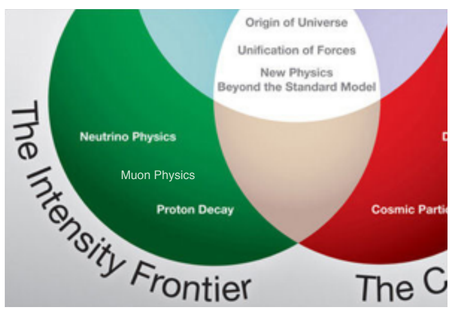


Taking Global Scale Data Handling to the Fermilab Intensity Frontier

Robert A. Illingworth, Marc Mengel, Andrew J. Norman, Adam L. Lyon
Fermilab Scientific Computing Division

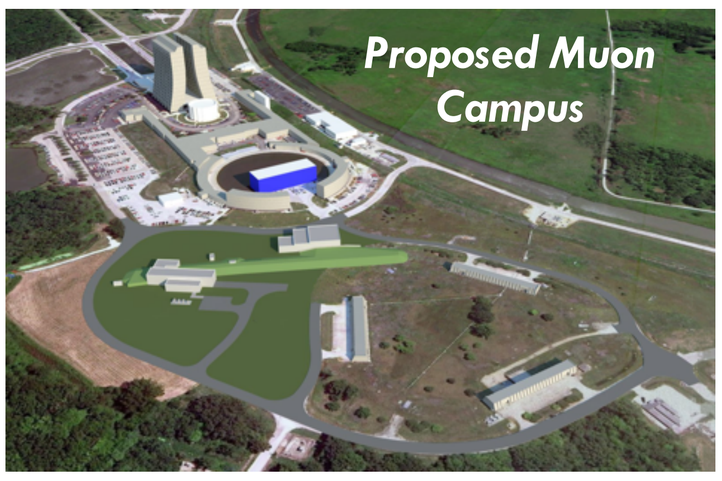
The problem: No data management

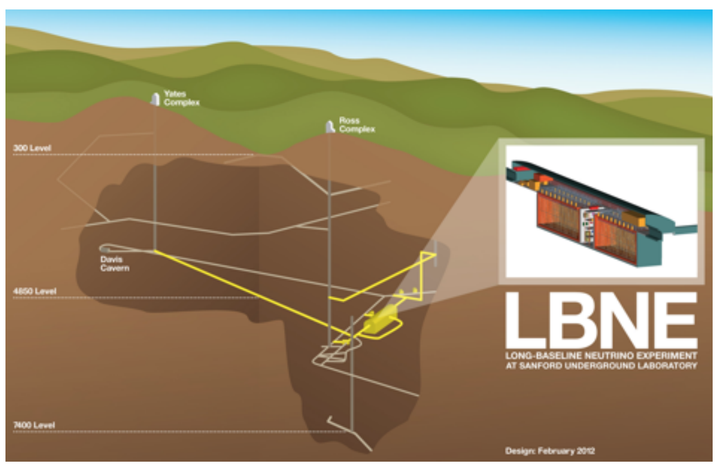
The Fermilab Intensity Frontier Experiments:

Operating: MINOS+, Minerva

Constructing: NOvA

Planning: Mu2e, g-2, LBNE

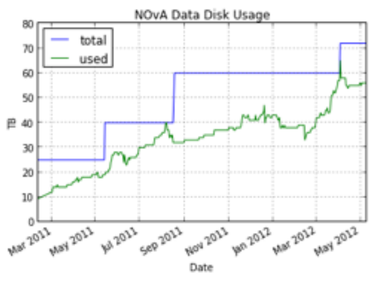


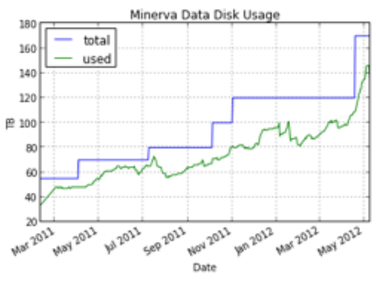


Current storage model:

KEEP MOST DATA ON DISK!

- Large Bluearc Network Storage
- Widely mounted on lab systems
- Accessible to FermiGrid workers
- 100,000s of files
- Only organized by directory tree
- No user-tools for archive storage and retrieval
- Inaccessible to offsite resources
- Notice "Just in Time" capacity planning in plots below



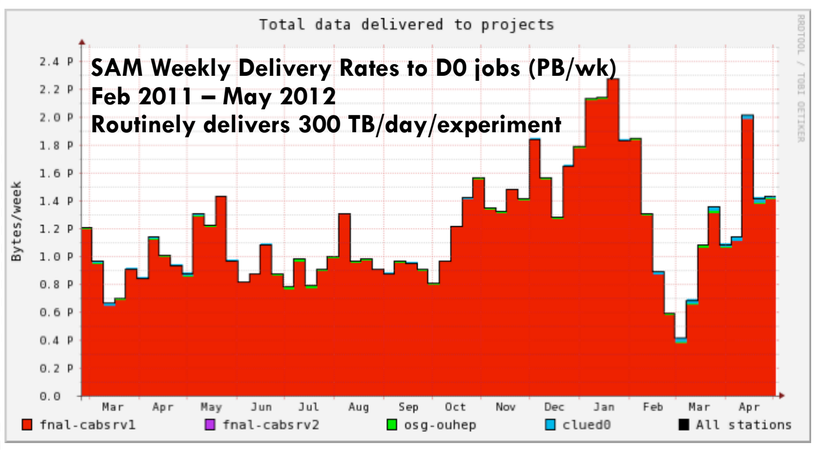


Our solution: SAM from Tevatron Run II

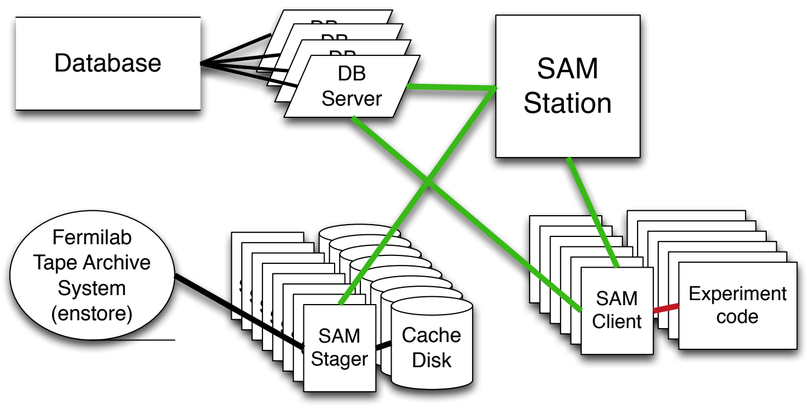
Use a comprehensive Data Management & Delivery System (SAM):

- **Extremely successful at Tevatron Run II Experiments (CDF & D0)**
- **In constant use for past 10 years**
- Metadata catalog, datasets
- Replica catalog
- Delivers data on demand from archival storage to caches world wide
- Automatically manages caches
- Closely integrates with Fermilab archive system (*enstore*)
- Closely integrates with dCache, SRM, gridFTP
- Client-server model with CLI, python client & C++ client
- Enables remote data processing

SAM Delivery Capabilities



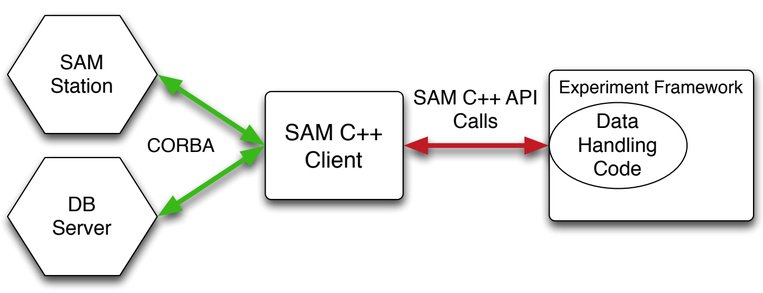
SAM Components



- Use Bluearc as cache
- Allows for running jobs offsite
- All advantages of a full data management system

Make Integration Easier:

Integration into D0 and CDF frameworks was difficult and time consuming

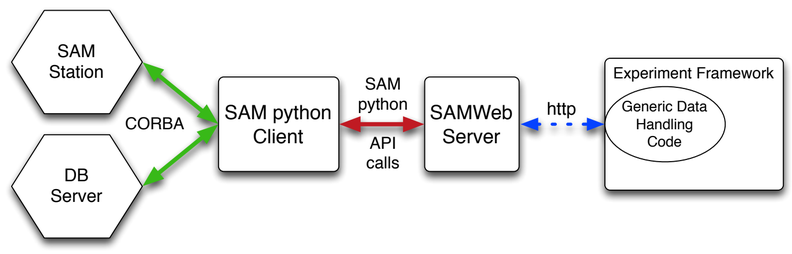


- Required many experts
- Coupled framework to SAM client
- Each job needs a SAM client instance – changes are hard to deploy

Improving the solution: SAMWeb and SAMfs

SAMWeb (prototype in use):

Improving integration with experiment's framework



Framework communicates to SAM client with http protocol via SAMWeb

- No SAM code in the framework so no coupling
- SAMWeb server at Fermilab
- Can scale up with more servers
- Client only deployed as SAMWeb backend
- No client needed at remote sites
- Easy integration, Easy running batch onsite or remote

Make Interaction easier:

Want to open a few files locally for testing or debugging

```
# To list files satisfying query
sam translate constraints --dim \
  "run_number 155945 and data_tier reconstructed"
```

- Neither intuitive nor easy for users
- Moving files to local disk is hard

SAMfs (in early design):

Make SAM behave like a file system with FUSE

- Navigate metadata as directories

```
ls -F /samfs/run
ERROR-supply-a-run-number-like-200000@

> ls -F /samfs/run/268211  # will list sub-runs
111/   112/   113/

> ls -F /samfs/run/268211/111 # will list data-tiers
raw/   reconstructed/

> ls -F /samfs/run/268211/111/raw  # will list files
raw_nu_268211_111_1.root* raw_nu_268211_111_2.root*
raw_nu_268211_111_3.root*
```

- Navigate pre-defined datasets

```
> ls -F /samfs/dataset
ERROR-supply-a-user-name@

> ls -F /samfs/dataset/lyon  # list my datasets
my-special-run-15593/  oscillated-nu-bars-5/
bad-tracking-examples/

> ls -F /samfs/dataset/lyon/my-special-run-15593/
reconstructed_nubar_15593_112_4.root*
reconstructed_nubar_15593_112_5.root*
```

- Load file to cache with "get"

```
> ls -F /samfs/run/268211/111/raw
raw_nu_268211_111_1.root* raw_nu_268211_111_2.root*
raw_nu_268211_111_3.root*

> ls -F /samfs/get/run/268211/111/raw # prohibit ls with get
ERROR-trying-to-list-directory-with-a-GET-path@

> cat /samfs/get/run/268211/111/raw/raw_nu_268211_111_2.root
# binary output here

> root  # Open file directly in root
TFile* f = new TFile("/samfs/get/run/268211/111/raw
268211_111_2.root")
# ... back to shell

> ls -F /samfs/run/268211/111/raw  # @ == soft link == cached
raw_nu_268211_111_1.root* raw_nu_268211_111_2.root@
raw_nu_268211_111_3.root*
```

SAM, SAMWeb & SAMfs:

A comprehensive data management system easy to integrate and easy to use and ready for the Intensity Frontier

Read the full paper by scanning this QR code

Fermilab
Fermi National Accelerator Laboratory
Office of Science / U.S. Department of Energy
Managed by Fermi Research Alliance, LLC