FNAL SRM Implementation: Status and Plans

CERN Storage Management Workshop
05 April 2005
Version 1.0

Rob Kennedy (FNAL)
Timur Perelmutov (FNAL)
with contributions from
Abhishek Singh Rana (PPDG)

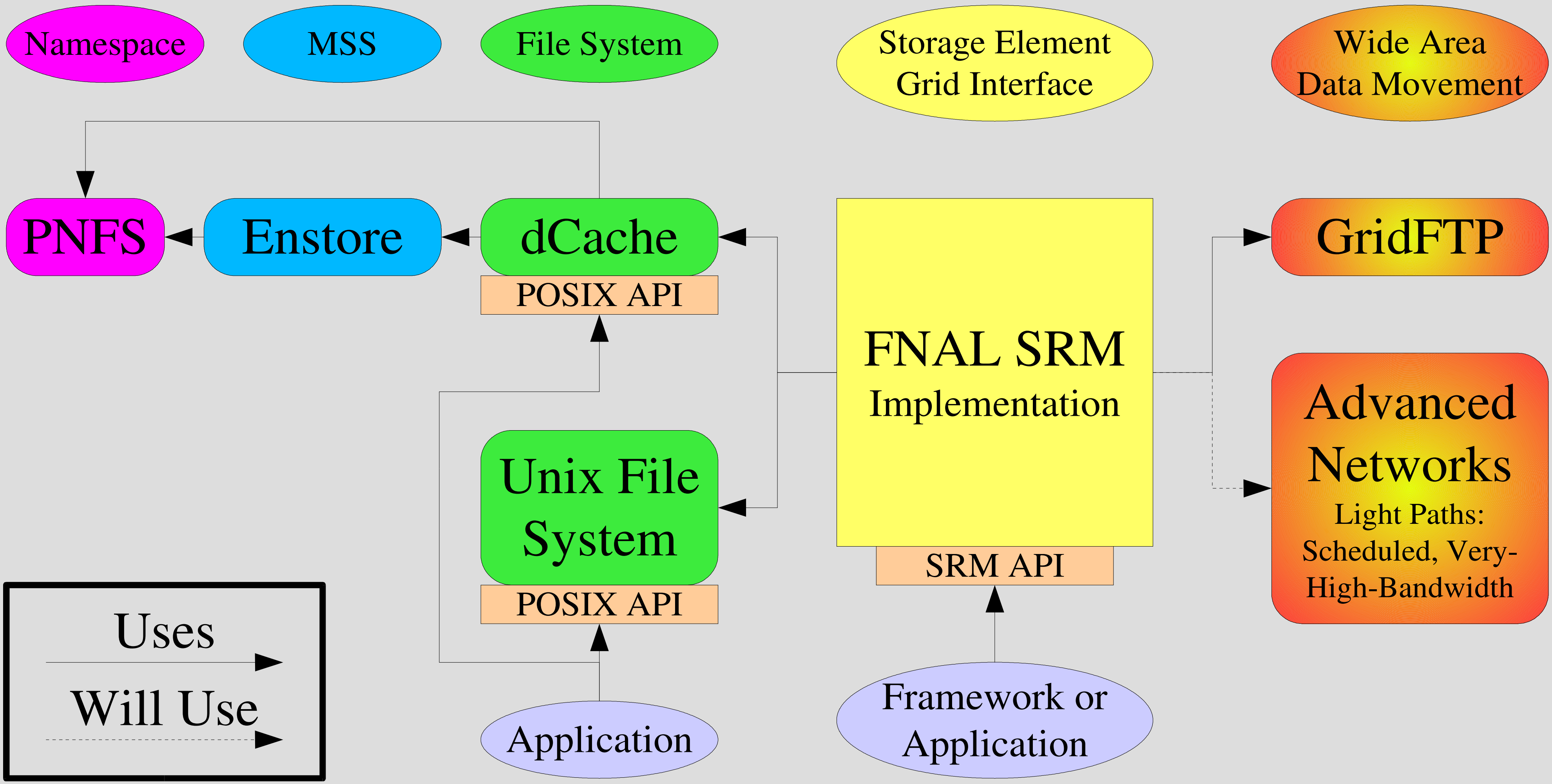
Outline

- ***FNAL SRM Imple. Overview***
  - Description, Experience
  - Development system context
- ***Features and Plans***
  - Implicit Space Reservation
  - SRM V2.1 API
  - Auditing, Accounting, ...
  - Client Shell Tools
  - VO Authorization Module
  - Parameterized interface
  - Light Path integration
  - V3.X API, GridFTP V2, ...
- ***Improvements***
  - DB-related performance
  - Scheduler edge-cases
  - Documentation
- ***Challenges***
  - Explicit Space Reservation
  - Multi-VO accounting
- ***Summary***
  - Status/plans summary
  - Lessons learned
  - FNAL SRM Link

FNAL SRM: Overview

- SRM V1.1 API + a little bit more
- Web Services-based, Java.
- Open source, BSD license.
- FNAL SRM implementation is separate from dCache, though has been developed to work with dCache
- Storage system underneath FNAL SRM is parameterized: Unix file system example/reference. Others....
- Inter-operation (v1.1 API level)
  - proven with CERN SRM-Castor.
  - detailed testing with LBL DRM.
  - more to come when FNAL SRM evolves to the v2.1 API
- SRM-dCache successful in SC2. CERN-T0 to US-CMS-T1 (to SRM-dCache and then to tape).
- Smaller-scale use >1 year on "public" FNAL dCache.
- Work driven by US-CMS T1, US-CMS T2, and LCG dCache so far.
- Also to be deployed on OSG.
- Expanding FNAL SRM team to 2+ FTEs (5+ FTEs overall in group)
- Current dev priorities:
  - implicit space reservation
  - v2.1 API
  - experience-driven improvements

FNAL SRM: System Context



Features and Plans 1

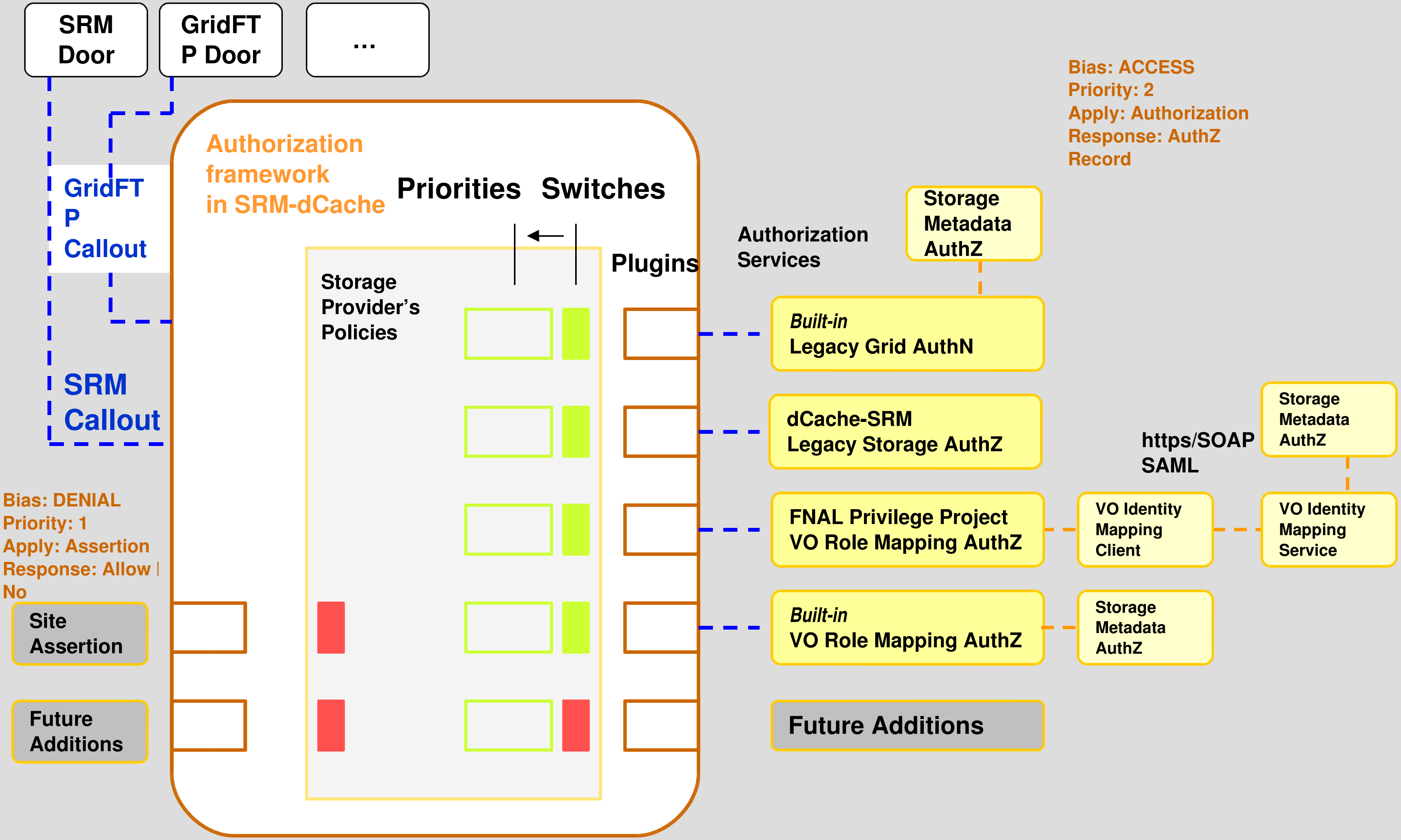
- Implicit Space Reservation: Initial version deployed. More to do.
  - Space reserved for a single file transfer, implicit in get/put/copy.
  - Reservation is not long-term, but just for time of transfer.
- SRM V2.1 API: Prototyped. Initial version being developed.
  - First feature we will deploy: srm-ls. High priority for our clients.
  - Expect this to be ready in 1-2 months, features released as available.
- Auditing, Accounting, Monitoring: SRM schema is “growing”.
  - Already have GRIS-GIIS path working: total space and space used.
  - Will coordinate with parallel efforts to define initial schema.
  - Need SRM DB interface, query tools, path to MonAlisa for full A&A.
- Broader Client Shell Toolset: Depends on our V2.1 API work
  - V1.1 tools already in LCG dCache distribution (Michael Ernst)
  - Preparing for distribution of V1.1 tools in OSG channel as well.

Features and Plans 2

- VO Authorization Module Integration: Work done by Abhishek Singh Rana in context of SRM-dCache.
  - Pluggable Authorization Mechanism architecture
  - Priorities and switches allow fine-tuning of authorization policy
  - Integrated with Site-Wide VO Authorization Service
  - This is the long-term Grid solution, adapting to existing practice. US-CMS T1 uses scripts and VO tools now, until this is ready.
  - Discussions begun: mapping high-level storage authorization policies to performant low-level mechanisms (ACLs, ...)
- Storage Authorization is superset of existing VO Authorization
  - Some storage systems (dCache, ...) need to know “where” in the SE namespace that a username-role is authorized to work, for instance.
  - Critical in a multi-VO SE, but still important in a VO-specific SE. One physics group should not be able to affect another's data storage.

VO Authorization Module Architecture

(slide by Abhishek Singh Rana)



Features and Plans 3

- Parameterized Storage: Initial version deployed with Unix f/s.
  - Example today is just a reference implementation, a little incomplete.
  - We may expand example, help adaptation to other storage systems
  - Possible fast track for others to a web services SRM interface. More interesting of course when FNAL SRM implements V2.1 API.
- Light Path Integration: Tracking FNAL advanced network dev.
  - FNAL SRM to allow Grid frameworks to use Light Paths, scheduled-access very-high-bandwidth networks: Starlight + Lambda Station.
- GridFTP v2: Will plan adaptation as soon as its available.
  - Adds distributed file system scalability, file integrity checking to FTP.
  - More flexibility in working with firewalls, private networks, proxies.
- SRM V3.X API: Will plan adaptation when API is available.
  - We are following this. Timur and Don Petravick in GGF SRM group.

FNAL SRM: Improvements

- DB-related Performance: SEs maintain much state information
  - We know how to make storage perform, but are still learning how to make database holding SRM state and meta-data more performant.
  - Consulting DBAs, starting to deploy improvements
  - Expect this to be an iterative process.
- Scheduler Edge-cases: SRM is more than a jacket on gridFTP
  - FNAL SRM contains scheduler for transfers, with persistent state.
  - Some edge-cases unanticipated, but happen in field tests.
  - Have designed how to address these, but not yet implemented.
  - Has been staff-time limited to date (should be less so soon).
- Doc Organization: We have much. Need to organize, extend.
  - Evolve doc to support future large-scale remote use.
  - More doc of internals design to promote collaborative development.

FNAL SRM: Challenges

- Explicit Space Reservation: *Its hard and complicated!*
  - Unknown distribution of space amongst files – not known beforehand how to allocate space on multiple disk nodes with varying free space.
  - In dCache, disk node may be selected on basis of file attributes (File Family, Storage Group), client network address, etc, all of which are unknown at the time of the space reservation.
  - Many and complex error conditions to handle. Unwinding needed?
- Multi-VO Auditing and Accounting: *Its an open-ended task!*
  - Every VO has its own idea of what the SE should track and monitor.
  - Our approach: start small and modest, grow schema carefully, maintain performance and inter-operability as much as possible.
  - Need for SE inter-operability: VOs in principle need to audit across many difference SEs. Add basic A&A interface to SRM API?

FNAL SRM: Summary

- FNAL SRM implementation proven in field
  - SRM-dCache for US-CMS T1 was successful in SC2
- FNAL SRM at API V1.1, going to V2.1
  - More features, functionality in 1-2 months
- Planning for wider use: SRM service, SRM client shell tools
  - SRM-dCache use is expanding: LCG, US-CMS, OSG channels.
  - Can be used with other storage systems too.
  - Expanding staff, and "external" contributors welcome (open source).
- *"Storage is hard."*
  - Storage maintains state where compute worker nodes do not.
  - Access to storage meta-data must be performant, not just access to data.
- *"Agree to APIs and protocols, compete on implementations."*
  - Healthy variety of storage solutions here at workshop. Customers win! 8^)
- FNAL-SRM link: http://www-isd.fnal.gov/srm