

# Preparing Seismic Applications for Exascale using Scientific Workflows

Scott Callaghan, Philip J. Maechling, Karan Vahi, Ewa Deelman, Fabio Silva, Kevin R. Milner, Kim B. Olsen, Robert W. Graves, Thomas H. Jordan, and Yehuda Ben-Zion

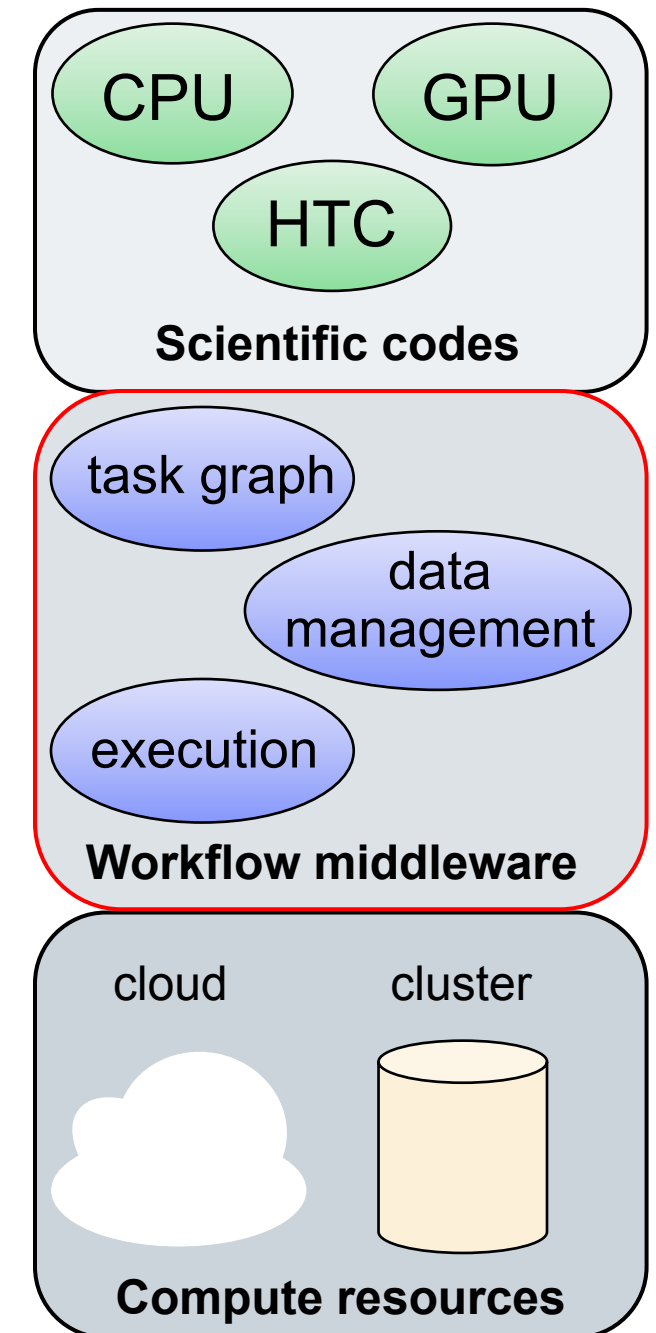
Wednesday, May 24, 2023  
GC11 – Solid Earth

# *Outline*

- What is a workflow?
- What are key workflow concepts?
- How do workflows benefit real-world seismic applications?
- What are the community challenges as we look ahead to exascale?
- Before we get started –
  - Who here uses a released workflow tool?
  - Who here has written their own workflow tool?

# *What is a workflow?*

- Series of computational tasks with dependencies between them
  - Short, long, serial, parallel, ...
- Capture executables, parameters, input, output
- Workflow process and data are usually independent
  - Can rerun same workflow on different data
  - System-independent: can run same workflow on different systems
- Workflow logic independent from scientific codes



# *Workflow Shared Concepts*

- Many workflow tools, but common concepts between them
- Representation of workflow tasks and their data
  - Can be specified through API, annotations, GUI
  - Explicit or implicit data roles
- Workflow prepared to run on certain hardware
- Schedule and run the workflow, honoring dependencies
  - May include remote job submission and data transfer
  - Some support interactivity and notebooks
- User monitors workflow execution
  - May include error handling and retry provisions

# *Why Use Scientific Workflows?*

- Automated management of task execution
- Support for distributed execution
- Data management
- Metadata management
- Error recovery
- Portable description of pipeline

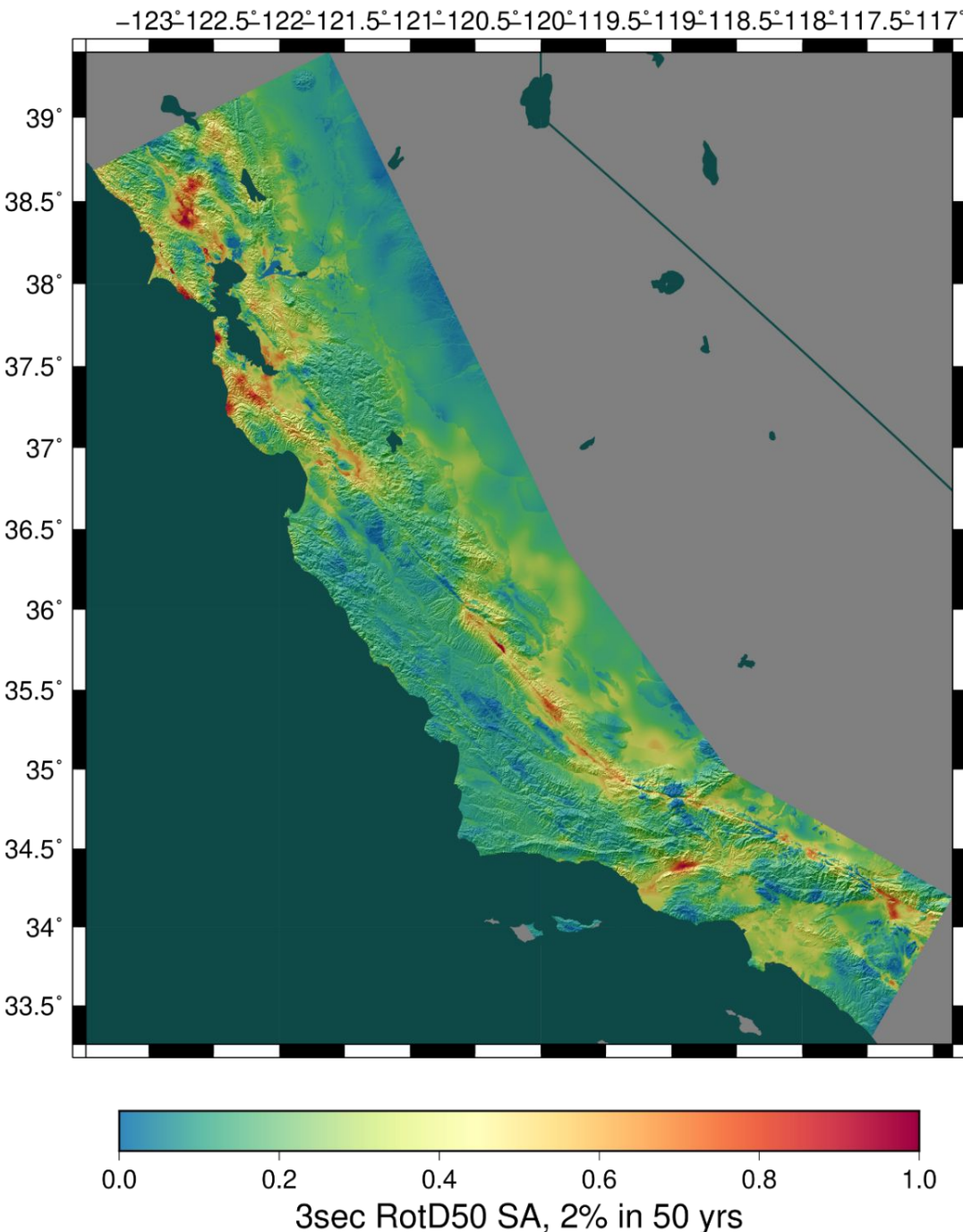
# NAS report

- US National Academy of Sciences commissioned a report on automated research workflows (ARWs) in 2020
- “The common goal of researchers implementing ARWs is to **accelerate scientific knowledge generation, potentially by orders of magnitude, while achieving greater control and reproducibility in the scientific process.**”
- “The tools and techniques being developed under the large umbrella of ARWs promise to transform the centuries-old serial method of research investigation... Simultaneously, ARWs provide a way to satisfy pressing demands across fields to increase interoperability, reproducibility, replicability, and trustworthiness by **better tracking results, recording data, establishing provenance, and creating more consistent metadata than even the most dedicated researchers can provide themselves.**”





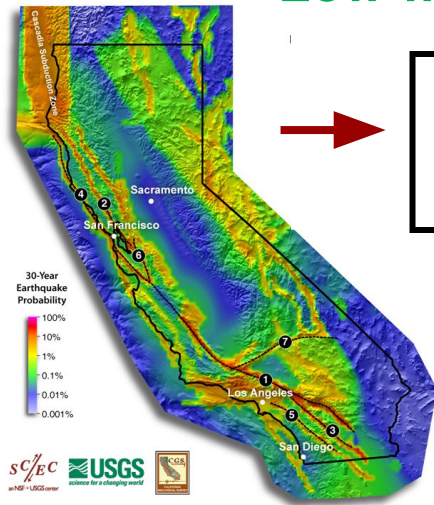
# *Real-World Scientific Workflows*



- Southern California Earthquake Center's CyberShake platform as example of scientific workflows in action
- 3D physics-based probabilistic seismic hazard analysis platform
- Simulation-based alternative to empirical ground motion models (GMMs)
- Reciprocity used to reduce computational cost
  - 670 instead of 720,000 regional wave propagation sims
- Approach used in California and SW Iceland
- Continue to improve models and codes

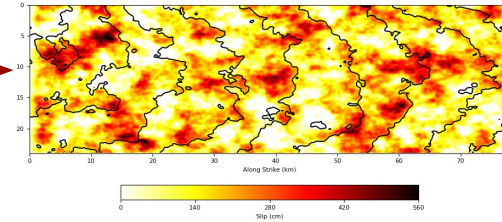
# Broadband CyberShake workflow

## Low-frequency CyberShake workflow



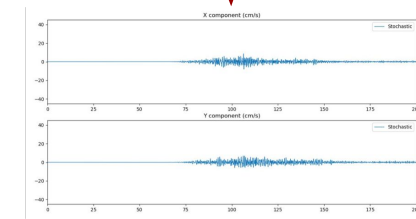
Uniform California Earthquake Rupture Forecast

**Graves & Pitarka kinematic rupture generator**



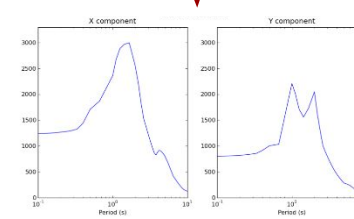
600,000+ events

**High-frequency seismogram synthesis (BBP)**

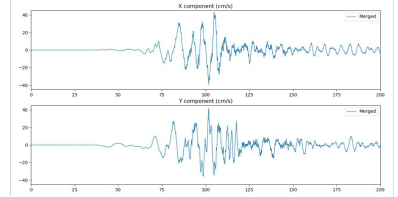


1-50 Hz seismograms

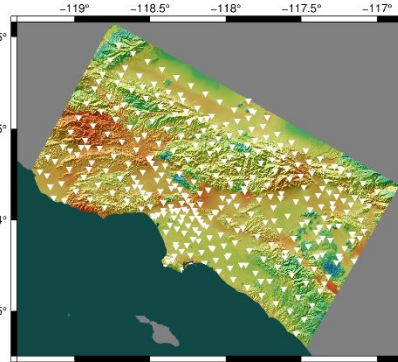
**Merge and combine**



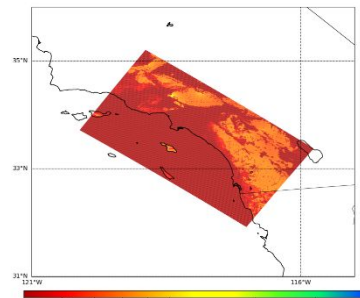
RotD50, PGA, PGV



0-50 Hz broadband data products



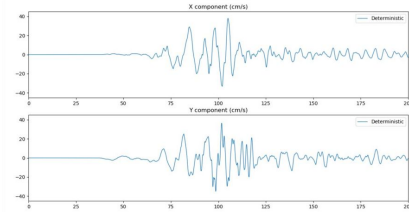
**UCVM**



Velocity Mesh

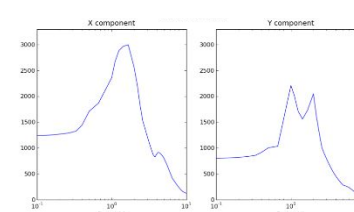
**AWP-ODC-SGT wave propagation**

**Low-frequency seismogram synthesis**



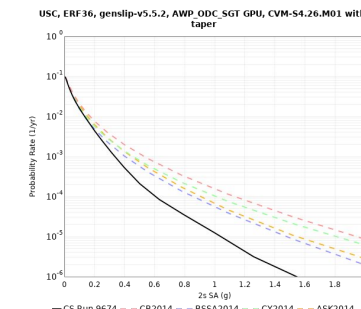
0-1 Hz low-frequency seismograms

**Intensity measures**

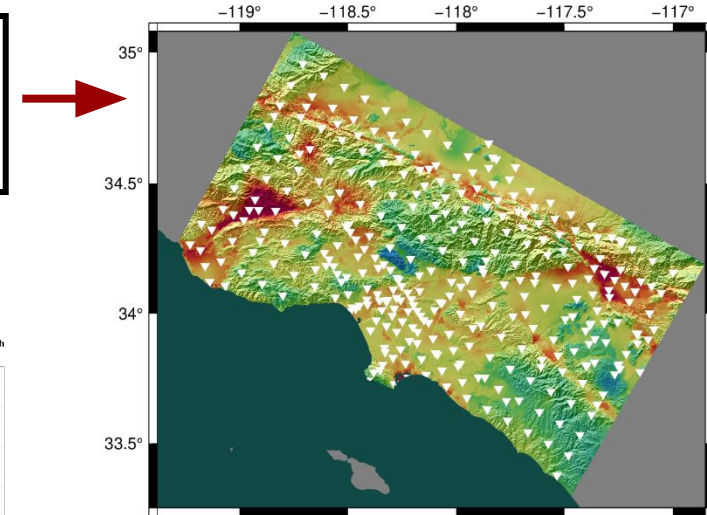


RotD50, PGA, PGV

**Aggregate data products**



Hazard Curve



2sec SA, RotD50, 2% in 50 yr  
**Hazard Map**



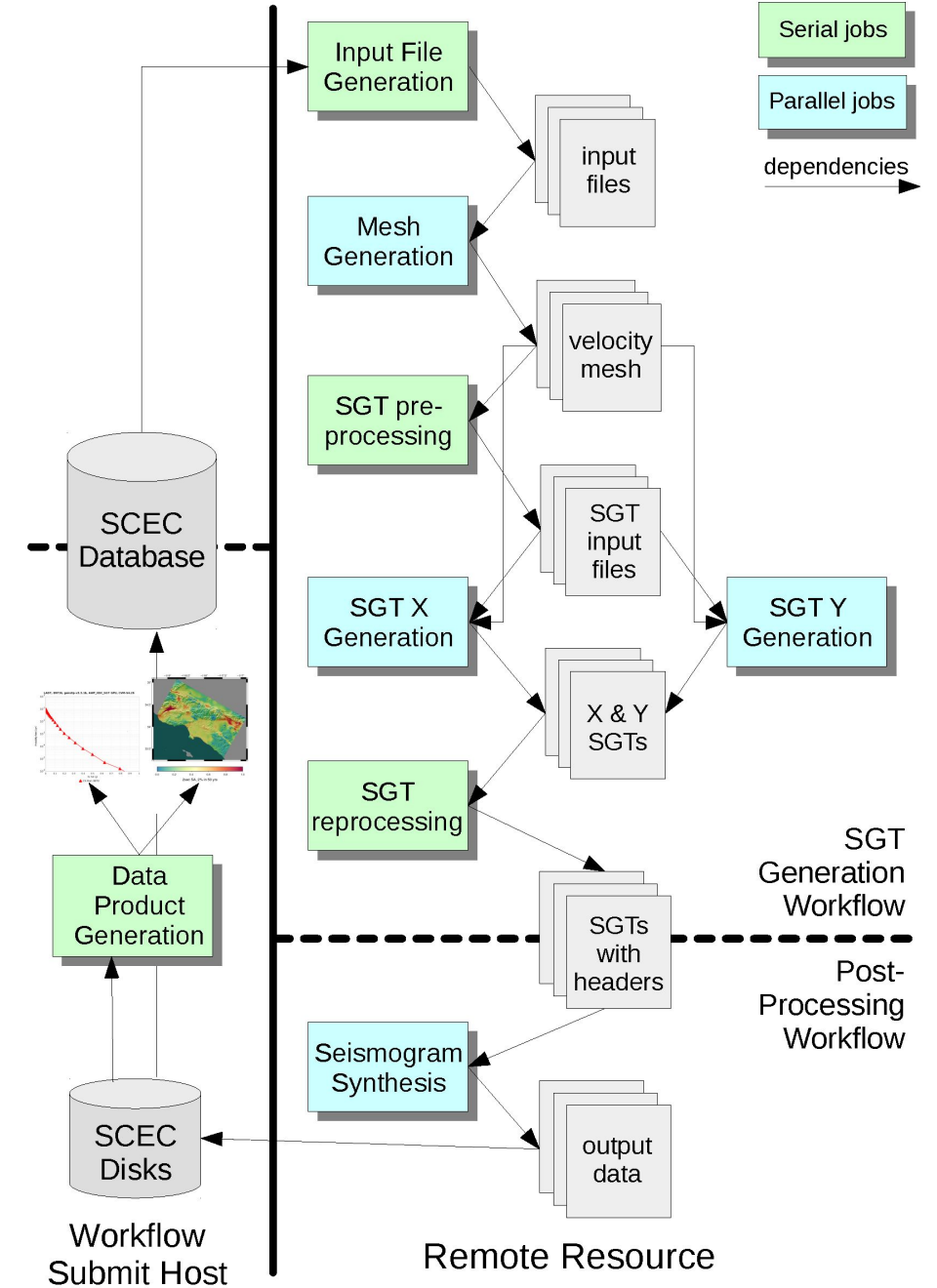
# *CyberShake Computational Requirements*

<b>CyberShake Stage</b>	<b>Number of Tasks</b>	<b>Node-Hours</b>	<b>Output Data</b>
Velocity mesh creation (parallel)	1	10 CPU	300 GB
Wave propagation (parallel)	2	80 GPU	1500 GB
Low-frequency seismogram synthesis (parallel)	1	1000 CPU	38 GB
High-frequency seismogram synthesis (serial)	77,000	1000 CPU	187 GB
Total, 1 site (including small jobs)	77,020	2090	2025 GB
<b>Total, full region</b>	<b>25.8 million</b>	<b>700,000</b>	<b>680 TB</b>

- Large computational and data requirements
- Mix of large parallel CPU and GPU jobs with HTC
- High degree of automation required to support continuous execution

# CyberShake Workflow Framework

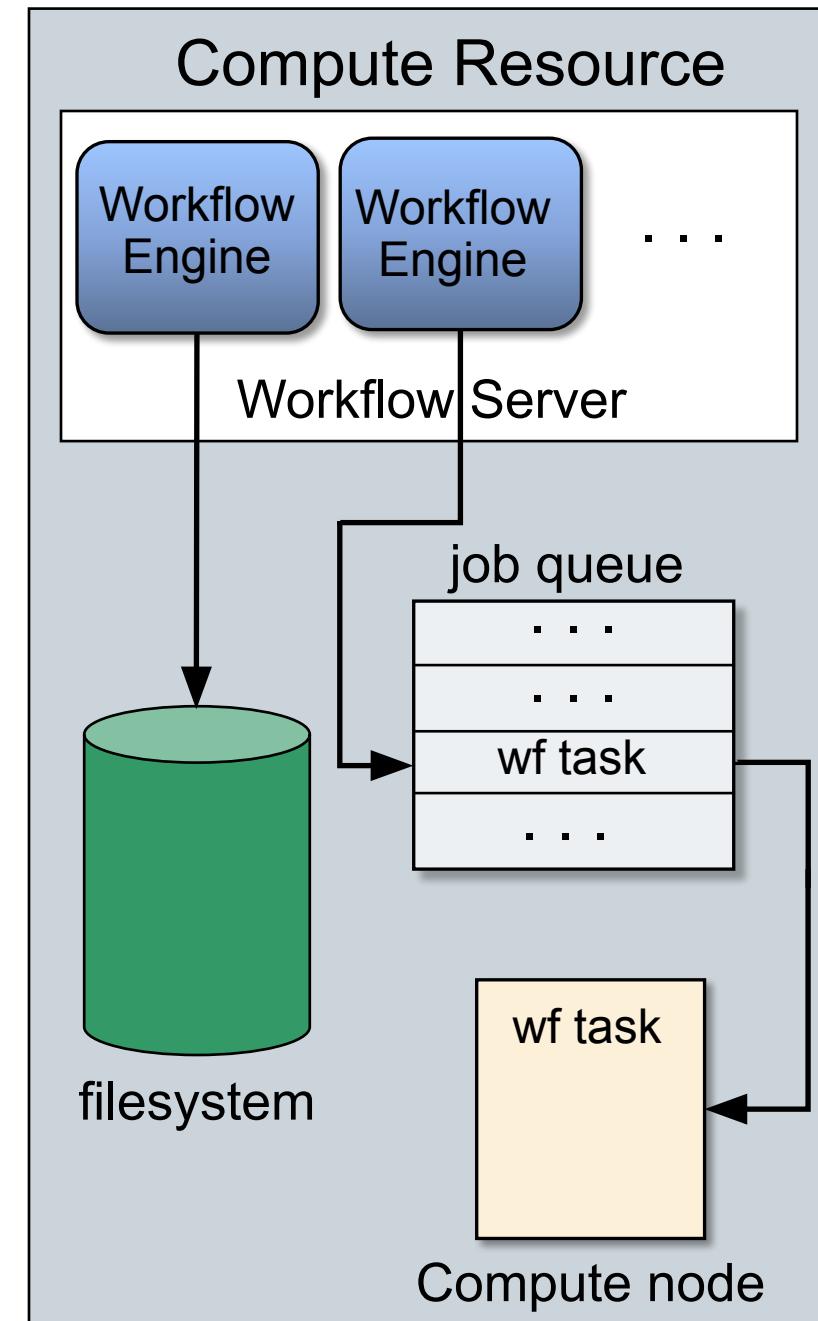
- Pegasus-WMS
  - Use API to create description of workflow
    - Tasks with dependencies
    - Input/output files
  - Plans workflow for execution on specified systems
  - Adds jobs to manage data
  - Wraps executables to track runtime metadata
- HTCondor
  - Manages real-time execution of jobs
  - Submits jobs to remote systems, checks on success
  - Monitors dependencies
  - Checkpoints workflow
- Globus used to transfer data



Schematic of CyberShake low-frequency workflow

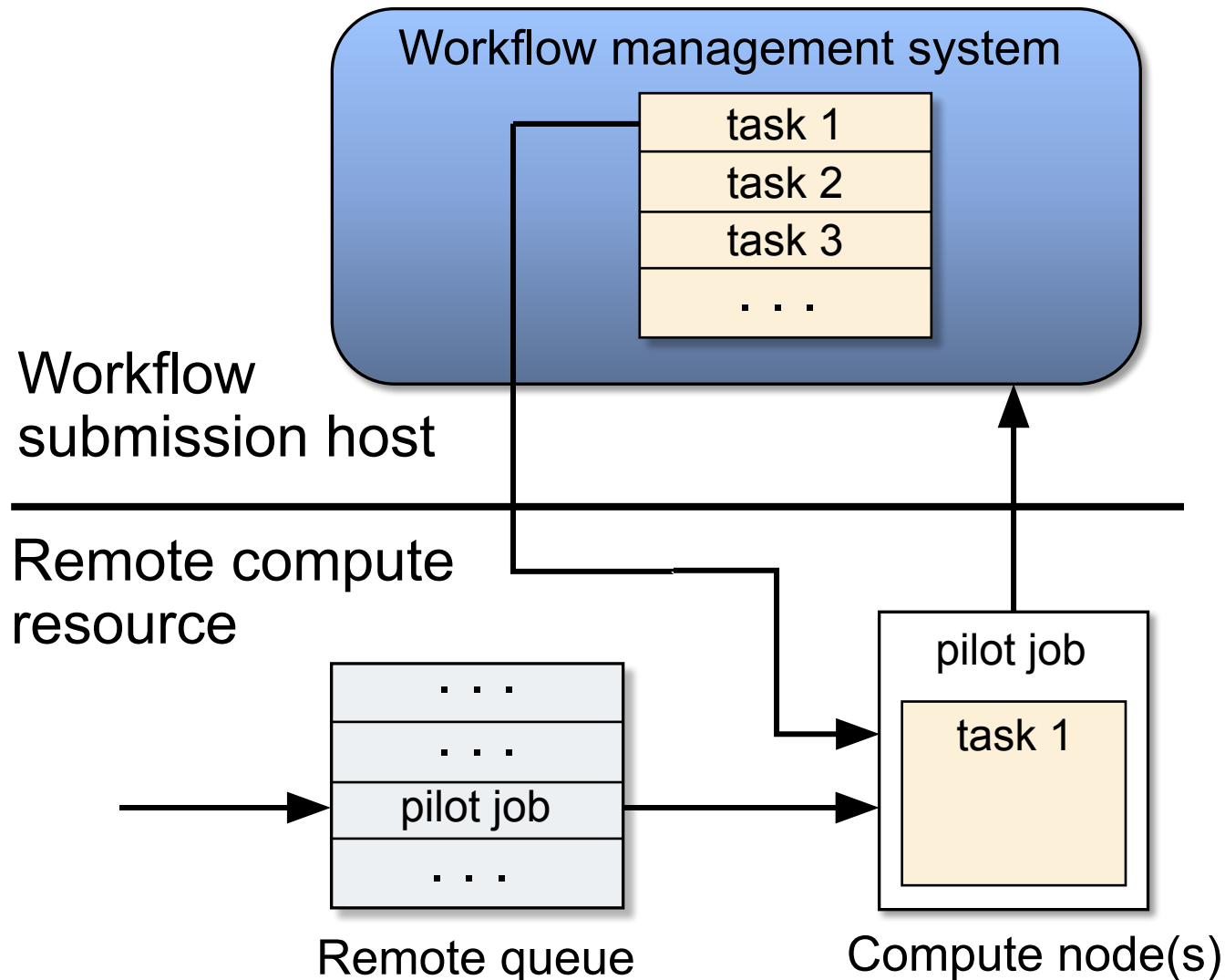
# *CyberShake Challenges: Automated Job Submission*

- CyberShake studies take months to run
  - Automated job submission is a must!
- Many HPC systems require two-factor authentication
  - Manual token entry conflicts with automated job submission
- Could orchestrate workflows from cluster
  - Limited support for distributed execution
  - Restricted to center-supported workflow systems
- We preferred solutions with independent workflow submission host

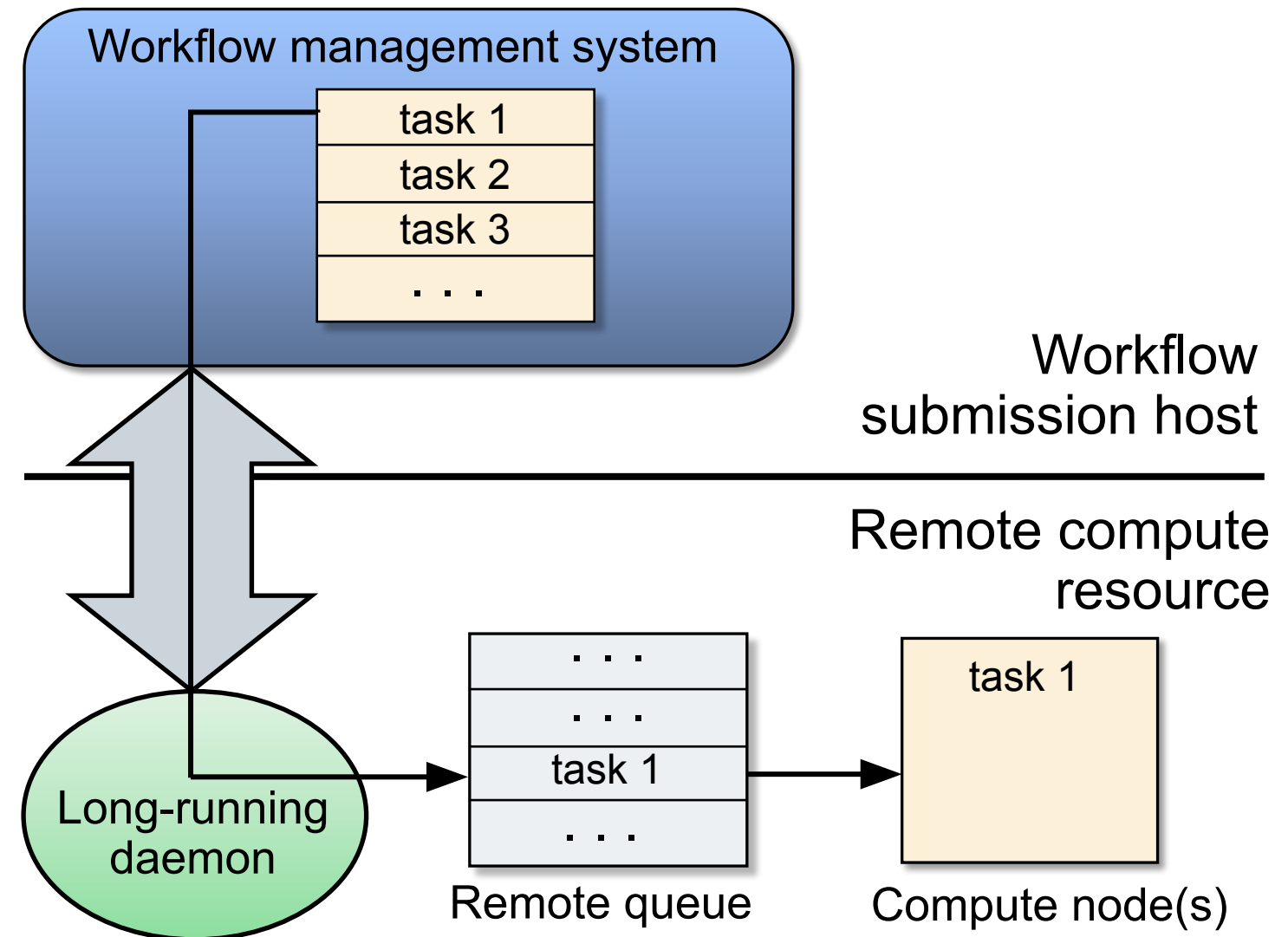


# Job Submission Solutions

- Pull-based: get resources first, then 'pull' work onto them



- Push-based: jobs sent when ready
  - Daemon required to set up connection





# CyberShake Challenges: Job Execution

- Workflows include heterogeneous jobs

- Serial, parallel, CPU, GPU

Summit Scheduling Policy

- Targeted OLCF *Summit* (#5)

- *Summit* prioritizes large jobs
- Want to run in bins 1 or 2
- Largest tasks in the workflow are ~1.5% of the system

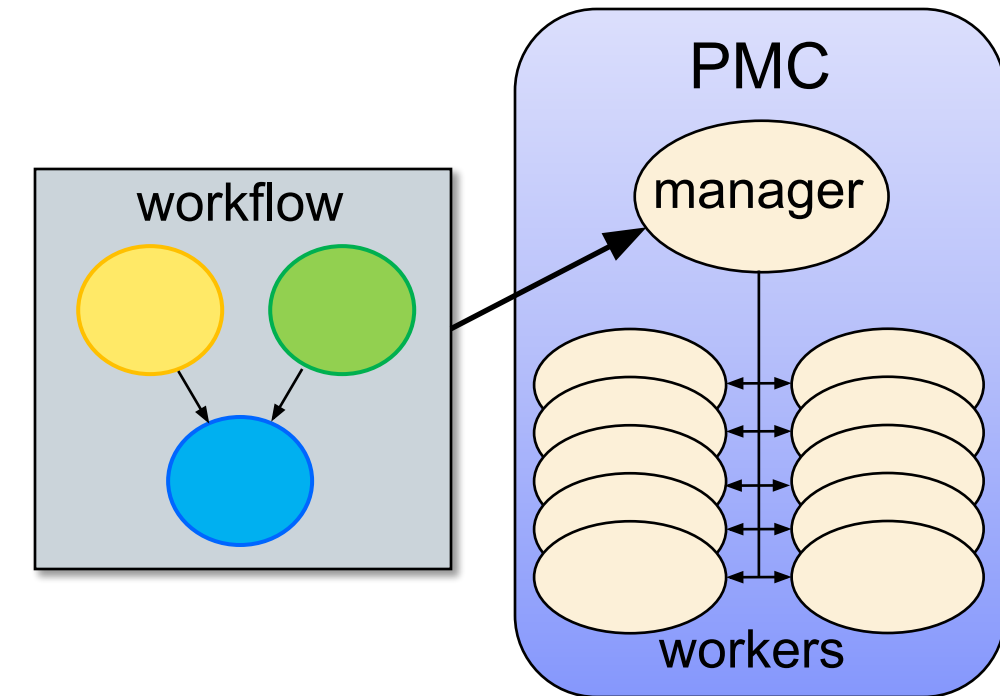
Bin	Min Nodes	Max Nodes	Aging Boost
1	2765 (50%)	4608	15 days
2	922 (20%)	2764	10 days
3	92 (2%)	921	0
4	46 (1%)	91	0
5	1	45	0

- Use pull-based approach

- Create large (~1000 node) pilot jobs, then fill with tasks
- Created process to monitor task queue on workflow submission host and submit pilot jobs when enough tasks were ready

# CyberShake Challenges: Job Execution

- High throughput needed for small serial tasks
  - 77,000 tasks per site for broadband calculations
- Can't place jobs directly in the *Summit* queue
  - Schedulers aren't designed for this kind of load
  - Scheduler cycle is ~5 minutes
  - Policy limits number of jobs in queue
- Bundle high throughput jobs using Pegasus-mpi-cluster (PMC)
  - MPI wrapper around tasks
  - Uses manager-worker paradigm to execute tasks
  - Preserves dependencies
- Push-based approach for PMC jobs

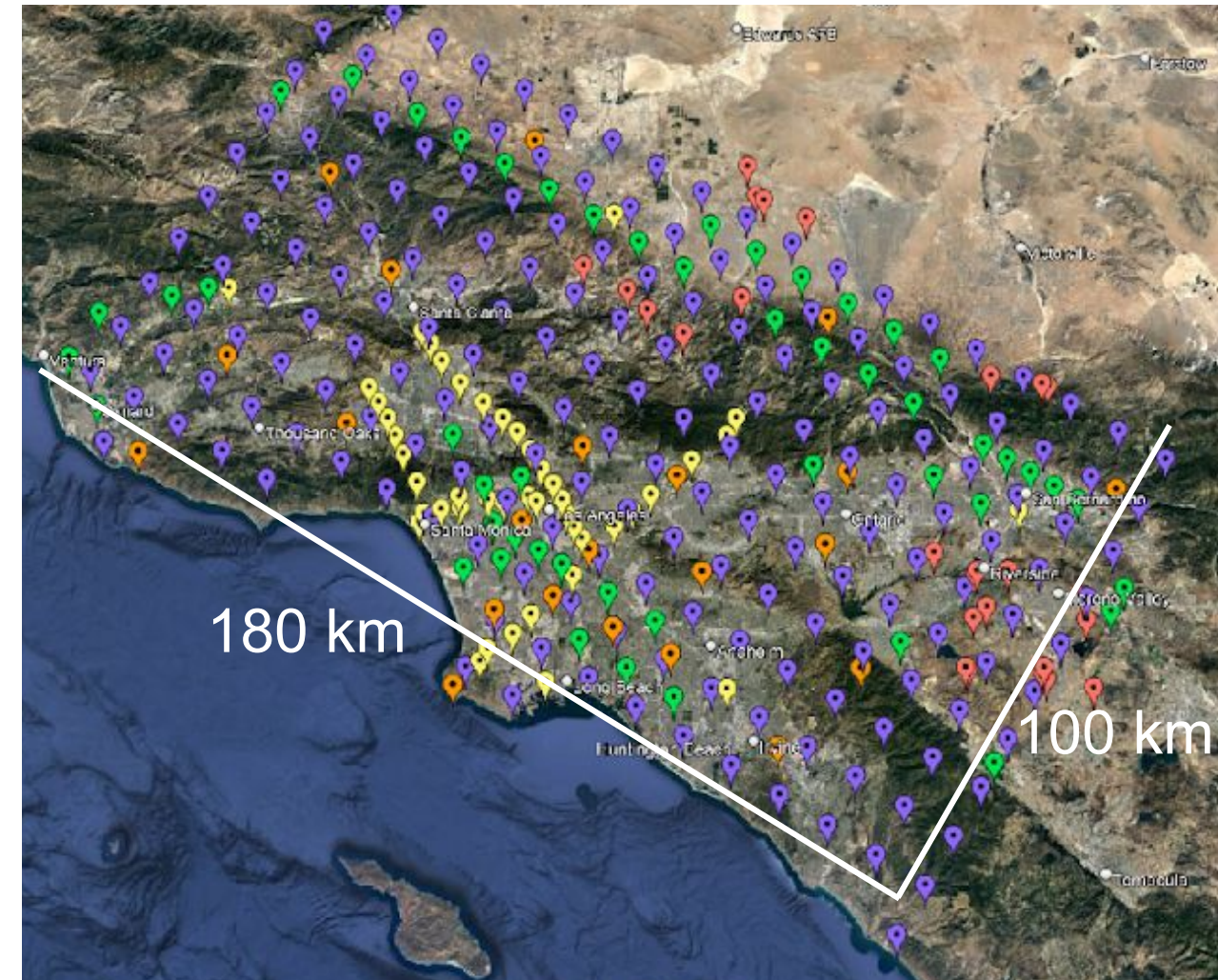


# *CyberShake Challenges: Data Management*

- Millions of data files
  - Workflows stage files needed for executables
    - Supports running distributed workflows
  - Output data staged back to archival storage
- Data integrity
  - Automated checks to detect file errors early
    - Correct number of files, correct size, NaNs present, zero-value checks, MD5 sums
  - Included as new jobs in workflow
- Preparing output data for easy access later
  - Work in progress

# Study 22.12

- CyberShake study for 335 sites in Southern California
- Both low-frequency (0-1 Hz) and broadband (0-50 Hz) hazard models
  - Broadband approach validated against historic earthquakes
- Updated rupture generation process
  - Reduced slip/risetime correlation
  - Reduced shallow rupture speeds
  - Increased hypocentral density from 4.5 → 4 km
- Improved near-surface 3D velocity model
  - Added  $V_{s30}$ -derived merged taper to create more realistic velocity profiles

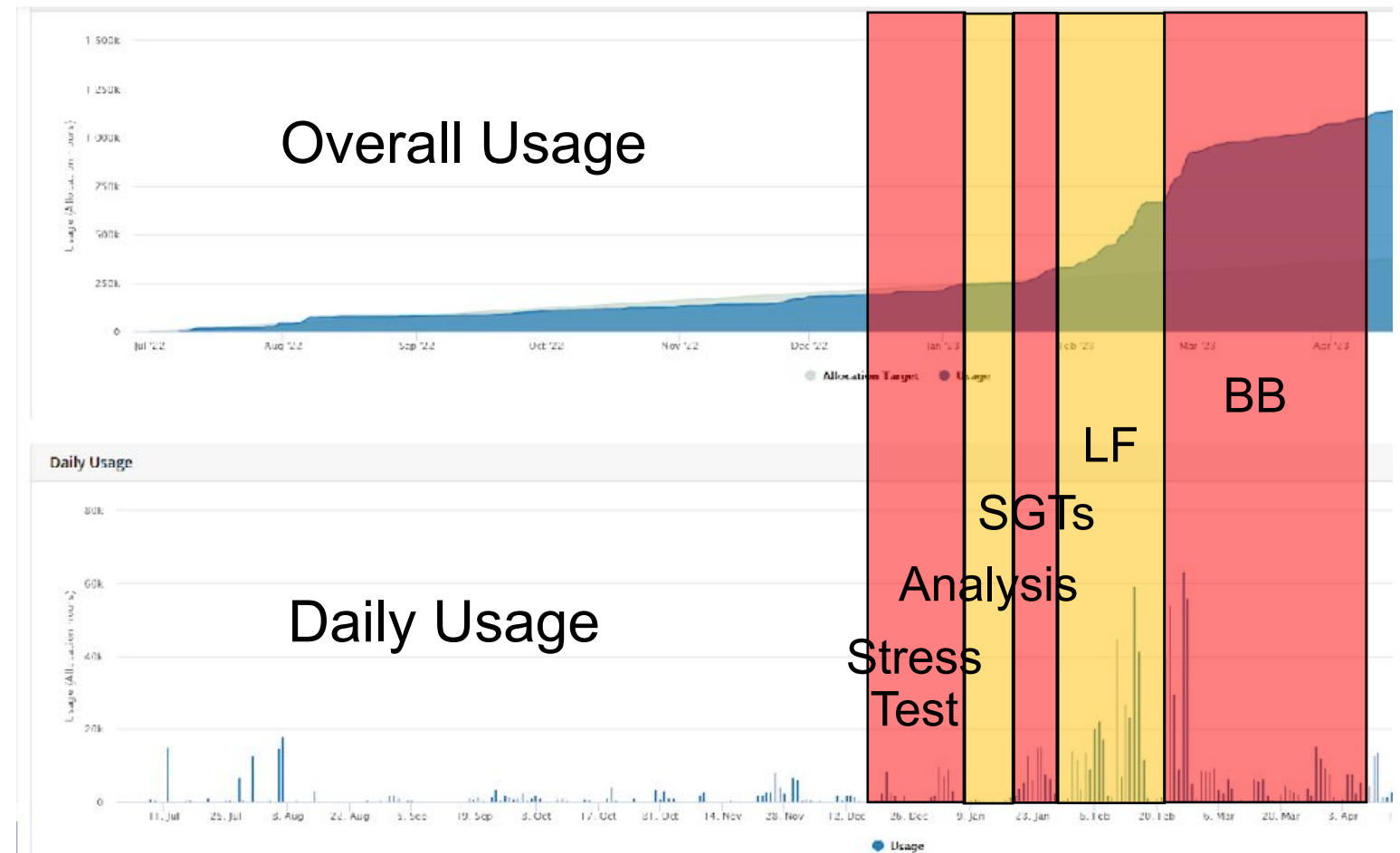


Study 22.12 site map



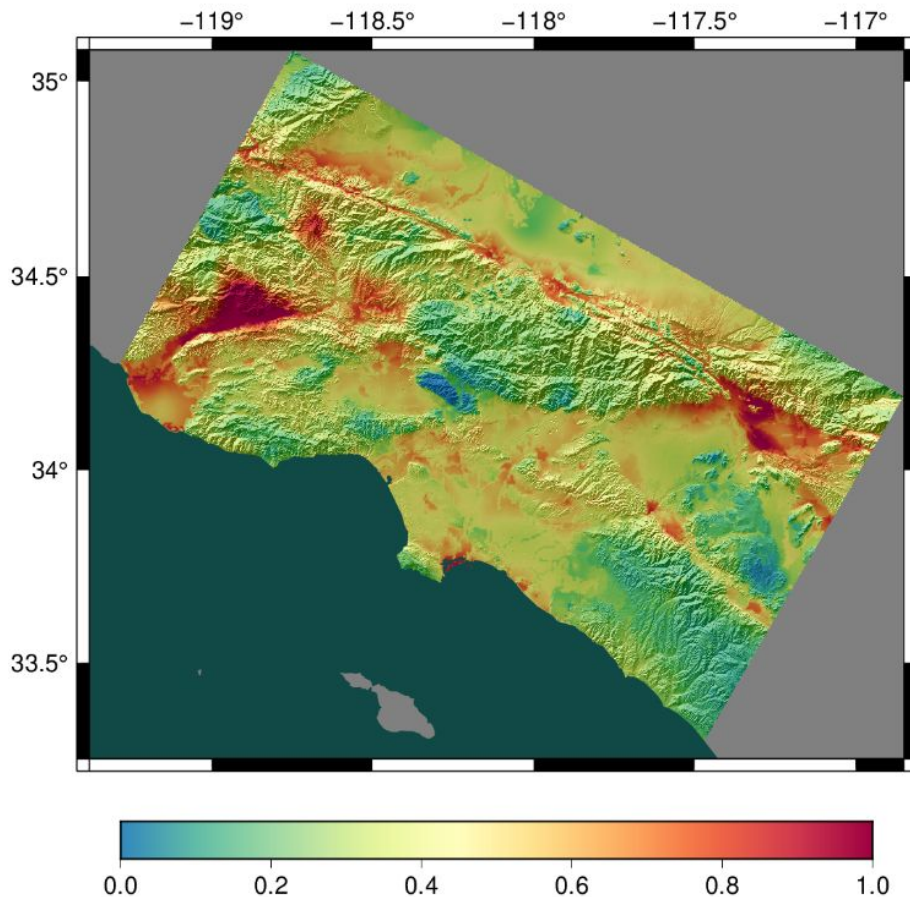
# Study 22.12 Statistics

- Makespan of 108 days
- Used 772,000 node-hours on OLCF *Summit*
  - Averaged 442 nodes
  - Max of 3382 (73% of *Summit*, ~17x MareNostrum 4, ~50% MareNostrum V)
- Workflow tools managed:
  - 28,120 tasks (11,431 remote)
  - ~2.5 PB total data
  - 74 TB / 19M files transferred and archived

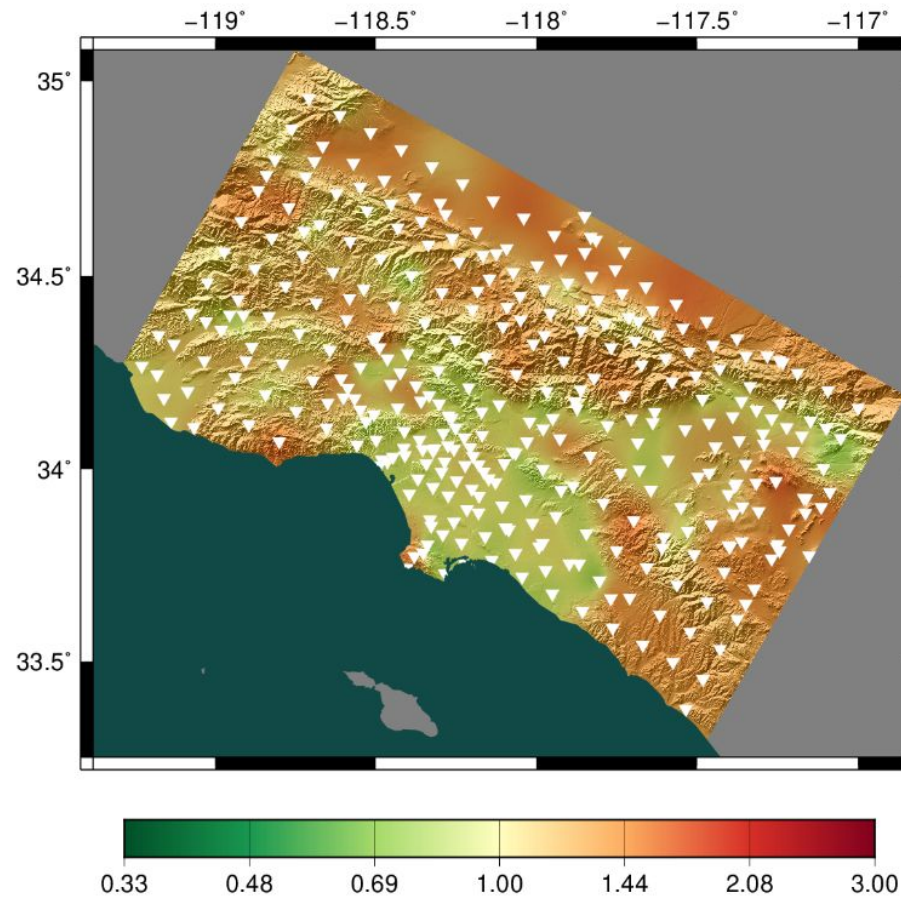


# Study 22.12 Results

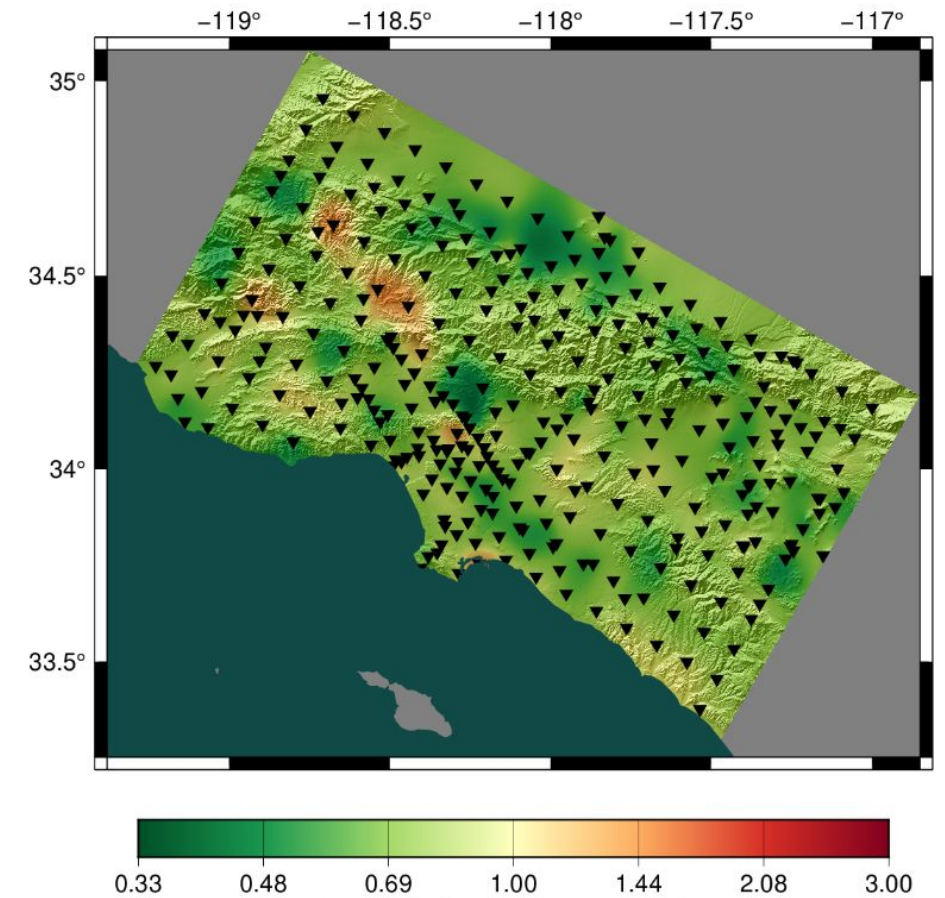
## Study 22.12



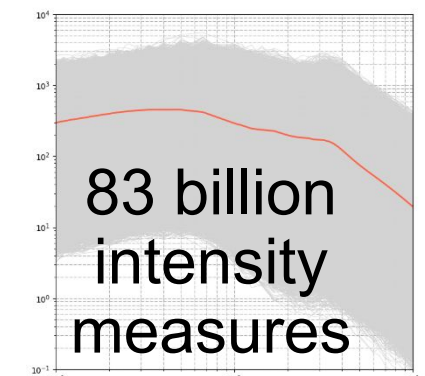
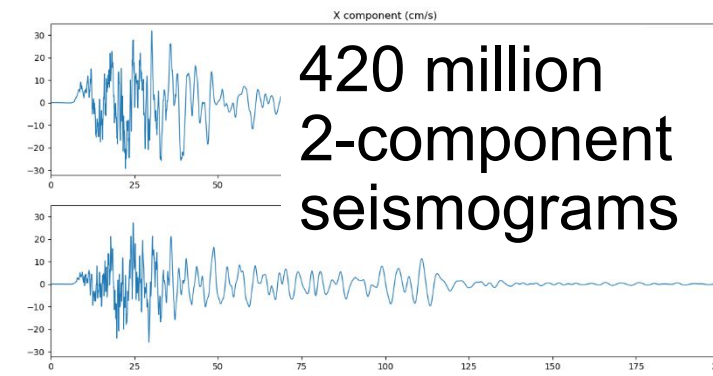
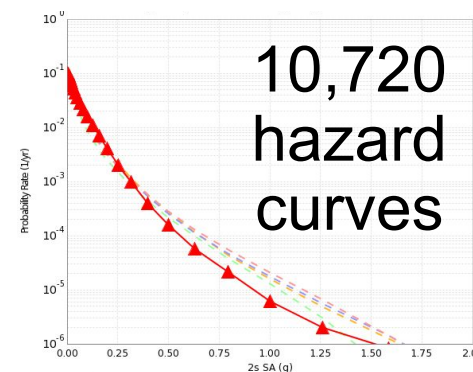
## Ratio, new/old CyberShake



## Ratio, 22.12/GMMs

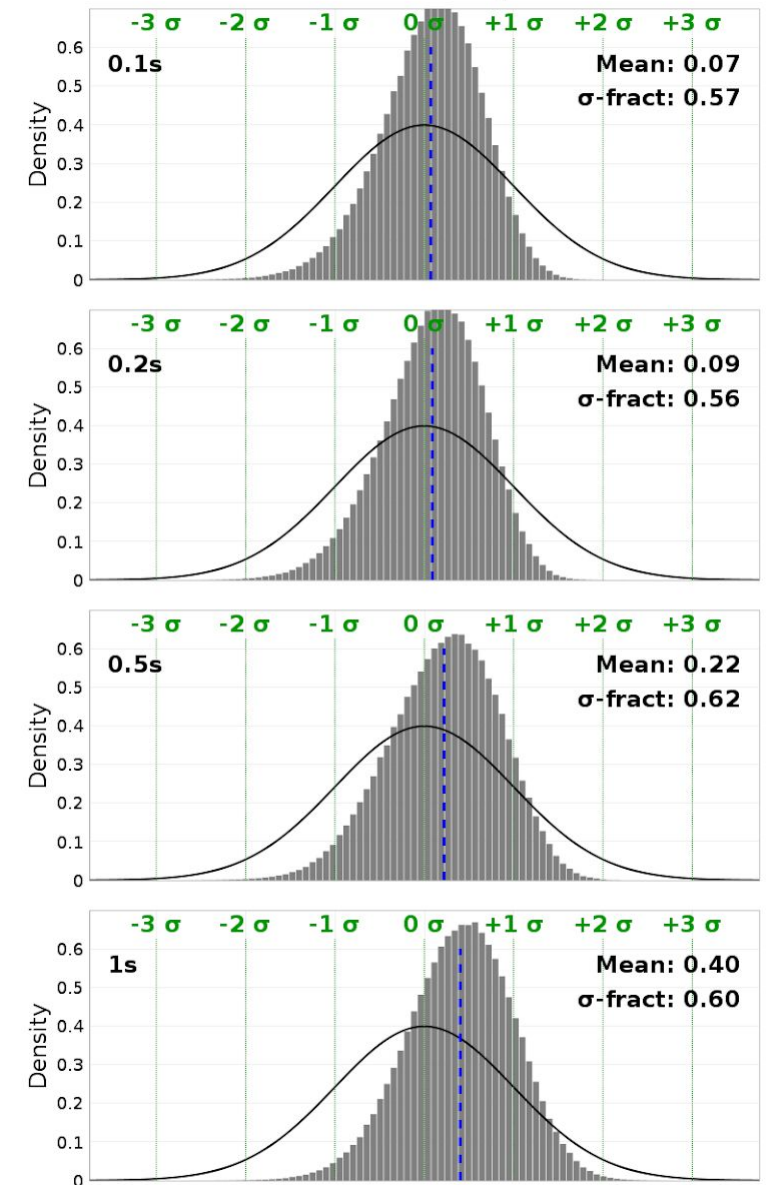


These hazard maps were generated using:



# *Future Science Directions*

- Increase deterministic frequency to 2 Hz
  - Frequency-dependent attenuation
  - Small-scale velocity heterogeneities
- Include nonlinear simulations
  - Reciprocity is by definition linear
  - Identify subset of events for full nonlinear simulations
  - Apply pseudo-nonlinearity to reciprocity results
- Streamline process of integrating new codes
  - Goal is to support multiple codes for each stage
  - Supports improved quantification of uncertainty



Study 22.12 Broadband vs. GMM ASK2014 z-scores



# *Future Computational Challenges*


- Workflows to support hybrid linear/nonlinear approach
  - Must bookkeep and combine site-based and event-based seismograms
- Largest systems are becoming more specialized
  - For example, not feasible to run GPU and CPU jobs on OLCF *Frontier*
  - Will require distributed workflows
- Improved data management and delivery
  - DOIs for data products
  - Continue to develop data access tool for direct data product access
- Support execution by other researchers
  - Containers?



# *Looking Ahead*

- Questions for us to consider as scientific workflow simulations move to exascale systems
- How do we take best advantage of upcoming opportunities?
- Hope to encourage discussion!

# *Utilizing Exascale Systems*



How can applications best utilize exascale machines to perform cutting-edge science?

- Full-system hero runs?
- Ensembles?
- Combination?

# *Application Resiliency*



How should applications and workflows manage component failures?

- Things will break more frequently on exascale systems
- Can we/should we do better than ‘turn it off and turn it back on again’?



# *Exascale Data*



How do we automate data and metadata management to ensure our simulation results are useful to the community?

- During simulations?
- Support for FAIR principles through workflows?
  - Archiving, indexing, DOIs, ...





CWL project lists 334 “existing workflow systems”

# *Workflow Community*



How should the workflow community work towards common goals, given the number of available tools and approaches?

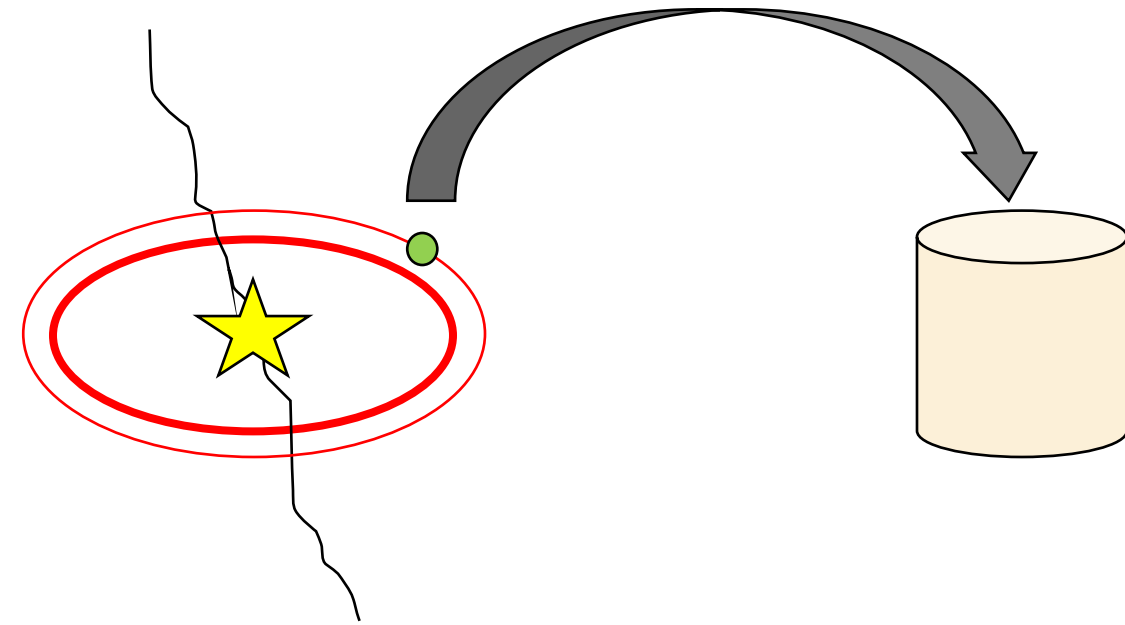
- Umbrella and standards groups?
  - ExaWorks
  - eFlows4HPC
  - Common Workflow Language
- How to reach out to new users with existing tools?



# *Urgent Computing*

Can we use workflow tools to fully automate urgent computing simulations?

- Use sensors to trigger simulations?
- Operational aftershock forecasting
- Simulation-informed ShakeMaps



# *Upcoming HPC Trends*



How can upcoming HPC trends help move our simulations forward?

- Composable computing?
- ML/AI?



# *Reproducibility Crisis*



Are current tools sufficient, or are we still missing key elements to achieve simulation reproducibility?

- How do we improve on the 40% of earth/environmental scientists who can't reproduce their own results? (*Nature*)
- How do we get to meta-FAIR?

# *Software Sustainability*



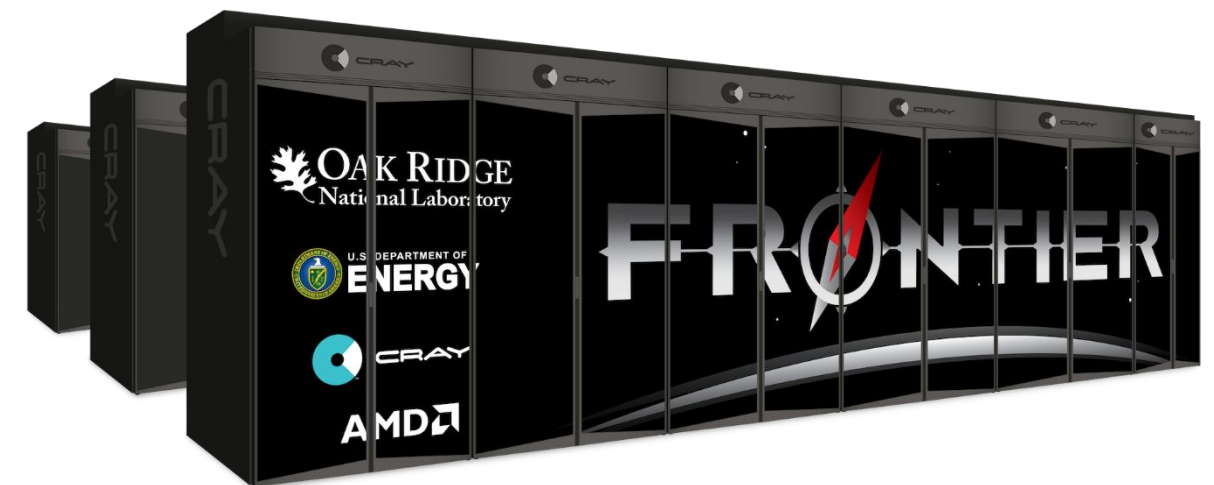
**How do we support the software development needed for exascale computing?**

- Science codes?
- Workflow tools?
- Data management tools?



# *Final Thoughts*

- We live in exciting computational times!
- Excellent opportunity to consider what's next
- Workflow tools can help us navigate exascale challenges
- NAS report: “Realizing the potential of ARWs could accelerate the pace of scientific discovery by orders of magnitude and thereby expand the research enterprise’s contribution to society.”



*Thanks!*

