PRINCESS Build and Release

Michael Reposa, Steve Marotta

24-MAY-2017

# Overview

We maintain a set of Windows and Linux scripts to manage the building and distribution of the PRINCESS challenge problem solutions to ensure we deliver a consistent release to Lincoln Labs for evaluation each time. These scripts are part of the PRINCESS source tree:

\princess\release

You will need the following third-party software installed to build and create PRINCESS releases:

* Windows or Linux
* SBT
* Java 8
* Scala 2.11

# Creating PRINCESS Releases

Running the scripts produces a sub-directory containing everything needed to deploy and run each PRINCESS challenge problem solution.

## Deploying All Challenge Problems (Lincoln Labs Release)

The complete build and release process for all PRINCESS challenge problem solutions is started by running the “ll-release” script, which in turn invokes other scripts found in the same location, as shown below:

* ll-release
  + buildall
  + cpt-release
  + cp1-release
  + cp2-release
  + cp3-release

Running “ll-release” builds everything, then creates the latest release of the Challenge Problem Toolkit, and then creates the latest release of each PRINCESS challenge problem. The results of a successful release will be in

\princess\releases

\princess-cp1\_1.0.0-2.11

\princess-cp2\_1.0.0-2.11

\princess-cp3\_1.0.0-2.11

**Note that when you run ll-release, it overwrites the current contents of the \releases directory each time.**

## Deploying Challenge Problem 1

The build and release process for a single PRINCESS challenge problem solution is only slightly more complicated.

* buildall
* cpt-release
* cp1-release

These steps build everything, create the latest release of the Challenge Problem Toolkit, and create the latest release of PRINCESS challenge problem 1. The results of a successful release will be in

\princess\releases

\princess-cp1\_1.0.0-2.11

**Note that when you run cp1-release, it overwrites the current contents of the \releases\princess-cp1\_1.0.0-2.11 directory each time.**

## Deploying Challenge Problem 2

The build and release process for a single PRINCESS challenge problem solution is only slightly more complicated.

* buildall
* cpt-release
* cp2-release

These steps build everything, create the latest release of the Challenge Problem Toolkit, and create the latest release of PRINCESS challenge problem 3. The results of a successful release will be in

\princess\releases

\princess-cp2\_1.0.0-2.11

**Note that when you run cp2-release, it overwrites the current contents of the \releases\princess-cp2\_1.0.0-2.11 directory each time.**

## Deploying Challenge Problem 3

The build and release process for a single PRINCESS challenge problem solution is only slightly more complicated.

* buildall
* cpt-release
* cp3-release

These steps build everything, create the latest release of the Challenge Problem Toolkit, and create the latest release of PRINCESS challenge problem 3. The results of a successful release will be in

\princess\releases

\princess-cp3\_1.0.0-2.11

**Note that when you run cp3-release, it overwrites the current contents of the \releases\princess-cp3\_1.0.0-2.11 directory each time.**

## Deploying the Challenge Problem Toolkit

The build and release process for a single PRINCESS challenge problem solution is only slightly more complicated.

* buildall
* cpt-release

These steps build everything, create the latest release of the Challenge Problem Toolkit, and create the latest release of PRINCESS challenge problem X. The results of a successful release will be in

\princess\cptoolkit\latest

**Note that when you run cpt-release, it overwrites the current contents of the \princess\cptoolkit\latest directory each time.**