

# Runbooks

## Table of contents

1 Install Apache Forrest.....	2
2 Generate the docs.....	2
3 Use "doc" tags!.....	3

Part of making your modules ops ready is providing user documentation. CTL was developed to help that handoff and provides a "runbook" document that becomes your reference manual for your module's users.

The [ProjectBuilder](#) module used to develop and build your module also contains support to generate a web site based on those module definitions. The generated web pages reads the `type.xml` file and generates a document suitable as a reference manual for your module, its attributes, commands, their options and anything else that makes up the definition of your module.

As the figure above illustrates, the tags in the `type.xml` files for your modules defined in in the `modules` source directory, are processed and along with some standard templates, an [Apache Forrest](#) document is generated.

Information is organized into a standard structure, divided up into sections, navigable via a top level table of contents.

- Displays information about your module library in HTML or PDF. Runbook can be deployed to any webserver.
- Gives an overview of your module's content.
- Describes all the attributes in your module along with their defaults.
- Describes all the commands in your module. This includes usage summary, command option description.
- You can embed your own information and graphics via Apache Forrest doc tags.

## 1. Install Apache Forrest

The Runbook generation depends on Apache Forrest 0.8.

Follow these steps:

1. Download and install Forrest v.0.8
2. Confirm the `FORREST_HOME` environment variable is defined. (e.g., `echo $FORREST_HOME`).

## 2. Generate the docs

The [ProjectBuilder](#) module includes the [generate-forrest-docs](#) command to generate the Runbook. Run the `generate-forrest-docs` command accepting the defaults:

```
ctl -m ProjectBuilder -c generate-forrest-docs
Base directory where module source files reside [/tmp/ctier/ctl/src]
Target directory where build files are generated [/tmp/ctier/ctl/target]
```

```
Library name? [default]
```

```
--snip--
```

The first time you run the command, a Forrest site will be created using some ProjectBuilder templates in a new directory called `docs` under your base directory (and parallel to your `modules` directory). You can run the `generate-forrest-docs` command any time. Subsequent calls to the command will essentially re-run "`forrest site`".

Besides building a forrest site, the `generate-forrest-docs` command also copies a PDF formatted copy of each module's reference into its doc directory: `modulename/doc/type.pdf`. This reference is accessible after your module has been deployed and may be useful to end-users of your module.

### 3. Use "doc" tags!

While the `generated-forrest-docs` command does not rely on anything besides the content of your `type.xml` tags, additional information can be quite helpful.

You can insert your own documentation by using `<doc> . . . </doc>` tags. The `generate-forrest-docs` command and templates attempt to keep your docs close to the related element of your `type.xml` document.

Because you are free to use any Forrest doc tags you can create some professional looking reference pages (including, tables, images, notes, warnings, etc.).

[Next: Packaging#](#)