Flux on LC clusters

Ryan Day
LC Operational Resource Management

Flux development team: Al Chu, James Corbett, Jim Garlick, Mark Grondona, Dan Milroy, Chris Moussa, Tom Scogland, Jae-Seung Yeom

July 26, 2022
What is flux?

- Flux is the future of resource management on LC clusters.
- Flux is hierarchical. Every flux ‘job step’ can be a full flux instance with the ability to schedule more job steps on its resources.
- Flux has a rich API that makes it easy to launch flux instances from within scripts.
- Flux can be used now on LC systems.
Flux is the future of LC resource management

LC clusters will run a Flux system instance.

- Coral 2 systems (El Capitan) will run Flux as the scheduler.
- CTS-2 systems will run Slurm as the scheduler initially, but will transition to Flux.
What is flux?

- Flux is the future of resource management on LC clusters.
- Flux is hierarchical. Every flux ‘job step’ can be a full flux instance with the ability to schedule more job steps on its resources.
- Flux has a rich API that makes it easy to launch flux instances from within scripts.
- Flux can be used now on LC systems.
Flux is hierarchical: Launching steps in Slurm

Get allocation

Allocation (4 nodes)

Launch steps

Job step (2 nodes)

Job step (2 nodes)

Job step (1 node)

Job step (1 node)
Flux is hierarchical: Launching instances in Flux

1. Get allocation (launch instance)
2. Launch instance
   - Instance (4 nodes)
3. Launch instance
   - Instance (2 nodes)
4. Launch instance
   - Instance (2 nodes)
5. Launch instance
   - Instance (1 node)
6. Launch instance
   - Instance (1 node)
Flux is hierarchical: ATS node diagram
Flux is hierarchical: ATS node diagram

Flux instance

Flux instance

Flux instance

Local storage

memory

2 cores

18 cores

network

Sierra Node

gpu
gpu
gpu

gpu

50 GB/s per channel (bidirectional)

150 GB/s aggregated bandwidth (bidirectional)

X Bus (64 GB/s)

170 GB/s aggregate peak bandwidth

PCIe Gen4 x4

PCIe Gen4 x8 - CAPI

Flux is hierarchical: ATS node diagram

Flux instance

Flux instance

Flux instance
MuMMI implements a complex and dynamic workflow

**Single Macro Simulation**
C++ (with MPI); MOOSE; ddcMD
24 CPU cores/node; 2400 MPI tasks
242 GB per day

**ML-based Selection**
Python; ML frameworks; FAISS
24 CPU cores
>1000 decisions per minute

**CG Setup**
Python; Custom; GROMACS
24 CPU cores each
1.5 hr each

**FIFO; real-time tracking & update**

**CG Simulation**
C++ (with CUDA); ddcMD
1 GPU + 1 CPU core each
1.04 μs and ~6.5 GB per day

**In situ CG Analysis**
Python; Custom
3 CPU cores each
>2K frames per day

What is flux?

- Flux is the future of resource management on LC clusters.
- Flux is hierarchical. Every flux ‘job step’ can be a full flux instance with the ability to schedule more job steps on its resources.
- Flux has a rich API that makes it easy to launch flux instances from within scripts
- Flux can be used now on LC systems
Usability: Submitting a Job

- **Slurm**
  
  - `srun -N2 -n4 -t 2:00 sleep 120`

- **Flux CLI**
  
  - `flux mini submit -N2 -n4 -t 2m sleep 120`

- **Flux API:**
  
  ```python
  import json, flux, job
  from flux.job import JobspecV1
  
  f = flux.Flux()
  j = JobspecV1.from_command(command=["sleep", "120"],
                           num_nodes=2,
                           num_tasks=4)
  
  j.set_duration(120)
  resp = flux.job.submit(f, j)
  ```

What is flux?

- Flux is the future of resource management on LC clusters.
- Flux is hierarchical. Every flux ‘job step’ can be a full flux instance with the ability to schedule more job steps on its resources.
- Flux has a rich API that makes it easy to launch flux instances from within scripts.
- Flux can be used now on LC systems.
Now:

- RZalastor (only 4 nodes)

Coming soon (mid to late August):

- Corona (16 nodes this Thursday, then the rest in a couple of weeks)
- Tioga
You can also start flux in a Slurm allocation on any cluster:

[day36@rzalastor2:~]$ salloc -N4 --exclusive
salloc: Granted job allocation 220682
sh-4.2$ srun -N4 -n4 --pty --mpibind=off flux start
sh-4.2$ flux mini run -n4 hostname
rzalastor16
rzalastor15
rzalastor17
rzalastor14
Submitting a simple job

“flux mini” commands work similarly to sbatch, srun, etc:
% cat myjob.script
#!/bin/sh
hostname
date
flux mini run –N 2–n 32 my_mpi_app
% flux mini batch –N 2 myjob.script

Or you can try our Slurm wrappers:
% module use /usr/global/tools/flux_wrappers/modulefiles/
% module load flux_wrappers
% which srun
/usr/global/tools/flux_wrappers/bin/srun

Sbatch, salloc, and squeue wrapper scripts are also available
Where to find out more

- Man flux-mini, man flux-jobs, etc.
- https://github.com/flux-framework/Tutorials
- https://hpc-tutorials.llnl.gov/flux/
- Email lc-hotline@llnl.gov with questions, bugs, or to get in touch with the workflows team.
This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.