SRM concepts in StoRM

Luca Magnoni INFN-CNAF

SRM2.2 deployment workshop - Edinburgh

13 November 2007

- StoRM Overview
- Storage Area, Token and Description in StoRM.
- Approachable rules
- Free and available space
- Conclusion

StoRM overview

StoRM is a storage resource manager for disk based storage systems, implementing the SRM interface v2.2.

- It is designed to take advantage from **high performing cluster file system**, as GPFS from IBM and Lustre from ClusterInc., but it supports also every standard POSIX FS.
- It allows **direct access** (through the protocol *file://*) to the storage resource, as well as other standard grid protocol as *gsiftp* and *rfio*.
- Authentication and authorization are based on the **VOMS** credential.
- Permission enforcing are based on setting **physical ACLs** on files and directories.

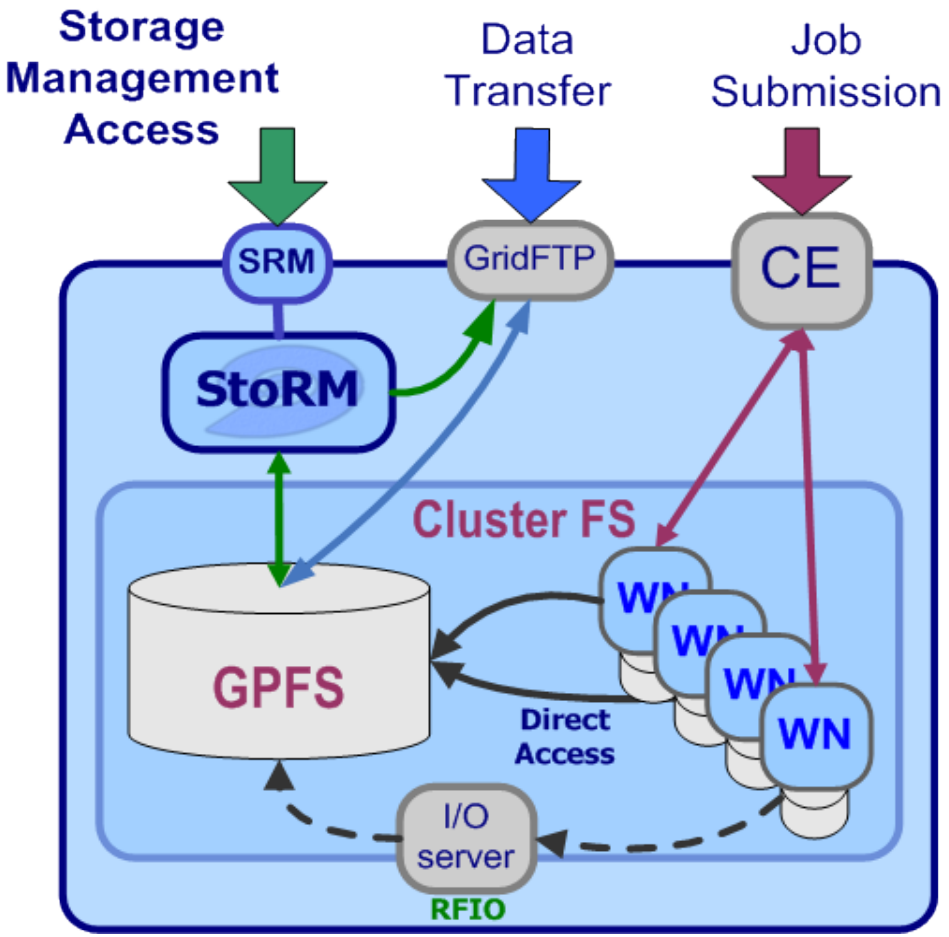
StoRM and cluster fs

- StoRM takes advantage of aggregation functionalities provided by dedicated systems, such as parallel and cluster file systems.
- A cluster file system allows large numbers of disks attached to multiple storage servers to be configured as a single file system.
- A cluster file system provides:
  - Transparent parallel access to storage devices while maintaining standard UNIX file system semantics.
  - High-speed file access to applications executing on multiple nodes of a cluster.
  - High availability and fault tolerance.

StoRM in a site

Current StoRM features:

- Support for different file system provided by a **driver** mechanism. Easy to expands.
- It's able to works on different file system type **at the same time**.
- Support for **file** protocol.
- As well as for other standard protocol as **rfio** and **gridftp**.



Storage Area, Space Token and Description in StoRM

StoRM support both **dynamic space reservation** and **static space reservation**.

- Dynamic space reservation are implemented relying on **file system advanced features** (as the case of GPFS).
- To define areas of storage reserved for particular VO (static reservation) the site admin have to act both on **StoRM configuration** and on **storage configuration**.

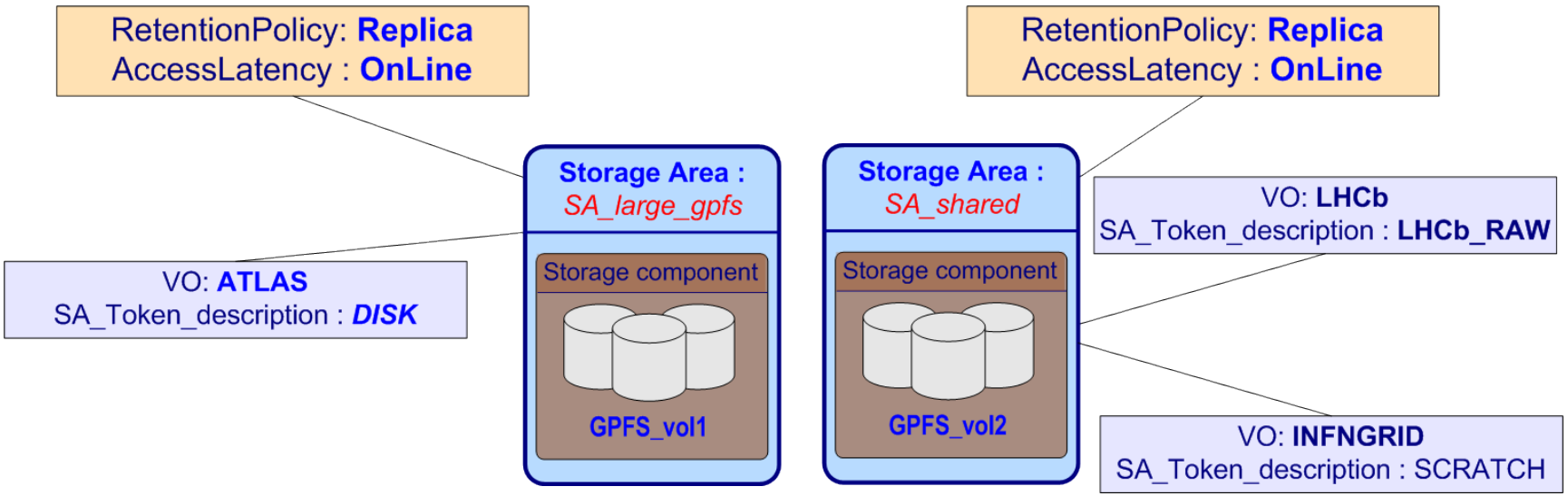
Defining Storage Area, Space Token and Description

When a site admin decide to reserve a **specific area of storage** for a particular VO he defines the corresponding **Storage Area** in StoRM:

- Editing the **StoRM namespace** configuration with all the related information:
  - **SA Path**.
  - **SA Token Description**.
  - **SA Storage class details**.
  - **SA Approachable rules**.

Once the **SA Description** is defined, StoRM creates an appropriate **Space Token** for data access operation.

Storage Area samples



Approachable rules

Approachable rules are regular expression in term of **VOMS FQANs and DN** that define a coarse grain access control on the Storage Area.

- All users: $< dn > * < /dn >$
- All user with VOMS credentials:
  $< dn > * < /dn >< vo - name > * < vo - name >$
- **DN rules** (all user named John):
  $< dn > CN = John < /dn >$
- **VO rules** (all users belonging to *infngrid*):
  $< dn > * < /dn >< vo - name > infngrid < /vo - name >$

Free and available space

- Site admin can use the file system **quota** to set the limit of usage of storage.
- StoRM can relies on **quota information** and **file system metadata** to get the remaining **free and available space**.
- Free and available space size for storage area are retrieved by SrmGetSpaceMetaData().

Conclusion

- Storage Area, Token and Description are defined through an appropriate StoRM **namespace configuration**.
- Currently (StoRM v1.3.18) each SA is **addressed by path, approachable rules and by Storage Area token**.
- Storage components depends on the underlying storage system configuration.
- StoRM allow to share a SA by different VOs.

StoRM

StoRM

Storage Resource Manager

http://storm.forge.cnaf.infn.it

Antonia Ghiselli
Alberto Forti
Luca Magnoni
Riccardo Zappi

Ezio Corso