

# Cluster in the Cloud

---

Easy, Scalable, Heterogeneous



**CLUSTER IN  
THE CLOUD**

Matt Williams  
Research Software Engineer  
University of Bristol

# The problem

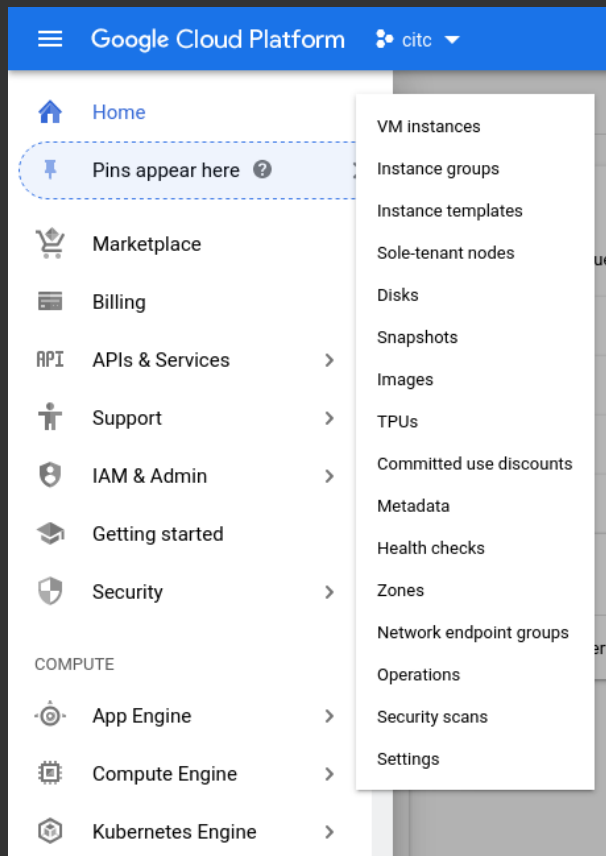
---

- Researchers having cloud credits

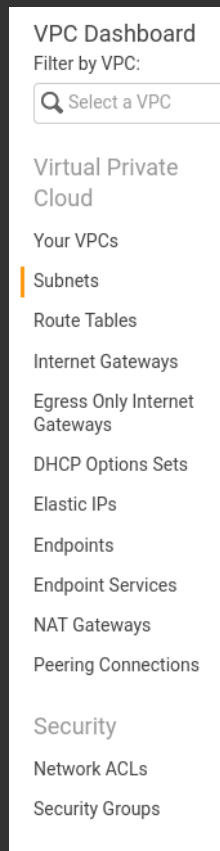
# The problem



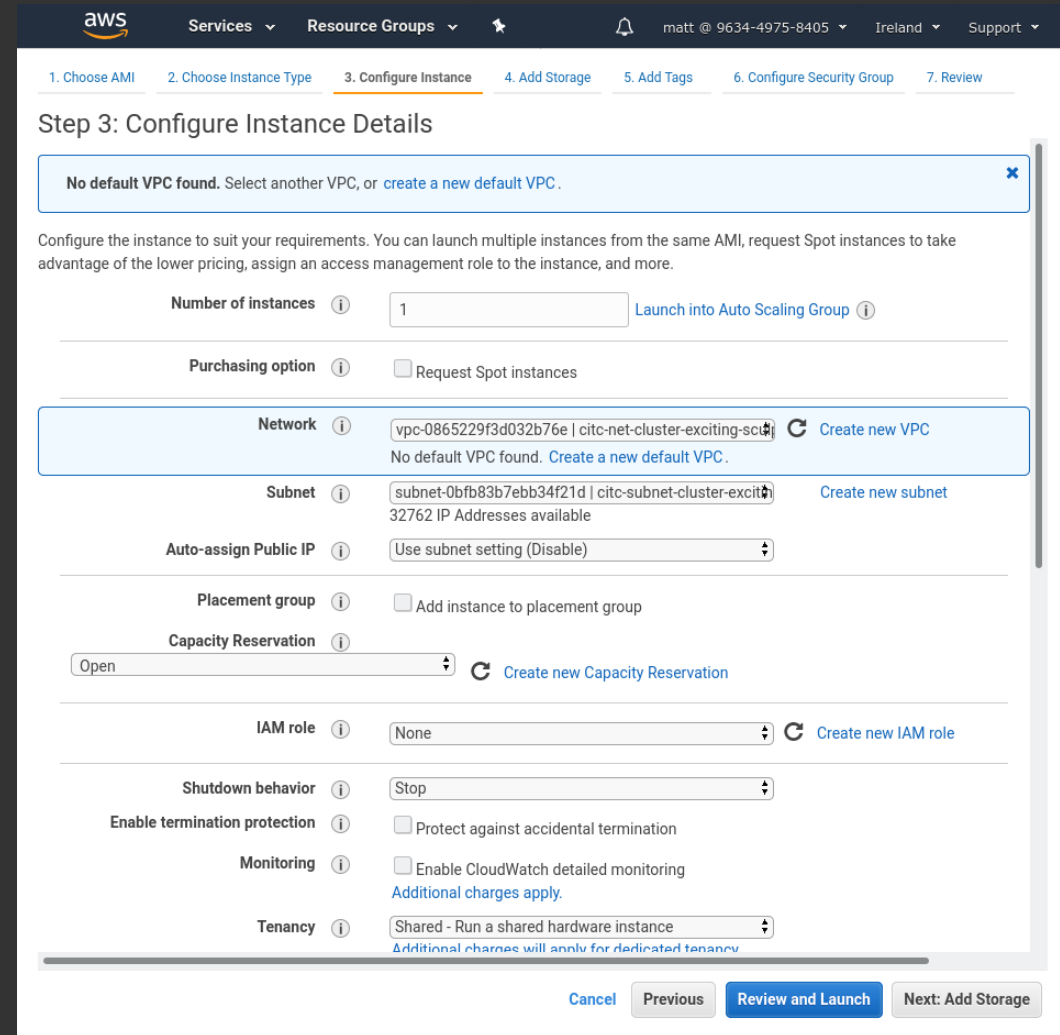
- Researchers having cloud credits
- Presented with:



Google Cloud Platform navigation menu showing categories like Home, Marketplace, Billing, APIs & Services, Support, IAM & Admin, Getting started, Security, and COMPUTE services such as App Engine, Compute Engine, and Kubernetes Engine.



AWS VPC Dashboard showing a list of Virtual Private Clouds (VPCs) and associated resources like Subnets, Route Tables, and Internet Gateways.



AWS Step 3: Configure Instance Details. A notification states: "No default VPC found. Select another VPC, or create a new default VPC." The configuration includes fields for Number of instances (1), Purchasing option (Request Spot instances), Network (vpc-0865229f3d032b76e), Subnet (subnet-0bfb83b7ebb34f21d), Auto-assign Public IP (Use subnet setting (Disable)), Placement group (Add instance to placement group), Capacity Reservation (Open), IAM role (None), Shutdown behavior (Stop), Enable termination protection (Protect against accidental termination), Monitoring (Enable CloudWatch detailed monitoring), and Tenancy (Shared - Run a shared hardware instance).

# The problem 🙄

---

- What they already know:
  - Their field of research
  - Python/R/GROMACS/Relion
  - sbatch/qsub
- We can't expect researchers to be professional sysadmins
  - The intersection is well handled by *Research Software Engineers*

# The solution



- 
- Give them what they are used to, but in a cloud environment
  - They don't have to know the difference
  - Except:
    - No queuing
    - Only pay for what they use
  - *Cluster in the Cloud*

# Cluster in the Cloud

---



An automatically-provisioned Slurm cluster



Uses Terraform to create:

- Networking
- Shared file system
- Management/login VM (e.g. `t3a.medium`)

A

Uses Ansible to configure the management VM and compute image

# Key Features

---

- 1. Familiar:** known environment for researchers with Slurm, JupyterHub etc.
- 2. Versatile:** Allows any number of any combination of instance types in a cluster
- 3. Dynamic:** They are started only when needed
- 4. Cheap:** Base cost is just one VM plus storage
- 5. Cross-cloud:** Works on AWS, Google Cloud and Oracle
- 6. Open source:** Under the MIT license and is free to use.

# Technical details: Terraform

---



- Terraform is used to create the skeleton
- <https://github.com/clusterinthecloud/terraform>
  - AWS: ~500 LOC
  - Google: ~400 LOC
  - Oracle: ~450 LOC
- Written from scratch for each platform



# Technical details: Ansible

---



- ~1.5K lines of Ansible
- <https://github.com/clusterinthecloud/ansible>
- Configures:
  - Mounting shared filesystem
  - LDAP for user management
  - Slurm
    - Including node start/stop scripts
  - Monitoring (Grafana)
  - Base software set
  - And more...
- Covers both the management VM and compute image

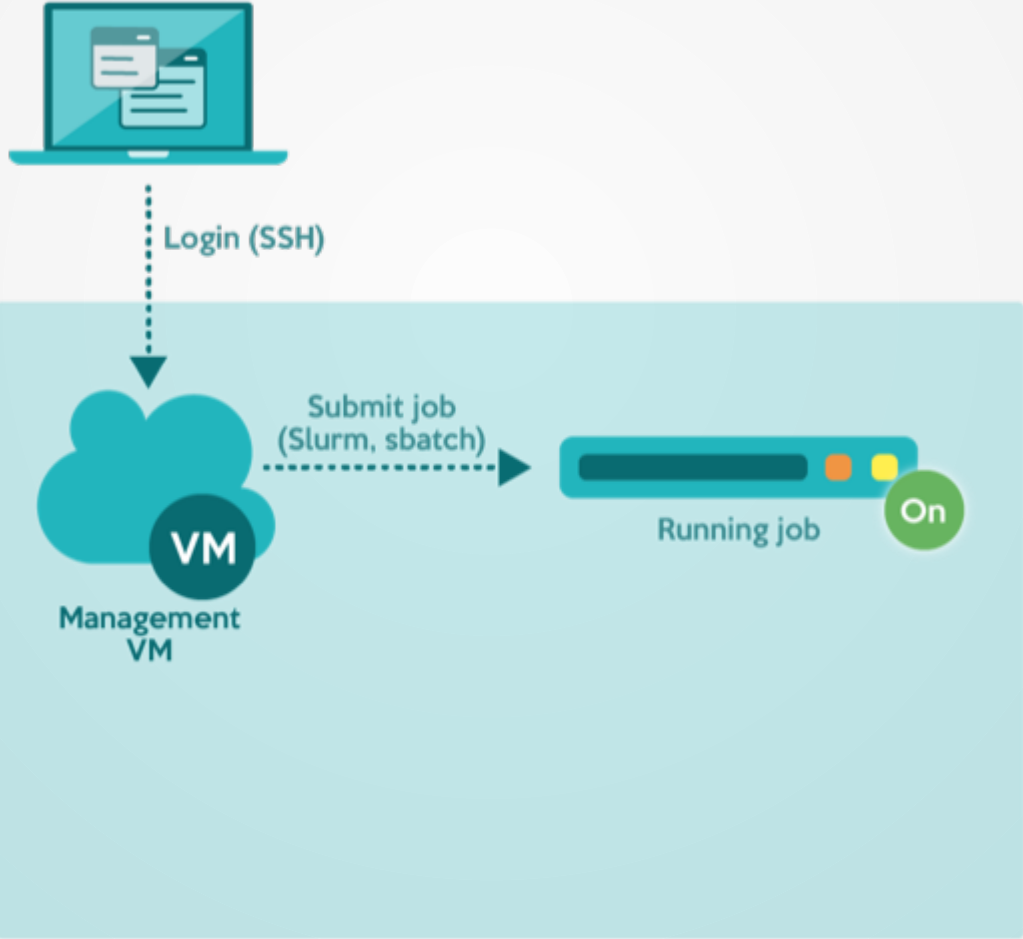
# Scaleable

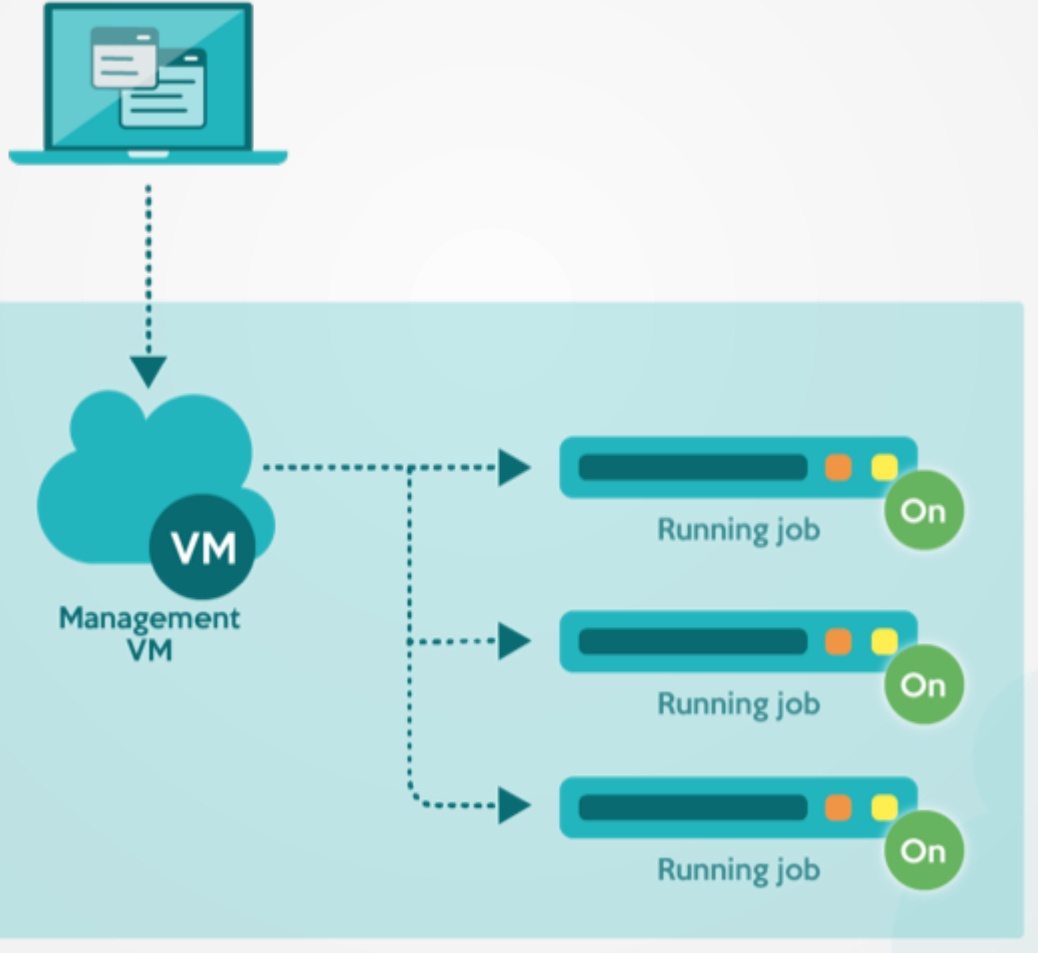
---

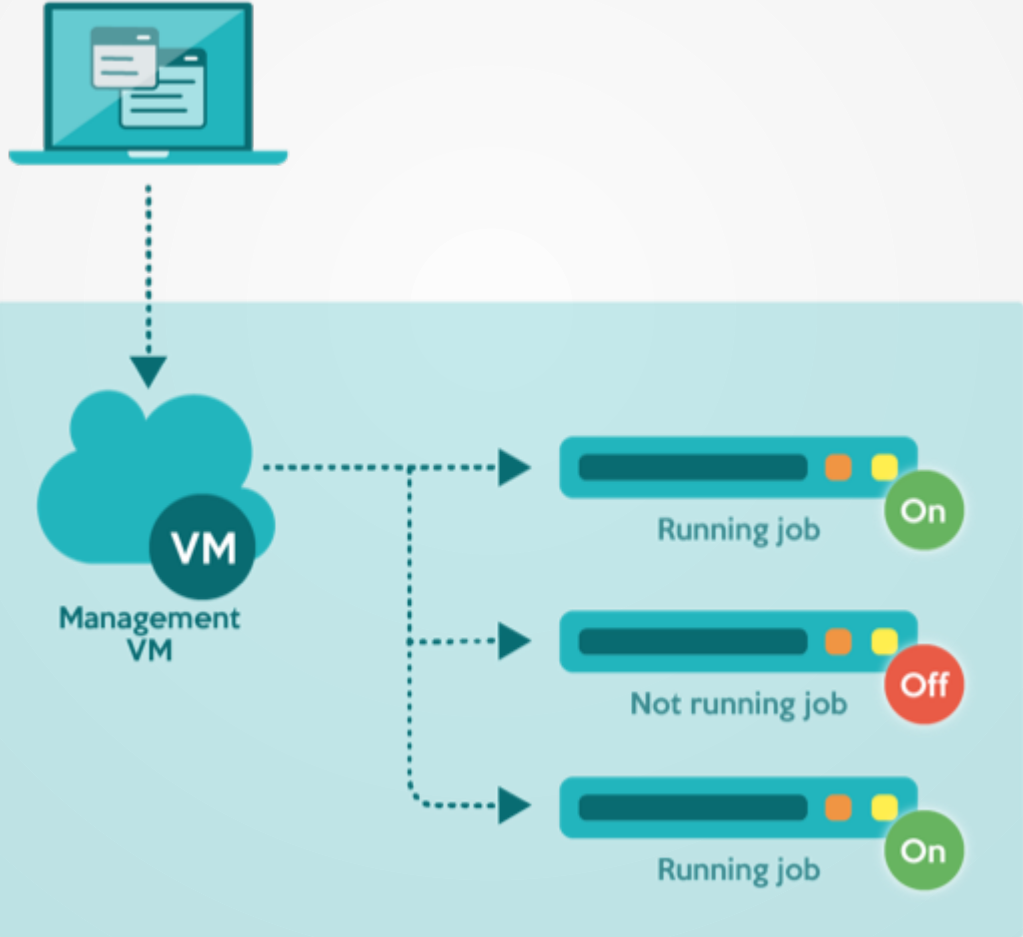
- At initial configuration you create any number of *potential* nodes of each desired type:
  - e.g. 1000 32-core, 1000 16-core, 1000 GPU etc.
- On job submission Slurm
  1. Chooses a node type
  2. Creates an appropriate instance from an image (via a Python script calling the cloud API)
  3. Runs the job
  4. Destroys it (after a timeout)



Management  
VM





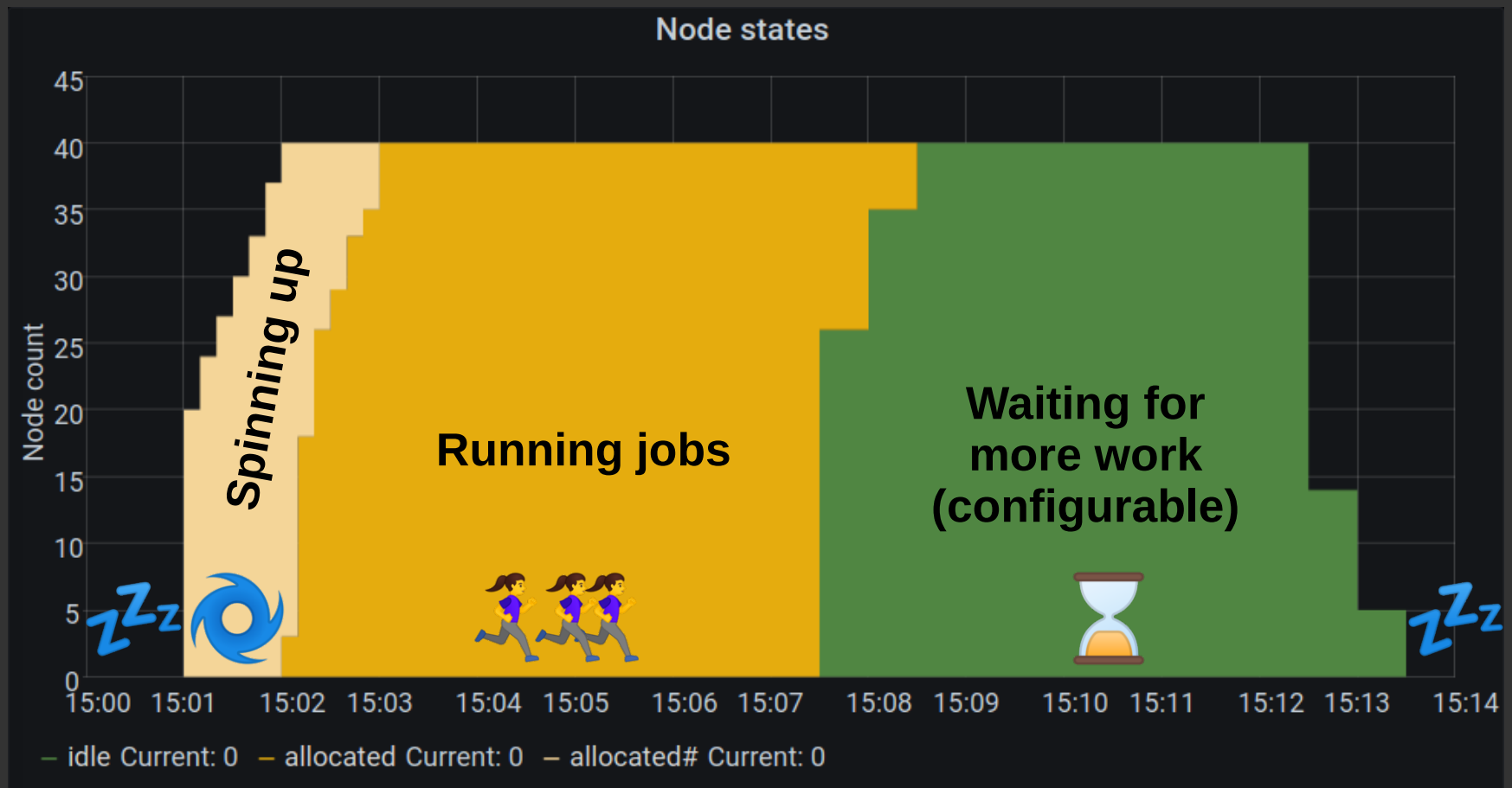




Management  
VM

# Node states

- Example 40-node array job, 5 minute runtime





# Timings

---

- Full system test ~17 minutes on AWS
  1. Create cluster from scratch, including node images
  2. Run test job
  3. Check other system statuses
  4. Tear down whole cluster
- Job submit → job start: 1 minute

# Performance characteristics

---



- ✓ Best-suited to heterogeneous high-throughput tasks
  - Pipelines needing different node type for different parts
  - Can be much more specific than the average on-premise cluster
  - Always access to latest hardware, e.g Graviton 2
- ✗ At present is not optimised for multi-node workloads
  - No fast interconnect support (work underway)
  - Only cheap shared storage
- ✓ Great for teaching clusters and benchmarking
- ✓ Suitable for Dask, Spark, Singularity

# Users

---

- **Smoking cessation:** A General Mechanism for Signal Propagation in the Nicotinic Acetylcholine Receptor Family [10.1021/jacs.9b09055](https://doi.org/10.1021/jacs.9b09055)
- **Vaccine delivery:** Synthetic self-assembling ADDomer platform for highly efficient vaccination by genetically encoded multiepitope display [10.1126/sciadv.aaw2853](https://doi.org/10.1126/sciadv.aaw2853)
- **COVID-19:** Free fatty acid binding pocket in the locked structure of SARS-CoV-2 spike protein [10.1126/science.abd3255](https://doi.org/10.1126/science.abd3255)
- **Molecular dynamics:** Molecular Simulations suggest Vitamins, Retinoids and Steroids as Ligands binding the Free Fatty Acid Pocket of SARS-CoV-2 Spike Protein (under review)
- **Other projects:**
  - Carbon sequestration
  - Radiotherapy research

# Future plans

---

- Improved web UI for managing users, node types, apps, benchmarking results etc.
- High-performance networking
- Pluggable storage solutions
- Easy backups to cloud storage
- Support for other clouds (Azure, OpenStack)

# Thank you

---

Find out more at  
[cluster-in-the-cloud.readthedocs.io](https://cluster-in-the-cloud.readthedocs.io)

Thanks to AWS, Google and Oracle for supporting development and to the Bristol RSE team