

# Upgrading to Apache Forrest 0.6

## Table of contents

1 Introduction.....	2
2 New Features.....	2
3 Run a clean target after upgrade.....	2
4 General upgrade tips.....	2
5 New location of \$FORREST_HOME.....	2
6 Copyless.....	3
7 Project sitemaps take precedence.....	3
8 Private skins might need changes to document2html.xsl.....	4
9 Various additions to skin configuration and new external DTD.....	4
10 forrest.antproxy.xml is obsolete in favor of Ant's <import> task.....	4
11 Deprecation of .html and .html.....	4
12 Deprecation of .cwiki filename extension to become .jspwiki.....	4
13 New forrestbot.....	4
14 New DTDs.....	5
15 SystemIdentifiers for DTDs changed to forrest.apache.org.....	5
16 Projects can use a local CatalogManager.properties.....	6
17 Skins renamed and deleted.....	6
18 Wholesite.html/pdf.....	7
19 To be continued.....	7

## **1. Introduction**

This page describes changes to Apache Forrest that affect people who are upgrading to the 0.6 version. If you have other issues, then please discuss on either the [dev](#) (./mail-lists.html#forrest-dev) or [user](#) (./mail-lists.html#forrest-user) mailing lists. As more experience is gained, this document will be updated.

(If you are upgrading from a version prior to 0.5 then you may want to see the notes for the [previous](#) (./docs/upgrading\_05.html) upgrade. However you should be fine with the notes from this 0.6 document.)

## **2. New Features**

The following list shows some of the key new features for Forrest 0.6 (for the full list of changes, see the [change log](#) (./changes.html) ).

- Copyless, so now used in place.
- Project sitemaps take precedence.
- Now using Subversion (SVN) for source control.
- New skinconf capabilities and new external skinconf DTD.
- New skins use CSS instead of tables. Old skins deprecated.
- Forrestbot.

## **3. Run a clean target after upgrade**

To avoid any issue with old classes being loaded, run a 'forrest clean' in your project directory, after you upgraded to this version.

## **4. General upgrade tips**

Synchronise your project's skinconf.xml and forrest.properties files.

Take advantage of the separation of concerns. In a new workspace, create a fresh 'forrest seed' site, then tweak its forrest.properties and skinconf.xml until it reflects your old site. When it is ready, replace your project's skinconf.xml and forrest.properties files. Any remaining issues would concern other aspects of your configuration, such as site.xml and your actual content.

## **5. New location of \$FORREST\_HOME**

To use the new Forrest, run 'build.sh' or 'build.bat' as normal, then change the FORREST\_HOME environment variable to point to forrest/src/core instead of the old

## Upgrading to Apache Forrest 0.6

forrest-0.5 location (.../build/dist/shbat). Also make sure PATH gets updated to use the new \$FORREST\_HOME/bin

### 6. Copyless

In essence, Forrest does not create a dist anymore, and uses itself in place. No more useless copying to a separate build space, no more backcopying of your changes, all is used live.

It improves the build process a lot. Development turnaround time is excellent. You can even tweak the main forrest core stylesheets and see changes immediately.

### 7. Project sitemaps take precedence.

With Forrest 0.6 it is now possible for projects to "plugin" to our sitemaps, without needing to copy the main sitemaps and keep them synchronised. This will enable hassle-free update to future Forrest versions.

If your old project did not use any customised \*.xmap files, then your upgrade process will be easy (you can safely skip this section).

If your project did use custom \*.xmaps, with matchers for special circumstances (for example special doctypes that you were handling) then you will need to be prepared to make some changes. Hopefully with the new functionality of Forrest, you can do away with your customisations altogether and just use the Forrest default sitemaps. Try that first. If not, then read on...

Prior to 0.6 it was possible to replace \*any\* of the xmaps by placing your own versions in your project directory, these were then copied over to replace the Forrest ones at build time. However, with the move to [copyless](#) this no longer happens, instead there is now an extension mechanism for the sitemap (as opposed to a replacement mechanism).

When upgrading to Forrest 0.6 you need to copy customised matchers in any of your projects \*.xmap files into your projects sitemap.xmap file. Any matchers in your project \*.xmap files that duplicate ones in Forrests own \*.xmaps can now be removed. This will ensure that future enhancements to these matchers in Forrest will automatically be included in your project.

The good news is that this process makes upgrading to future versions of Forrest much simpler, because your project \*.xmap files will contain only your customisations. As a result there will no longer be a need to merge your custom xmaps with the updated ones in upcoming versions of Forrest.

See [Using project sitemaps](#) (../docs/project-sitemap.html) for more details.

## 8. Private skins might need changes to document2html.xsl

Moved all references to `//skinconfig` out of the `document2html.xsl` into the `site2xhtml.xsl` file. If you have your own skins that were referencing `"$config"` or `"//skinconfig"` in the `document2html.xsl` then you need to make similar changes. For further information, see Issue [FOR-146](http://issues.cocoondev.org/secure/ViewIssue.jspa?key=FOR-146) (<http://issues.cocoondev.org/secure/ViewIssue.jspa?key=FOR-146>) .

## 9. Various additions to skin configuration and new external DTD

Various capabilities have been added to the skinconfig. See the new descriptions in the 'forrest seed' site `src/documentation/skinconf.xml` and synchronise yours.

For example, to use different colors (e.g. the light blue of the old krysalis skin), CSS colors can be specified in `skinconf.xml`

There is now an external DTD which makes it much easier to keep your `skinconf.xml` synchronised.

## 10. forrest.antproxy.xml is obsolete in favor of Ant's <import> task

Projects that use `forrest.antproxy.xml` via an Ant build task to invoke Forrest, will receive an error message directing them to this document. Please see the [Invoking Forrest from Ant](#) (`../docs/your-project.html#invoking_from_ant`) documentation for instructions on how to use the `<import>` task to integrate Forrest with your build system.

## 11. Deprecation of .html and .ihtml

The `.html` input file format has been deprecated and will likely be removed in the next release. Please convert all `.html` files to `.html` files. If you do 'forrest seed' there is a sample `html` source file.

## 12. Deprecation of .cwiki filename extension to become .jspwiki

Use the `.jspwiki` filename extension rather than old `.cwiki` extension. Clarifies that the purpose is the JSPWiki source format.

## 13. New forrestbot

The `forrestbot` and the `forrestbot` web interface have been completely rewritten. There is no direct way to convert old configurations to new configurations. Please see the [forrestbot documentation](#) (`../docs/forrestbot.html`) for instructions to create buildfiles that work with the

## Upgrading to Apache Forrest 0.6

new forrestbot.

### 14. New DTDs

Updated all v1.2 DTDs to become v1.3 DTDs (forward compatibility: v1.2 docs will work fine as V1.3). The main change is the addition of a @class attribute to every element, which enables the "extra-css" section in the skinconf to be put to good use.

Updated the v2.0a DTDs to become v2.0 DTDs (forward incompatibility: v1.2/1.3 docs are not forward-compatible as V2.0).

#### Changes from V1.2 to V1.3

=====

document	- Addition of class attribute all elements
faq	- Addition of class attribute all elements
changes	- Addition of class attribute all elements
howto	- Addition of class attribute all elements
todo	- Addition of class attribute all elements

#### Changes from V2.0a to V2.0

=====

document	- Addition of class attribute, all elements
	- Addition of label attribute to note and warning elements (consistent with v1.2/1.3)
faq	- Class attribute, all elements
changes	- New DTD
howto	- New DTD
todo	- New DTD

#### Changes from V1.3 to V2.0

=====

document	- Renamed <link> to <a>
	- Removed <fork> and <jump>.
faq	- Renamed <part> to <faqsection>
	- @title attribute on <faqs> is now a nested <title> element
changes	- Same changes as in document between 1.3 and 2.0
howto	- Same changes as in document between 1.3 and 2.0
todo	- Same changes as in document between 1.3 and 2.0

### 15. SystemIdentifiers for DTDs changed to forrest.apache.org

Everyone should still continue to use the [Catalog Entity Resolver](#) (../docs/catalog.html) and that certainly still operates at the core of Forrest using the well-defined PublicIdentifiers. However, some impoverished XML tools do not, so they need to be able to get the DTDs from the website. Some other tools rely on the System Identifier rather than the Public Identifier. See Forrest Issue [FOR-107](#)

(<http://issues.cocoondev.org/secure/ViewIssue.jspx?key=FOR-107>) .

In previous versions of Forrest, and maybe in your application if you copied the fresh-site xdocs, there were inconsistent SystemIdentifiers. Some used local filenames, others used [apache.org/forrest/dtd/](http://apache.org/forrest/dtd/) URIs. In v0.6 we changed to use System Identifiers at [forrest.apache.org/dtd/](http://forrest.apache.org/dtd/) as resource URLs. You do not need to change them because you are using the entity resolver, but to remain consistent, it probably is best.

Beware if you have a catalog where the mapping is not declared properly. The Catalog Entity Resolver will miss the local mapping and happily go to the network to get the DTDs. That would cause Forrest to appear to be slow for you, yet it will still operate properly. While we are on that topic, if you use the XSLT document() function - there is a [Xalan bug 28341](http://issues.apache.org/bugzilla/show_bug.cgi?id=28341) ([http://issues.apache.org/bugzilla/show\\_bug.cgi?id=28341](http://issues.apache.org/bugzilla/show_bug.cgi?id=28341)) for DTD resolution via document() - please help to fix it. Also see the [Cocoon FAQ](http://cocoon.apache.org/2.1/faq/faq-xslt.html) (<http://cocoon.apache.org/2.1/faq/faq-xslt.html>) .

## 16. Projects can use a local CatalogManager.properties

You can add a local CatalogManager.properties to your project.classes-dir to define your additional catalogs for DTDs and other entities.

If you do not add such a configuration file, then there will be a harmless message on startup "CatalogManager.properties not found".

## 17. Skins renamed and deleted

Skins now have a naming convention. The default skins are [described](#) (skins.html) .

The old skin names are automatically mapped to the new names. You should change your forrest.properties to use the new names.

The following skins were renamed (the old names still work, but may not in future releases).

0.5 Name	0.6 Name
krysalis-site	crust
tigris-style	tigris

The following skins were deleted (the old names still work, but may not in future releases).

0.5 Name	0.6 Alias
forrest-css	crust
avalon-tigris	tigris

## *Upgrading to Apache Forrest 0.6*

The old "forrest-site" skin is retained for a little while longer, but is deprecated, so please move to one of the other skins.

### **18. Wholesite.html/pdf**

Instead of using names "site.html/pdf" to create an aggregate page of your entire site, use "wholesite.html/pdf". The old names are still supported for backwards compatibility but they may not be in the future.

### **19. To be continued...**

...as more issues are discovered/remembered :) Please send feedback to the [mailing list](#) ([../mail-lists.html#forrest-dev](#)) .