover all the different parts of an application, like reports, forms, charts, etc. and once complete, the UI changes can be written back to the application's theme style settings, either as a new theme style or by updating the currently active theme style.

Summary

Oracle Application Express 5.0 makes building modern, responsive business applications easier than ever before. With the Universal Theme, Template Options and Theme Styles, developers can now take full advantage of the power of HTML5 and CSS3 without being a UI designer. The new APEX Theme Roller allows developers to customize theme styles to meet specific corporate UI requirements, create a new color scheme or page layout - all without having to write a single line of HTML or CSS.

About the Author

Marc Sewtz is a Senior Software Development Manager at Oracle Corporation in New York. With over eighteen years of industry experience, Marc held roles in Consulting, Sales and Product Development and now manages a global team of Software Developers and Product Managers in the Oracle Application Express (APEX) development team, part of the Oracle Database Tools group. Marc and his team are responsible for features such as the development of Mobile Web Applications with APEX, Reporting and Charting, Tabular Forms, Oracle Forms to APEX conversion and integration with Oracle Business Intelligence Publisher. Marc has a Master's degree in computer science from the University of Applied Sciences in Wedel, Germany.

Contact:
Marc Sewtz
Senior Software Development Manager
Oracle USA Inc.
120 Park Avenue, 26th Floor
New York, NY 10017
USA

Phone:	+1 212 508 7951
Email:	marc.sewtz@oracle.com
Web:	apex.oracle.com
Blog:	http://marcsewtz.blogspot.com/
Twitter	@msewtz