

# StoRM on Amazon S3

Magnoni Luca INFN-CNAF

EGEE'08 Conference - Istanbul

25 September 2008

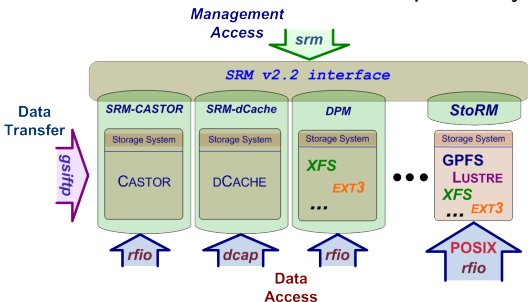
## Guidelines for this work:

- Investigation on emerging technologies as storage cloud solutions
- Understand if it is feasible a smooth integration in Grid
- How to leverage the cloud facilities

- The StoRM service
- Amazon Simple Storage Service
- StoRM on Amazon S3

# Storage Resource Manager (SRM)

SRM services manage space and files in Grid storage resources through a standard interface (**the SRM interface**) to hide storage characteristics and to allow interoperability.



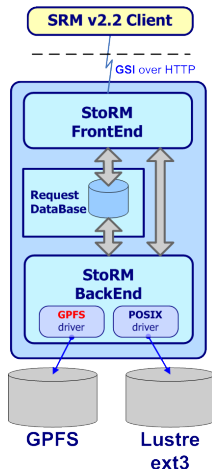
# StoRM key concepts

StoRM is a SRM implementation with these key concepts:

- **One SRM service for different storage systems.**  
Sites can change storage system without care about the SRM layer
- **Simple, configurable and highly scalable.**  
Easy enough to be the best substitute for a classic-SE and scalable enough to satisfy a Tier1-scaled centre
- **Efficient.**  
High performance on SRM requests execution
- **Secure.**  
Layered security mechanism, VOMS based and highly configurable

# StoRM architecture

- The **Front End** exposes the service interface and manages user authentication
- **Database** is used to store SRM request data and the internal StoRM metadata
- The **Back End** is the core of StoRM, it executes all synchronous and asynchronous SRM requests and interacts with file systems through a **driver mechanism**



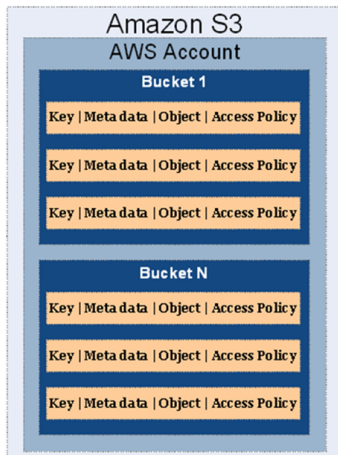
# Amazon Simple Storage Service (S3)

- *"Amazon S3 is storage for the Internet. It is designed to make web-scale computing easier for developers"*
- Simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web
- The idea behind Amazon S3 is to provide quality Internet-based storage in a scalable, reliable, fast, inexpensive and simple way
- Pay only for what you use

# Amazon S3 data model

S3 provides a straightforward flat object store model.

- Data is owned by **Amazon AWS account**
- **Bucket** is the container for Objects
- **Object** is the fundamental entity
- **Key** is the unique identifier for an object in a bucket





# Amazon S3 core concepts 1/2

- **Simple Amazon S3 Application Programming Interfaces (API)**
  - Uses standards-based REST and SOAP interfaces
  - Operation of Buckets, Objects and S3 service
- **Data access**
  - Default download protocol is HTTP
  - HTTPS also available
  - BitTorrent protocol is supported for high-scale distribution
  - *"Built to be flexible so that protocol or functional layers can easily be added. Additional interfaces will be added in the future."*
- **Data availability**
  - Amazon S3 achieves high availability by replicating data across multiple servers within Amazon's data centers
- Content-MD5 for consistency check

# Amazon S3 core concepts 2/2

## ● Authentication

- Access Key ID (a 20-character, alphanumeric string)
- Secret Access Key (a 40-character string)

## ● S3 Requests validation

- HMAC-SHA1 based signature attached to the request as parameter
- Support query string authentication

## ● Access control

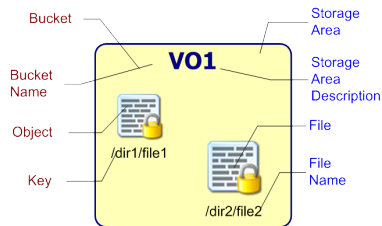
- ACL on Buckets and Objects
- Operations: READ, WRITE , ACL management
- Subject: AWS users, AWS group or Anonymous

# Grid data concepts in S3

- Site owns the AWS account, and it is only used by StoRM
- Buckets correspond to storage areas
- Objects are not hierarchical, but keys can be organized replicating the usual namespace adopted for file names
- Listing operation can be done on key prefix

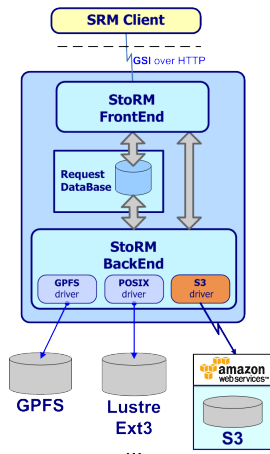
*AWS S3 domain*

*Data Management domain*



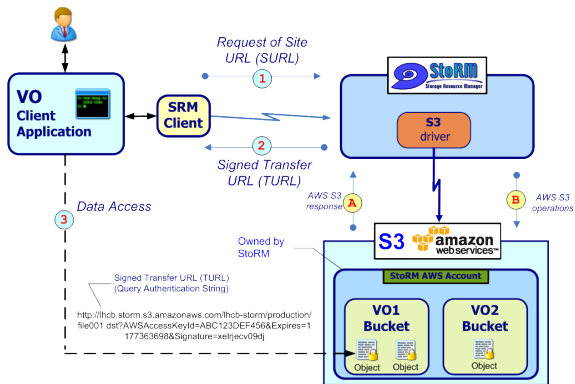
# StoRM driver for Amazon S3

- Integration requires new driver for Amazon S3
- StoRM interacts with S3 as a usual file system through the driver interface
- Driver uses S3 functionalities to reproduce classic file system capabilities

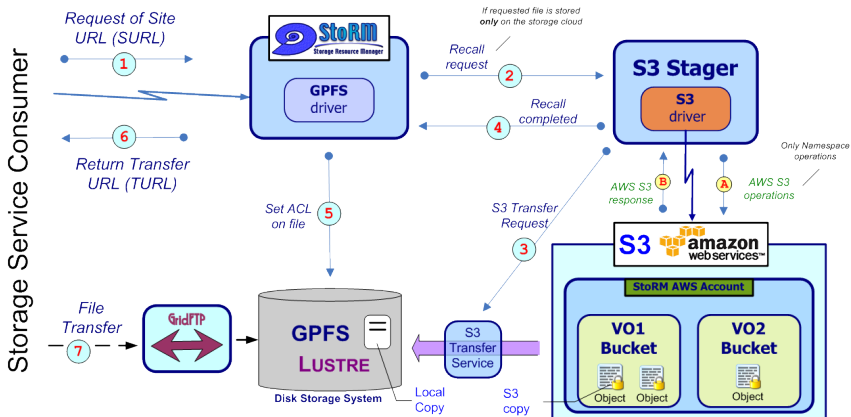


# Using S3 with StoRM: simple integration

- Namespace and Authorization are managed by StoRM interacting with S3
- Access to data is performed by user through specific client built on S3 API
- Authentication made via *query authentication string*



## Using S3 with StoRM: hierarchical integration 1/2



## Using S3 with StoRM: hierarchical integration 2/2

- S3 can be used to maintain a reliable copy of data
- StoRM drives stage/recall of data
- Data can be accessed directly via GPFS/Lustre/POSIX file system through standard Grid protocols and services as usual
- Use of multiple connections to S3 to maximize transfer throughput

# Conclusion

- Pro
  - S3 is a flexible and easy to use solution
  - Suitable for different integration in Grid
- Const
  - Different data access method supported by S3 and Grid applications, at the moment
  - Cost for the large volume of data existing in HEP
  - Lack of a well defined SLA in term of data retention



# Acknowledgements

- This work has been done by INFN CNAF within Research and Development group lead by Antonia Ghiselli
- People involved: S. Andreozzi, L. Magnoni and R. Zappi
- Suggestions are welcome.
- For news: <http://storm.forge.cnaf.infn.it>