

Roller GSOC 2011 - mid term project review

Proposal and design

The purpose of the project is to add a mobile theme capability to Roller and at least one mobile theme. The project design was started on the Roller wiki. Here is the proposal:

<https://cwiki.apache.org/confluence/display/ROLLER/Proposal+HTML5+Mobile+Template+for+Apache+Roller>

The design's evolution is documented on the Roller mailing list and via a series of blog entries. Below is a list, with links, of each of the blog entries so far and my summary of the changes discussed.

Blog: Mobile Theme setting up and its Use Cases

<http://rollermobile.blogspot.com/2011/05/mobile-theme-setting-up-and-its-use.html>

Changes discussed:

- UI change - admin must pick mobile theme
- New Weblog property - defaultMobileTheme
- Rendering change - PageServlet sets weblog.editorTheme

Blog: Adding Meta-data in theme.xml

<http://rollermobile.blogspot.com/2011/06/adding-meta-data-in-themexml-to.html>

Changes discussed:

- New theme meta-data value - type in theme.xml

Blog: Tag to define template type in theme.xml

<http://rollermobile.blogspot.com/2011/06/tag-to-define-template-type-in-themexml.html>

Changes discussed:

- Type values: standard, mobile

Blog: Creating custom template

<http://rollermobile.blogspot.com/2011/06/creating-custom-template.html>

Changes discussed:

- Change to template management
 - Each template has a standard and a mobile parts

Blog: Editing templates

<http://rollermobile.blogspot.com/2011/06/editing-templates.html>

Changes discussed:

- Change to template UI

Blog: Changing Mobile Theme template

<http://rollermobile.blogspot.com/2011/07/changing-mobile-theme-template.html>

Changes discussed:

- Design for selecting mobile theme vs. standard theme

Actual changes

Implementation started well and made good progress. Below is a list of the changes, grouped into related blocks and AFTER each block, a comment about the changes.

```

M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/SharedTheme.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/SharedThemeFromDir.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/SharedThemeTemplate.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/WeblogCustomTheme.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/WeblogSharedTheme.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/StaticTemplate.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/StaticThemeTemplate.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/Template.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/Theme.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/WeblogTemplate.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/ThemeMetadata.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/ThemeMetadataParser.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/ThemeMetadataTemplate.java
M /weblogger-webapp/src/main/webapp/themes/sotto/theme.xml
M /weblogger-webapp/src/main/webapp/themes/basic/theme.xml
M /weblogger-webapp/src/main/webapp/themes/brushedmetal/theme.xml
M /weblogger-webapp/src/main/webapp/themes/fauxcolly/theme.xml
M /weblogger-webapp/src/main/webapp/themes/frontpage/theme.xml

```

Those changes add a new property named type to themes and to templates. So far so good.

```

A /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/rendering/mobile/MobileDeviceRepository.java

```

That change adds mobile device detection. Good.

```

A /weblogger-webapp/src/main/webapp/themes/mobile
A /weblogger-webapp/src/main/webapp/themes/mobile/_day.vm
A /weblogger-webapp/src/main/webapp/themes/mobile/basic-custom.css
A /weblogger-webapp/src/main/webapp/themes/mobile/basic-preview.png
A /weblogger-webapp/src/main/webapp/themes/mobile/permalink.vm
A /weblogger-webapp/src/main/webapp/themes/mobile/searchresults.vm
A /weblogger-webapp/src/main/webapp/themes/mobile/sidebar.vm
A /weblogger-webapp/src/main/webapp/themes/mobile/theme.xml
A /weblogger-webapp/src/main/webapp/themes/mobile/weblog.vm

```

Beginnings of a mobile theme. Good.

```

M /weblogger-business/src/main/resources/sql/createdb.vm
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/Weblog.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/pojos/wrapper/WeblogWrapper.java

```

Here we add a new column "mobilethemename" to the table "weblog." This could be viewed as a problem because there may be more types of themes in the future, e.g. touch screen, TV set-top box, etc. and we don't want to have to add a new column to the weblog table every time we add one.

```

M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/ThemeManager.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/themes/ThemeManagerImpl.java

```

These changes add `getEnabledStandardThemeList()` and `getEnabledMobileThemeList()`. This could also be seen as a problem because it hardcodes the names of specific theme types into Java classes. It would be better to have one method with the signature:

`getEnabledThemeList(String type)`

```

M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/WeblogManager.java
M /weblogger-business/src/main/java/org/apache/roller/weblogger/business/jpa/JPAWeblogManagerImpl.java

```

This changes the `getPageByLink()` to `getPagesByLink()` which returns multiple pages, because different types of themes share the same link.

```
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/rendering/servlets/PageServlet.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/rendering/servlets/PreviewServlet.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/rendering/servlets/SearchServlet.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/rendering/util/WeblogPageRequest.java
```

These changes modify the rendering Servlets to check the incoming request and to determine which type of theme and templates should be used to render the response.

```
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/core/CreateWeblog.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/core/CreateWeblogBean.java
M /weblogger-webapp/src/main/webapp/WEB-INF/jsp/core/CreateWeblog.jsp
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/editor/ThemeEdit.java
M /weblogger-webapp/src/main/webapp/WEB-INF/jsp/editor/ThemeEdit.jsp
M /weblogger-webapp/src/main/webapp/WEB-INF/jsp/editor/ThemeEditSharedThemesOnly.jsp
```

These changes enable users to pick both a standard and mobile theme at theme creation time. Here are "hardcoding" knowledge of the mobile theme type into the UI, which is not ideal but is hard to avoid without some serious UI design and dev work.

```
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/editor/StylesheetEdit.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/editor/TemplateEdit.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/editor/TemplateEditBean.java
M /weblogger-web/src/main/java/org/apache/roller/weblogger/ui/struts2/editor/Templates.java
M /weblogger-webapp/src/main/webapp/WEB-INF/jsp/editor/TemplateEdit.jsp
```

These changes make it so that, when a new template is created there are actually two templates that are created, one of type standard and one of type mobile. That means two records are created in the webpage template or each "template" created via the web UI.

Commentary

Shelan has done a great job of seeking feedback and documenting his evolving design thoughts via the Roller mailing list and his blog. The proposal on the wiki is not as up to date and complete as it could be, but that could be fixed by deleting out of date material and perhaps, adding links to key design discussions on the mailing list.

Implementation has gone well so far, with lots of good progress, but there are two course corrections that should be made. First, as we discussed on the mailing list, we should avoid hard-coding mobile-specific fields into the database. Second, the design for storage of different type of templates seems too complex and introduces too many changes to Roller code. There are some recommended new designs below to adjust these problems.

Recommended design changes

These are recommended changes to the Roller Mobile design. First, a fix for the mobile-specific field in the Weblog object and corresponding website table. We want to allow each weblog to have a set of themes, each with a name so we introduce a WeblogThemeAssoc class and a corresponding rol_weblogtheme table like this:

```
create table rol_weblogtheme (
  id          varchar(48) not null primary key,
  weblogid    varchar(48) not null,
  name        varchar(255) not null,
  custom      $db.BOOLEAN_SQL_TYPE_FALSE not null,
  #columnNotNullWithDefault('type' 'varchar(16)' 'standard')
);
```

Second, a new design for weblog templates. Instead of creating a WeblogTemplate for each type of template (e.g. standard, mobile, etc.), we allow each WeblogTemplate to have multiple "template code" objects. In effect, each WeblogTemplate object will have two WeblogTemplateCode objects, one for type standard and one for type mobile. With this approach, we shouldn't have to make as many changes to Roller code, especially in the Struts UI actions.

```
create table rol_templatecode (  
    id                varchar(48) not null primary key,  
    templateid varchar(48) not null,  
    template      $db.TEXT_SQL_TYPE not null,  
    #columnNotNullWithDefault('type' 'varchar(16)' 'standard')  
);
```