Performing large-scale seismic hazard analysis using Pegasus workflows

Scott Callaghan
Computer Scientist
Southern California Earthquake Center

February 25, 2021
Seismic Hazard Analysis

• What will peak earthquake shaking be over the next 50 years?

• Useful information for:
  • Building engineers
  • Disaster planners
  • Insurance agencies

• Estimates produced by
  1. Assembling a list of earthquakes
  2. Determining how much shaking each event causes
  3. Combining the shaking levels with probabilities

Two-percent probability of exceedance in 50 years map of peak ground acceleration
CyberShake Project

• Developed by the Southern California Earthquake Center (SCEC)

• For each site of interest:
  • Simulate each of 500,000 earthquakes
  • Determine maximum shaking from each
  • Combine with probabilities to produce curve

• Repeat process for multiple locations
### CyberShake Computational Requirements

<table>
<thead>
<tr>
<th>Simulation</th>
<th>CPU compute hours</th>
<th>GPU compute hours</th>
<th>Output data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 location</td>
<td>64,700</td>
<td>2,500</td>
<td>539 GB</td>
</tr>
<tr>
<td>Regional study</td>
<td>56,000,000</td>
<td>2,100,000</td>
<td>457 TB</td>
</tr>
</tbody>
</table>

- High degree of automation required for around-the-clock execution
  - Rely heavily on scientific workflow tools: Pegasus-WMS and HTCondor
- Typically target large NSF and DOE-funded supercomputers
  - Workflows orchestrated from SCEC server at USC
- Challenges associated with this scale
  - Remote job submission
  - High-throughput tasks
CyberShake requires execution of thousands of remote jobs

**Push-based**
- When tasks are ready to run, send them to resources
- SSH: keys must be accepted on remote system
- rvGAHP: daemon on remote system connects to workflow submit host
  - Can be used on systems with two-factor authentication

**Pull-based**
- Uses “pilot jobs” or HTCondor glideins
- Acquire resources first, then look for tasks
- Results in additional overhead
- Can bundle jobs
High-Throughput Tasks

- Added new capability to CyberShake to calculate higher frequency results
  - Makes results more useful for building engineers
- Requires execution of 75,000 additional tasks
  - Serial
  - Short duration (2 sec – 30 min)
- Can’t submit these tasks directly to the scheduler
- Use Pegasus-mpi-cluster (PMC)
PMC

• MPI wrapper around serial or thread-parallel tasks
  • Master-worker paradigm
  • Preserves dependencies

• Simple for workflow user
  • Job starts up on cluster, starts PMC
  • Specify tasks as usual, Pegasus does wrapping

• Uses intelligent scheduling
  • Core counts
  • Memory requirements

• Writes rescue file
CyberShake Study Metrics

- Study conducted over 128 days
- Consumed 6.2 million node-hours
  (120M core-hours/13,650 core-years)
  - Averaged 2,018 nodes
  - Max of 16,219 nodes (~280,000 cores)
- Ran 39,285 Pegasus jobs across 3 systems
- Pegasus managed 1.2 PB of data
  - 157 TB of data transferred by pegasus-transfer
  - 14.4 TB of final data products staged to USC storage
- Our workflow software stack scales!
Future Directions

• New CyberShake hazard results with higher-frequency codes
  • Planning regional study later this year to rely heavily on PMC

• Integrating improved physics
  • New workflow jobs
  • Larger data management requirements

• Coscheduling of CPU and GPU jobs on same nodes
  • Improves resource utilization
  • Advertise different types of slots using glideins

• Target new systems
  • Run on 12 different clusters in 14 years
Our Pegasus Feedback

• In general, we would benefit from tools which allow us to manage multiple workflows.
• In CyberShake, the unit of work is hazard calculations for a single site, but this requires the execution of multiple workflows. We would benefit from additional tools to manage the campaign.
• When running across multiple systems, we face challenges load-balancing, especially for bottleneck stages like database writes. We would benefit from ways to control the number of jobs at various stages across workflows.
Thank You!