ICAT Job Portal

ICAT Workshop in Lund, Sweden
12/13 March 2013

Kevin Phipps and Steve Fisher
Scientific Computing Department, RAL
Introduction

- Started as project for Lasers for Science Facility (LSF) at RAL
- Project running for about 1.5 years
- Funded by Harwell Imaging Partnership
LSF team

- Small team of scientists
- Develop and support analysis software
- Limited IT support
- ~10 analysis nodes with GPUs (Linux)
- No cataloguing of datasets – finding datasets takes longer as more datasets added
The Aim

• Build a batch and interactive job portal
• Use tried, tested, scalable and preferably open source components
• Hide away the underlying complexity from the end user
Project Status

• Prototype demonstrated mid-2012
• Work since then mainly to make it more generic, configurable and installable
• Usable system deployed for LSF within next few months
Architecture Overview

User's PC
- Web browser
- Remote Desktop client

https
- ICAT Job Portal webapp
- Torque batch server

Submit batch job
- Assign interactive job

Worker Node 1
- Prepare job
- Torque worker node
- Assign batch job

Worker Node n
- Facility software
- Run batch job
Head Node Architecture

JEE Application Server
- ICAT
- IDS
- Job Portal

Submit batch job

Torque batch server

Head Node

Metadata database

File storage

Jobs database

XML Job Descriptions and Job Dataset Parameters
Installation and Configuration

• Puppet Automation Software used to install head node and worker nodes
• Start with a clean Ubuntu 12.04 LTS installation and a network connection on all machines
• Installation takes an hour or two and includes: Java Development Kit, Glassfish Application Server including ICAT, IDS and Job Portal Software, MySQL database server and required databases, Torque batch system, Ganglia monitoring and the facility specific software
• Facility software updates rolled out automatically to all worker nodes by Puppet
‘Admin’ User Tasks

- Create XML Job Descriptions
- Create XML files for each dataset type picking out dataset features relevant to Job Options
- Write wrapper scripts for each application – saving and loading datasets from IDS and recording provenance
- A Python library of utilities is being created to help with the creation of wrapper scripts
## Job Portal Main Panel (Datasets)

### Datasets

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>20120524_0002_0001_632c1e9-9f32-4a39-a649-855ed5592c27</td>
<td>744nm laser</td>
<td></td>
</tr>
<tr>
<td>20120626_0004_0001_0bb8360-dc7d-4c13-84ca-72a638650334</td>
<td>colocalisation in T47D 3 Affibody 639 nm laser</td>
<td></td>
</tr>
<tr>
<td>20120525_0005_0001_e421c3c-37eb-4e0f-baea-bf656ed31688</td>
<td>T47D 3 Affibody 639 nm laser</td>
<td></td>
</tr>
<tr>
<td>20120525_0005_0001_6e28e0b-5e99-45a4-93d7-61a952a35912</td>
<td>T47D 3 Affibody 639 nm laser</td>
<td></td>
</tr>
<tr>
<td>20120524_0002_0001_c1b5d6c5-9e6e-fafa-be3f-89e529e4f12e</td>
<td>T47D 3 Affibody 639 nm laser</td>
<td></td>
</tr>
<tr>
<td>20120524_0002_0001_da5e37d-0d56-406f-9e06-59c57680d1d</td>
<td>T47D 3 Affibody 639 nm laser</td>
<td></td>
</tr>
</tbody>
</table>

### endDate
- 2012-11-27T14:18:17Z

### experiment_type
- Undefined

### id
- 7201

### instrument
- OctopusSM3

### location
- Dummy investigation 1/20120524_0002_0001_aeae73d8-e-d7c4-4b6c-a599-6e62eb4f829e

### name
- 20120524_0002_0001_aeae73d8-e-d7c4-4b6c-a599-6e62eb4f829e

### nchannels
- 1

### nframes
- 671

### sampledescription
- T47D 3 Affibody 639 nm laser

### startdate
- 2012-11-27T14:16:21Z
Job Options

XML Job Description on Head Node

```
<jobType>
  <name>MSM Viewer Project</name>
  <executable>/usr/local/nasa/bin/run_nasa_viewer</executable>
  <multiple>false</multiple>
  <type>interactive</type>
  <datasetTypes>project</datasetTypes>
  <jobOptions>
    <name>View</name>
    <groupName>View type</groupName>
    <type>boolean</type>
    <programParameter>-reg-beads</programParameter>
    <condition>numHeadFiles>0 & & numChannels>1</condition>
  </jobOptions>
  <jobOptions>
    <name>Track method</name>
    <type>enumeration</type>
    <programParameter>--trackmethod</programParameter>
    <values>Simple</values>
    <values>SLH</values>
    <values>Biggles</values>
    <values>Simulation</values>
  </jobOptions>
  <jobOptions>
    <name>Regular expression for images in directory</name>
    <type>string</type>
    <programParameter>-image-pattern</programParameter>
  </jobOptions>
</jobType>
```

Job Options Form in Web Browser

MSMM Viewer Project Options

- View type: View, View beads, View white lights, View reg residual frames, View reg model frames
- Track method
- Show variance image instead of image
- Do not load traces
- Read features/tracks from hdf5 files (slow)
- Set min, max for colour scale
- Regular expression for images in directory
- Do not clean levels/stats (default=0) (min=0) (max=10)
- Min number of detected features per frame range of a level/state (default=2)
- Threshold for the Chauvenet's outlier test (default=2) (min=1) (max=5)
- Set the (real) EM gain by hand
- Quantum efficiency (default=0.910000026) (min=1.0) (max=1.0)
- Set the (real) electron/ADU by hand
- A unique identifier of the EMCCD (default=Command:Line)
- Quit immediately after initialisation completes
- Add a string to the view window title

Submit Close
### Job Status Panel

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Worker Node</th>
<th>Batch Filename</th>
<th>Submitted</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>81.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>qmybdzrphr.sh</td>
<td>01-03-2013 14:41:54</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>78.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>icfhlkvhjf.sh</td>
<td>12-02-2013 13:51:57</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>77.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>agefhjzfwhf.sh</td>
<td>12-02-2013 13:51:51</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>76.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>xezhuyccms.sh</td>
<td>12-02-2013 13:40:39</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>75.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>fcebrhyxvp.sh</td>
<td>12-02-2013 13:40:29</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>74.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>iqlkkvabkk.sh</td>
<td>12-02-2013 10:50:48</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>73.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>dnfsmuakvy.sh</td>
<td>12-02-2013 10:48:21</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>64.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>ahtpltkhzc.sh</td>
<td>11-02-2013 15:27:14</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>65.sig-10.esc.rl.ac.uk</td>
<td>sig-12.esc.rl.ac.uk</td>
<td>phqrzrbcki.sh</td>
<td>11-02-2013 15:27:14</td>
<td>COMPLETED</td>
</tr>
</tbody>
</table>
Multiple Dataset Handling

- Jobs can accept a single or multiple datasets (specified in XML Job Description)
- Multiple datasets can be selected in the Portal
- Multiple datasets can be submitted to a job specified as accepting multiple datasets as input
- A separate batch job can be submitted for each dataset (with a single click)
- With multiple datasets selected, Job Options Form offers only options common to all datasets
Command Line Interface

• RESTful web service and Python client added for job handling
• Alternative to using web browser
• May become preferred interface for more proficient users
• Enables scripted interaction with Job Portal
Provenance Support in ICAT

Diagram:
- Application (name & version) -> Job
- Job -> InputDataset
- Job -> OutputDataset
- Job -> InputDatafile
- Job -> OutputDatafile
- InputDataset -> Dataset
- OutputDataset -> Datafile
- InputDatafile -> Datafile
- OutputDatafile -> Datafile
Provenance in Job Portal

• Job Portal will include a tool to visualise dataset provenance
• When a new dataset is added to ICAT the wrapper script must register the provenance information
• Relevant to PanData WP 6 – Data Provenance
Future Developments

• Improvements following user feedback
• LSF secured funding for 30 new dedicated nodes on the EMERALD GPU cluster at RAL
• Port installation to different OS/any OS?
• Make it batch system independent
• Find new groups of users for it