Upgrading to Forrest 0.5

$Revision: 1.8 $

1. Introduction

This page describes changes to Forrest that affect people who are upgrading from 0.4 and earlier to the 0.5 version. Please post your upgrade experiences to the forrest-dev mailing list. As more experience is gained, this document will be updated.

2. New Features

This new version includes many changes, as it as been under development for seven months (in part, awaiting the Cocoon 2.1 release). The following list shows some of the key new features for Forrest 0.5 (for the full list of changes, see the change log).

- Significantly faster (100% - 300%) command-line rendering
- Docbook support. The DTDs are shipped. Document types are automatically detected by the sitemap and there is a basic stylesheet which converts DocBook documents to intermediate forrest document-v12 for standard rendering. For rendering as full DocBook, see this FAQ entry.
- Configuration of URLs to ignore with cli.xconf configuration.
- Skin enhancements.
- The document-v12 DTDs are added. You can still use the old document-v11 DTDs if you would rather. However it is easy to upgrade - just change the document type declarations in you project's xdocs and run '`forrest validate-xdocs`'.
- Automated handling of documents in Wiki syntax. See the fresh-site example which are generated when you '`forrest seed`' a new project.
- The '`forrest backcopy`' target copies all types of content that you may have edited under the webapp while doing a '`forrest run`'.
- Flag for email obfuscation.
- Support for raw un-processed content. See this FAQ entry.
- Element `<abstract>` is now rendered in the HTML output.
- Images scale properly in PDF output. See also this FAQ entry.
- Configurable table-of-content depth with new 'toc' element in the `skinconf.xml` configuration.

3. Upgrading the sitemap

In brief, Forrest 0.5 is *mostly* backwards-compatible for sites that do not define a custom sitemap. Between 0.4 and 0.5, the Forrest sitemap was completely rewritten. Instead of a single sitemap.xmap doing everything, a 'driver' sitemap.xmap now delegates to a number of mounted subsitemaps handling different functional areas. The new sitemap is fully described in the Sitemap Reference.

Users that have overridden and augmented the Forrest 0.4 sitemap.xmap (run '`forrest overrides`' to see if your project has) will need to:

1. Move their overridden sitemap (`src/documentation/sitemap.xmap`) out the way
2. Copy the new sitemap, $FORREST_HOME/context/sitemap.xmap, to `src/documentation`
3. Reapply your customizations to the new sitemap.xmap. Customizations can be determined by comparing your modified sitemap.xmap with original Forrest 0.4 sitemap.

4. Version-specific sitemaps

Forrest 0.5 now supports *version-specific sitemaps*. Ie., if your project's overridden sitemap is called `sitemap-0.5.xmap`,

then this sitemap will be used in preference to `sitemap.xmap`, or any other sitemap-*.xmap variants.

So if, for example, one has:

```
src/documentation/sitemap.xmap
src/documentation/sitemap-0.5.xmap
```

Then Forrest 0.4 will use sitemap.xmap, and Forrest 0.5 will use sitemap-0.5.xmap. This means users don't need to all upgrade to 0.5 at once.

As Forrest 0.5 has been split into multiple sitemaps, this version-specific behaviour is triggered for any *-0.5.xmap file. For example, if one has:

```
src/documentation/sitemap.xmap
src/documentation/forrest-0.5.xmap
```

Then Forrest 0.4 will use sitemap.xmap, and Forrest 0.5 will use Forrest's own sitemap.xmap, plus the user-defined forrest-0.5.xmap file.

The same system prevents future compatibility problems, so Forrest 0.5 should continue to work when future (incompatible) sitemaps are present:

```
src/documentation/sitemap.xmap
src/documentation/sitemap-0.5.xmap
src/documentation/sitemap-0.6.xmap
src/documentation/sitemap-0.7.xmap
```

Note:

If your `forrest.properties` defines the `forrest.validate.sitemap.{includes,excludes}` properties, these will have to be updated to prevent validation of sitemaps from unused versions. If undefined, Forrest will only validate sitemaps from the current version.

5. Excluding URLs: filterlinks.xsl removed and cli.xconf added

In Forrest 0.4 certain URLs could be excluded from command-line processing by overriding and editing `filterlinks.xsl` and so excluding the link nodes.

Forrest 0.5 uses the re-written command-line from Cocoon 2.1.1 which, apart from being much faster, **does not use filterlinks.xsl**. Instead, patterns for command-line inclusion and exclusion can be specified in cli.xconf, as described in this FAQ entry. There is already a default cli.xconf but you can over-ride that with your own if needed by copying `resources/conf/cli.xconf` from the Forrest distribution into your project's top-level directory.

6. Skin invocation changes

Users with custom skins *may* need to update them, depending on what use they make of passed-in XSLT parameters. If your custom skin does not appear to work with 0.5, look in Forrest's `sitemap.xmap` for occurrences of `{forrest:skin}`, and check that the callee (your XSLT) is expecting what the caller (the sitemap) is passing it. Forrest skins can be used as a reference.

7. Upgrading skinconf.xml

Look at the differences for the sample skinconf.xml as a guide. You will need to update the internal DTD and add several new elements.

8. Upgrading forrest.properties

Look at the differences for the sample forrest.properties as a guide. Note that project.sitemap has changed to project.sitemap-dir and note that cli.xconf now controls some settings that were previously controlled by forrest.properties, such as project.start-uri

9. Run a clean target after upgrade

To avoid any issue with old classes being loaded, run a '`forrest clean`' just after you upgraded to this version.

10. SVGs should omit DOCTYPE declarations

FIXME (forrest-dev):

Is this note still relevant?

In Forrest 0.5, sites that render SVGs may encounter ClassCastExceptions:

```
javax.xml.transform.TransformerException: java.lang.ClassCastException
    at org.apache.xalan.transformer.TransformerImpl.transformNode(TransformerImpl.java:1326)
    at org.apache.xalan.transformer.TransformerImpl.run(TransformerImpl.java:3329)
...
```

There appears to be a bug in Cocoon 2.1.1's SVG component where the DTD associated with SVGs cannot be resolved. The workaround is to edit your `src/documentation/resources/images/*.svg` files, and comment out the **<!DOCTYPE ...** > declaration.

11. To be continued...

..as more issues are discovered/remembered :) Please send feedback to the mailing list.