Maestro Workflow Conductor: A vision for the future of HPC Workflow

Computing Expo

Francesco Di Natale
Software Engineer
Maestro Project Lead
Computer Scientist (ASQ)

September 30, 2020

LLNL-PRES-810817

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

maestro

What is Maestro? What can Maestro do?

Maestro Workflow Conductor is an open-source HPC software tool and library that automates software processes

- Automation of multi-step computational workflows both locally and on supercomputers
  - A parameter sweep of a simulation model (setup, simulate, post-process)
- Parses a human-readable specification that is self-documenting and portable from one user and environment to another
- Makes it easy to setup and run computational based studies by abstracting away the details of running on HPC clusters
- The core design tenants of Maestro focus on:
  - encouraging clear workflow communication and documentation
  - consistent execution allowing users to more easily focus on science

Maestro handles core functions of running a user's workflow

1. Run submission and monitoring

Maestro submits, monitors, and restart jobs. Maestro can also manage the amount of jobs submitted to the scheduler at a given time.

2. Workspace management

Maestro manages the study workspace creating files and ensuring data doesn't overwrite steps/studies.

3. Workflow Provenance

Maestro captures workflow provenance of what is run including the sampled parameters, study spec, and inputs.

Maestro centers around the concept of studies for defining step-wise workflows

- A list of steps with their dependencies specified
- Parameters to apply to the list of steps
- Fixed value substitutions (variables)
- A study specification is a documented artifact of a user workflow that can be run and repeated
- A user can write a study by hand or write a programs to algorithmically generate study specifications.

```yaml
description:
    name: hello_world_singleparam
    description: A simple single parameter 'Hello World' study.

env:
    variables:
        OUTPUT_PATH: ./sample_output/hello_world

study:
    - name: hello_world
      description: Say hello to someone!
      run:
          cmd: |
              echo "Hello, $(NAME)!" > hello_world.txt

global.parameters:
    NAME:
        values: [Pam, Jim, Michael, Dwight]
        label: NAME.%%
```

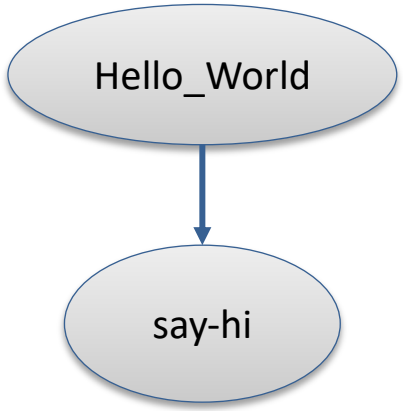
A simple “Hello World” Maestro study specification.

```
description:
  name: Hello_World
  description: Say hi to everyone!

study:
  - name: say-hi
    description: Echo hello, world to a file.
    run:
      cmd: |
        echo "Hello, world!" > hi.txt
      depends: []
```

Study overview

User specified steps to be executed

Maestro DAG



To run ”hello.yaml”, simply execute the command line “maestro run hello.yaml”

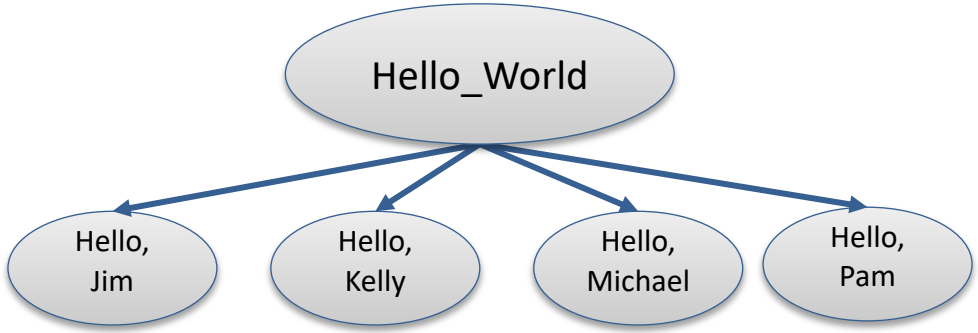
A simple “Hello World” Maestro study specification.

```
description:
  name: Hello_World
  description: Say hi to everyone!

study:
  - name: say-hi
    description: Echo a friendly greeting.
    run:
      cmd: |
        echo "Hello, $(NAME)!" > hi_$(NAME).txt
      depends: []

global.parameters:
    NAME:
          values: [”Jim”, ”Kelly”, “Michael”, “Pam”]
          label: NAME.%%
```

Study overview

User specified steps to be executed

User specified parameters

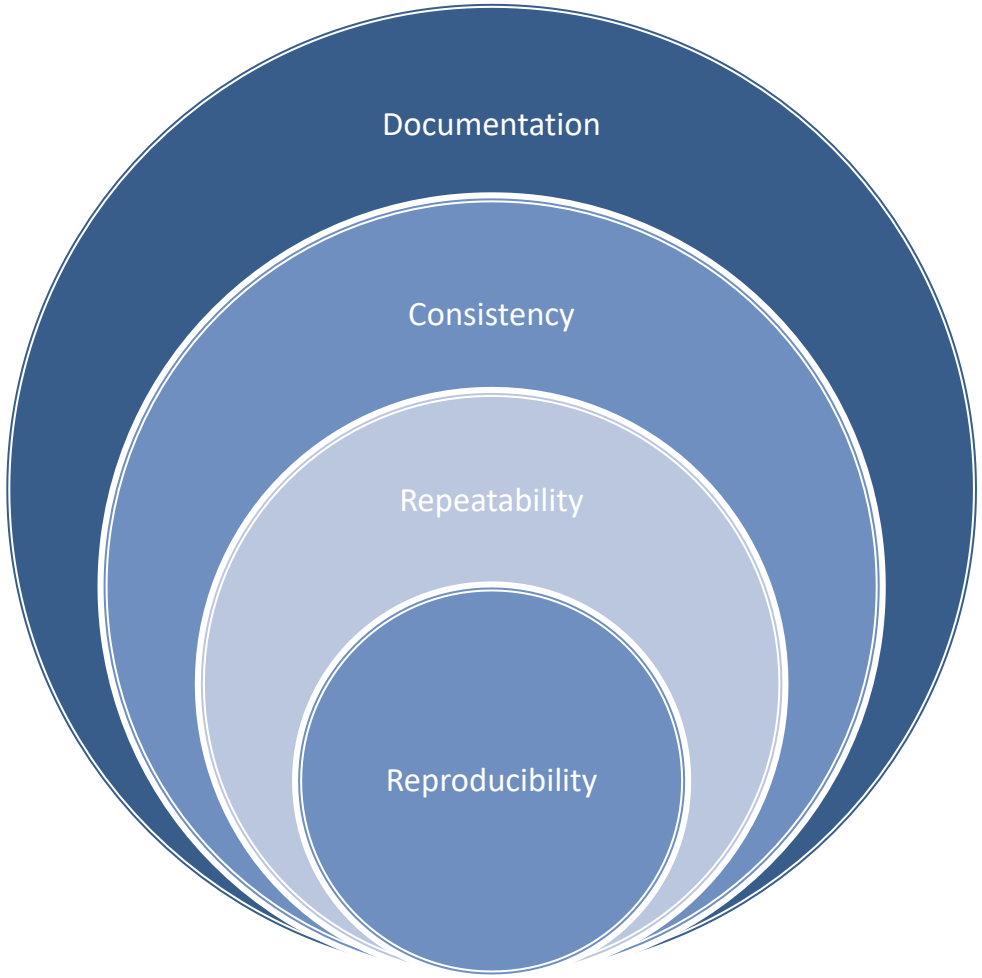


Adding a parameter to a study is straight-forward, simple, and easy.

maestro

How is Maestro designed?

Maestro's core principles center around reproducibility

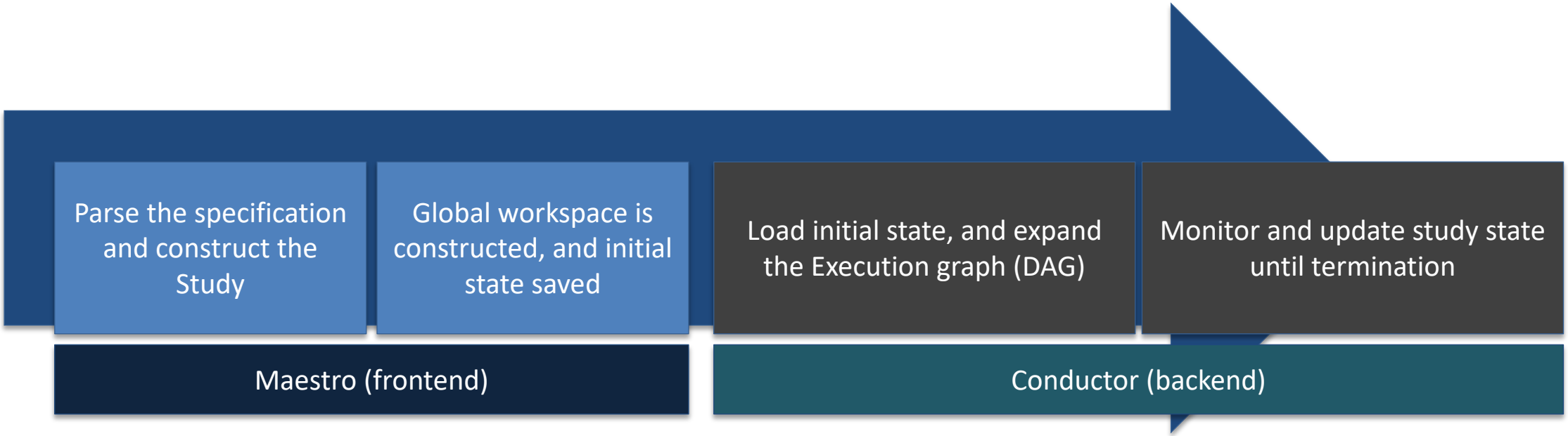


- Self-documentation
  - Should be documented and easy to document.
- Consistency
  - Should be run the same way every time it's run.
- Repeatability
  - Should be easy to repeat.
- Reproducibility
  - All the above are pre-requisites.
  - Different than repeatability.
  - Requires more extensive metadata capture.

Maestro studies allow users to break workflows down into composable pieces

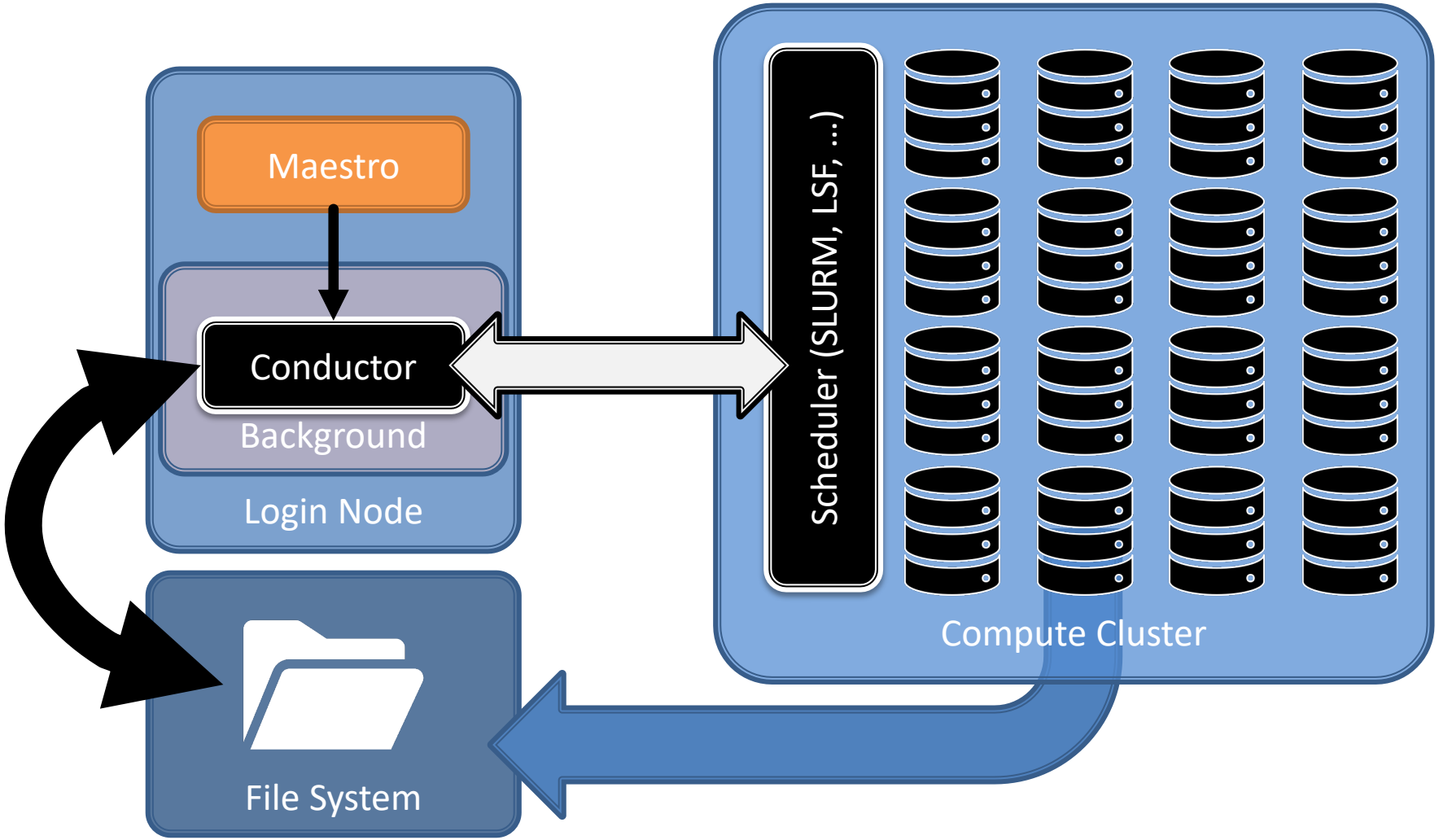
Workflow Overview • Name • Description • Other metadata	description: name: simple_workflow description: A simple workflow.
Study Steps specify • What gets run • The order in which things are run • Used to define multistep workflows	study: - name: run-sim description: Submit the simulation. run: cmd: **/usr/gapps/code input.in –def res $(RES)** - name: post-process description: Post process simulation run: cmd: **python process.py –p $(run-sim.workspace)** depends: [run-sim]
Parameter/sample space	global.parameters: RES: value: [2, 4, 6] label: RES.%%

Maestro is split between the frontend command line utility and the backend Conductor daemon



- The benefit to having this modular design is that the various components can be swapped out to deliver various benefits.
  - Different specifications could be supported
  - Different backends utilizing varying technologies can be seamlessly used

Maestro is split between the frontend command line utility and the backend Conductor daemon



Maestro's Software Engineering Strategy and Vision

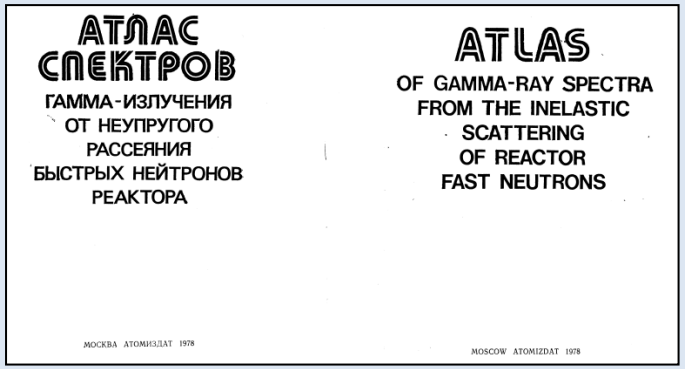
- A strong focus on user centered design and development
  - Meet requirements in as **lightweight**, **transparent**, and **general** a manner as possible
  - Negotiate requirements to provide features that encourage ease of use and best practices
  - Provides as much flexibility as possible leaving workflow decisions to the user
- Development of a community that shares a common workflow vocabulary and collaborates around central core of best practices
  - The study specification provides a consistent, step oriented, workflow structure for discussion
- An emphasis on flexibility, maintainability, and expandability
  - Enable users to utilize technologies, but not couple users to them
  - Use sound software system design and architecture to promote sustainability
  - Enable the creation of a community driven ecosystem

Where is Maestro being used?

Maestro is being used to compare nuclear data measurements to compiled libraries

Al-Tuwaitha Nuclear Research Facility, Iraq

- IRT-5000 reactor "decommissioned" in Operation Desert Storm
- IAEA shared databook with LBNL, LLNL
- LBNL created online electronic database

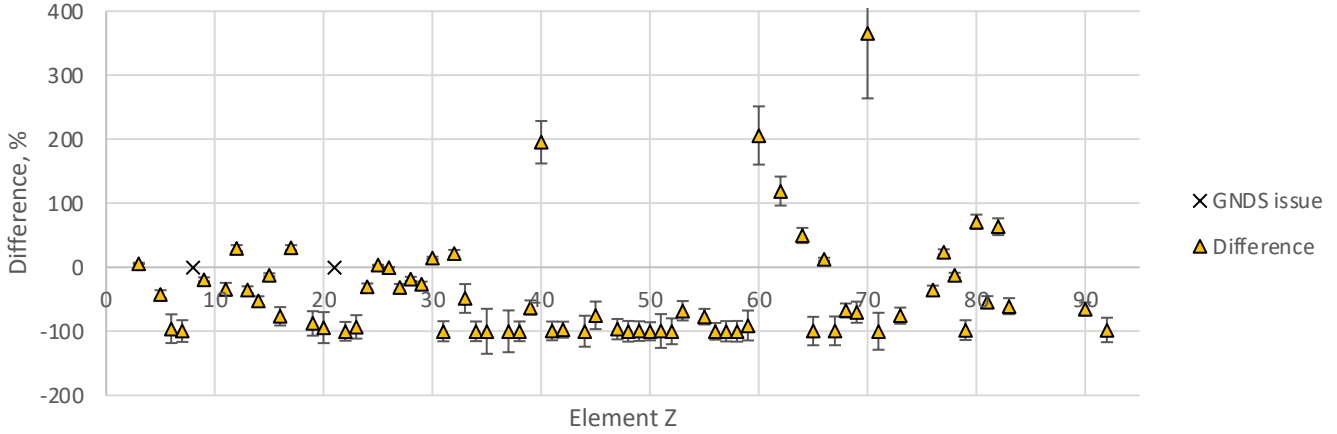

АТЛАС СПЕКТРОВ
ГАММА-ИЗЛУЧЕНИЯ ОТ НЕУПРУГОГО РАССЕЯНИЯ БЫСТРЫХ НЕЙТРОНОВ РЕАКТОРА

МОСКВА АТОМИЗДАТ 1978

ATLAS
OF GAMMA-RAY SPECTRA FROM THE INELASTIC SCATTERING OF REACTOR FAST NEUTRONS

MOSCOW ATOMIZDAT 1978

- Compared data in "Baghdad Atlas" to data libraries
  - Gamma-rays produced in neutron-inelastic reactions
  - Data libraries include ENDL and ENDF used in applications



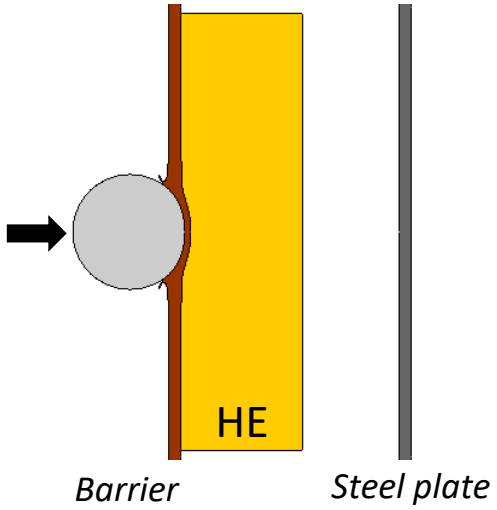
- Maestro used to run ~70 Mercury simulations with GNDS (ENDL 2009.3) data and post-process results to get gamma intensity
- Next: Add plotting call to Maestro and test additional data evaluations such as ENDFB-VIII

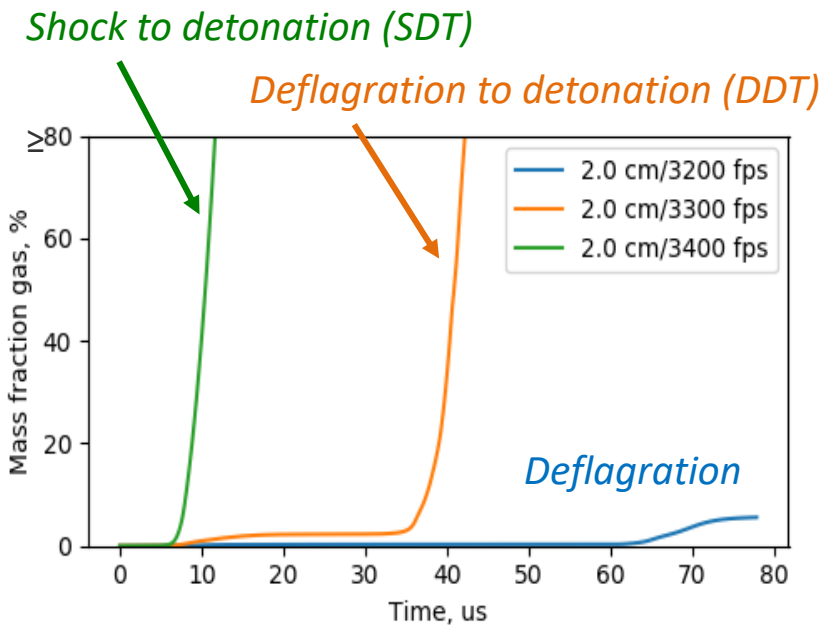
Study of fragment impacts on explosives is using Maestro to sweep across parameters

- High Explosive Response to Mechanical Stimulus (HERMES) model used to examine response of high explosive (HE) materials to mechanical insults
  - Package in ALE3D
  - Maestro with pgen used to sample fragment size and speed for different geometries

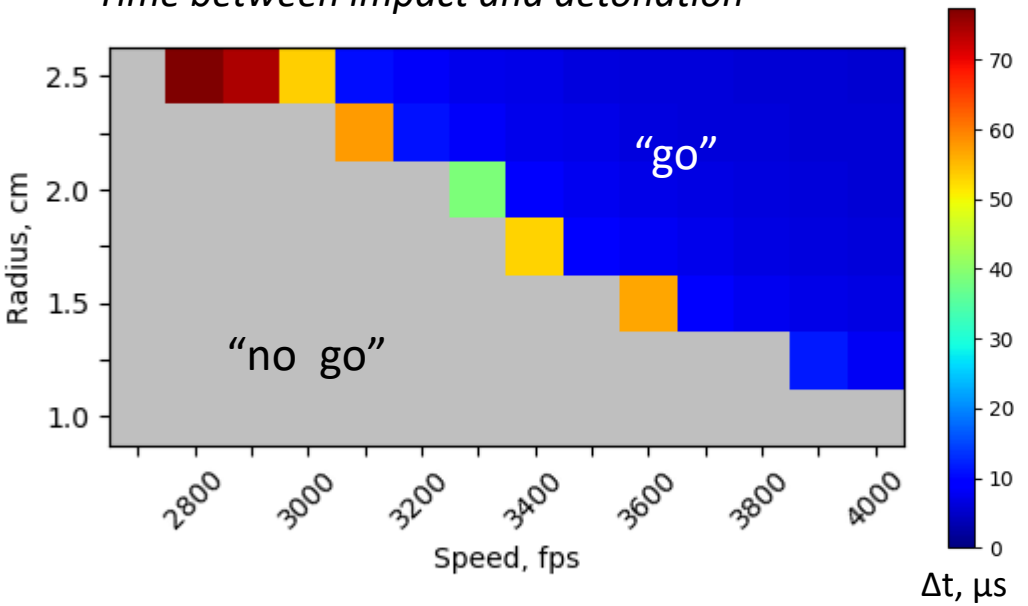
Example

2 cm radius steel sphere, 3400 fps, at t = 6 μs





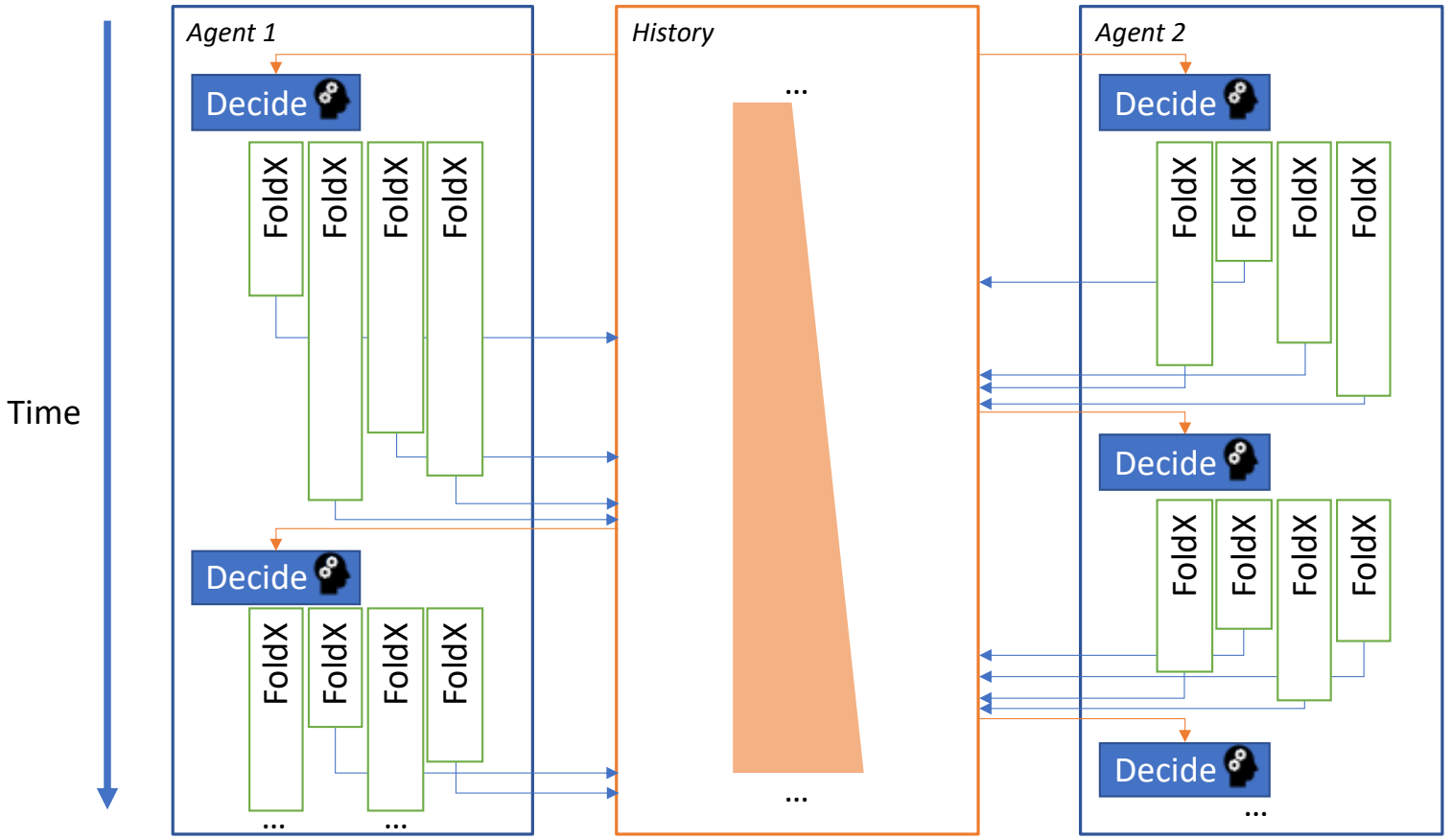
Time between impact and detonation



- Next steps: automate post-processing and job submission with Maestro to define "go/no go" boundary

Maestro is being used to train a decision-making loop for finding antibodies to SARS-CoV-2 (COVID-19)

- Agents are spun up and alternate between decision making and executing calculations
- The individual studies place their structure and results into the history
- Decision makers choose new mutations to run calculations



Maestro is improving user productivity in a wide variety of ways

- Generation of perturbed simulations of a shaped-charge jet and creating synthetic radiographs to feed a deep learning model along with scalar data from the simulations
  - Train the model to link images back to input parameters (surrogate modeling)
- Pipelining of cardiac simulations and testing of the hyperparameters for an ML model that generates non-invasive cardiac images based on EKG input data
  - Led to a patent on the model for generating images
- The ATOM Modeling Pipeline (AMPL) has used Maestro to predict the safety and pharmacokinetic properties of over 26 million drug-like compounds (GS-CAD)
  - When mixed with binding affinity calculations, can be used to recommend experimental drugs in the battle against COVID-19
  - Dataset released this week: https://covid19drugscreen.llnl.gov/info

We are excited to work with the user community in helping to develop and grow their workflows

- **Maestro GitHub**
  - https://github.com/LLNL/maestrowf
- **Maestro Issue Tracker**
  - https://github.com/LLNL/maestrowf/issues
- **Maestro Documentation**
  - https://lc.llnl.gov/confluence/display/MAESTRO
  - https://maestrowf.readthedocs.io
- **Mailing List**
  - maestrowf@llnl.gov
- **Try Maestro**
  - pip install maestrowf

Maestro encourages a supportive and collaborative community for both Maestro developers and users.

Get involved!

- Provide feedback/use cases
- Submit tickets
- Become a developer
- How are you using Maestro?
  - Tell your story ☺.
- Hang out and join the discussion!