Re-Thinking Experiments

Accelerating Research through Reproducible Experiments at Scale

Vivek Katial

09/12/2019

Slides

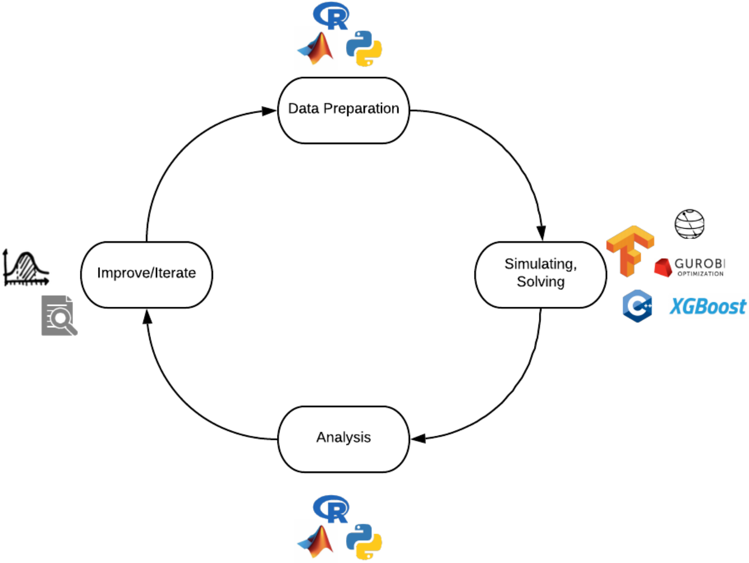
- Check out the slides at https://tinyurl.com/rpoe9jw

About Me

- Vivek Katial (vkatial@student.unimelb.edu.au)
  - PhD Candidate (Optimisation on Quantum Computers)
  - Data Scientist (3 years)

Motivation

Running Experiments and Developing ideas is Complex



Issues

1. Data Prep:
   - Want to test many different parameters
2. Simulating,Training, Solving:
   - Track various metrics
   - Use different algorithms or solvers
3. Analysis:
   - Work needs to be reproducible
4. Re-iterate:
   - Need to be able to track parameters, metrics, etc...

- Other issue is that this all needs to **scale**

Introducing “EZ-EXPERIMENTr”

- **EZ-EXPERIMENTr** is a toolkit under-development with the aim to help researchers run automated experiments on an HPC cluster.
- How it works?
  - *SPARTAN* to run heavy-duty experiment code
  - *Melbourne Research Cloud (MRC)* to host server that tracks experiments
  - *Singularity Containers* to make each experimental run fully reproducible
  - *GitHub* to manage and version code
  - `yaml` to configure and specify a run of each experiment
  - *mlflow* to track parameters, metrics at scale

Cloud Infrastructure

SPARTAN and MRC

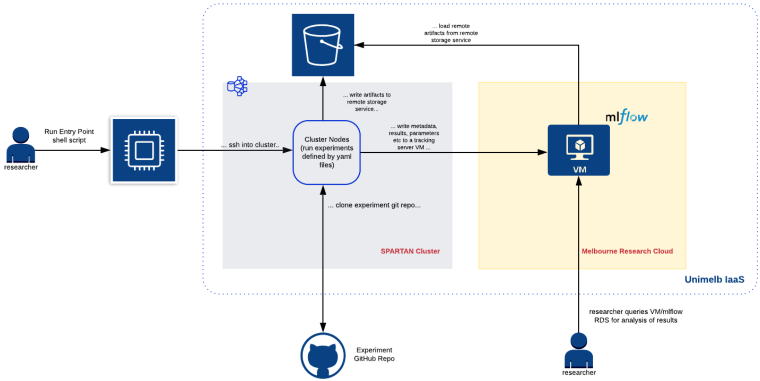
- Spartan is High Performance Computing (HPC) system operated by Research Platform Services (ResPlat) at The University of Melbourne.
- Why?
    - Computing takes too long!
    - Lack of memory and speed
- The Melbourne Research Cloud (MRC) provides Infrastructure-as-a-Service (IaaS) cloud computing to the University of Melbourne researchers.
- Why?
    - Quickly deploy virtual machines
    - Similar to AWS, Microsoft Azure, GCP

Singularity Containers

- Singularity is a container technology. Essentially a container is a curated collection of resource requirements to run an application.
- Specified by a simple text file which is easily documented and version-controlled through tools like Git.
- Allow for the reproducibility of entire environments and also make applications portable.
- Container technologies like Docker are widely used in industry, however often in research environments (where arguably reproducibility is most important) users don't have root access to the computing resources they use (particularly in HPC).
- Singularity is a container-technology designed for use in HPC environments.

Example Container

Architecture



Concepts

Experiments and Runs

- The tool is organized around the concept of *experiments* and *runs*.
- An experiment can be considered the wider project.
  - Each experiment is defined by a **GitHub** repository
- A run is an execution of one-configuration of that experiment.

  A run consists of:
  - Code Version
  - Parameters
  - Metrics
  - Artifacts

Define an Experiment

- Each experiment is defined by a text file
- The text file specifies:
  - One configuration of the experiment
  - Parameters
  - Path to scripts
- Easy to understand

```
experiment:
  repository: "aqc-three-sat-sim" # GitHub Repostiry
  github_url: "https://github.com/vivekkatial/aqc-three-sat-sim"
  cluster_uri: "/data/cephfs/punim1074/"
  cluster_scratch_dir: "/scratch/punim1074/"
  cluster_provider: "unimelb_SPARTAN"
  singularity_image_uri: "ubuntu@115.146.94.33:aqc-three-sat-sim/portable-image.img"
```

An example of a run

- Each experiment is defined by a text file
- The text file specifies:
    - One configuration of the experiment
    - Parameters
    - Path to scripts
- Easy to understand

```yaml
experiment:
  name: "three-sat"
  author: "Vivek Katial"
  tracking-uri: "http://<mlflow-server-ip-address>:5000"
  seed: 1032918

initialise:
  source: "src/generate-instances.R"
  params:
    n_qubits: "5"
    k: "4"
    n_sat: "3"

build_hamiltonians:
  source: "src/run-time-evolution.R"
  params:
    time_T: "100"
    t_step: "0.1"
    num_energy_levels: "4"

results:
  source: "src/produce-plots"
```