Forrestbot

Table of contents

1. Overview

Forrestbot lets you automate building & deploying websites. There are implementations to get source from various locations, build it, then deploy it somewhere. It can notify you afterwards, and it keeps a log of the build process. Each workstage has multiple implementations; you can use one or more of each.

2. Using Forrestbot

Simply execute:

```
forrest -f mybuildfile.xml
```

The next section explains how to create your buildfile.

3. Creating a buildfile

A sample buildfile:

```
<project name="mysampleproject" default="main">
        <property name="notify.email.host" value="smtp.myhost.com"/>
        <property name="notify.email.to" value="me@domain.com"/>
        <property name="notify.administrator" value="Your Name
&lt;me@domain.com>"/>
        <property name="getsrc.cvs.user" value="anoncvs"/>
        <property name="getsrc.cvs.password" value="anoncvs"/>
        <property name="getsrc.cvs.root" value="/home/cvspublic"/>
        <property name="getsrc.cvs.host" value="cvs.myhost.com"/>
        <property name="getsrc.cvs.module" value="myproject"/>
        <property name="deploy.scp.dest"
value="username@myhost.com:/var/www/mydomain/htdocs"/>

        <!-- here we specify to use two notification implementations -->
        <target name="notify" depends="notify.local, notify.email"/>

        <!-- here we specify to deploy with the scp implementation -->
        <target name="deploy" depends="deploy.scp"/>

        <!-- the default implementation for getsrc is getsrc.cvs, which is what we
want -->

        <!-- assumes FORREST_HOME has been set as an environment variable -->
        <property environment="env"/>
        <import file="${env.FORREST_HOME}/tools/forrestbot/core/forrestbot.xml"/>
</project>
```

First, set properties needed by the workstages you are going to use. Here, we set properties that will be used by notify.email and getsrc.cvs. Next, specify what implementations will be used by each workstage.

Workstage	Implementations
getsrc	• getsrc.local • getsrc.cvs (default) • getsrc.svn
build	• build.forrest

deploy	• deploy.local (default) • deploy.scp • deploy.cvs • deploy.svn
notify	• notify.local (default) • notify.email

If you want to do more advanced processing for your project, you can override the 'main' target, which by default is `<target name="main" depends="getsrc, build, deploy, notify"/>`, create your own implementation of a workstage, or use any other ant tasks to do additional work.

Many workstages use usernames and passwords. You may want to keep them out of your project's xml file (especially if you store that file in CVS or SVN). A nice way to do this is make a simple buildfile (e.g. my-settings.xml) that just sets those properties (don't include it in CVS/SVN!). Then in your project buildfile, have `<import file="my-settings.xml"/>`.

3.1. Workstage Properties

Each workstage implementation is configurable with properties. The following tables describe each property and whether or not you are required to set it in your buildfile.

3.1.1. Misc Properties

Property	Description	Default Value	Required?
ant.project.name (you specify this by <project name="____"> in your buildfile)	This must be unique for each project.		Yes

3.1.2. getsrc.clean-workdir

This should be executed before a getsrc implementation is executed. For example, `<target name="getsrc" depends="getsrc.clean-workdir, getsrc.svn"/>`

3.1.3. getsrc.local

Property	Description	Default Value	Required?
getsrc.local.root-dir	Absolute path to the project's root directory on the local computer. Use **location=** instead of **value=** for this <property>		Yes

3.1.4. getsrc.cvs

Property	Description	Default Value	Required?
getsrc.cvs.user	CVS username		Yes

getsrc.cvs.password	CVS password		Yes
getsrc.cvs.root	CVS root directory	/home/cvsroot	Yes
getsrc.cvs.host	CVS host	cvs.apache.org	Yes
getsrc.cvs.module	CVS module name (an alias, or full path) to the directory that contains forrest.properties	${ant.project.name}	Yes
getsrc.cvs.tag	CVS tag or branch name		No

3.1.5. getsrc.svn

Property	Description	Default Value	Required?
getsrc.svn.url	Full repository URL for project (this directory must contain forrest.properties)		Yes
getsrc.svn.revision	Revision number to fetch	HEAD	No

3.1.6. build.forrest

Property	Description	Default Value	Required?
build.work-dir	Directory to temporarily hold working files	work	No
build.log-dir	Directory to hold log files	logs	No

3.1.7. deploy.local

Property	Description	Default Value	Required?
deploy.local.dir	Path to deploy site to. Relative paths will be relative to ${bot.home}	sites/${ant.project.name	No

3.1.8. deploy.scp

${user.home}/.ssh/known_hosts must properly recognize the host, so you should manually make an ssh connection to the host if you never have before.

Property	Description	Default Value	Required?
deploy.scp.dest	Full destination reference in the format user@host:/directory/pat		Yes
deploy.scp.password	Password for user@host		No. You will be prompted for it if it is not set.

3.1.9. deploy.cvs

This is only available on *nix operating systems.

Property	Description	Default Value	Required?
deploy.cvs.user	CVS username to use when committing changes		Yes
deploy.cvs.password	CVS password		Yes
deploy.cvs.root	CVS root	/home/cvs	Yes
deploy.cvs.host	CVS host	cvs.apache.org	Yes
deploy.cvs.module	CVS module	${ant.project.name}	Yes
deploy.cvs.commit-mess	Message to use when committing. You probably want to put a machine name or person's name here.	Automatic publish from forrestbot	No

3.1.10. deploy.svn

Property	Description	Default Value	Required?
deploy.svn.user	SVN username to use when committing changes		Yes
deploy.svn.password	SVN password		Yes
deploy.svn.url	Full repository URL		Yes
deploy.svn.commit-mess	Message to use when committing. You probably want to put a machine name or person's name here.	Automatic publish from forrestbot	No

3.1.11. notify

These settings are used by all notify implementations.

Property	Description	Default Value	Required?
notify.administrator	Name and email address of the forrestbot administrator		Yes
notify.on.failure	On a build failure, notification will happen if this is true.	true	No
notify.on.success	On a succesful build, notification will happen if this is true.	true	No

notify.log	Log file		No. Set by other workstage(s).
notify.deploy-location	Deployed location		No. Set by other workstage(s).
notify.completion-status	Result of the build		No. Set by other workstage(s).

3.1.12. notify.local

No properties.

3.1.13. notify.email

Property	Description	Default Value	Required?
notify.email.host	SMTP host through which the email will be sent.	localhost	Yes
notify.email.to	Email address to send notification to.	${user.name}@localhost	Yes
notify.email.from	From: address in the email	Forrestbot	No, but some mailers may require a valid email address.

4. Forrestbot design

Forrest and forrestbot use ant buildfiles extensively. Ant 1.6's import task is used to import multiple buildfiles into a single build. The following is the flow of control when running forrestbot:

- Your buildfile
  - forrestbot.xml
    - workstage buildfiles
    - forrest.build.xml

The workstage buildfiles set up the properties and files so that the main forrest buildfile (forrest.build.xml) will run. After it is run, other workstages buildfiles can implement reporting, deployment, or other post-build activities.

Your buildfile can specify which workstages you want to use, set properties for them, and do any additional pre- and post-processing.