

Technology of the Characterisation Virtual Laboratory (C-DeVL project)

Dr Chris Hines (chris.hines@monash.edu)
Dr Lance Wilson (lance.wilson@monash.edu)
For assistance with CVL help@massive.org.au



Sea Devil (C-DeVL) *Melanocetus johnsonii*

Today

What is the Characterisation Data Enhanced Virtual Laboratory?

Technology:

- Technology to support federation
- Technology to support desktops and identity

What is the CVL? Our view:

- A program of work to connect Australian Characterisation instruments with data management environments, tools and analysis pipelines on the Australian research cloud.
- 100+ Instruments of \$250m value, 2,700+ users of CVL and CVL-developed software

The researcher view:

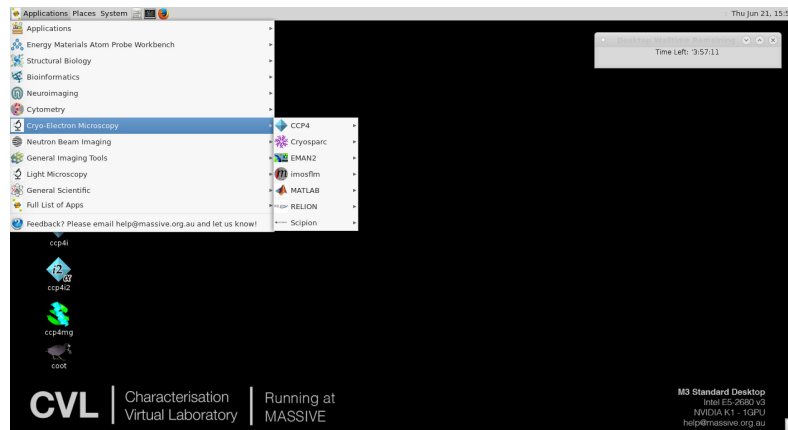
- A place where a researcher has their characterisation data and software tools and infrastructure in the one place

Characterisation Virtual Laboratory:

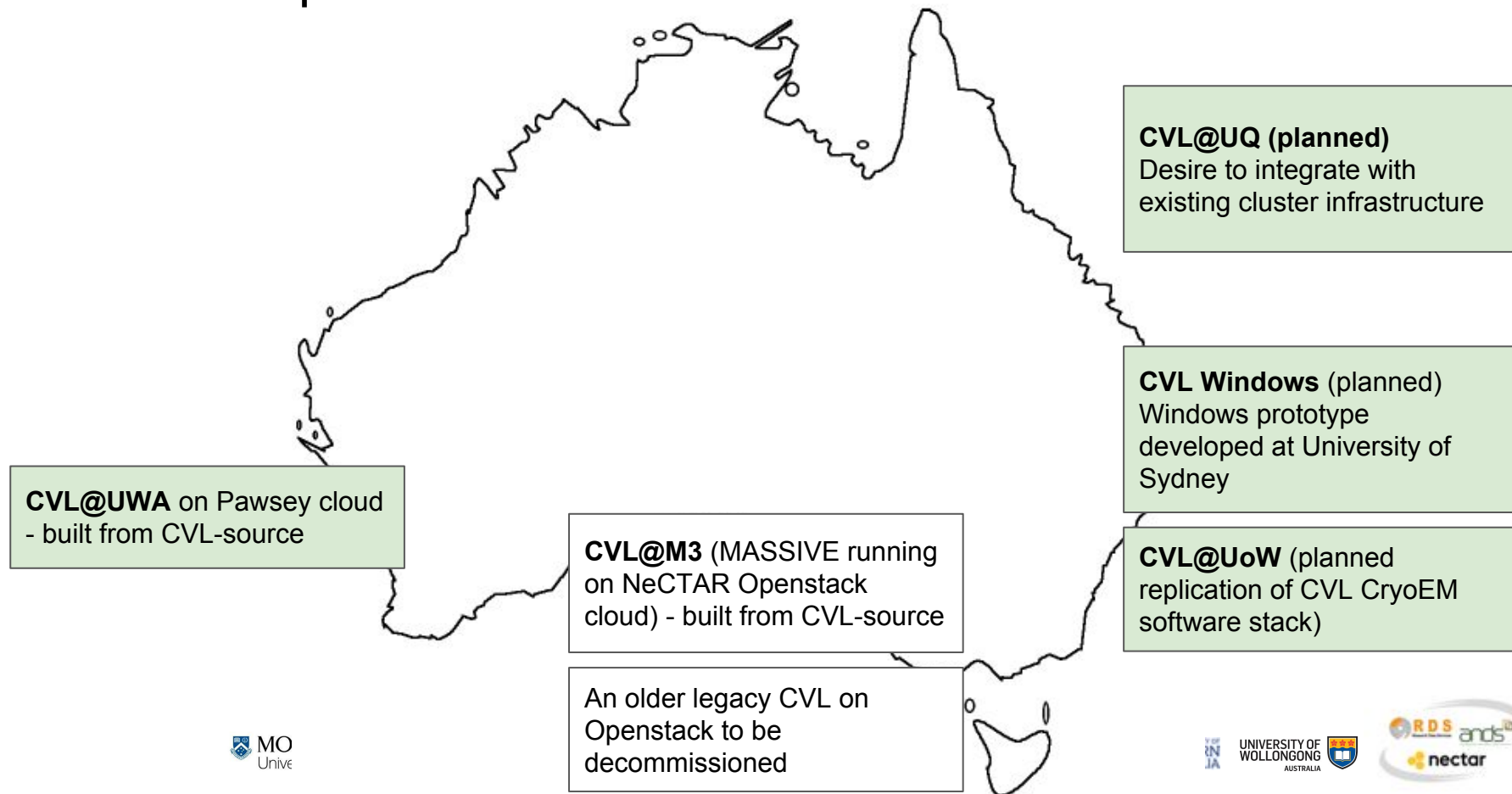
Start with the “data at the instrument” and upwards approach

C-DeVL program (in absolute summary):

1. Connect instruments
2. **Create rich online environments**
 - a. **Federate the CVL across multiple nodes**
3. Training and FAIR data program



Current and planned instances of Federated CVL



Federation Principles from Consultation

- Single user portal
- Single sign up/registration (AAF authenticated)
- Single helpdesk
- Shared software stack (in as much as is possible considering technical and licensing challenges)
- Single desktop user interface and branding UI (menus, background, software modules)
- A consistent user experience, whilst ensuring that the best possible capability (hardware, software specialisation, instrument integration) of each node is exposed

Additional to the desktops are the instrument integration components of:

- Local initiatives to capture and 'push' data to this environment
 - Not considered within scope. However, flagged as essential and likely considered in future.
- Domain support, expertise and contribution to the software stack
 - Leveraging specialised knowledge across the network.

CVL Node Types - Technology Stacks

CVL Instance - “From Source on Openstack”

‘CVL in a box’

Cluster
Operations

Desktop System

Job Scheduler
(SLURM)

M3 LDAP

Software Repo.

CVL Build

NeCTAR OpenStack Cloud

CVL Instance - “Leverage Existing HPC”

CVL Consistent User Experience

Desktop capability
and identity

AAF-ID
translation

Research
Software

Node-specific build (likely an existing HPC system)

Cluster
Operations

Desktop System

Job Scheduler

Site-specific
LDAP / ID

Node-specific build

HPC / Cloud

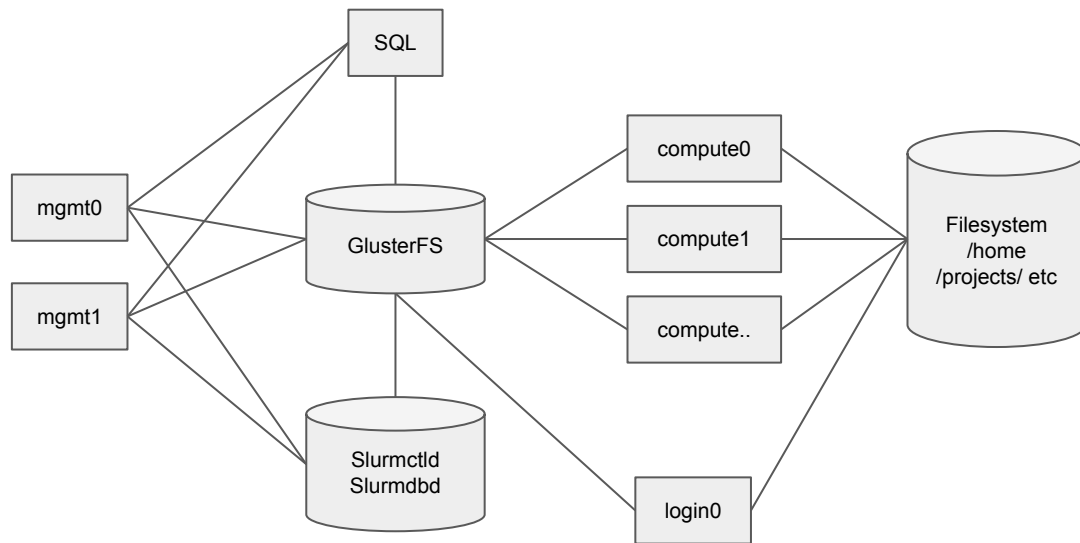
How does CVL deliver desktops?

Cluster configuration and deployment technologies:

- Openstack
- Ansible
- Git

Desktop delivery technologies:

- Strudel-desktop
- Strudel-web



Ansible (What do we use it for?)

1. Define Your configuration
2. Let ansible get it there

```
[lancew@m3-login1 ~]$ cd clusterbuild/
[lancew@m3-login1 clusterbuild]$ . ../ansible_local/bin/activate
(ansible_local) [lancew@m3-login1 clusterbuild]$ ansible -i m3inventory 'm3g*' -m shell -a "uptime"
[ERROR]:

m3g014 | SUCCESS | rc=0 >>
  11:37:08 up 2 days, 21:21,  1 user,  load average: 0.03, 0.04, 0.05

m3g015 | SUCCESS | rc=0 >>
  11:37:08 up 2 days, 21:09,  1 user,  load average: 4.03, 4.04, 4.05

m3g017 | SUCCESS | rc=0 >>
  11:37:08 up 24 days, 19:52,  1 user,  load average: 10.08, 5.36, 5.52
```

Ansible (What do we use it for?)

1. Define Your configuration
2. Let ansible get it there
3. Move along nothing to see here

Git (How we keep our services consistent)

All our cluster configs have

1. A Shared component
2. A cluster specific component (passwords, which shared components to use)

Git submodules (or how do we allow differences?)

Confusing as heck but ...

Multiple teams can update shared roles

Each cluster needs to QA changes before accepting an update

The submodule allows versioning of the shared roles.

How do we make changes? (Code review & People)

One person makes changes and submits

Another reviews and accepts

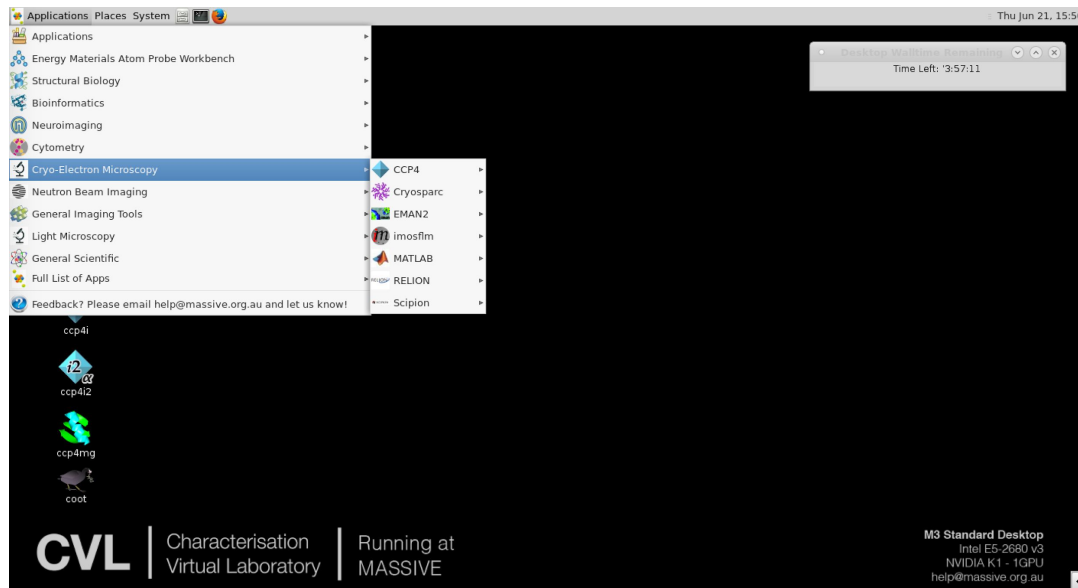
Either one deploys

If you break it, cake it.

<http://isocake.readthedocs.io>

Federation of CVL

- Auth Once
- Access multiple clusters (not everyone gets all clusters)
- Clusters have identical environments (singularity containers)



Federation Nuts and Bolts

1. Strudel-web asks permission to access a cluster
2. you login via AAF
3. Lookup the cluster map (usually ldap) for AAF Id to cluster username
4. OAuth2 gives permission for Strudel-web to access the cluster with your username
5. Strudel-web gets a short lived SSH Credential (i.e. its access expires)

Federation Buy-in

- Each cluster keeps its own set of usernames and groups
- Just because you have access to one cluster doesn't give you access to all of them ... yet?

Next Gen Federation Maybe ...

Strudel V2 asks permission to access all your CVL clusters at once

Lookup usernames for you on all clusters concurrently

Authorise access to all clusters at the same time

Allows us to build tools to copy between clusters (run jobs where most appropriate)