Developing and Operating MyTardis at Monash University

James Wettenhall and Manish Kumar
Monash eResearch Centre
https://www.monash.edu/eresearch

In collaboration with:
Amr Hassan, Wojtek Goscinski, Samitha Amarapathy, Keith Schulze, Chaitanya Manapragada, Grischa Meyer, Steve Androulakis, Jason Rigby, Andrew Perry, Anitha Kannan, Lance Wilson, Andrew Mehnert, Andrew Janke
What is MyTardis?

• A Django web application for storing, sharing and publishing research data

• Its particular focus is on:
  • Integration with scientific instruments
  • Integration between research storage and computing infrastructure
  • Large data storage, access, collaboration and publication

Source: [http://www.mytardis.org/about/](http://www.mytardis.org/about/)
What is Django?

• A high-level Python web framework that enables rapid development of secure and maintainable websites
• High-profile sites that use Django include:
  • Disqus, Instagram, Knight Foundation, MacArthur Foundation, Mozilla, National Geographic, Open Knowledge Foundation, Pinterest, and Open Stack

MyTardis: the early days ...
Store.* Integration Model

MyData desktop app

MyTardis Server

Storage

Store.* - Web portal / SFTP

HPC

Facility Overview

Facility Managers / Researchers

Researchers
Store.Monash – A Scalable MyTardis Deployment

- 54 Instruments
- 9 Million Data Files
- 0.6 PB Managed Storage
- 55,000 Datasets
- 1568 Users
- 856 Groups
Types of instruments integrated:

- **Microscopy**
  - Advanced light microscope
  - Fluorescence and Confocal microscope, Multiphoton microscope
  - Cryo-electron microscope,
  - Scanning electron microscope
  - Transmission electron microscope
- **Flow Cytometry**
  - Cell Sorter, Analyzers
- **Imaging**
  - X-ray tomography, MRI, PET/MRI, PET/CT
- **Proteomics**
  - Mass Spectrometers
- **Gene Sequencers**

Data Usage:
- **Electron Microscopy** 77%
- **Optical Microscopy** 13%
- **Biomedical Imaging** 1%
- **Other** 5%
- **X-Ray Tomography** 3%
- **Cytometry** 1%
Store.Monash – Addressing Scalability

What Stack Do We Start With?

One Server

- Static and media files on server's FS
- Database server
- gunicorn running Django behind nginx
- Cron for periodic tasks

Source: https://blog.bitlabstudio.com/from-single-server-to-cloud-c6d4d0e957a
Store.Monash – Addressing Scalability

Source: https://blog.bitlabstudio.com/from-single-server-to-cloud-c6d4d0e957a
OpenStack Orchestration with Heat

Heat Basic Workflow

Source: https://www.oreilly.com/library/view/openstack-orchestration/9781783551651/ch05s04.html
Store.Monash Heat Workflow

- Download the current Heat template
- Add/modify resources (VMs)
- Perform a dry run
- Apply the updates:
  - `openstack stack update`
- Accept the new VMs’ SaltStack keys on the SaltMaster:
  - `sudo salt-key -accept`
- MyTardis and its dependencies can be installed with:
  - `salt '*'new_vm_identifier*' state.apply`
Configuration Management with SaltStack

Source: https://www.mirantis.com/blog/introduction-to-salt-and-saltstack/
Store.Monash – Scaling Out

• We can use Heat / SaltStack to scale out!

• But how do we know when we need to scale out?

• **Monitoring with Nagios:**
  • Size of Celery queue (asynchronous tasks waiting to be processed)
    • If too large, add Celery VMs
  • Response time from RESTful API:
    • If too slow, add API VMs
  • Upload speed of large files via SCP
    • If too slow, add SCP servers / CPU cores
So we can add RESTful API VMs or Celery VMs, but what are they?

- **VMs with MyTardis which run one or more services:**
  - RESTful API VMs run gunicorn
  - Celery Worker VMs run celery worker processes
  - The Celery Beat VM submits scheduled tasks

- **HAProxy ACLs route URL patterns to VM flavours:**
  - Web UI VMs
  - RESTful API VMs
  - Download VMs (construct TARs on the fly)
  - All flavours of web VMs run gunicorn
Store.Monash – Monitoring

• Detecting the need to scale out is not the only reason for monitoring

• I/O wait
  • When using hybrid disk/tape storage, we want to be alerted when MyTardis is unexpectedly recalling data from tape

• Single points of failure
  • We only have one “Celery Beat” scheduler, so we want to be alerted immediately if it stops submitting regularly scheduled tasks

• Database replication failover
  • We want to be alerted if/when our master PostgreSQL server fails and a slave is promoted to master

• Unscheduled outage
  • Send appropriate communications to users
Managing your Instrument Data with MyData and MyTardis
Managing your Instrument Data with MyData and MyTardis

1. How your data is organized?
2. Add Instrument to MyTardis
3. Configure your MyData
Initial Consultation with Data Collection Facility

• Low barrier of entry for ingesting data
  • MyData
  • Metadata entry is not required to begin uploading

• Work closely with storage providers
  • How much data (GB or TB) will be collected per week and how often will it be accessed after ingestion?
Initial Consultation with Data Collection Facility

• How is your data structure currently?
  • How can you identify who should have access to each dataset?
  • Are you willing to restructure your data into “User” or “User Group” folders?
  • For new data only? Or for legacy data too?

• MyData
  • A desktop application (http://mydata.readthedocs.io) which supports many different folder structures.
  • We are also developing a “Drag and Drop” solution.
MyData – Drag and Drop
MyData – Drag and Drop

The 'dragged_dir' folder will be uploaded to https://store.erc.monash.edu

Please complete the required fields below:

Email: registeredUser@organisation.edu

[Buttons: Cancel, Upload]
MyData – Drag and Drop

The 'dragged_dir' folder will be uploaded to https://

Message
Adding to upload queue...

OK
# MyData – Drag and Drop

<table>
<thead>
<tr>
<th>Id</th>
<th>Folder (dataset)</th>
<th>Location</th>
<th>Created</th>
<th>Experiment</th>
<th>Status</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>data</td>
<td>/Users/chait</td>
<td>2018-07-26</td>
<td>Test Instrument - Test User1</td>
<td>2 of 2 files uploaded</td>
<td>testuser1</td>
</tr>
</tbody>
</table>

Uploaded 2 of 2 files.
MyData – Drag and Drop

The dataset is now available on the MyTardis server and is collected into the “Test Instrument – Test User1” experiment, because the email address associated with the dragged folder matched the email address of “Test User1”:
MyTardis New Features

Customizable Login Page and Social Auth / OpenID Connect Auth

New User? If you don’t have an account yet, logging in with any of these methods will create an account for you. Logging in with the Australian Access Federation is the preferred authentication mechanism. If you don’t have an Australian Access Federation account, you can Log In with your Google Account, but this will require an additional activation step.

Source code: https://github.com/monash-merc/Monash_OpenID_Login
MyTardis New Features

Account Migration (e.g. from LDAP to AAF)

Migrate My Account

This process allows you to migrate your data and settings from one account to another within Store.Monash. Please note that this process will automatically **deactivate your old account**. All your data, groups and instrument associations, and settings will be migrated.

For more details, please refer to the [Store.Monash Account Migration Instructions](#)

**Step 2:** Are you sure you want to migrate all the data, groups and instrument associations, and setting between the following accounts?

**Legacy Authentication**

**Old User**

- Username: wettenhj
- Email: James.Wettenhall@monash.edu

**New User**

- Username: wettenhj
- Email: James.Wettenhall@monash.edu

[Submit] Migrate [Cancel]
MyTardis New Features
SFTP Authentication via SSH Keys

SSH Keys
Add and manage your public SSH keys for accessing Store.Monash via SFTP.

You don’t have any public keys registered. Please add keys using the Add key button above.
MyTardis New Features

SFTP Authentication via SSH Keys

Generate an SSH key pair

Store.Monash can generate an SSH key pair for you. Only the public part of the key pair will be stored by Store.Monash and you will be provided with a one time opportunity to download the private part of the key pair.

Name:

james-test

Generate & Download

Important: Keep your private key in a safe place. Anyone with access to the private key will be able to access your data on Store.Monash.

Close
MyTardis New Features

SFTP Authentication via SSH Keys

SSH Keys
Add and manage your public SSH keys for accessing Store.Monash via SFTP.

<table>
<thead>
<tr>
<th>Keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>james-test</td>
</tr>
<tr>
<td>Type: SSH-RSA</td>
</tr>
<tr>
<td>Fingerprint: 7b23dc39f9cc10f086bd97076664e13f</td>
</tr>
<tr>
<td>Date added: 2018-10-14</td>
</tr>
</tbody>
</table>

+ Add key
+ Generate key

Delete
MyTardis New Features

SFTP Authentication via SSH Keys

![Cyberduck SFTP Configuration](image-url)
Characterisation Virtual Laboratory - CVL @ M3
MyTardis “Push To” CVL @ M3

Dataset
Push To Demo
From the experiment: Test Push To

Preview Images

1 File
Download Dataset: TAR SFTP MD5

Search: Enter part or all of a filename, then

Select: All / None

Download Selected Files

- eRA-favicon.png (4.0 KB)
MyTardis “Push To” CVL @ M3

Login to MeRC Authentication Translator
MeRC is used by Monash University as a service to translate SAML tokens to x509 and openssh certificates

Please select your organisation below, you will be redirected to complete the login process.

Search for your organisation

AAF Virtual Home
AARNet
AIMS

Continue to your organisation

Remember my organisation

Australian Access Federation
Current AAF status
Contact AAF support

Username
wettenhj

Password
*************

Don’t Remember Login
Clear prior granting of permission for release of your information to this service.

Login

MeRC is used by Monash University as a service to translate SAML tokens to x509 and openssh certificates

Forgot your password?
Need Help?
MyTardis "Push To" CVL @ M3

Authorisation required

MyTardis is requesting permission to act on your behalf

MyTardis would like to:
- Access your MASSIVE and/or CVL account

MASSIVE (M3)

Enter a destination for your files:

Push
MyTardis “Push To” CVL @ M3

The requested item will be pushed to MASSIVE (M3). You will be notified by email once this has been completed. Data will be pushed to

/home/wettenhj

[TARDIS] Data pushed successfully

[wettenhj@m3-login2 ~]$ find /home/wettenhj/mytardis-data/
/home/wettenhj/mytardis-data/
/home/wettenhj/mytardis-data/Push To Demo
/home/wettenhj/mytardis-data/Push To Demo/eRA-favicon.png
MyTardis Search

- Search is disabled by default, because it requires setting up an Elasticsearch index
- Instructions for enabling search are available here:
- On Store.Monash, we:
  - run the Elasticsearch index service (v1.7.5) on each PostgreSQL database server node and
  - run an HAProxy load balancer for the Elasticsearch index services on the database connection pooling node (PgPoolServer).
Searching for “diffraction” shows both a matching experiment and a matching dataset (which happens to be within the matching experiment):
MyTardis Search

Investigating a Scalability Puzzle

- Having noticed that search wasn’t working as expected for Store.Monash system administrators or for facility managers with access to thousands of experiments, we investigated and found the following:
- As access control in MyTardis is done at the experiment level, finding the intersection between the experiments a user has access to and the matching results from Elasticsearch was sending a complex where clause to our Elasticsearch backends.
- The solution was to increase:
  - `index.query.bool.max_clause_count`
- from its default value of 1024 in:
  - `/etc/elasticsearch/elasticsearch.yml`
**Automatic extraction of metadata and thumbnail images**

### Bioformats

#### Datafile Metadata

For the file `Test.nd2`

<table>
<thead>
<tr>
<th>Series Metadata</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Preview</strong></td>
<td><img src="image" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dimension Order</th>
<th>XYCZT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emission Wavelength</td>
<td>Channel 2: 665</td>
</tr>
<tr>
<td>Emission Wavelength</td>
<td>Channel 1: 515</td>
</tr>
<tr>
<td>Emission Wavelength</td>
<td>Channel 0: 450</td>
</tr>
<tr>
<td>ID</td>
<td>Image0</td>
</tr>
<tr>
<td>Name</td>
<td>Test.nd2</td>
</tr>
<tr>
<td>Voxel Size X</td>
<td>0.0021485569422</td>
</tr>
<tr>
<td>Voxel Size Y</td>
<td>0.0021485569422</td>
</tr>
<tr>
<td>Samples per Pixel</td>
<td>Channel 0: 1</td>
</tr>
</tbody>
</table>
Automatic extraction of metadata and thumbnail images

### Datafile Metadata

For the file: Specimen_001_BM M10_015.fcs (54.3 MB)

<table>
<thead>
<tr>
<th>Parameter (Unit)</th>
<th>Stain</th>
<th>Range (Unit)</th>
<th>Bits (Unit)</th>
<th>Decades (Unit)</th>
<th>Gain (Unit)</th>
<th>Voltage (Unit)</th>
<th>Channel Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSC-A</td>
<td></td>
<td>262144</td>
<td>32</td>
<td>0.0</td>
<td>1.0</td>
<td>478</td>
<td></td>
</tr>
<tr>
<td>FSC-H</td>
<td></td>
<td>262144</td>
<td>32</td>
<td>0.0</td>
<td>1.0</td>
<td>478</td>
<td></td>
</tr>
<tr>
<td>SSC-A</td>
<td></td>
<td>262144</td>
<td>32</td>
<td>0.0</td>
<td>1.0</td>
<td>263</td>
<td></td>
</tr>
<tr>
<td>B670-A</td>
<td>PI</td>
<td>262144</td>
<td>32</td>
<td>0.0</td>
<td>1.0</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>

Flow Cytometry
MyTardis has been supported by:

http://www.mytardis.org/
https://github.com/mytardis/mytardis
https://mytardis.readthedocs.io
store.star.help@monash.edu