The FIFE project: supporting non-LHC Experiments at Fermilab

Ken Herner

HOW 2019

18 March 2019

Fabric for Frontier Experiments (FIFE)

- Standardize computing framework for non-LHC FNAL experiments
- Offer a robust, shared, and modular set of tools for experiments, including:
  - Job submission and monitoring
  - Workflow management software
  - Data management and transfer tools
  - Continuous Integration service
- Work closely with experiment contacts during all phases: from experiment workflow development and testing to production campaigns. Standing meetings; regular workshops and tutorials
  - Provide an on-boarding process for an experiment's production team

DES

NOvA

g-2

Some experiments currently using FIFE tools

FIFE Project Benefits

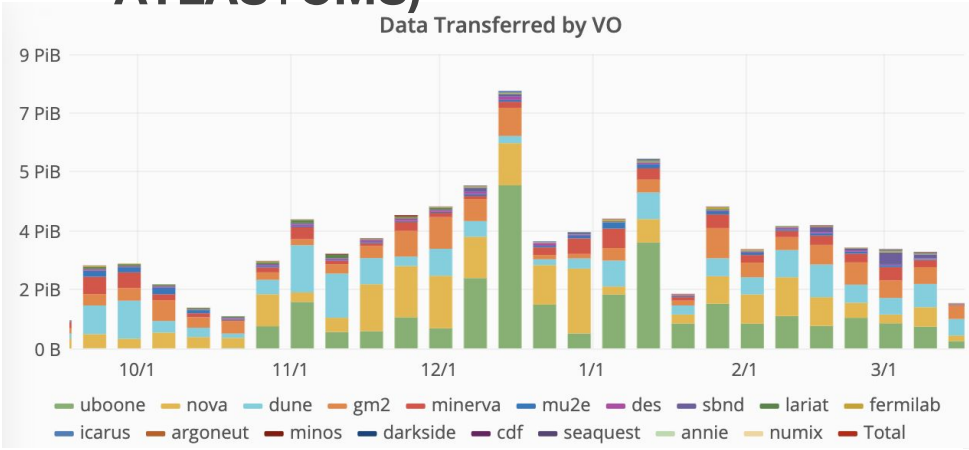
- Can leverage each other's expertise and lessons learned
- Allow to identify and address common problems and **provide common solutions**
  - Greatly simplifies life for those on multiple experiments (same toolset)
- Shift burden from experiments to gain access to new resources
  - OSG
  - European Grid
  - HPC
- Simplify transitions from one common service to another in transparent ways:
  - From FermiGrid to HEP Cloud
  - From GUMS & VOMS-Admin to a new authorization registry service (FERRY)
- FIFE Toolset includes
  - Job submission (jobsub), Workflow management (POMS)
  - Data handling and distribution
  - Continuous integration
  - Integration with new technologies and resources: GPUs and HPC
- Many experiments use ART, module software framework forked from CMSSW
- **Also of note: LArSoft, common framework for LAr TPC experiments**

FIFE Vision

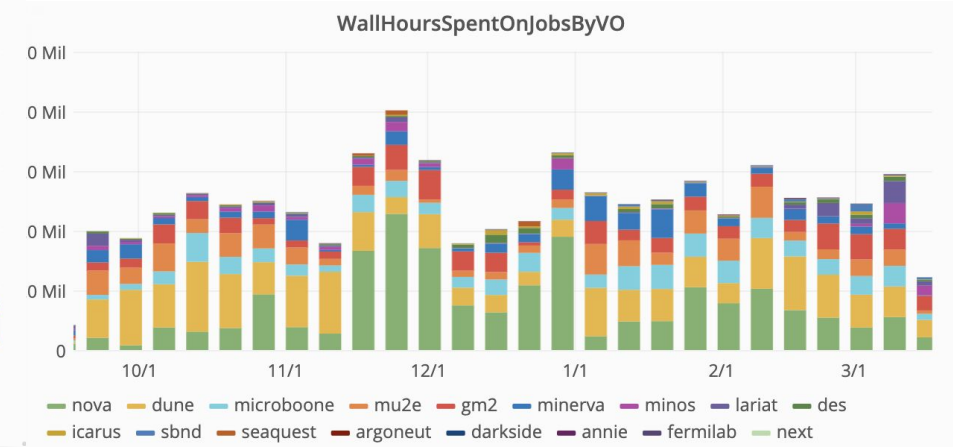
- Mature set of tools used by more that 600 users from different experiments and projects
- All tools are still “alive”: new features/improvements are being implemented.
- Layered architecture: could easily replace an obsolete tool with a better one while keeping UI the same
- Monitoring that encompass all the aspects: from individual file transfer to campaign execution
- Well defined on-boarding process for all new experiment.
- Unified users’ support

Supported Experiment Data and Job Volumes

- 35 experiments; more than 600 unique users
- 29 sites in 9 countries
- About 160K jobs per day; 170M hours per year
- **Combined numbers approaching scale of LHC (6-7x smaller wrt ATLAS+CMS)**



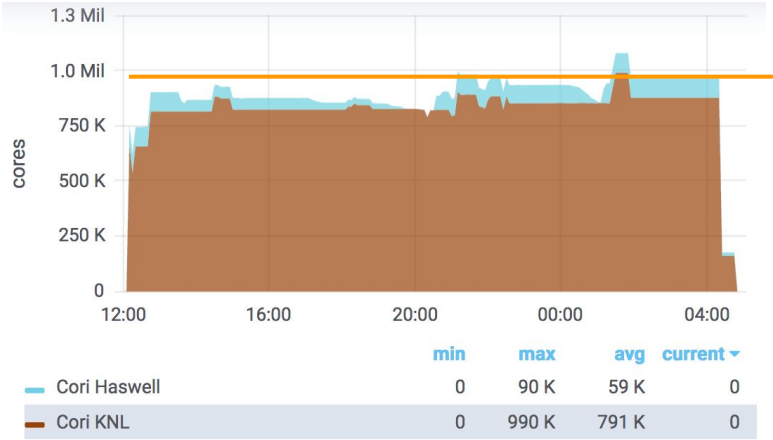
Total weekly data transferred by experiment, last 6 months



Total weekly wall time by experiment, last 6 months

HEPCloud Demonstration Success Story for long-baseline: NOvA @ NERSC

- NOvA experiment uses Feldman-Cousins procedure as part of final analysis.
- Using standard HTC resources, analysis of full dataset would have taken several weeks; impossible to complete in time without significantly expanding resources.
- NOvA was able to use 2M hardware threads simultaneously at NERSC via HEPCloud, achieving an overall speedup factor of about 150.
- Results presented at Neutrino 2018.



One million!

Short-Baseline Neutrino Program

Important precursor to DUNE

Consists of **MicroBooNE** (running), **ICARUS** (commissioning this fall), **SBND** (2020); all LArTPC

Similar near-far detector concepts as NOvA/DUNE

ICARUS very interested in adopting Rucio ASAP

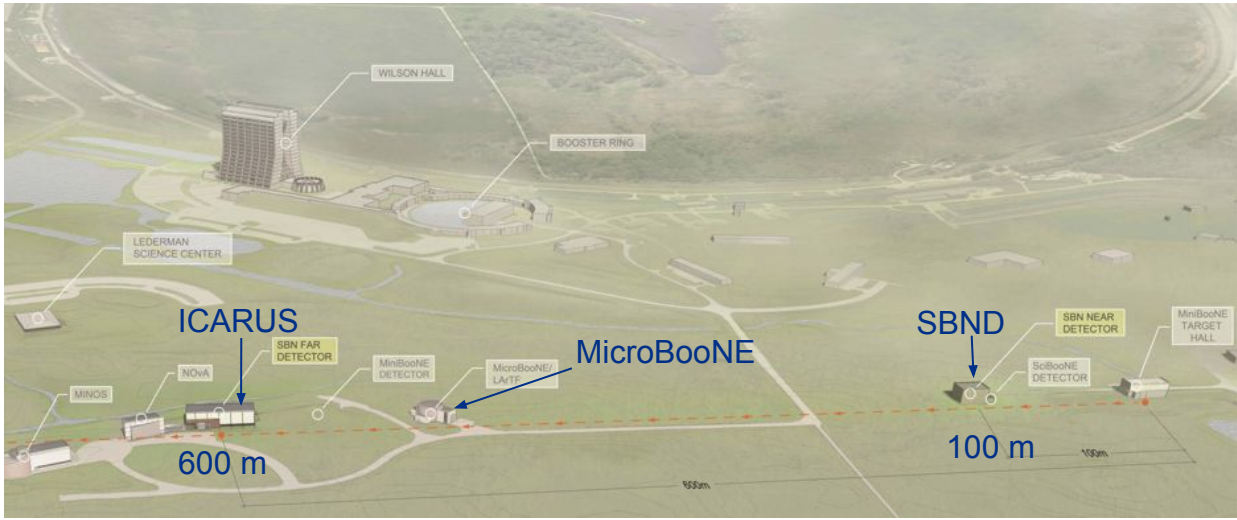
Common challenges:

Memory footprint

Learning how to effectively use multiple cores =>

Effectively using HPC resources

Storage/data model (all are multi-PB experiments)



Muon Experiments

g-2: Probe hints of anomalous magnetic moment of the muon

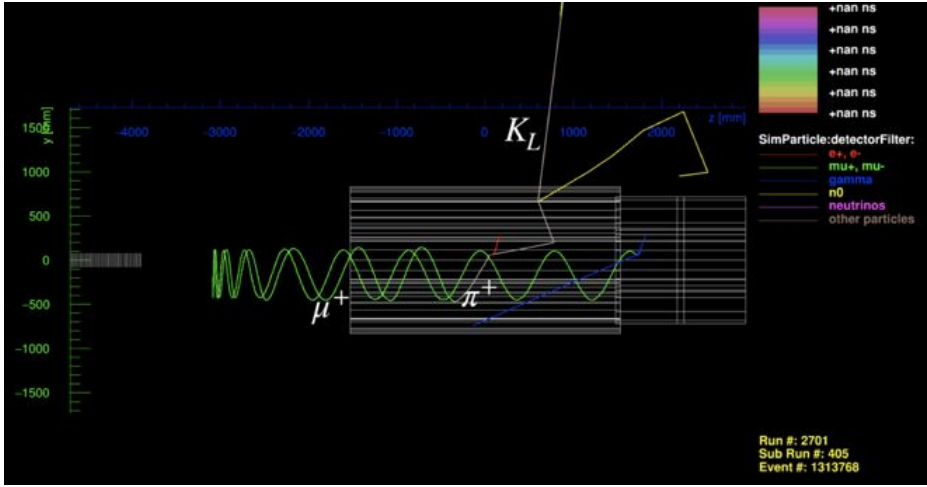
Currently taking data

Daily production CPU efficiency
Year to date

Mu2e: Look for muon -> electron decay (lepton flavor violation)

Design and construction continuing

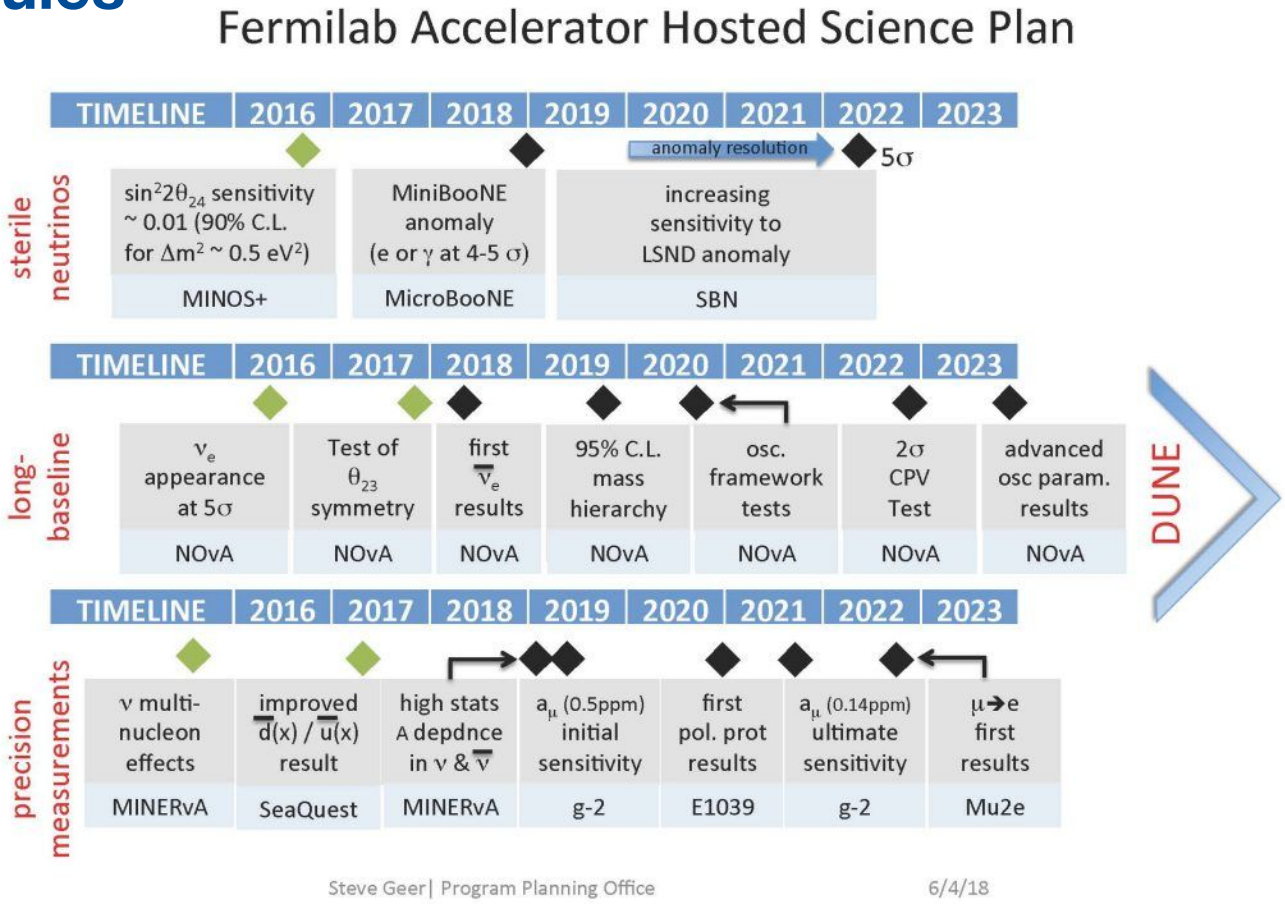
One of the earlier experiments to get onto OSG (50 M+ hours in 2015-2016)



Experiment schedules

Significant overlap between running experiments in 2019-21

Additional challenges include DUNE Near Detector design/construction



Tools to ease the crunch

Auxiliary File Delivery via OSG's StashCache

Example of the problem: several neutrino experiments use O(100 MB)"flux files" for simulating neutrinos in rock, etc.

- Several files are used per simulation job
- Total input is few GBs
- Not well-suited to CVMFS for several reasons (each job uses random subset; can thrash worker node cache)
- Transferring all the time sub-optimal (plus no caching at all that way)

StashCache service developed and widely used by OSG to solve just such a problem:

FNAL experiments using the service so far: DES, DUNE, Microboone, Minerva, Mu2e, Nova, SBND (all are Stash + CVMFS, access via /cvmfs/expt.osgstorage.org)

FIFE assists experiments with setting up and populating area.

One issue we've seen: *inconsistent ways of configuring such repositories between US and EGI sites.* Makes it take longer to get this working. Closer communication as far as Stash and CVMFS configs would help here

StashCache Usage

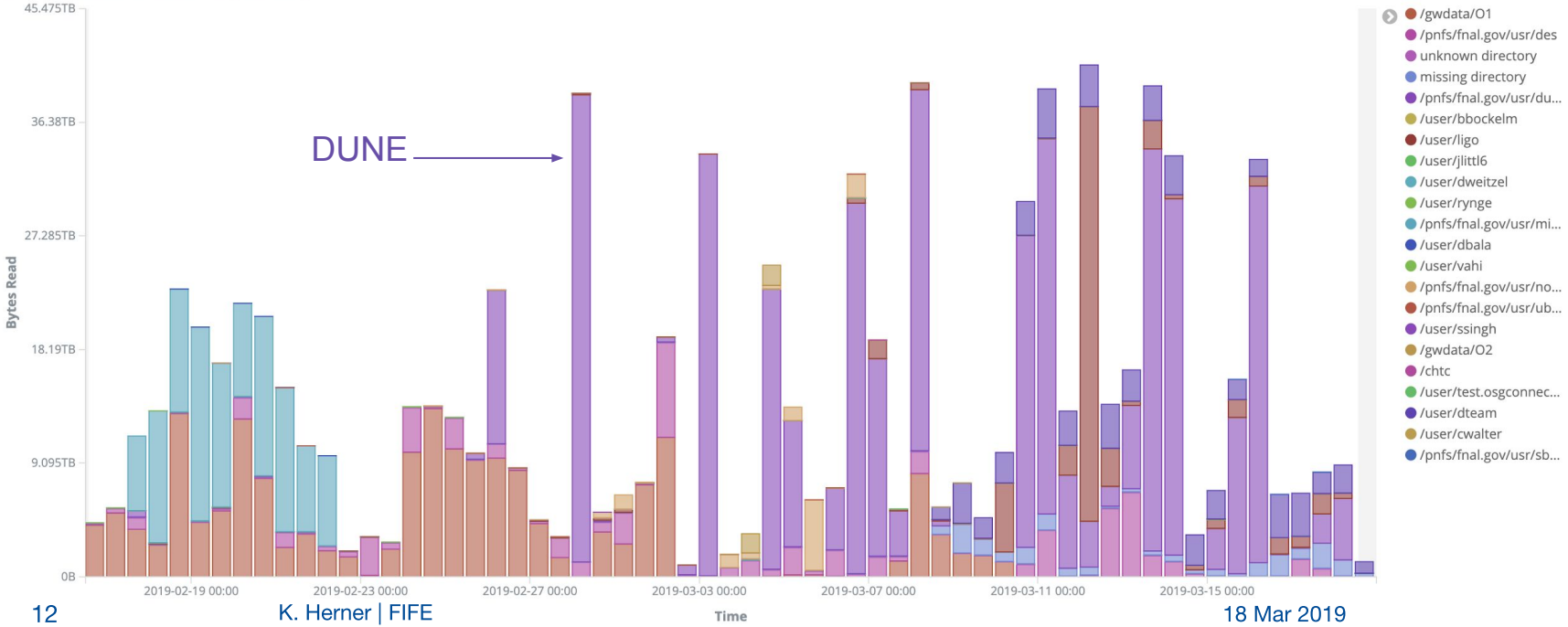
DUNE: largest StashCache user in past 30 days

3 of top 5 are FNAL expts

XRootD StashCache By Logical Directory

Directory	Bytes
/pnfs/fnal.gov/usr/dune	390.114TB
/gwdata/O1	180.212TB
/pnfs/fnal.gov/usr/minerva	99.352TB
/pnfs/fnal.gov/usr/des	63.204TB
/user/ligo	58.211TB

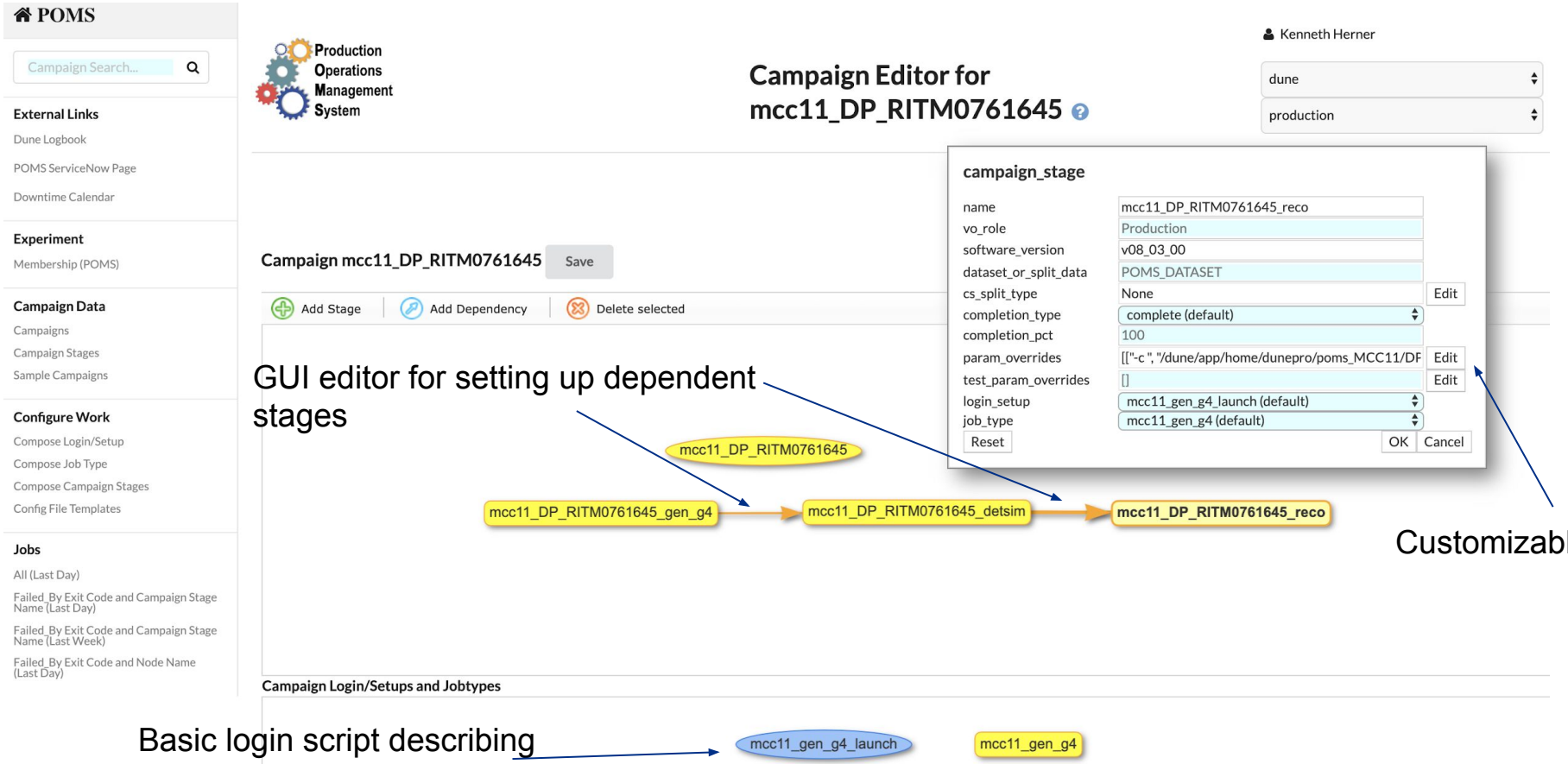
XRootD StashCache Transfers by Directory



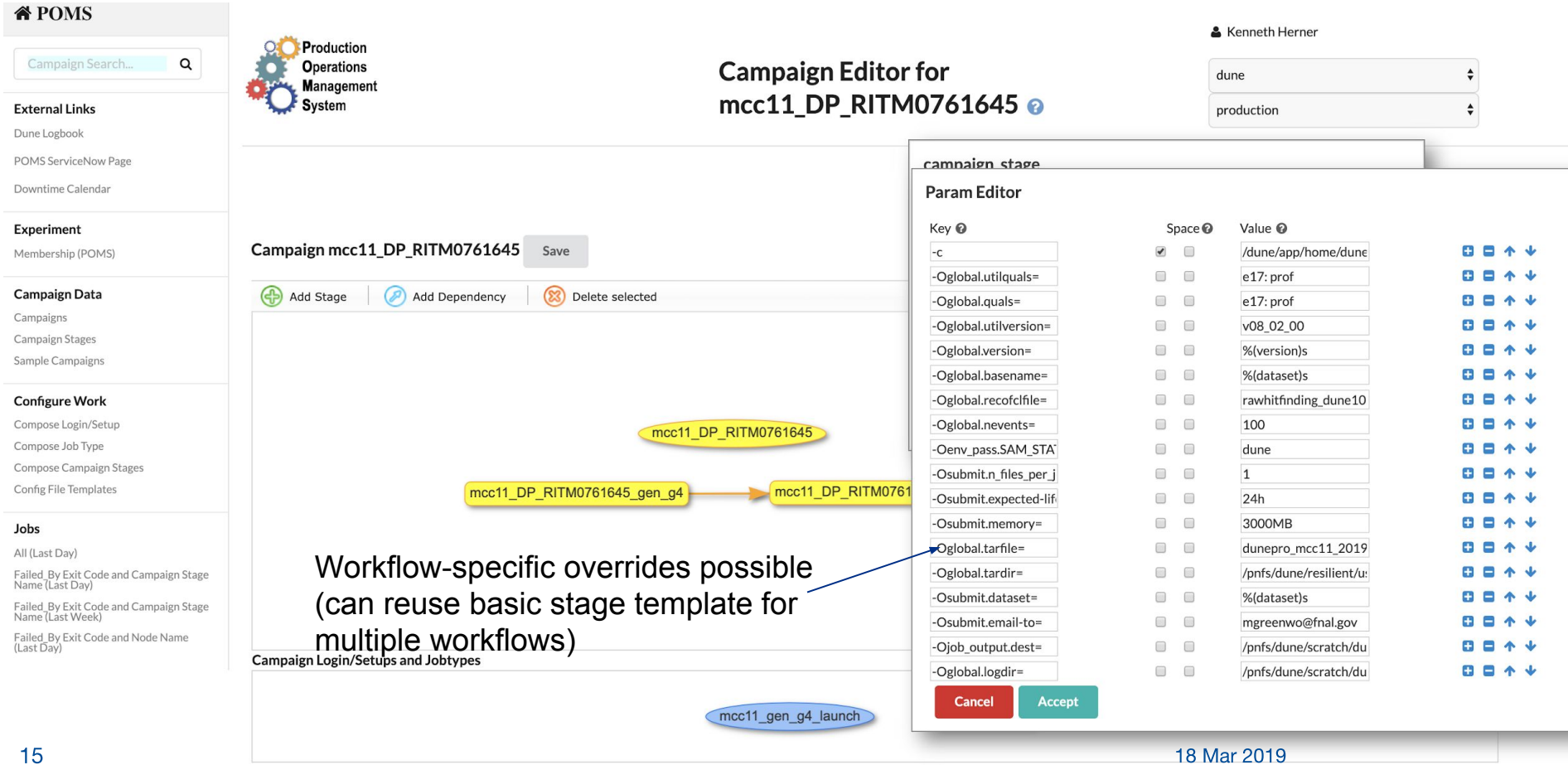
Workflow Management with POMS

- Started working on in-house Production Operation Management Service (POMS) suitable for production workflows for FIFE experiments (now opening to analyzers as well- same interface!)
- Tracks what processing needs to be done (“Campaigns”) and job submissions from a campaign
- Make automatic job submissions and set automatic recovery options
  - Allows dependent workflow stages
- Interface with SAM to track files processed by submissions and Campaigns
- Provides “Triage” interface for examining individual jobs/logs and debugging failures.

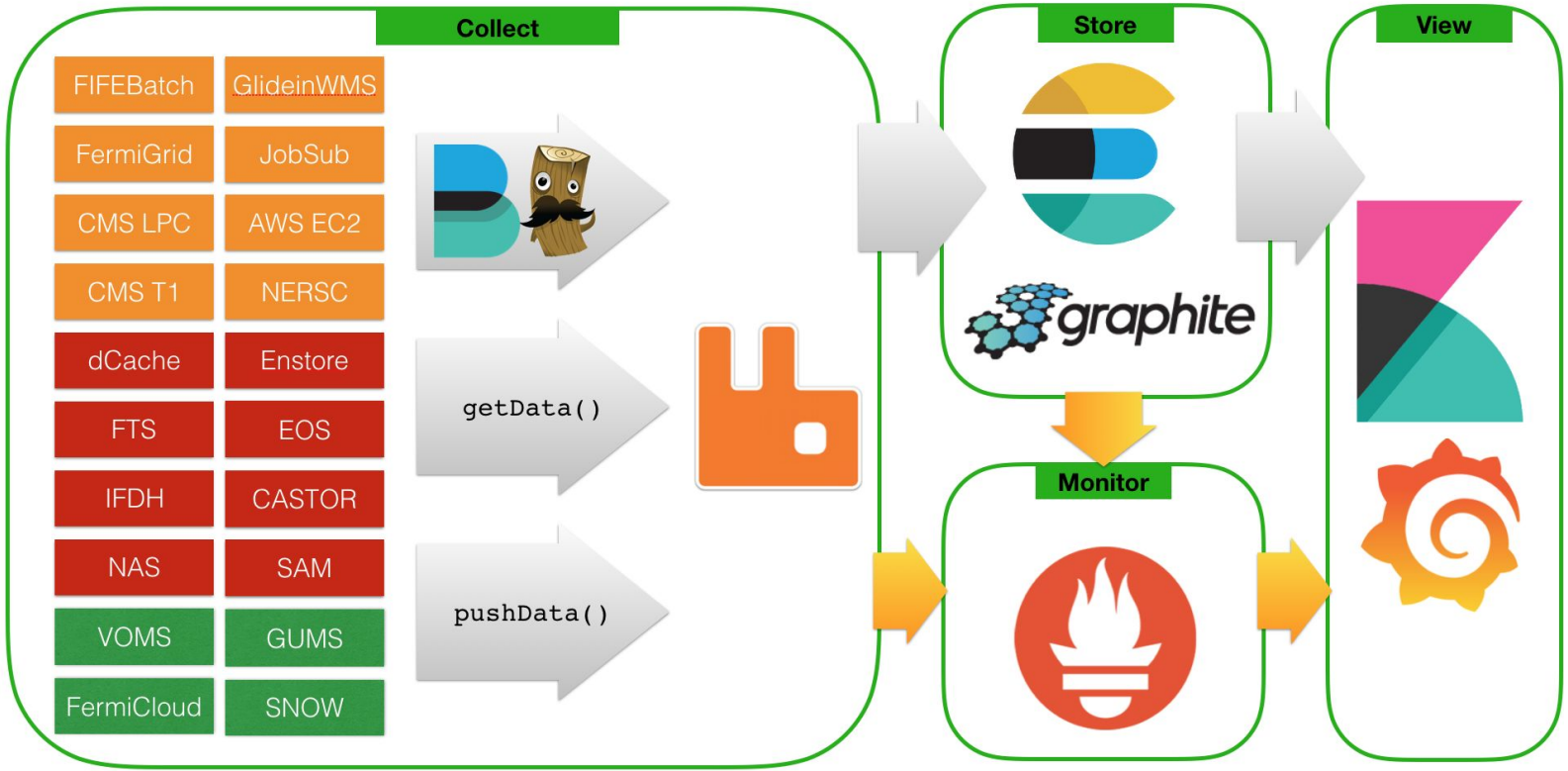
Workflow Management with POMS (2)



Workflow Management with POMS (3)

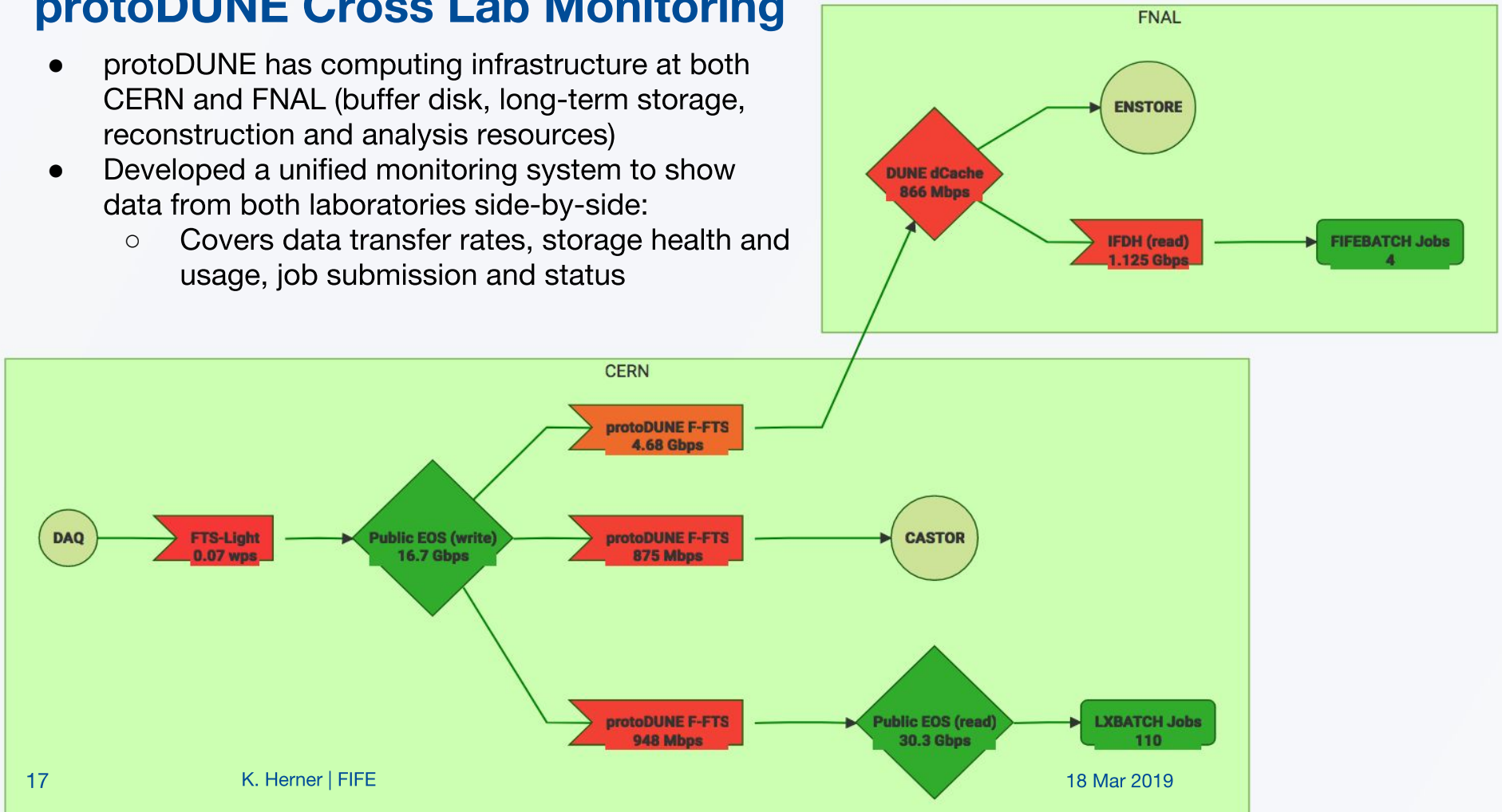


Monitoring (Landscape Program)



protoDUNE Cross Lab Monitoring

- protoDUNE has computing infrastructure at both CERN and FNAL (buffer disk, long-term storage, reconstruction and analysis resources)
- Developed a unified monitoring system to show data from both laboratories side-by-side:
  - Covers data transfer rates, storage health and usage, job submission and status



Short and Intermediate Term plans

- **Containers** (Docker, Singularity, etc.) are becoming more important in increasingly heterogeneous environments (including GPU machines). Help shepherd users through this process and create some common containers for them
- Seamlessly integrate dedicated, opportunistic, HPC, and commercial computing resources
- **Lower barriers to accessing computing elements around the world in multiple architectures**
  - Help to connect experimenters and computing professionals to drive experiment SW to increased multithreading and smaller memory per core footprints
  - Federated identity management (reduced access barriers for international partners)
- Integrate new technologies (e.g. Rucio)

Summary

FIFE project provides valuable service for FNAL non-LHC experiments. Makes life much easier for those on multiple experiments

Plans include increasing containerization and transition to other

Following existing approaches saves time, effort, enables easy international cooperation

Eagerly seeking common approaches as widely as possible in future projects (Rucio, HPC, IRIS-HEP, etc.)

BACKUP

Efficient User Code Distribution

- Until January, 2018 analysers were able to access their Bluearc code dev area
  - Was NFS mounted on all FermiGrid nodes.
- After Bluearc has been unmounted analyzers have started to use dCache scratch area to upload tarballs and stage them from dCache to worker nodes. It turned out to be a bad idea:
  - too much hammering on single pools when many jobs of the same type start
  - can completely block access to files owned by other users on a given pool
- Interim solution: stage tarballs in “resilient” dCache area (files replicated 20x on different pools):
  - Advantages: keeps the system running, doesn’t block users, more efficient.
  - Disadvantages: tarballs use 20x space; need careful cleanup policies
- Longer-term solution: looking into automated distribution via CVMFS.
  - User tarball created, hashed, stored on job submission server at submit time. Avoids repeats copying if contents unchanged

Off-line Production Shifter Dashboards

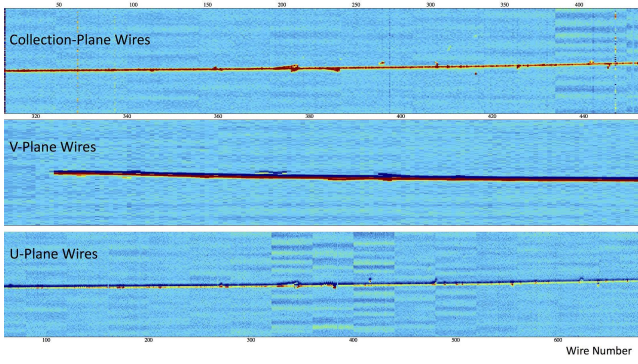
- Customizable dashboards for each experiment
- Display information experiments deem relevant to shifters in a single place
- Could drill down to more details

The FIFE tools in action: ProtoDUNE

ProtoDUNE using SAM, F-FTS, jobsub, GlideinWMS, POMS (workflow management), CI, Landscape

Using common tools makes it easy for people to contribute quickly

Still lots to do for ProtoDUNE of course

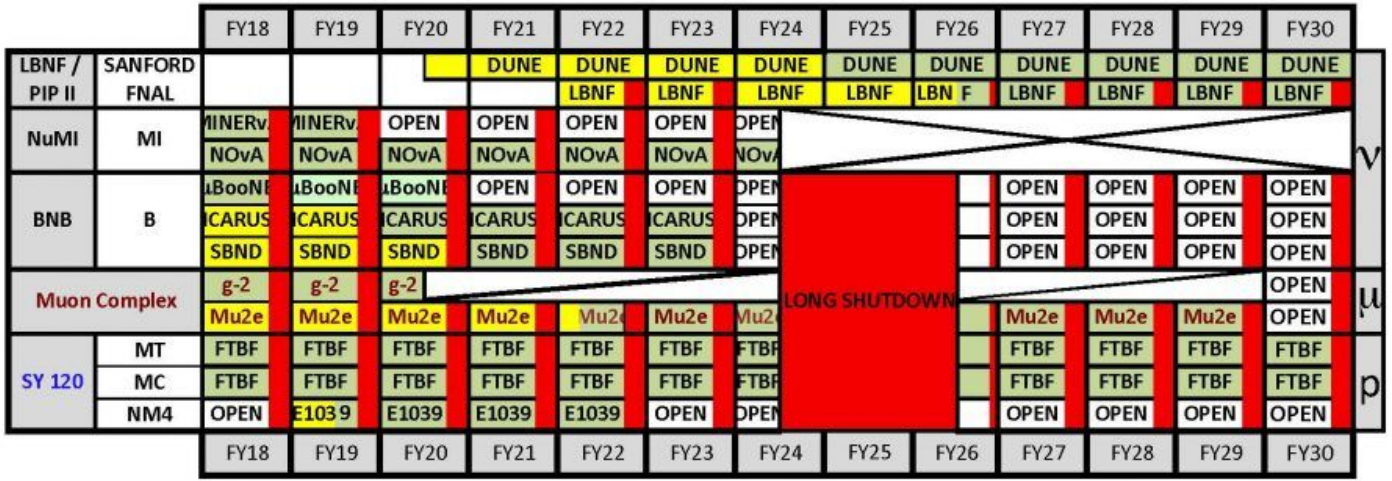


Red only because no jobs in queue ;)

Fermilab Experiment Schedule

Fermilab Program Planning 5-April-18

LONG-RANGE PLAN



NOTES:
1. Mu2e estimates 4 year running starts mid-FY22 after 18 months commissioning. Assume, with contingency, 5.5 years data taking.
2. DUNE: 1st 10kT detector module commissioned in FY24. Runs without beam FY25 to mid-FY26.
3. NOvA runs as long as possible [in the spirit of PAC Nov 2017].
4. Assume NuMI in nubar mode through FY19 - facilitates 12E20 POT for MINERvA [PAC Nov 2017]. Assumption may need revision.
5. Assume g-2 completed before Mu2e commissioning start mid-FY20. Very tight. Needs scrutiny.
6. Assume E1039 fully approved & commissioned by mid-FY19.
   Experiment estimates 2 yrs run. Add 1 yr contingency. [Stage 1 approval PAC June 2013, update July 2017]
7. FY19 and FY20 MicroBooNE running subject to future PAC review [PAC July 2017].

CI Example

Multiplatform Continuous Integration for LarCI

Build	Start Time	Build Type	setup_environment	checkout	build	unit_test	install	ci_tests
uboone_ci/498 (LArSoft uBooNE)	2018-07-05 12:11:37	d16 c2:prof	✓	✓	✗	?	?	?

Build	Start Time	Build Type	checkout	build	unit_test	install	ci_tests	ci_validation
lar_ci/2775 (uBooNE)	2018-07-05 11:48:16	slf6 e15:prof	✓	✓	✗	✓	✗	!
lar_ci/2774 (uBooNE)	2018-07-05 10:18:34	slf6 e15:prof	✓	✓	✗	✓	✗	!

Build	Start Time	Build Type	setup_environment	checkout	build	unit_test	install	ci_tests
uboone_ci/497 (LArSoft uBooNE)	2018-07-05 09:43:09	slf7 e15:prof	✓	✓	✓	✗	✓	⚠
uboone_ci/496 (LArSoft uBooNE)	2018-07-05 09:36:04	slf7 e15:prof	✓	✓	✓	✗	✓	⚠
uboone_ci/495 (LArSoft uBooNE)	2018-07-05 09:17:23	slf6 e15:prof	✓	✓	✓	✗	✓	✓
uboone_ci/494 (LArSoft uBooNE)	2018-07-05 09:06:57	slf6 e15:prof	✓	✓	✓	✗	✓	✓

Build	Start Time	Build Type	checkout	build	unit_test	install	ci_tests	ci_validation
lar_ci/2773 (uBooNE)	2018-07-05 08:57:44	slf6 e15:prof	✓	✓	✗	✓	✗	!
lar_ci/2770 (uBooNE)	2018-07-05 07:49:52	slf6 e15:prof	✓	✓	✗	✓	✗	!

Build	Start Time	Build Type	setup_environment	checkout	build	unit_test	install	ci_tests
uboone_ci/493 (LArSoft uBooNE)	2018-07-04 21:40:36	slf6 e15:debug	✓	✓	✓	!	✓	✗