

AvalonDistributionLayout

Supporting the new-style URLs

I'm in the process of getting components built by maven. In that case all that's needed is some info in project.xml and the command `maven:deploy -Dversion=1.1.1 -Dtag=MYCOMP_1_1_1`. However this is work in progress, so right now some manual steps are involved. This is a basic explanation of what you need to do:

- WARNING*: all scripts and commands are *untested*!

Adding a new component

Two basic options:

1) keep supporting placing the new libraries in <http://www.apache.org/dist/avalon/excalibur/components/>, and symlink

This works like:

```
# first put files up at avalon/excalibur/components/ like you're used to
```

```
ssh www.apache.org
```

```
# symlink to the component location as excalibur-$COMP_NAME/source and
```

```
# excalibur-$COMP_NAME/binaries
```

```
umask 0002
```

```
export COMP_NAME="xmlutil" # example, edit this!
```

```
export VERSION="1.1" # example, edit this!
```

```
cd /www/www.apache.org/dist/avalon/
```

```
mkdir excalibur-$COMP_NAME
```

```
cd excalibur-$COMP_NAME
```

```
ln -s ../excalibur/components/$COMP_NAME source
```

```
ln -s source binaries
```

```
# extract jarfiles into the 'jars' directory
```

```
mkdir jars
```

```
unzip binaries/$COMP_NAME-$VERSION.zip '*.jar' -d jars
```

useful might be a script like this:

```
=== export-component.sh ===
```

```
#!/bin/sh
```

```
#
```

```
# Make linking a little easier
```

```
#
```

```
# usage: export-component.sh componentname version
```

```
#
```

```
export COMP_NAME=$1
```

```
export VERSION=$2
```

```
echo building new-style structure for $COMP_NAME version $VERSION ...
```

```
cd /www/www.apache.org/dist/avalon/
```

```
mkdir excalibur-$COMP_NAME
```

```
cd excalibur-$COMP_NAME
```

```
ln -s ../excalibur/components/$COMP_NAME source
```

```
ln -s source binaries
```

```
# extract jarfiles into the 'jars' directory
```

```
mkdir jars
```

```
unzip binaries/$COMP_NAME-$VERSION.zip '*.jar' -d jars
```

```
# cleanup
```

```
unset COMP_NAME
```

```
unset VERSION
```

2) just place the files in the new location

```
ssh www.apache.org
```

```
cd /www/www.apache.org/dist/avalon/
```

```
mkdir -p excalibur-$COMP_NAME/source
```

```
cd excalibur-$COMP_NAME
```

```
ln -s source binaries
```

```
mkdir jars
```

```
unzip binaries/$COMP_NAME-$VERSION.zip '*.jar' -d jars
```

New release for an existing component

in the event that you're doing a new release of something that was already released, life is a bit easier. You'll likely just want to upload the files like before to `avalon/excalibur/components/`, then do

```
cd /www/www.apache.org/dist/avalon/excalibur-$COMP_NAME
```

```
unzip binaries/$COMP_NAME-$VERSION.zip '*.jar' -d jars
```

Easy urls for the latest distro's

Not yet implemented for excalibur, but we should. For a particular subproject, have a symlink to the latest distribution files in `binaries/` and `source/` in the base directory, then symlink to those files, replacing the version with 'latest':

basiclink.sh

```
#!/bin/sh
```

```
#
```

```
# usage: basiclink.sh groupname distname version
```

```
export GROUPNAME=$1
```

```
export DISTNAME=$2
```

```
export VERSION=$3
```

```
cd /www/www.apache.org/dist/avalon/$GROUPNAME
```

```
ln -s binaries/$DISTNAME-$VERSION-bin* .
```

```
ln -s source/$DISTNAME-$VERSION-src* .
```

```
ln -s $DISTNAME-$VERSION-src.zip $DISTNAME-latest-src.zip
```

```
ln -s $DISTNAME-$VERSION-bin.zip $DISTNAME-latest-bin.zip
```

```
ln -s $DISTNAME-$VERSION-src.zip.asc $DISTNAME-latest-src.zip.asc
```

```
ln -s $DISTNAME-$VERSION-bin.zip.asc $DISTNAME-latest-bin.zip.asc
```

```
ln -s $DISTNAME-$VERSION-src.tar.gz $DISTNAME-latest-src.tar.gz
```

```
ln -s $DISTNAME-$VERSION-bin.tar.gz $DISTNAME-latest-bin.tar.gz
```

```
ln -s $DISTNAME-$VERSION-src.tar.gz.asc $DISTNAME-latest-src.tar.gz.asc
```

```
ln -s $DISTNAME-$VERSION-bin.tar.gz.asc $DISTNAME-latest-bin.tar.gz.asc
```

Sample usage

```
ssh www.apache.org
```

```
umask 0002
```

```
basiclink.sh framework Avalon 4.1.5
```

Hope that's enough info!
