Data Access with HPC

DOUG BENJAMIN
Argonne National Lab

U.S. DEPARTMENT OF ENERGY
Argonne National Laboratory is a
U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC.

ARC Data Access

- Local storage on HPC is essentially a cache managed by ARC code
  - Cache is configured with watermarks – deleting last recently used files from cache
  - Jobs typically run from scratch space which is cleaned up when job finishes
- Data In
  - aCT pulls jobs from PanDA and sends them ARC-CE with required input files
  - ARC CE queries Rucio for files and downloads them
    - Data is copied with gridftp, xrootd to ARC-CE with what ever is preferred protocol
- Data out
  - ARC CE uploads the job output files to preferred grid storage – in Event Service case – CERN Object Store, Grid RSE otherwise

Harvester Data Access

- All Large HPC's (DOE and NSF) in US have multi Data Transfer Nodes
  - Globus is available at all HPC centers https://www.globus.org/
- Data In/Out
  - PanDA instructions Rucio to transfer files to/from RSE (Datadisk) associated with HPC PanDA queue
    - Titan (BNL-OSG2_DATADISK) , NERSC – ALCF (SLACXRD_DATADISK)
  - Harvester running on an edge node uses plugins used to transfer data to HPC local shared storage
  - Harvester plugins clean up old space (sweeper)
  - At Titan (soon to be decommissioned) – Harvester run edge node – use Rucio client code
  - ALCF and NERSC – Harvester run on login node and use Globus Python SDK to trigger transfer via Globus

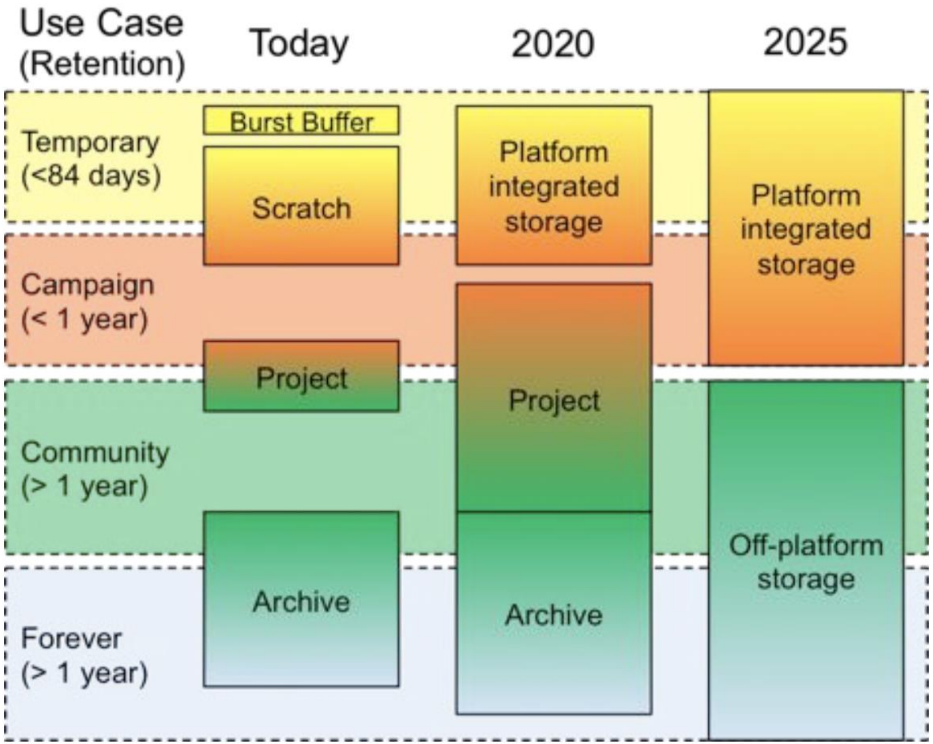
Going forward

- Is the one dual use Globus Endpoints/Rucio Storage Element robust enough for our current use?
  - We are working on second one at BNL.
  - We are also testing using OSRIS Object Store for Event Service output.
- Do we want to integrate the HPC Data Transfer Nodes into Rucio?
- When Event Streaming Service goes into production, how will the data flow into the HPC disks?
- What about when the HPC’s change the storage system?

Glenn Lockwood (NESC)

NERSC roadmap: Design goals

- **Target 2020**
  - Collapse burst buffer and scratch into all-flash scratch
  - Invest in large disk tier for capacity
  - Long-term investment in tape to minimize overall costs
- **Target 2025**
  - Use single namespace to manage tiers of SCM and flash for scratch
  - Use single namespace to manage tiers of disk and tape for long-term repository



Use Case (Retention)

Today

2020

2025

Temporary (<84 days)

Campaign (< 1 year)

Community (> 1 year)

Forever (> 1 year)

Burst Buffer

Scratch

Project

Archive

Platform integrated storage

Project

Archive

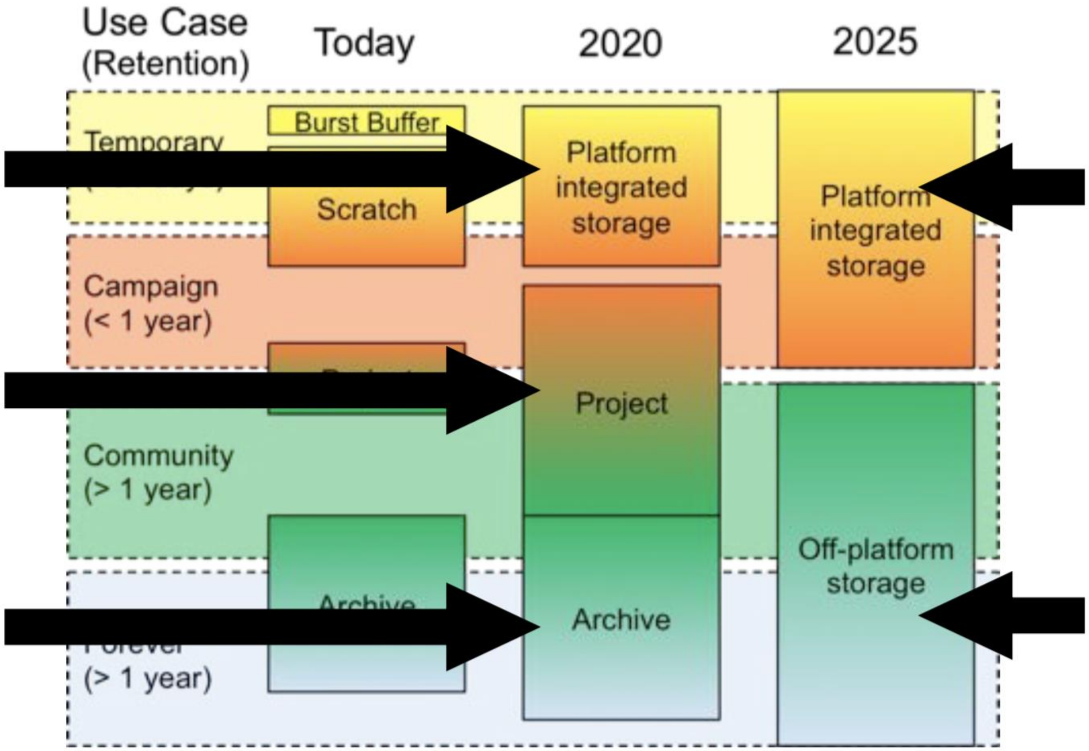
Platform integrated storage

Off-platform storage

https://indico.fnal.gov/event/18091/session/0/contribution/2/material/slides/0.pdf

Glenn Lockwood (NESC)

NERSC roadmap: Implementation



All-flash parallel file system on NERSC-9

> 150 PB disk-based file system

> 350 PB HPSS archive w/ IBM TS4500 +Integrated Cooling

Performance object store w/ SCM+NAND on NERSC-10

Archival object store w/ HDD+tape (GHI+HPSS? Versity? Others?)