Experiment Management from a Pegasus Perspective

Jens-S. Vöckler

Ewa Deelman

Outline

You know what FutureGrid is, but…

- What is an Experiment?
- What is Pegasus?
- How do the two connect?
- What extra are we building?

What Is a Scientific Experiment?

in a nutshell

1. Create a hypothesis
2. Design an *experiment* to prove or disprove
   – Document your setup (**apparatus**)
3. **Run** and **observe** (be ready to be surprised)
   – Ensure sufficient **sensors** (placement, granularity)
   – Document all observations (report)
4. Draw conclusions (paper, publication)
   – Others should be able to **repeat** the experiment

Experiments Using Computer Science

- The *apparatus* is often a (set of) program(s) and execution environment from the domain science
- The *experiment* often involves:
  1. Processing massive data with same code (proudly parallel)
  2. Complex processing in dependent steps (workflow)
- *Sensors* often constitute log files and monitoring

Pegasus is set to deal well with all of the above.

Pegasus
Workflow Management System

- Developed since 2001
- A collaboration between USC and the Condor Team at UW Madison (includes DAGMan)
- Used by a number of applications in a variety of domains
- Provides reliability
  - can retry computations from the point of failure
- Provides scalability
  - can handle large data (kByte…TB of data),
  - and many computations (1…10^6 tasks)

INFORMATION SCIENCES INSTITUTE
. . . agent of innovation . . .

USC Viterbi
School of Engineering

Pegasus Workflow Management System

- Automatically captures provenance information ☞ *apparatus, sensors*
- Can run on resources distributed among institutions, laptop, campus cluster (HPC), Grid, Cloud
- Enables the construction of complex workflows based on computational blocks
- Infers data transfers
- Infers data registrations

INFORMATION SCIENCES INSTITUTE
. . . agent of innovation . . .

USC Viterbi
School of Engineering

Pegasus WMS

- Provides a portable and re-usable workflow description ☞ *experiment, repeatability*
- Lives in user-space
- Provides correct, scalable, and reliable execution
  - Enforces dependencies between tasks
  - Progresses as far as possible in the face of failures
- Pegasus makes use of available resources, but cannot control them

INFORMATION SCIENCES INSTITUTE
. . . agent of innovation . . .

USC Viterbi
School of Engineering

Pegasus WMS runs Experiments

- Workflows capture hypotheses
  - – Abstract description independent of apparatus
  - – Can be shared to repeat experiments
- Multiple concurrent experiments
  - – Automatic batch-style execution
- Provenance captures apparatus
  - – Provenance helper *kickstart* to aide sensors

Room for Improvement

- Support for interactive steps
  - – Need to separate into multiple workflows
- Formal apparatus description
  - – Provenance is necessary but not sufficient
- Standardized sensor classes
  - – More sensors, better resolution
  - – Tie-ins with monitoring systems, etc.
- Repository of Experiments
  - – Sharing of DAX is ad-hoc for now

INFORMATION SCIENCES INSTITUTE
. . . agent of innovation . . .

USC Viterbi
School of Engineering

Next Steps

- Design repository for experiments
  - Attempt to make it useful to all FG EM efforts
- Improve capture of apparatus description,
- Improve capture of sensor data
  - Work with FG-Performance group
  - Might be useful beyond FG EM
  - Big disk AMQP sink for multiple sensor streams
    - Can be used to create repeatable experiments differently

Is there something you, the audience, would like to see for FG Experiment Management?