

Frequently Asked Questions

Questions

1. Building Forrest

- [I have checked out a working copy of xml-forrest CVS, now what?](#)
- [When running ./build.sh in cygwin, I get an error: cygpath.exe: *** can't create title mutex, Win32 error 6.](#)

2. Technical

- [How do I stop Forrest breaking on links to external files that may not exist, like javadocs?](#)
- [How do I link to raw files such as config.txt and brochure.pdf?](#)
- [What do the numbers in square brackets mean when Forrest lists each document that it has built?](#)
- [Image don't display in PDFs. How do I fix this?](#)
- [The tab link in my site incorrectly assumes that 'index.html' is present in the linked-to directory. How do I fix this?](#)
- [When generating PNG images from SVG, I get an error: Can't connect to X11 window server using ':0.0' as the value of the DISPLAY variable.](#)
- [How do I enable XSP processing?](#)

3. Older versions

- [When invoking Forrest 0.4 from the forrest.antproxy.xml, the build fails because \\${project.home} isn't resolved.](#)
- [After upgrading to 0.4 my HTML looks significantly different, and table widths are wrong. What happened?](#)
- [When building my project, I get an validation error: Document root element "site", must match DOCTYPE root "null"..](#)

4. General

- [What is the relation between site.xml and book.xml?](#)
- [How do I use DocBook as the xml documentation format?](#)
- [How can I help?](#)

Answers

1. Building Forrest

1.1. I have checked out a working copy of xml-forrest CVS, now what?

Run `./build.sh` to build forrest, then run `forrest` to generate the documentation for Forrest, built with Forrest. The documentation is generated in the `build/site/` directory. Logfiles are in the `build/tmp/context/WEB-INF/logs/` directory. When you thereafter update your cvs, then run `build clean` before building.

The [Using Forrest](#) guide provides further step-by-step assistance in getting started with Forrest for your project.

1.2. When running `./build.sh` in cygwin, I get an error: `cygpath.exe: *** can't create title mutex, Win32 error 6.`

This [appears to be a bug in cygwin](#). Please use the `.bat` script instead.

2. Technical

2.1. How do I stop Forrest breaking on links to external files that may not exist, like javadocs?

This can be done by overriding the `cli.xconf` config file, and defining patterns for URLs to exclude.

This means creating a directory `src/documentation/conf` (or wherever `${forrest.conf-dir}` points) and copying `$FORREST_HOME/WEB-INF/cli.xconf` to it. Then edit `cli.xconf`, and add any exclude sections you require at the end. The default `cli.xconf` ignores directory links and links containing 'apidocs' or starting with 'api/':

```
...
<!-- Includes and excludes can be used to limit which URLs are rendered -->

<exclude pattern="**/">
<exclude pattern="**apidocs**">
<exclude pattern="api/**">

<uri src="favicon.ico"/>
</cocoon>
```

This is just an example, and you should modify it appropriately for your site.

Note:

Wildcards may be used. Wildcards work as in Cocoon's sitemap. `foo/*` would match `foo/bar`, but not `foo/bar/baz` — use `foo/**` to match that.

Frequently Asked Questions

2.2. How do I link to raw files such as config.txt and brochure.pdf?

Place them in the `src/documentation/content` directory and they will get copied into the output tree where you can link to them. You can also have sub-directories there to reflect your xdocs tree. See the samples documents when you 'forrest seed' a new project for a demonstration of this ability.

2.3. What do the numbers in square brackets mean when Forrest lists each document that it has built?

```
* [37] index.html
* [0] images/project-logo.png
* [0] skin/images/printer.gif
* [32] components/index.html
* [31] components/resolver/index.html
...
```

This is the number of links that were gathered from that page.

2.4. Image don't display in PDFs. How do I fix this?

Forrest uses [Apache FOP](#) for rendering images. FOP cannot handle all image types natively, and requires third-party jars to be added. For an overview, see [FOP Graphics formats](#)

Specifically, to get PNGs working in PDFs with Jimi:

1. Download Jimi from <http://java.sun.com/products/jimi/>
2. Unpack the Jimi distribution and copy JimiProClasses.zip to `lib/core/jimi-1.0.jar` (or to `src/documentation/lib/jimi-1.0.jar` so that each developer does not need to modify their Forrest installation).
3. Rebuild Forrest (if you used the `lib/core/` option above).

Alternative you can use JAI (Java Advanced Imaging API). For more info, see [FOP Graphics Packages](#)

Note:
Due to Sun's licensing, we cannot redistribute Jimi or JAI with Forrest

2.5. The tab link in my site incorrectly assumes that 'index.html' is present in the linked-to directory. How do I fix this?

In `tabs.xml`, use `@href` instead of `@dir`, and omit the trailing `/`. That will leave which file to serve up to the sitemap. For example, if the 'User Manual' tab should link to

manual/Introduction.html, tabs.xml should contain:

```
<tab label="User Manual" href="manual"/>
```

and add this rule to the sitemap:

```
<map:match pattern="manual">  
  <map:redirect-to uri="manual/Introduction.html"/>  
</map:match>
```

2.6. When generating PNG images from SVG, I get an error: Can't connect to X11 window server using ':0.0' as the value of the DISPLAY variable.

If you are using JDK 1.4.0 or newer, you can enable *headless* operation by running Forrest with the `forrest.jvmargs` parameter set to `-Djava.awt.headless=true`, like this:

```
forrest -Dforrest.jvmargs=-Djava.awt.headless=true site
```

2.7. How do I enable XSP processing?

First consider whether your needs would be better met by Cocoon itself, rather than Forrest. Forrest was designed primarily with static sites in mind, and is not a general-purpose Cocoon distribution.

That said, there are valid reasons for wanting programmatically generated content, so here is how to enable XSP:

1. Download [jdtcore-2.1.0.jar](#), and copy it to the `$FORREST_HOME/WEB-INF/lib` directory (or `lib/core/` directory in the source distribution).
2. Override `sitemap.xmap` in your local project (copy `$FORREST_HOME/context/sitemap.xmap` to `src/documentation/conf/`), and add the following generator definition in the `map:generators` section:

```
<map:generator name="serverpages" pool-grow="2"  
  pool-max="32" pool-min="4" src="org.apache.cocoon.generation.ServerPag
```

3. Decide how you want to use XSP. For single files, you could just define a `*.xml` matcher:

```
<map:match pattern="dynamic.xml">  
  <map:generate src="content/xdocs/dynamic.xsp" type="serverpages"/>  
  ...  
  <map:serialize type="xml"/>  
</map:match>
```

You may instead wish to override `forrest.xmap` to define a general mapping for XSPs.

Frequently Asked Questions

See also the [AddingXSPToForrest](#) Wiki page.

3. Older versions

3.1. When invoking Forrest 0.4 from the `forrest.antproxy.xml`, the build fails because `${project.home}` isn't resolved.

This is a bug in 0.4. The following modification to `FORREST_HOME/forrest.antproxy.xml` should fix it:

```
--- forrest.antproxy.xml      7 Feb 2003 22:21:15 -0000      1.2
+++ forrest.antproxy.xml      22 Feb 2003 09:39:42 -0000
@@ -41,6 +41,7 @@
     </path>

     <target name="forrest">
+   <property name="project.home" location="."/>
     <java classname="org.apache.tools.ant.Main" fork="true">
       <classpath refid="forrest-classpath"/>
       <jvmarg value="-Dforrest.home=${forrest.home}"/>
```

3.2. After upgrading to 0.4 my HTML looks significantly different, and table widths are wrong. What happened?

Forrest now uses a HTML 4.0.1 DOCTYPE declaration, `<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">`. This causes browsers to render the page in standards-compliance mode, or rather, not in "quirks" mode. For some operations (like determining table column widths), quirks mode results in better looking pages. For the old pre-0.4 behaviour, create a custom sitemap and remove the line:

```
<doctype-system>http://www.w3.org/TR/html4/loose.dtd</doctype-system>
```

3.3. When building my project, I get an validation error: Document root element "site", must match DOCTYPE root "null"..

You are probably trying to build the project with an old version of Forrest (built before 2003-01-08) that is incorrectly trying to validate the `site.xml` file. If so, please update your Forrest installation.

Alternatively, you may be building with an up-to-date Forrest, but have overridden `forrest.validate.xdocs.excludes` in `forrest.properties`. With the introduction of `site.xml`, the above property must have `site.xml` listed to prevent an attempt at DTD-based validation.

4. General

4.1. What is the relation between site.xml and book.xml?

One `site.xml` file in your project root can replace all the `book.xml` files (one per directory) in your site. Internally, Forrest uses `site.xml` to dynamically generate `book.xml` files. However, Forrest first checks for the existence of a `book.xml` file, so backwards-compatibility is preserved. If a directory has a `book.xml` file, the `book.xml` will be used to generate the menu. This is useful in situations where `site.xml`-generated menus aren't appropriate.

4.2. How do I use DocBook as the xml documentation format?

There are two ways. Forrest can transform the DocBook format into the Forrest "document" DTD on-the-fly and then render that as normal. The stylesheet that does this transformation still needs a little work but it handles most things.

The other way is to use the DocBook stylesheets directly. The DocBook DTDs are shipped with Forrest and automatically handled. However, you will need to have the DocBook stylesheets on your system (they are too massive to ship with Forrest) and configure Forrest accordingly. You will need to create a "sitemap" as explained in [Using Forrest](#) and add matches to handle your DocBook documents, e.g.

```
...
<map:match pattern="**/resolver-*.html">
  <map:generate src="content/xdocs/{1}/resolver-{2}.xml"/>
  <map:transform
    src="file:///usr/share/sgml/docbook/xsl/xhtml/docbook.xsl"/>
  <map:serialize/>
</map:match>
...
```

You can also use a mixture of the two methods, some handled automatically by Forrest and some directly using DocBook stylesheets. You can also have a mixture of "document-v*" DTD and DocBook.

4.3. How can I help?

Join the forrest-dev [mailing list](#), and tell us what you'd like to see improved. We regard all feedback as valuable, particularly from newcomers—often, close proximity blinds software developers to faults that are obvious to everyone else. Don't be shy!

Frequently Asked Questions

More info on contributing can be found on the [Contributing to Forrest](#) page. Its always a good idea to check the Forrest [to do](#) list before diving in.