# Inca Workshop
# August 26, 2010

## Hands-on Tutorial #2
## Deploying the Default Installation

**Make sure your machine meets the requirements for installing Inca:**
**http://inca.sdsc.edu/releases/2.6/guide/userguide-one.html#USERGUIDE-REQUIREMENTS**

1. After logging in, you can make the rest of your deployment steps less tedious by setting an environment variable for the location of your installation.

```
% setenv INCA_DIST $HOME/myInca
```

2. Download the Inca installer.

```
% wget http://inca.sdsc.edu/releases/2.6/incaInstall.sh
```

3. Download the Inca software.

```
% sh incaInstall.sh $INCA_DIST all
```

4. Change into your Inca installation directory.

```
% cd $INCA_DIST
```

5. Create authentication certificates for the Inca deployment. Choose and enter a certificate password.

```
% bin/inca createauth
password> *******
Confirm password> *******
```

6. Configure the default installation. Since this step overwrites any existing installation, Inca asks for confirmation. Reenter the password you chose in the previous step.

```
% bin/inca default
password> *******
Preparing to deploy default Inca configuration...
Initializing Inca configuration...
** Warning: this will erase any collected reporter state
   on the Inca depot and configuration on the agent
Do you wish to continue (y/n)? y
```

You should now have an Inca Depot, Agent, and Consumer running on your machine. You can access the Consumer by opening a browser to http://localhost:8080. While the Inca system is getting underway, the Consumer may indicate that there are no suites and/or resources found. Refresh the display every minute or so until the Consumer displays the default configuration—a single suite named "defaultSuite" and three resources named "defaultGrid", "localResource" and "localSite", all of which refer to your local machine.

Select "defaultSuite" and "defaultGrid", then press the "Submit" button. The Consumer will display a table that shows the current status of the 10 series in defaultSuite. Each of these series runs once every 10 minutes, staggered so that no two series run simultaneously. Until a series runs, the Consumer table entry for it will be blank. Again, you can periodically refresh the display until all table entries are filled. You can click around in the display to see what other views of your data the Consumer makes available.

Take a look at the contents of $INCA_DIST/var. This directory contains the log file produced by each Inca component, along with additional component runtime data. Scanning though these log files can help to troubleshoot problems with your Inca installation.

The $INCA_DIST/etc directory contains many of the configuration files for your deployment. Here you'll find authentication certificate and key files and the hibernate.properties file that allows you to use different database engines (PostgreSQL, MySQL, etc.). The jetty.xml file, covered later in the Consumer tutorial, allows customization of the Consumer display. Two more files of interest can be found in the $INCA_DIST/etc/common subdirectory. inca.properties contains the property settings that allow you to specify default configuration information for the Inca components. The most likely settings you may want to change are the port numbers used by the Depot and Agent. log4j.properties allows you to specify which messages the Inca components include in their log files. By default, the Inca components limit the messages logged, in order to reduce the size of the log files. Changing the log4j.rootLogger and log4j.edu.sdsc.inca settings from "info" to "debug" before starting an Inca component will log much additional information that may assist in troubleshooting.

Finally, try viewing your Inca deployment configuration by running incat:

```
% bin/inca incat --agent localhost:6323
password> *******
```

You can walk through your configuration to see how your repositories, resources, and series are configured.

## Modifying Your Deployment

1. Create a directory to serve as a repository for the reporters you wrote in the first tutorial.

```
% mkdir $INCA_DIST/newReporters
% cp <your new reporters> $INCA_DIST/newReporters
% cd $INCA_DIST/newReporters
```

2. Use incpack to create a catalog for the repository. After it's created, you can take a look at the contents of the catalog using zless.

```
% setenv PERL5LIB ../Inca-Reporter-5.13398/lib/perl
% setenv PYTHONPATH ../Inca-Reporter-5.13398/lib/python
% perl ../Inca-Reporter-5.13398/sbin/incpack *
% zless Packages.gz
```

3. Create a file that we'll in use later in one of our series.

```
% echo >! /tmp/targetfile
```

4. Start incat to show your current deployment

```
% cd $INCA_DIST
% bin/inca incat --agent localhost:6323
password> ******
```

5. Add your new repository to the set used by your deployment.

- In incat's Repositories tab, press "Add …" at the bottom of the Repositories panel. This will open a dialog that tasks for the Repository URL.
- Enter **`file:///`*`home`*`/myInca/newReporters`** into the text box
- Press the "OK" button.

The Reporters panel of incat's Repository tab shows all the reporters available in any of your deployment's repositories. Unclick the "All Repositories" checkbox at the bottom of this panel to see only the contents of your new repository. You can select individual Reporters to see their catalog attributes in the Reporter Properties panel and press the "Show" button to see their source.

6. Now we'll add some macro definitions to the defaultGrid resource that we can use later to define new series.

- Select incat's Resource Configuration tab.
- In the Resource panel, select the defaultGrid resource.
- Press the "Add …" button at the bottom of the Macros panel. This will open a dialog that allows you to enter a new macro name and one or more values for the macro.
- Enter **`file`** as the macro name and **`/tmp/targetfile`** in the edit box at the top of the Value(s) panel, then press the "Ok" button. The new macro setting should appear in the Resource Configuration macro panel.
- Press the "Add …" button at the bottom of the Macros panel again. This time, name the macro **`targets`** and give it three values—**`ucsd.edu`**, **`utexas.edu`**, and **www.harvard.edu**. You give a macro multiple values by pressing enter between values, at which time the value in the edit box appears in the Value(s) panel. After entering the third value for the macro, press the "Ok" button. The macro setting should now appear in the Resource Configuration macro panel.

7. Next, add a new suite.

- Select incat's Suites tab.
- Press the "Add …" button at the bottom of the Suites panel. This opens a dialog that allows you to enter a new suite name and description. Enter **`mySuite`** for the suite name, then press the "Ok" button. (You can leave the description blank.)

8. Now we can add a ping series to the new suite.

- Press the "Add …" button at the bottom of the Series panel. This opens a dialog that allows you to define new series.
- In the top half of the panel, select the reporter grid.benchmark.performance.ping. This reporter measures the round-trip ping time to a host. Make sure that the defaultGrid resource is selected. Enter **`@targets@`** in the text box for the "host" argument. This refers to the targets macro you defined in step #7. Since this macro

has three values, Inca will create three separate series—one that measures the ping time to each of the three hosts.
- In the bottom half of the panel, use the schedule widgets to indicate that the series should be run every 5 minutes from a random starting time. This will ensure that enough data is generated to make later examination of the series interested.
- Enter **`ping_to_@targets@`** in the series Nickname text box. This reference to the targets macro will be expanded along with the one in the "host" argument, ensuring that the nickname for each of the three series Inca creates will be unique.
- After completing the above steps, press the "Ok" button. The new series should appear in the Series panel.

9. Add another series to test the version of Perl installed on the system.

- Press the "Add …" button at the bottom of the Series panel again
- Select cluster.lang.perl.version in the Reporter panel and change the schedule widgets to run the series every 5 minutes. Set the "log" argument to 5; this will give you a chance to see some of the log messages produced by the Reporter. Press "Ok".

10. We'll add one more series to illustrate the use of comparisons.

- Press the "Add …" button at the bottom of the Series panel again
- Select user.search.output.unit in the Reporter panel. This is a flexible reporter that searches the output of a specified program for a regular expression.
- Enter **`/bin/cat /tmp/targetfile`** in the text box for the "com" argument, a comma for the "delim" argument, and **`second`** in the text box for the "search" argument.
- Again, change the schedule widgets to run the series every 5 minutes.
- At the bottom of the dialog, enter

  ```
  errorMessage == '' || errorMessage =~ /call failed/
  ```

  in the Comparison text box. This indicates that Inca should consider the Reporter run to be successful either if the file contains the word "second" or if the file is empty (in which case the run produces an error message containing the text "call failed".)
- Press "Ok".

11. You now have three series defined in your mySuite suite. You can add additional ones if you like—for example, using the reporters you wrote. Afterwards, you're ready to apply all the changes you've make to your deployment.

- Select the "Commit" item in the Agent menu. This informs your Agent about the change you made and instructs it to start your new series.
- Quit incat. You don't need to save your changes unless you want a file that contains a copy of your deployment configuration.

After allowing a minute or two to let the changes take effect, open a browser to your consumer, http://localhost:8080. It should allow you to select and view information for your new suite.

12. The user.search.output.unit series is succeeding because the comparison specifies that an empty file is acceptable.  Change the contents of the file to a value that will cause the series to fail.

```
% echo first >! /tmp/targetfile
```

13. The next time the series runs (within five minutes—use the “Run Now …” button if you’re impatient) the Consumer display for it should change to indicate failure.  You can switch it back to success by replacing the file contents with the expected value.

```
% echo second >! /tmp/targetfile
```