Use Cases of Computational Reproducibility for Scientific Workflows at Exascale

SC2017, CRE Workshop

Line Pouchard, Carlos Gamboa, Shantenu Jha, Li Tang, Kerstin Kleese Van Dam, BNL

Sterling Baldwin, LLNL

Todd Elsethaggen, Bibi Raju, Eric Stephan, PNNL

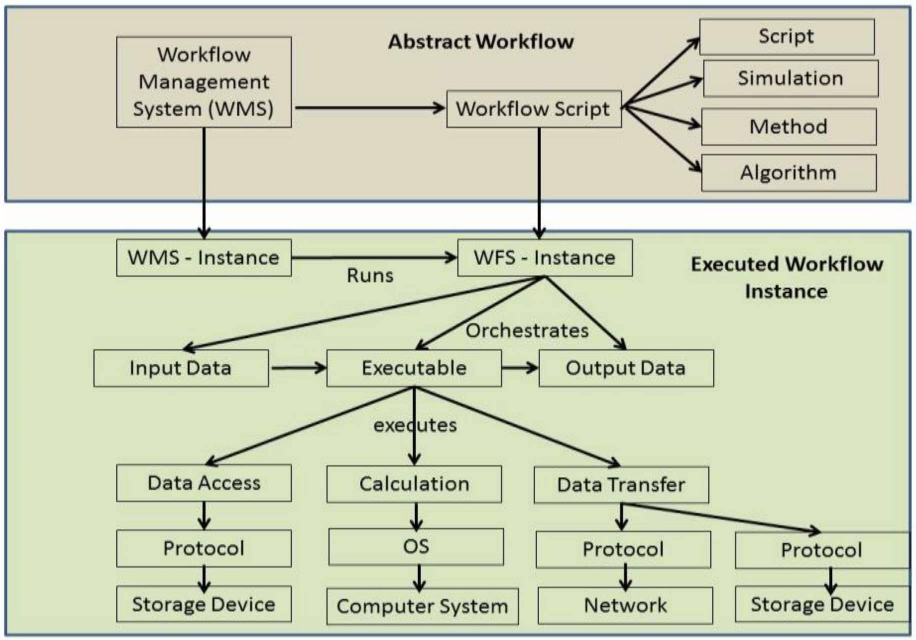
70 YEARS OF DISCOVERY

A CENTURY OF SERVICE

What kind of reproducibility?

- DOE recent guidelines for extreme-scale reproducibility (2014)
  - focus on more consistency in scientific experiments at extreme-scale (with well-understood, community-based methodologies)
  - achieve portable performance across architectures
- Workflow performance reproducibility: or lack thereof
- Bitwise reproducibility: the Accelerated Climate Modeling for Energy (ACME) Use Case

Workflow complexity and ecosystem

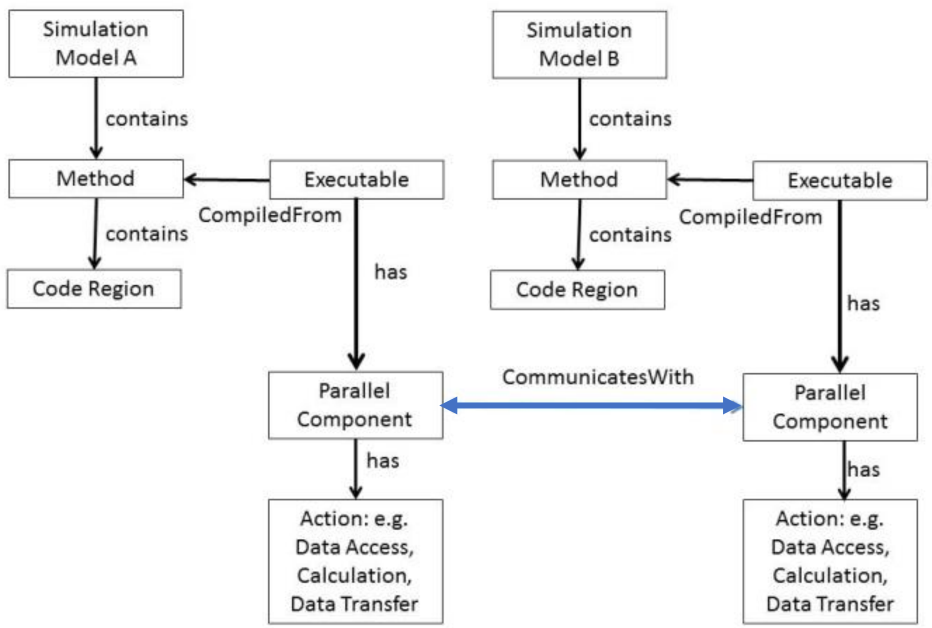


Workflows are complex, utilize shared resources, and mix programming models

Usage of shared resources can introduce further complexities

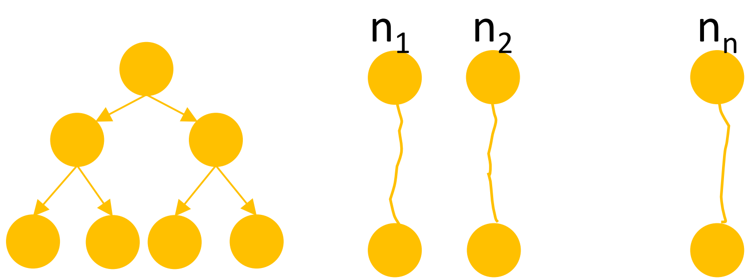
Reference: Kleese van Dam, K., et al., Enabling Structured Exploration of Workflow Performance Variability in Extreme-Scale Environments, Proc. 8th Workshop in Many-Task Computing on Clouds, Grids, and Supercomputers (MTAGS) collocated with SC 2015. DOI: 10.13140/RG.2.1.3311.9127

Interdependencies between parallel components at runtime



Performance of each individual component can affect overall performance of a workflow

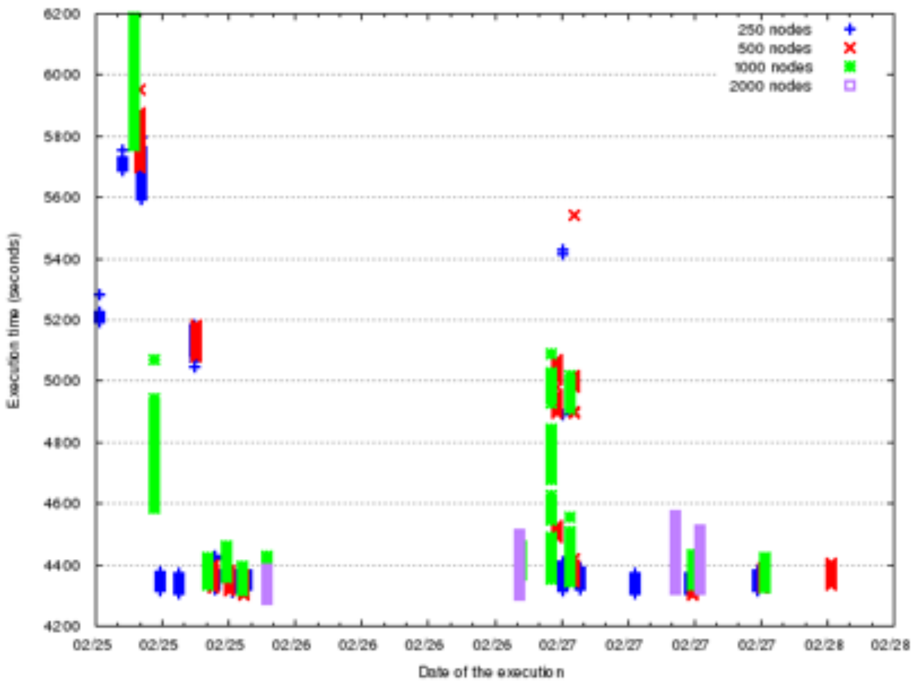
Performance metrics are measurements from workflow and system resources



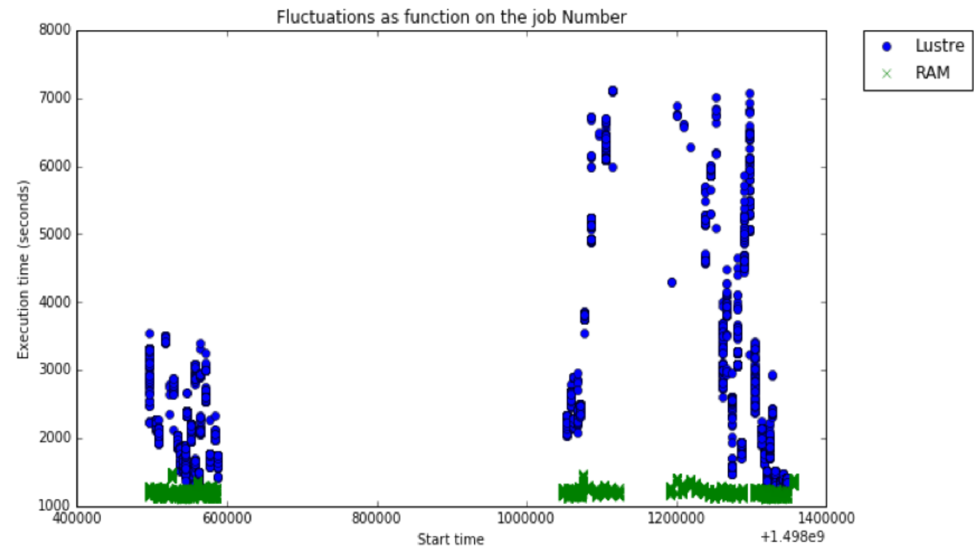
Challenges for the reproducibility of workflows

- Defining a specific reproducibility goal and attainable levels of reproducibility
- Defining the information needed to attain a specific reproducibility goal
- Defining the appropriate level of abstraction for monitoring workflows given:
  - the number of components
  - the complexity of the connections
  - the rate of execution for each component
  - communications

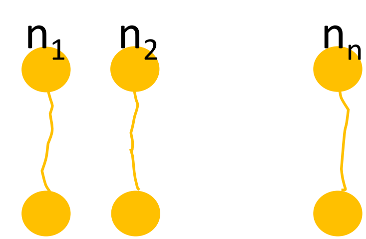
Lack of workflow performance reproducibility



Gromacs (MD)



AthenaMP (HEP)



https://github.com/radical-cybertools/radical.pilot

Approach to enhance reproducibility

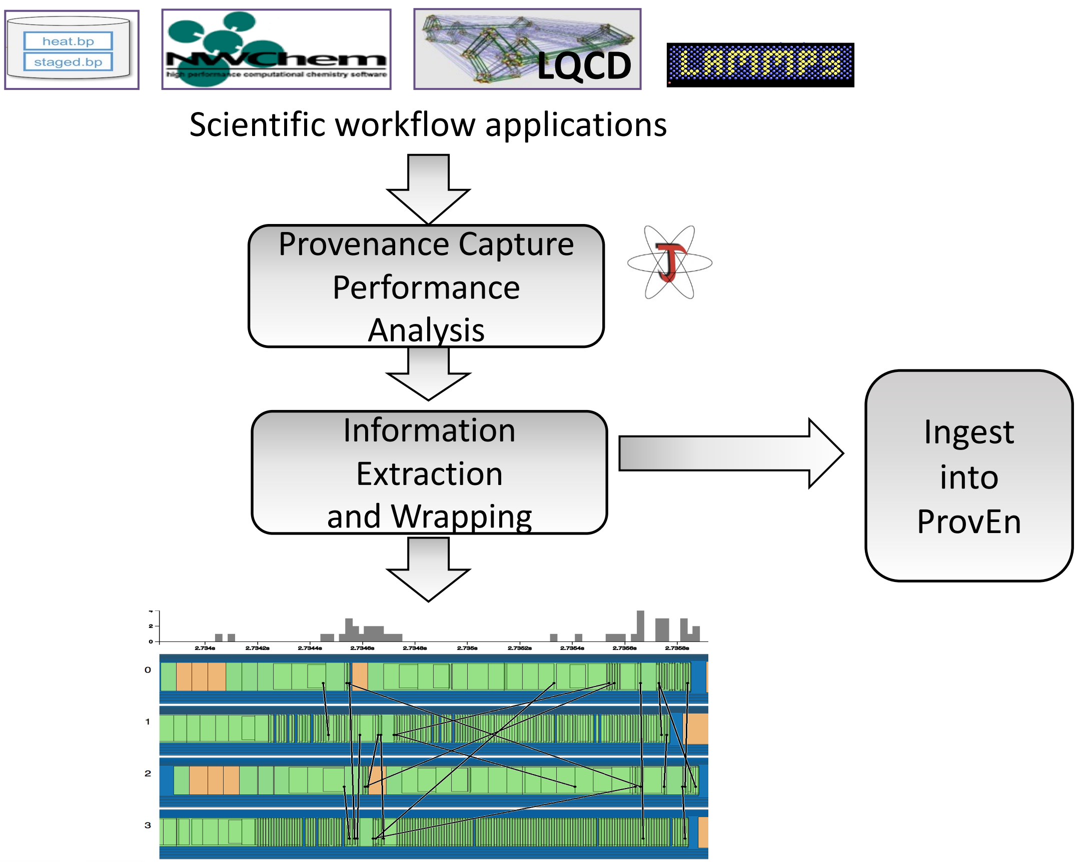
Extract pertinent and sufficient information to enable reproducibility goals

We capture provenance data, performance metrics, system environments, and runtime configurations with the Chimbuko and ProvEn frameworks

Performance and provenance data are extracted, related, and queriable

We add visualization and analytics to help explore areas of interest.

Chimbuko Performance Analysis Framework



A unique capability to capture, analyze and visualize individual workflow performance evolution over time

Systematically captures, stores and correlates provenance and performance metrics of workflows and systems

Ability to plug in various performance tools incl. ECP Tuning and Analysis Utilities

Performance tools extract metrics for single applications, not workflows

https://github.com/CODARcode/Chimbuko-feature-extraction

U.S. DEPARTMENT OF ENERGY | BROOKHAVEN NATIONAL LABORATORY | ECP EXASCALE COMPUTING PROJECT | CODAR | 70 YEARS OF DISCOVERY A CENTURY OF SERVICE

Provenance added to performance metrics

Execution environment

Executable names and version

Standard libraries

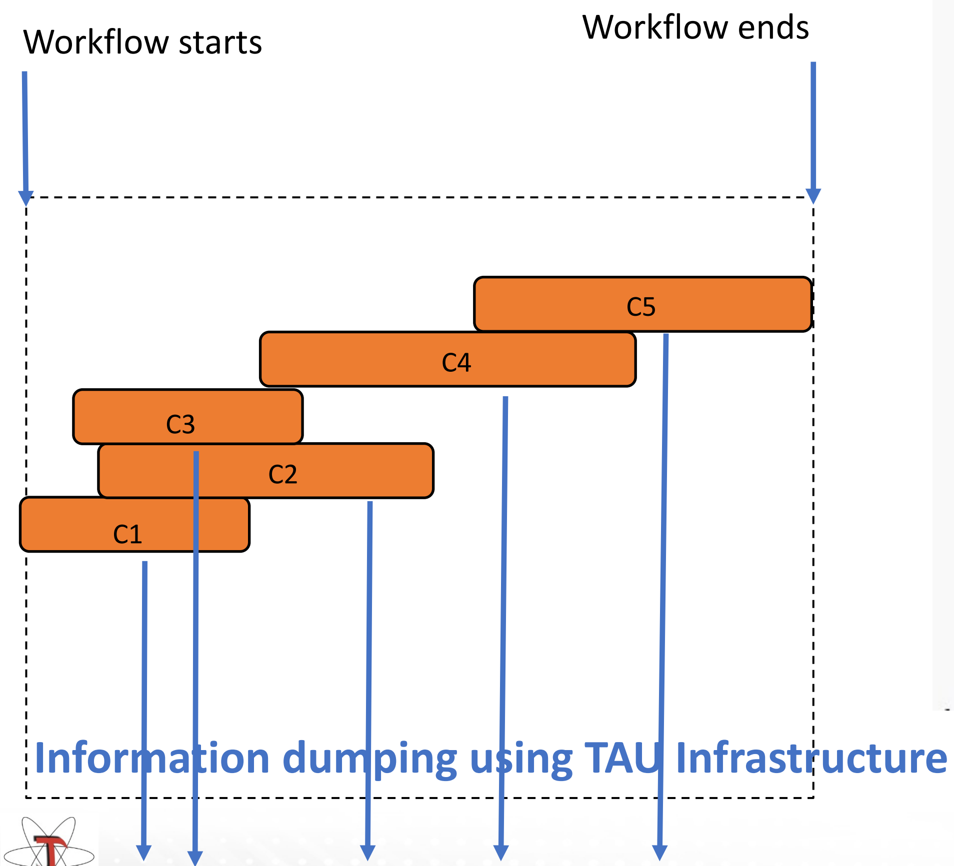

```
"./tau_data": {
  "metadata": {
    "Rank 0, Thread 0": {
      "CPU Cores": "8",
      "CPU MHz": "2593.813",
      "CPU Type": "Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz",
      "CPU Vendor": "GenuineIntel",
      "CWD": "/lfs1/home/pouchard/Example-Heat_Transfer",
      "Cache Size": "20480 KB",
      "Command Line": "./heat_transfer_adios2 heat 4 3 40 50 6 500",
      "Executable": "/lfs1/home/pouchard/Example-Heat_Transfer/heat_transfer_adios2",
      "Hostname": "hpc1.csc.bnl.gov",
      "Local Time": "2017-09-12T16:05:39-04:00",
      "MPI Processor Name": "hpc1.csc.bnl.gov",
      "Memory Size": "65931876 kB",
      "Node Name": "hpc1.csc.bnl.gov",
      "OS Machine": "x86_64",
      "OS Name": "Linux",
      "OS Release": "2.6.32-431.el6.x86_64",
      "OS Version": "#1 SMP Thu Nov 21 13:35:52 CST 2013",
      "Starting Timestamp": "1505246739744458",
      "TAU Architecture": "default",
      "TAU Config": " -adios=/home/pouchard/Install/adios-1.12.0
-pdt=/home/pouchard/Install/pdtoolkit-3.24 -c++=mpicxx -cc=mpicc -fortran=mpif90",
      "TAU Makefile":
"/home/pouchard/Install/tau-2.26.2p1/x86_64/lib/Makefile.tau-mpi-pdt",
      "TAU MetaData Merge Time": "6.2E-05 seconds",
      "TAU Version": "2.26.2p1",
```



```
"Metrics": [
  {
    "id": 1,
    "location-id": 1,
    "measurement": "Time",
    "units": "microseconds",
    "description": "This is a metric used to measure the applications."
  }
]
```


Units of measurement

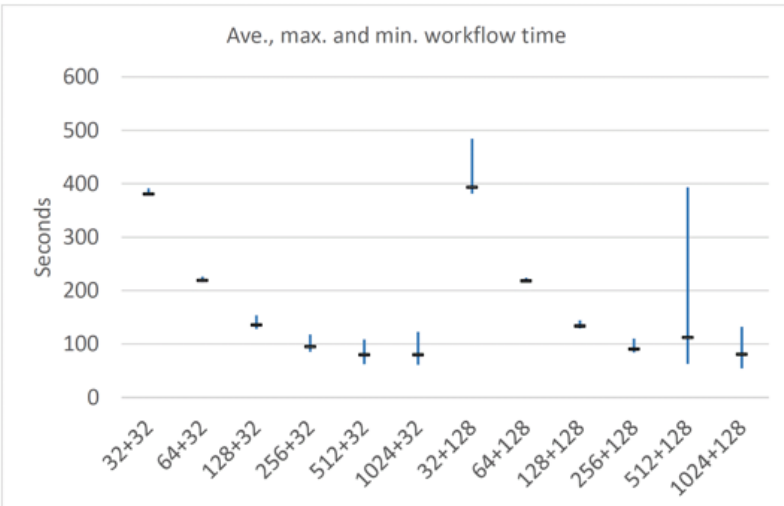
Information Extraction in Chimbuko




```
"Workflow Component": [
  {
    "id": 1,
    "name": "/lfs1/home/pouchard/Example-Heat_Transfer/heat_transfer_adios2",
    "location-id": 1,
    "application-id": 1,
    "start-timestamp": "1505246739744458",
    "end-timestamp": "1505246806066072",
    "Local-Time": "2017-09-12T16:05:39-04:00",
    "Processes": 12,
    "aggr_communication_calls": 204636,
    "aggr_communication_time": 64880892,
    "aggr_adios_time": 333156.58333333331,
    "aggr_execution_time": 1300862,
    "aggr_communication_collective_bytes": 118888.0,
    "aggr_adios_bytes": 2307168.0,
    "total_time": 66321614
  },
  {
    "id": 2,
    "name": "/lfs1/home/pouchard/
    "location-id": 1,
    "application-id": 2,
    "start-timestamp": "1505246739793351",
    "end-timestamp": "1505246806032319",
    "Local-Time": "2017-09-12T16:05:39-04:00",
    "Processes": 3,
    "aggr_communication_calls": 446,
    "aggr_communication_time": 8751554,
    "aggr_adios_time": 57819702.333333336,
    "aggr_execution_time": 437339,
    "aggr_communication_collective_bytes": 117172.0,
    "aggr_adios_bytes": 2304264.0,
    "total_time": 66238968
  }
```


Additional Features

Workflow performance fluctuations in classical MD

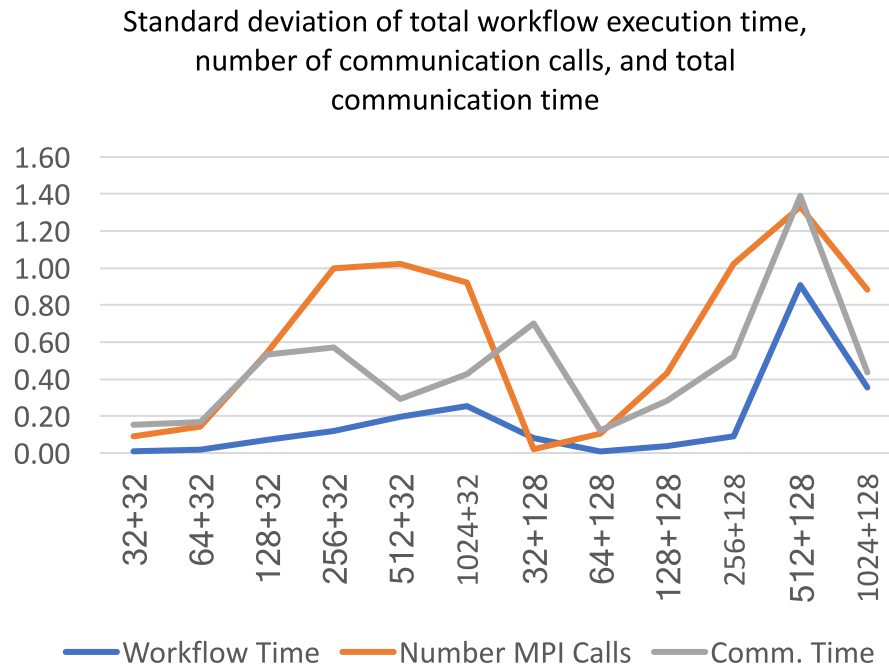


X: Number of nodes for each component
Y: Time to completion

Classical Molecular Dynamics particle simulator: LAMMPS
Execution:
- runs on Titan with 2 components
- calculation component => bulk of computation – first number
- small test component =>stage-write
- 10 runs for each node combination

Characteristics extracted with Chimbuko

- Same runs as previously
- Fluctuations in total execution time is shown with communication performance
  - comm OH, data transfer
- Number of communication calls is extracted
  - MPI calls - all aggregated
- Total communication time is aggregated
- Standard deviations exhibit similar trends but with larger fluctuations for the number of calls



X: Number of nodes for each component

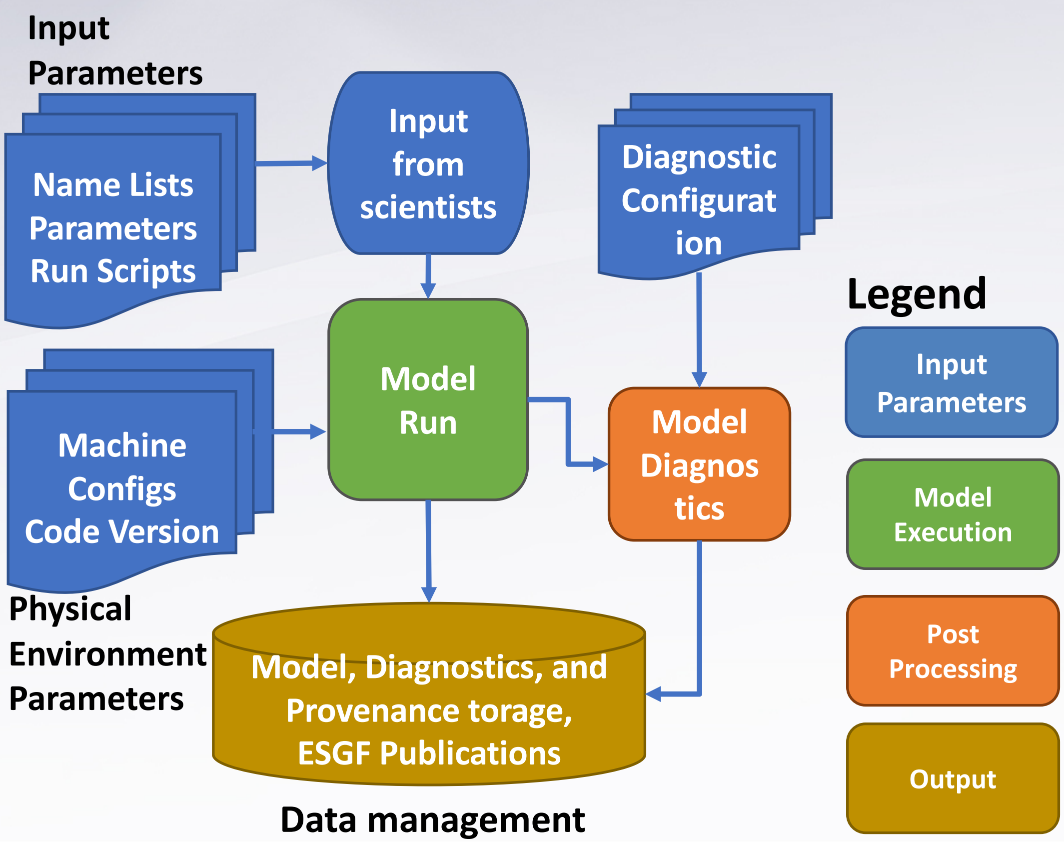
We need additional information:

communication behavior needs to be further explored

launching patterns and task graph

execution time: per component, per node

ACME
Accelerated Climate Modeling
for Energy



- Models answer scientific questions and inform policy
- independently verifiable data
- large, distributed development teams
- need to access previously successful run configurations

so far, sharing this information is largely done manually

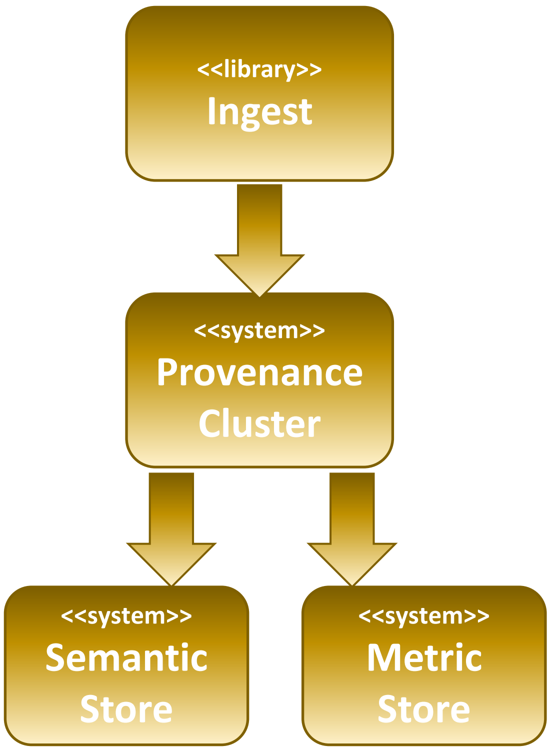
U.S. DEPARTMENT OF ENERGY

ACME Harvest List and run_acme script

https://github.com/ACME-Climate/ACME/blob/master/run_acme.template.csh

Harvest Item	Location
Git hash of the ACME code run	GIT_DESCRIBE
Component set and resolution	README.case in case root
Machine environment	env_case.XML
Machine specific information	env_mach_specific.xml; env_mach_pes.xml
Compiler	env_build.xml
Namelists	xxx_in files and user_nl_xxx files
XML files	from CaseDocs subdirectory
Environment variables set by	archive/…/CaseDocs.xxx/software_environment.txt
run_acme script if used	archive/…/build_environment.txt run_script_provenance directory in case directory
Compiler flags	Macros.make in case directory Depends.* SourceMods building logs from building directory

ProvEn: Workflow Performance Analysis Campaign Storage and Analysis



Captures performance analysis and provenance for reproducibility campaigns

Incorporates metrics, handles scalability, and undertake analytics

Provenance data stored in semantic store

Performance data stored in time-series db

Jupyter Notebook interface to support desktop analytics

Scalability and Reproducible Research

- Data selection
- In-situ analytics using anomaly detection are used in Chimbuko to reduce volumes of trace and performance profiles
    - Decision points, and ML parameters will have to be persisted
    - Encouraging the publication of make files and training sets
- Replacing resource-intensive on-demand queries with predictive modeling in ProvEn

Formalizing computational workflows to achieve reproducibility

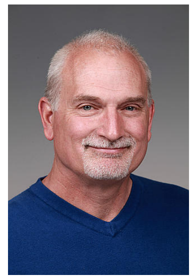
- Workflow Building Blocks (Radical Pilot) rather than monolithic systems
- Multiple levels of reproducibility as goals
- Tasks along the paths towards these goals themselves improve quality and reproducibility of the research

Team members

Computational Scientists

Application Scientists

Data Scientists



The authors gratefully acknowledge the funding support from the U.S. Department of Energy Office of Science/ Office of Advanced Scientific Computing Research as part of "Integrated End-to-end Performance Prediction and Diagnosis (IPPD)," and the Exascale Computing Project (17- SC-20-SC) "Co-Design Center for Online Data Analysis and Reduction." This manuscript has been authored by 1) employees of Brookhaven Science Associates, LLC under Contract No. DESC0012704, and 2) employees of Pacific Northwest National Lab under Contract DEAC05-76RL01830. This work was supported by the U.S. Department of Energy Office of Science/Office of Biological and Environmental Research under Contract DE-AC52-07NA27344 at Lawrence Livermore National Laboratory.

U.S. DEPARTMENT OF ENERGY

BROOKHAVEN NATIONAL LABORATORY

ECP EXASCALE COMPUTING PROJECT

CODAR

70 YEARS OF DISCOVERY

A CENTURY OF SERVICE