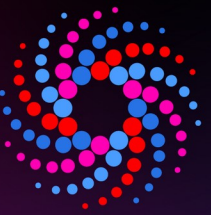


SC22
Dallas, TX | hpc accelerates.

Events as a Basis for Workflow Scheduling

David Marchant

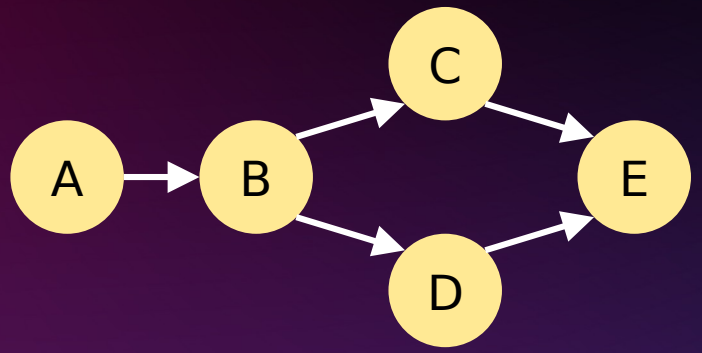
Premable

Funded by the European Union

- Part of the MUMMERING Project
- MUltiscale, Multimodal, and Multidimensional imaging for EngineeRING
- Marie Skłodowska-Curie Innovative Training Network (Grant Agreement no. 765604)

Scientific Workflow Management Systems

- Utilise DAGs or logical equivalent
- Very strong predictive system
- But also very limited adaptability



Apache Airflow

dask

kubernetes

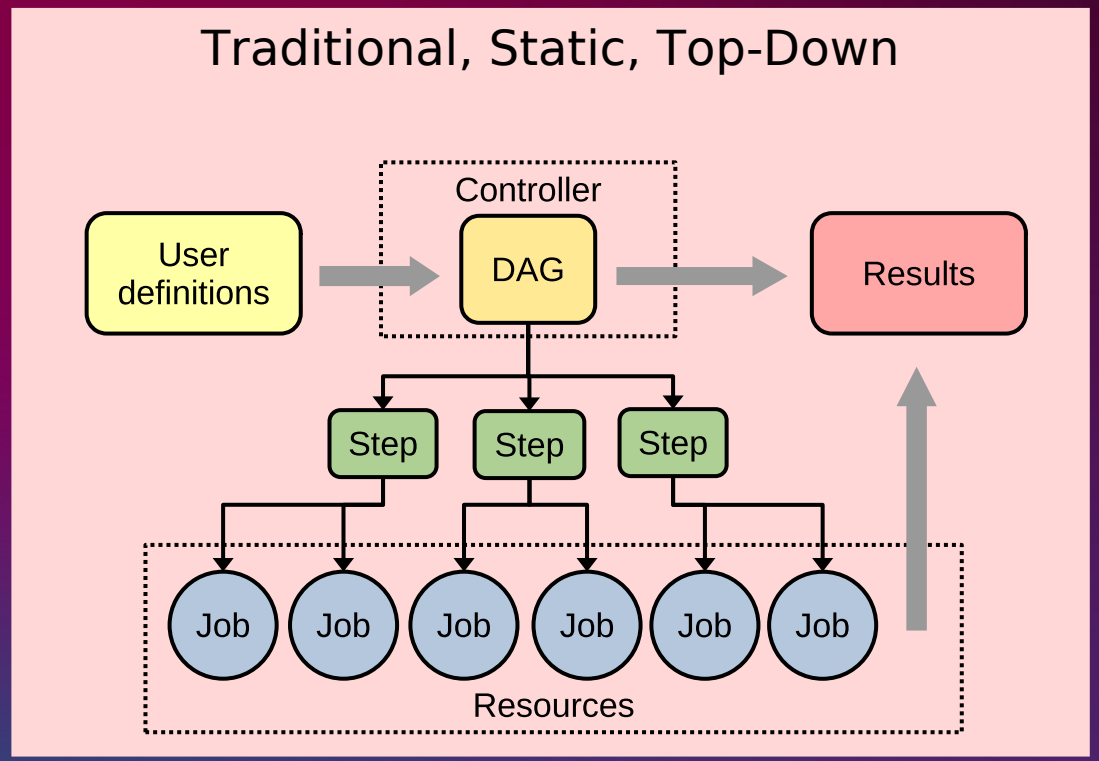
slurm workload manager

Apache Taverna

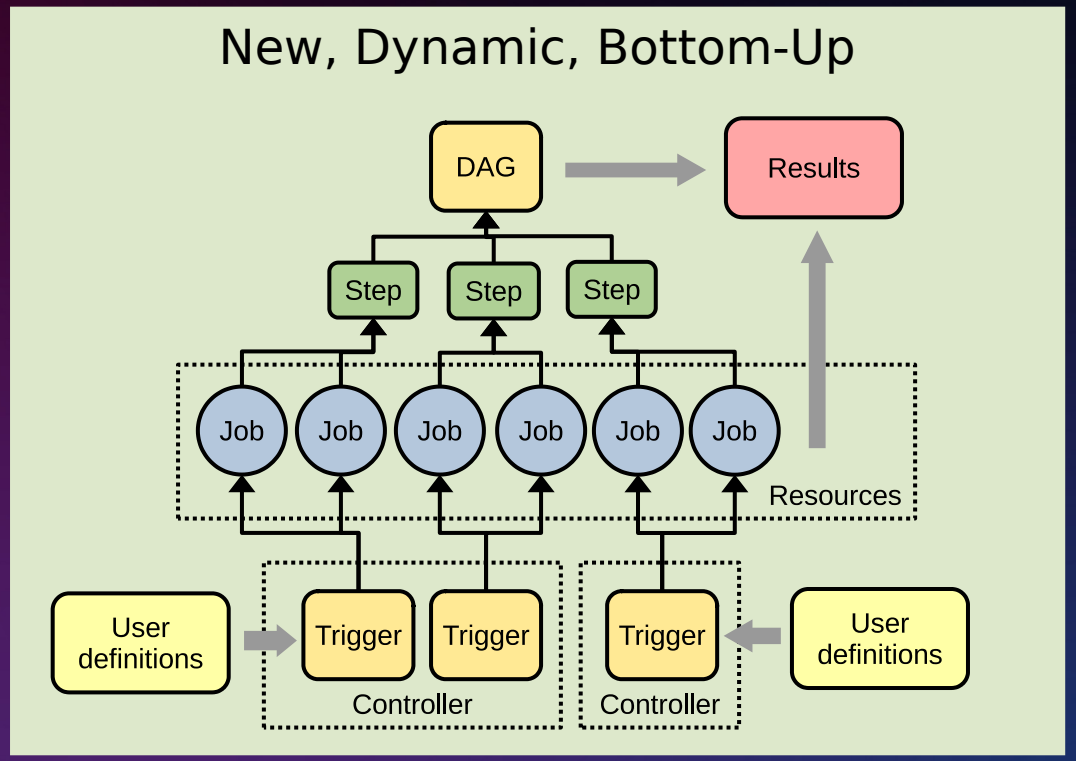
Kepler

WIFIRE

Top-down vs Bottom-up

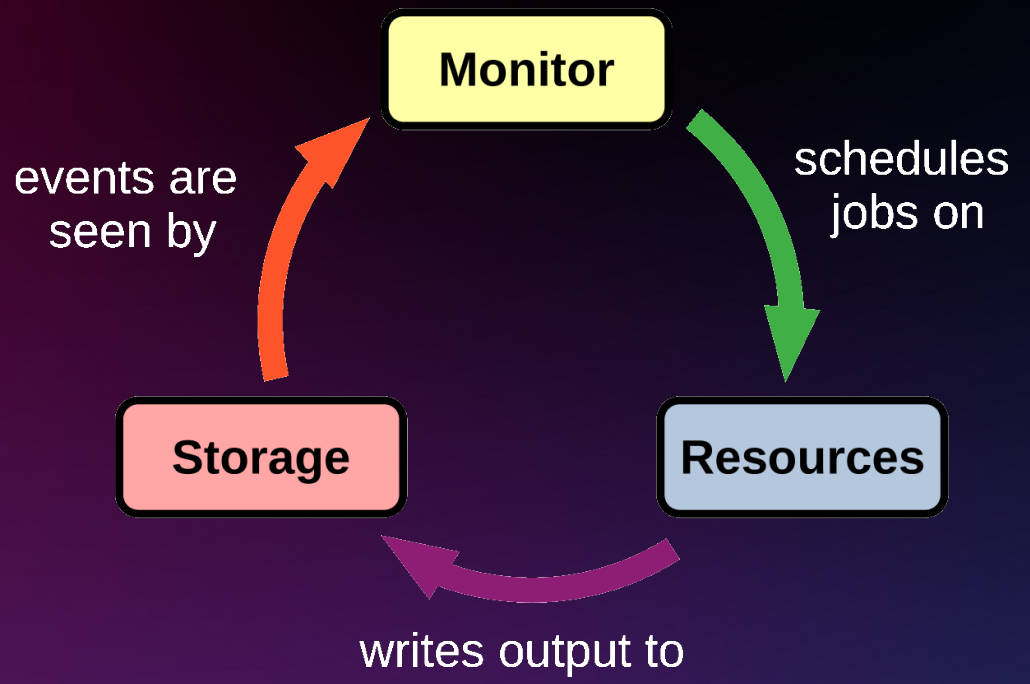


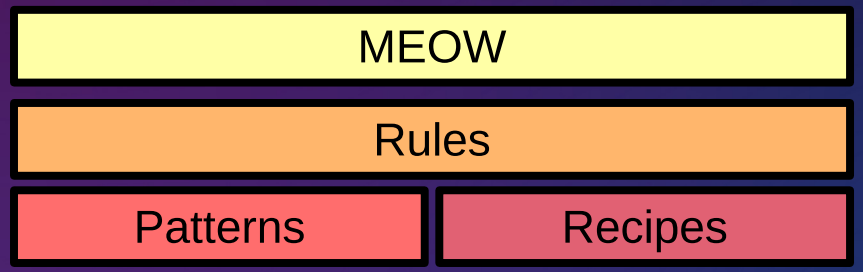
VS



MEOW

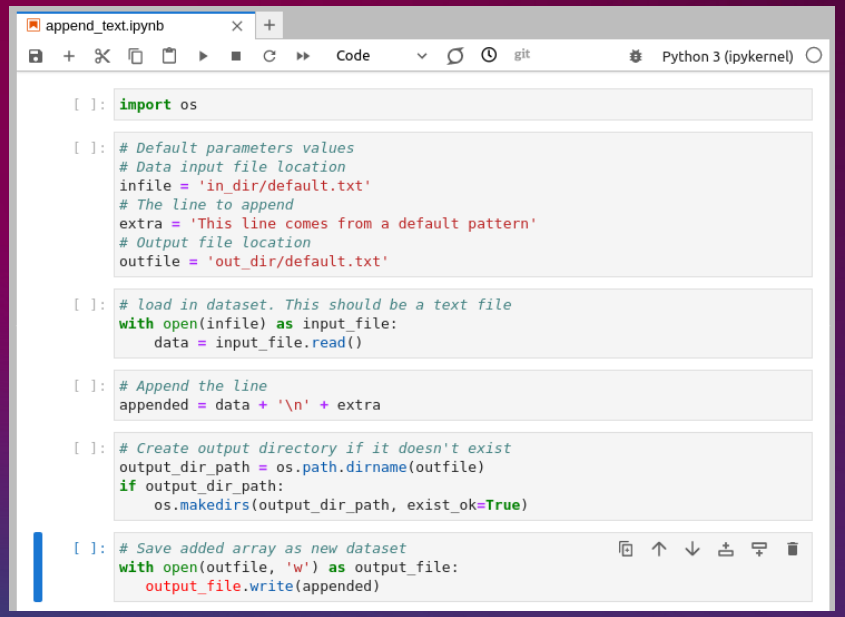
- Managing Event Oriented Workflows
- Rules-based system for isolated job scheduling
- Composed of Patterns and Recipes
- Workflows as an emergent property
- Workflow structure can be altered by adding, cancelling or modifying jobs or monitoring structures





mig_meow

- Python library for building MEOW objects
- Users define *Recipes* (the code to run) ... or *Patterns* (The conditions when to run)

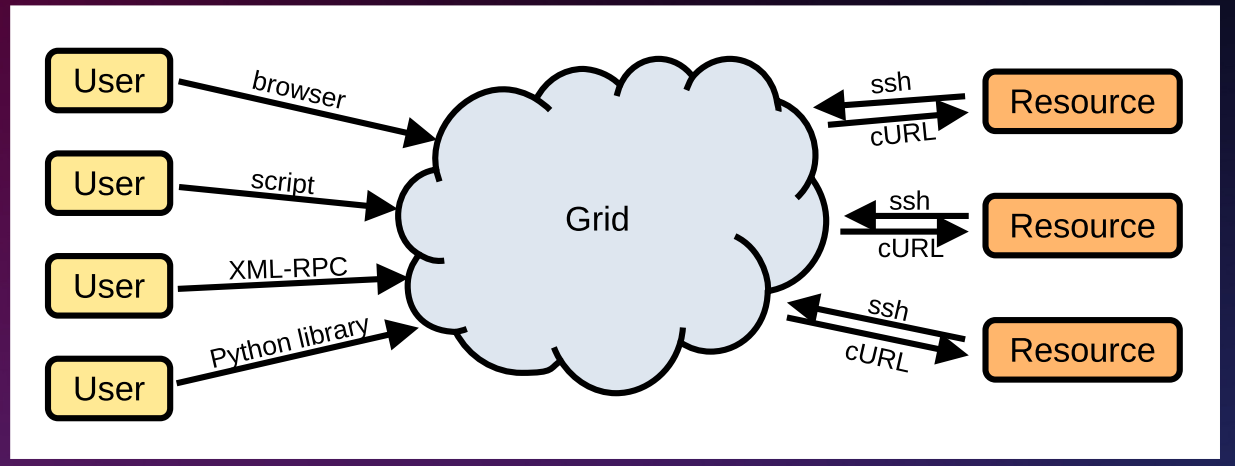



```
input_file: infile
input_paths:
- initial_data/*
output:
  outfile: '{VGRID}/int_1/{FILENAME}'
parameterize_over: {}
recipes:
- append_text
variables:
  extra: This line is overridden
```

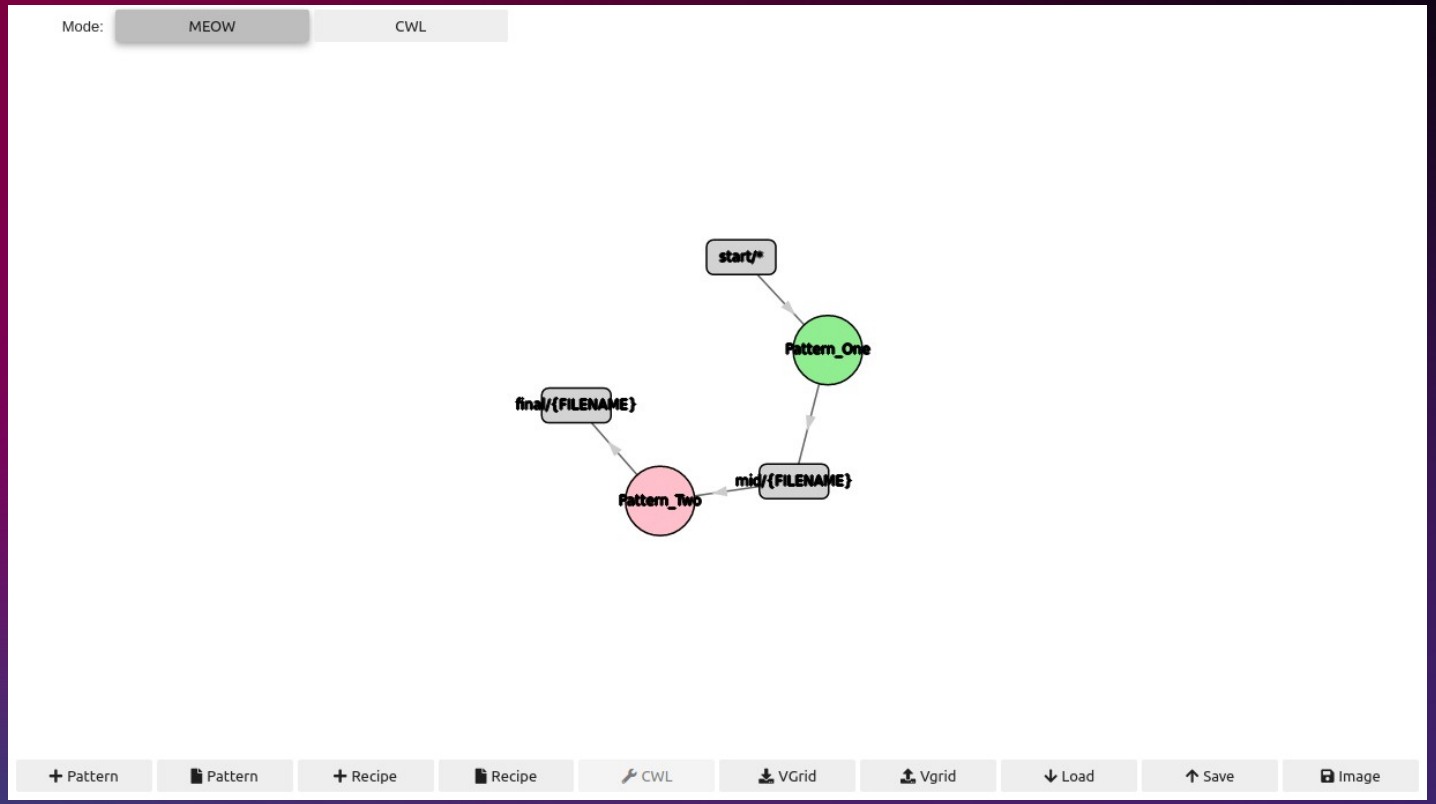
- Together these form a *Rule* (Scheduling in response to events)

Where can it be used?

- Minimum intrusion Grid
- Integrated into cloud-based Jupyter hub instances
- Also available as a stand alone package in any Python3 environment via *mig_meow* library
- Provides *Workflow_Runner*



Using the WorkFlow Widget to Predict



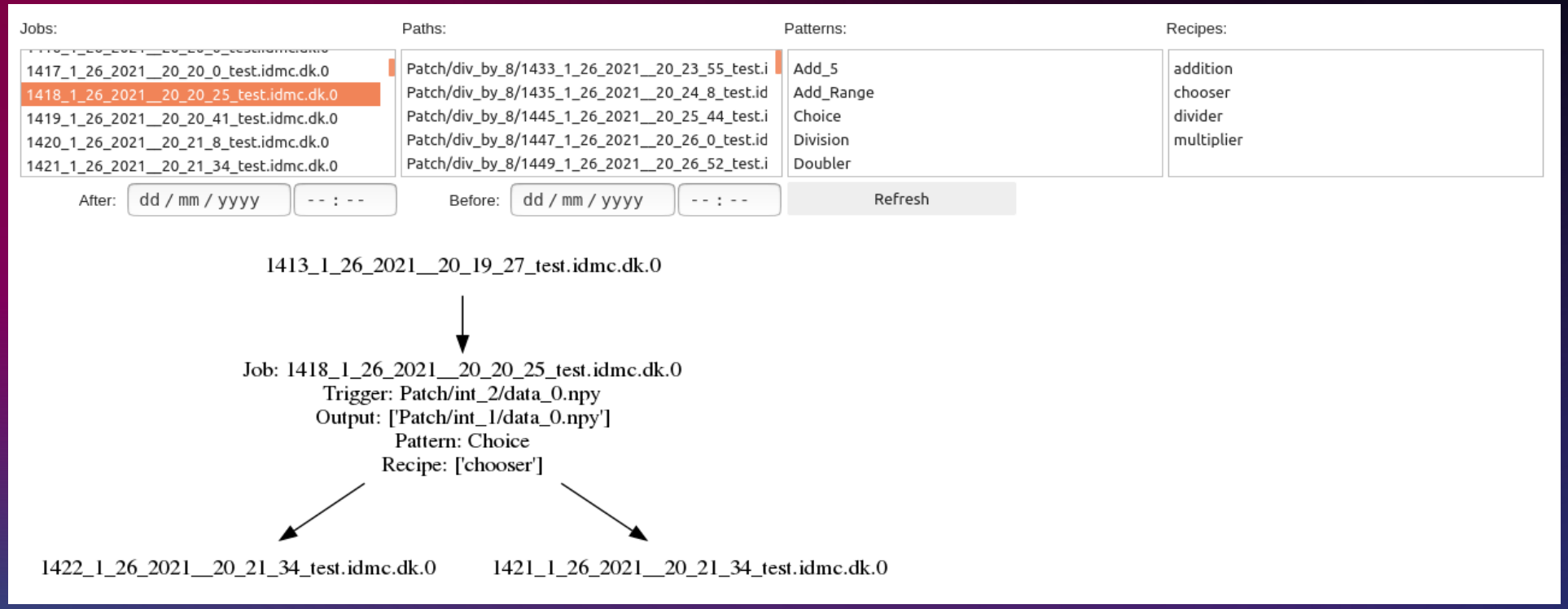
Using the Monitor Widget to Track

```
In [7]: meow.create_monitor_widget(vgrid='test_vgrid')
```

Displaying 1 to 10 of 37 total jobs for VGrid test_vgrid

Job ID	Status	Created at			
40_1_27_2020__14_53_48_migrid.test.0	FAILED	2020-01-27 14:53:48	+	↻	×
39_1_27_2020__14_44_48_migrid.test.0	FINISHED	2020-01-27 14:44:48	+	↻	×
38_1_27_2020__14_16_58_migrid.test.0	FAILED	2020-01-27 14:16:58	+	↻	×
37_1_27_2020__13_41_49_migrid.test.0	FAILED	2020-01-27 13:41:49	+	↻	×
36_1_27_2020__13_35_18_migrid.test.0	FAILED	2020-01-27 13:35:18	+	↻	×
35_1_27_2020__13_16_46_migrid.test.0	FINISHED	2020-01-27 13:16:46	+	↻	×
34_1_27_2020__13_9_43_migrid.test.0	FINISHED	2020-01-27 13:09:43	+	↻	×
33_1_27_2020__13_6_27_migrid.test.0	FINISHED	2020-01-27 13:06:27	+	↻	×
32_1_27_2020__12_50_12_migrid.test.0	FINISHED	2020-01-27 12:50:12	+	↻	×
31_1_27_2020__12_43_36_migrid.test.0	FINISHED	2020-01-27 12:43:36	+	↻	×

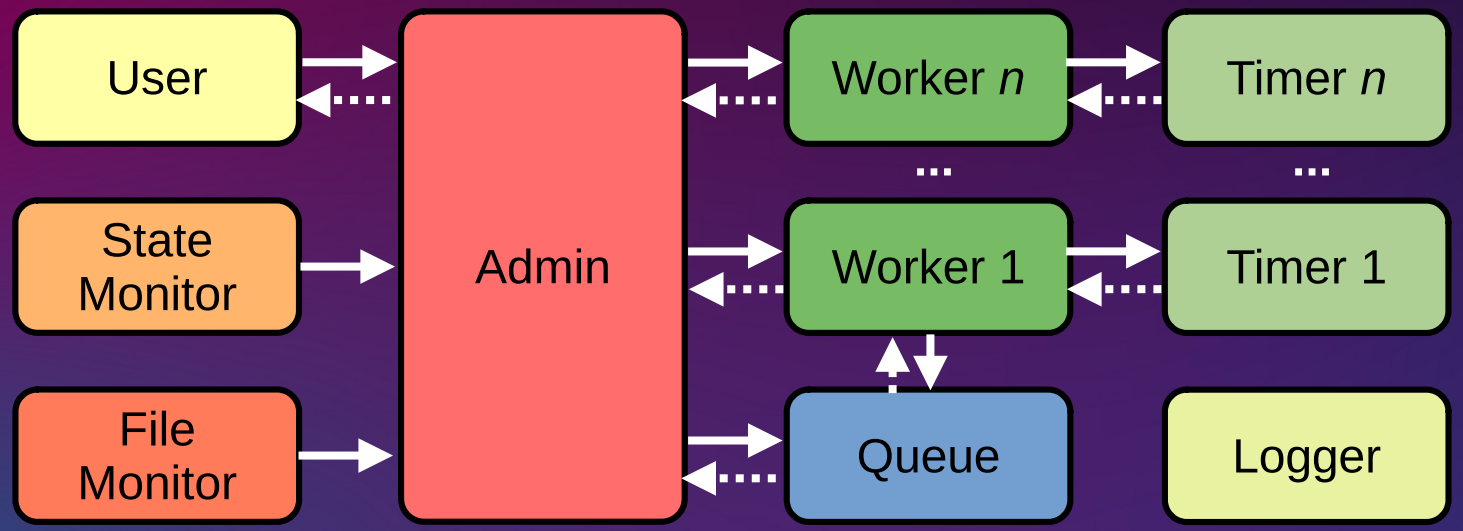
Using the Report Widget to Reflect



Workflow Runner Structure

- Designed to mimic MiG, albeit in a reduced capacity as we don’t have a whole grid to manage

- Originally intended illustration for users who lacked access to MiG so inherits structure warts and all



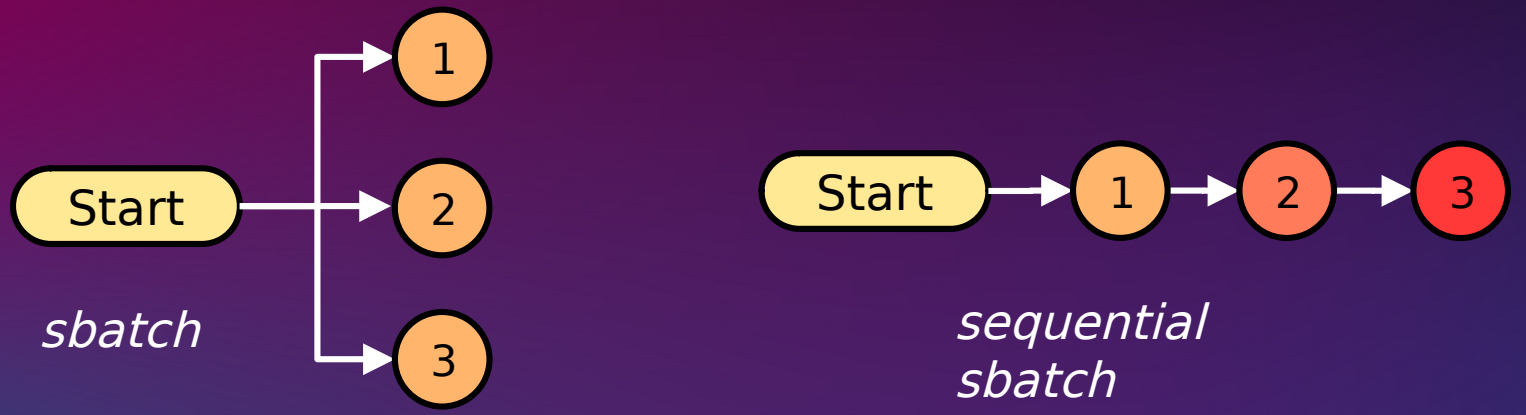
Notes on the timing tests

- All tests run repeated 10 times with mean taken as final result
- Run for 10 to 500 jobs
- Run both on a Laptop and ‘Threadripper’ desktop for scientific processing
- Threadripper:
  - 16 cores (32 hyperthreads)
  - Threadripper 1950X processor at 3.4GHz
  - 112 GB RAM

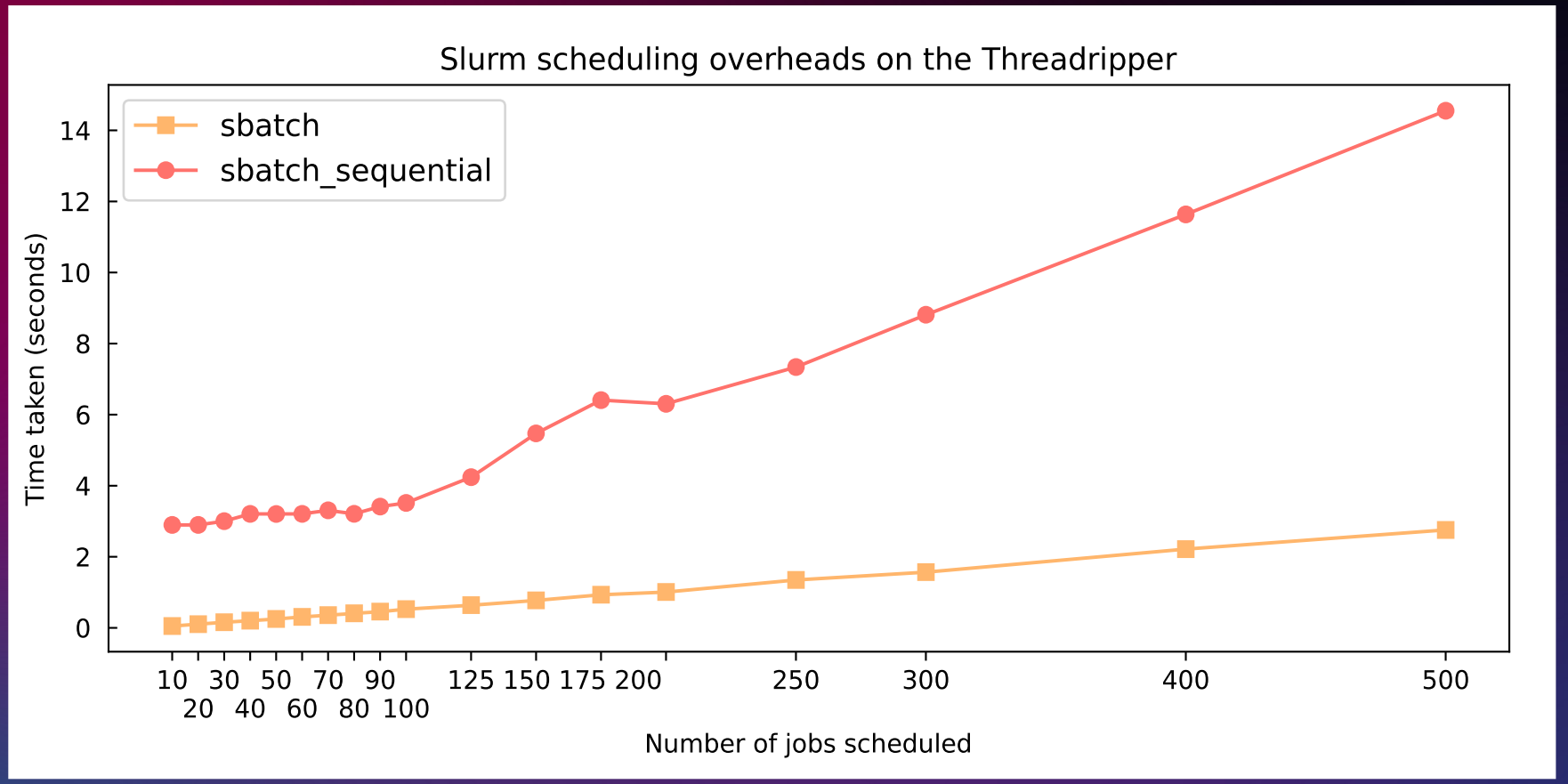
  Laptop:
  - 4 cores (8 hyperthreads)
  - i7-8550U at 1.8GHz
  - 8 GB RAM
- As tests are not explicitly parallelised (e.g only ever one scheduler), main performance difference is as desktop vs laptop

Slurm as a Baseline

- Not a fully fledged SWMS, but MEOW is currently light on non-scheduling features
- Widely adopted, both by researchers directly and within SWMSs. If we can get similar performance then MEOW is acceptable
- Main difference is Slurm scheduling is not ongoing



Slurm as a Baseline



Five Benchmarks

- Each test will schedule n jobs

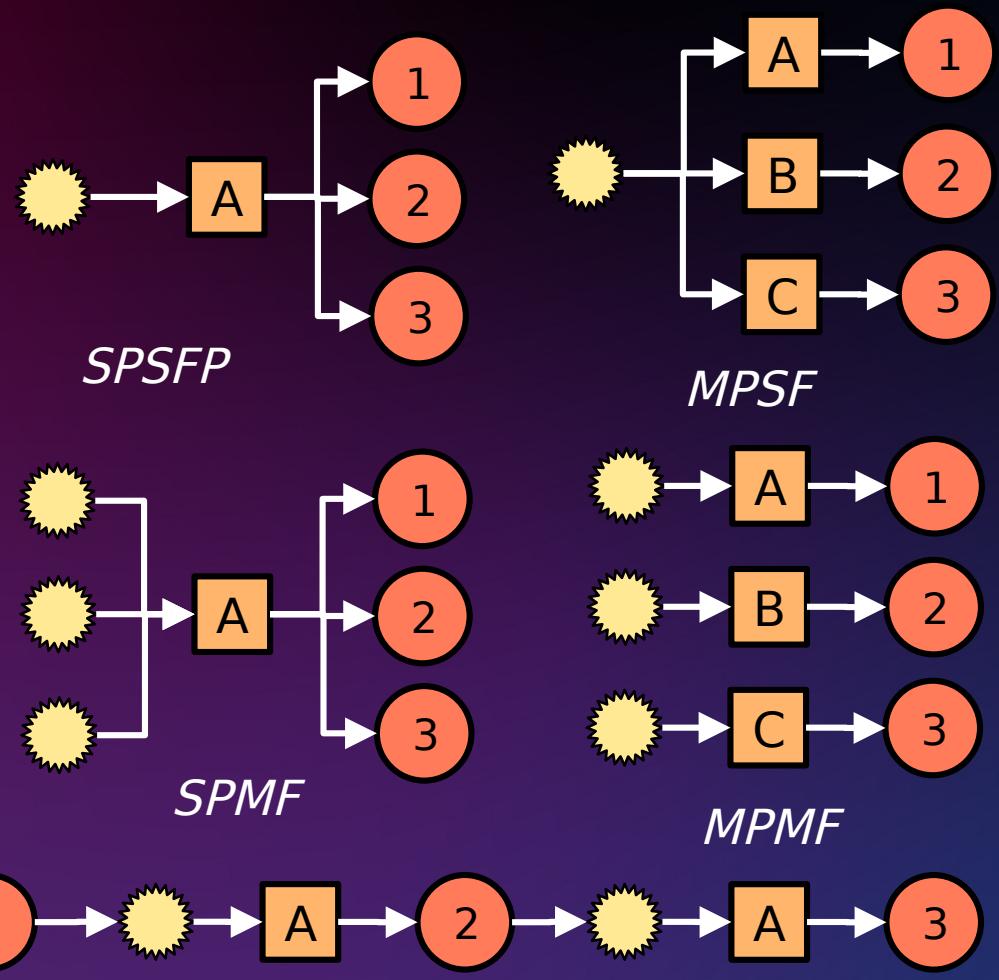
1)Single Pattern Single File Parrallel (SPSFP)

2)Multiple Patterns Single Files(MPSF)

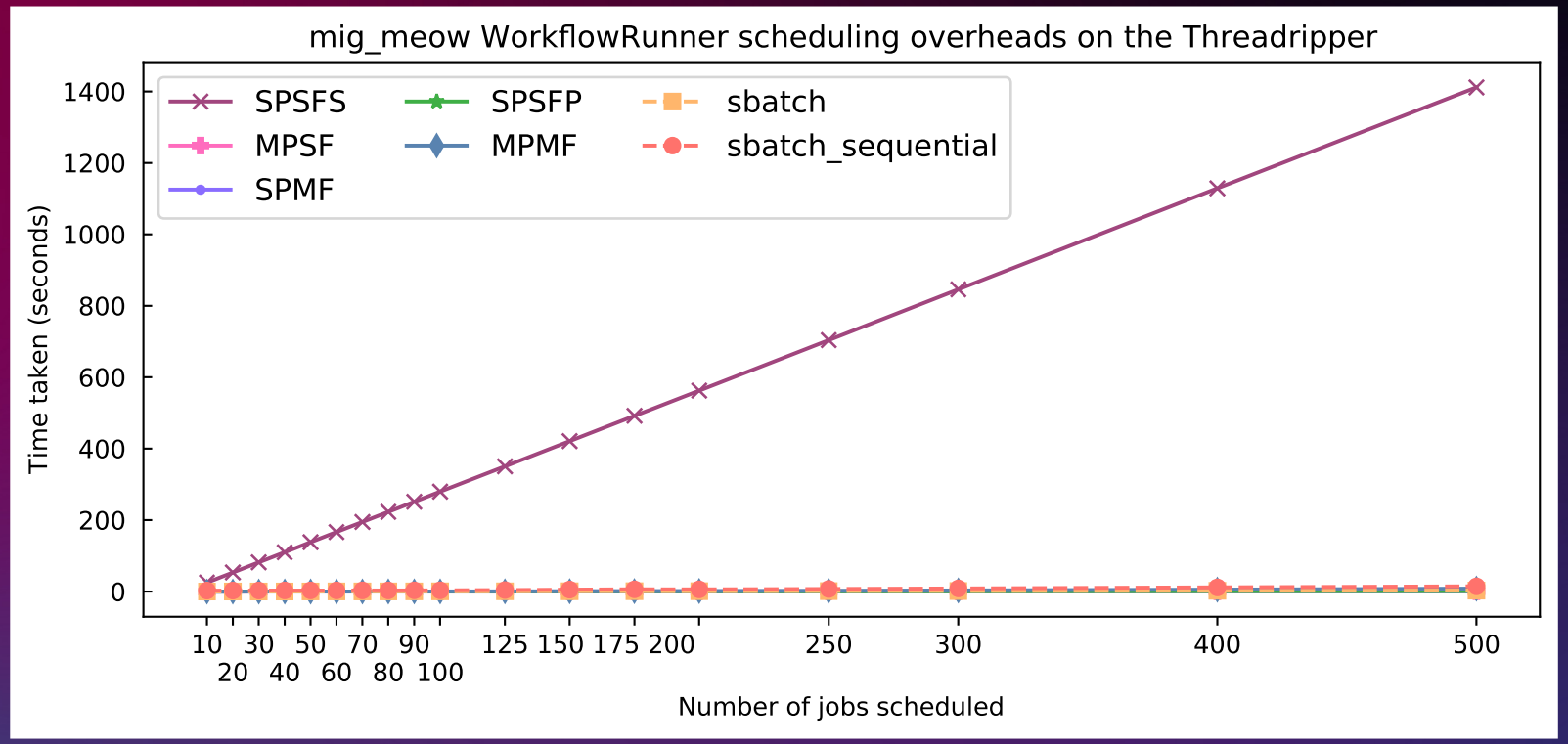
3)Single Pattern Multiple Files (SPMF)

4)Multiple Patterns Multiple Files (MPMF)

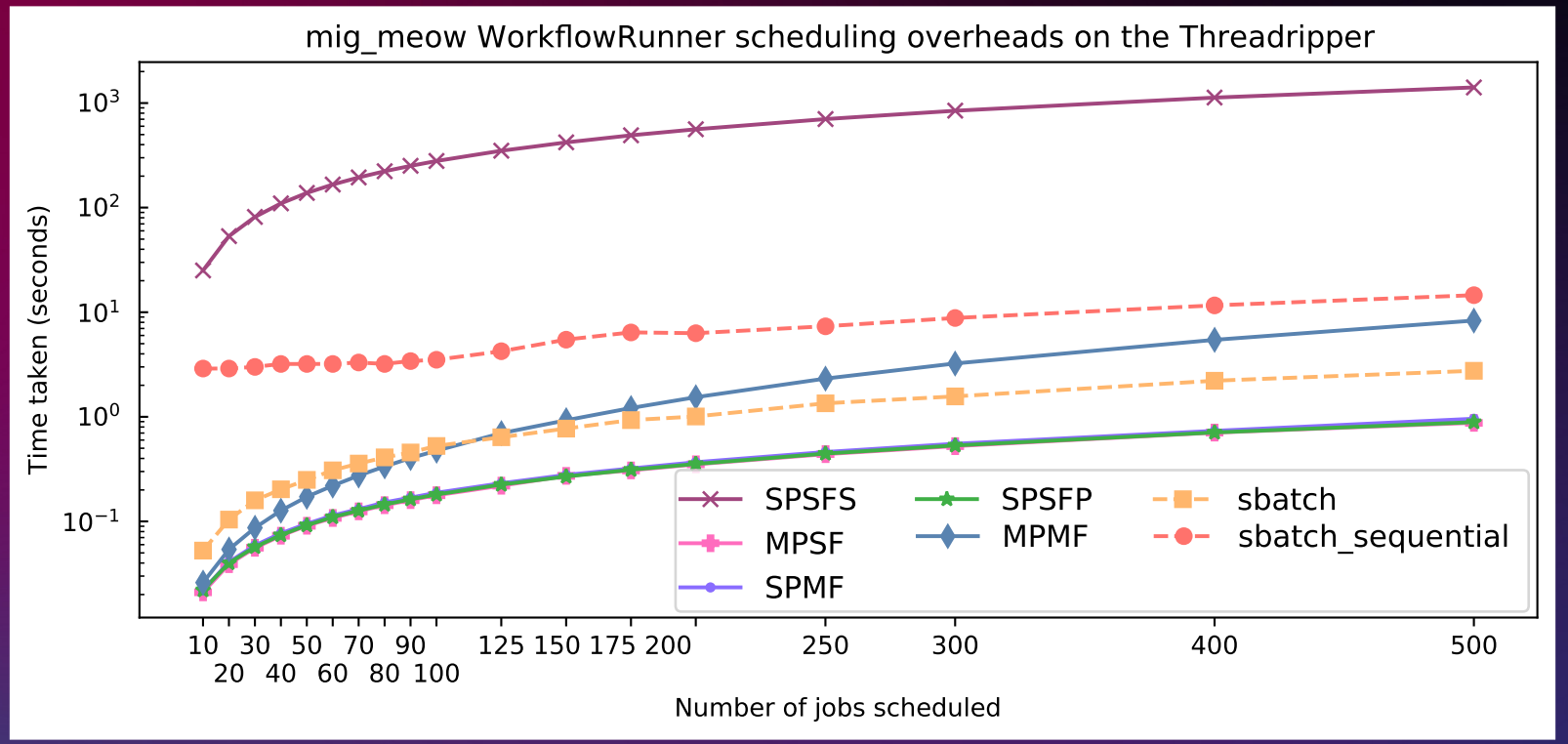
5)Single Pattern Single File Sequential (SPSFS)



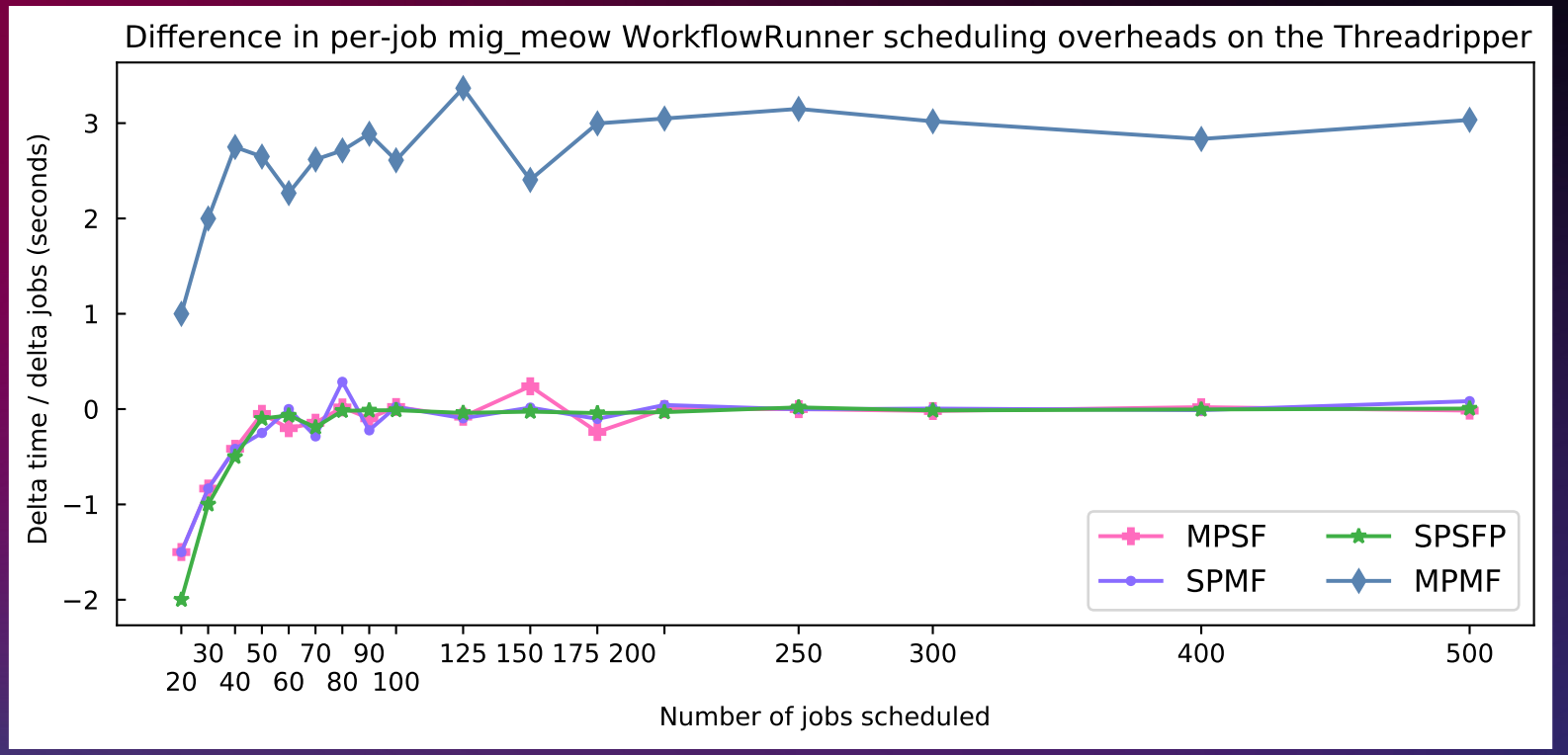
MEOW on the WorkflowRunner



MEOW on the WorkflowRunner



Scalability on the WorkflowRunner



Key Timings Takeaways

The Good

- WorkflowRunner generally beats Slurm. ~2.5x speedup in non-MPMF
- WorkflowRunner can be used as is for scheduling
- Per-job scheduling time is vanishingly small in any significant processing

The Bad

- Sequential Testing is always much slower

The Ugly

- MPMF is showing signs of quadratic growth

Concluding Remarks

- Events are plausible alternative to DAGs as a basis for workflow scheduling, but DAGs are still useful
- Available now as part of the MiG, or as stand-alone implementation in *mig_meow*
- Demonstrates acceptable scalability
- Well suited to extremely dynamic, distributed or heterogeneous systems (with work)
- More work needed to make managing event based analysis prettier and more useable