

# SRM, valutazioni e prospettive

Riccardo Zappi, INFN-CNAF

Workshop CCR e INFN-GRID 2009

11-15 Maggio 2009, Palau



# Outline

- Una breve introduzione
- L'evoluzione di SRM:
  - dal Classic SE fino a SRMv2.2, con MoU e Addendum vari
- La storia insegna ...
- Problematiche attuali
- SRM Collaboration F2F meeting
  - Cosa cambiare, come e perché'

## Perche' questa presentazione (1/2)

Un workshop senza SRM?

Si discute spesso di SRM poiche':

- E' complesso, e piu' se ne parla piu' lo si conosce.
- Non soddisfa le esigenze degli esperimenti.
  - periodicamente c'e' una nuova versione di SRM
  - oppure un MoU
  - oppure un Addendum of MoU
  - oppure cambiano i requirements degli esperimenti

## Perche' questa presentazione (2/2)

- La prossima settimana ci sara' un SRM Collaboration F2F Meeting (OGF)
  - I partecipanti sono coloro che definiscono le specifiche SRM.
  - Non saranno presenti rappresentanti degli esperimenti.
  - I requirements degli esperimenti saranno riportati da Dirk Dullman
- **Raccolta** di suggerimenti da riportare alla collaborazione SRM

Di cosa stiamo parlando?

## Cosa e' SRM

- SRM e' una interfaccia web service per la gestione della risorsa di Storage (Un servizio SRM e' un web service che implementa l'interfaccia SRM).
- I servizi SRM sono componenti middleware per gestire i file e lo spazio in modo dinamico.
- I files non sono piu' entita' **permanenti**, bensì' **dinamiche**, ovvero possono essere rimossi dal sistema in accordo alle specifiche dell'utente.

# Le origini: Il Classic SE

- Il Classic SE e' un FTP server ottimizzato con autenticazione e autorizzazione di Grid.
- Un ottimo sistema per integrare la risorsa di storage in Grid
- La maggiori limitazioni del Classic SE:
  - Non c'e' la possibilita' di interrogare il servizio per informazioni sullo stato (i.e. spazio disponibile)
  - Non c'e' modo di una vera e propria gestione dello spazio di storage
  - Non c'e' una separazione tra nomi logici e locazione fisica dei files (complicazione nella gestione dello storage).



# L'evoluzione: Nascita di SRM

Nasce dalle esigenze principalmente della comunità HEP nella quale vi è una generazione di una grande mole di dati che devono essere distribuiti per poi essere richiamati per essere analizzati.

Era desiderato un servizio che permettesse :

- di riservare dello spazio di storage per evitare i fallimenti per esaurimento di spazio.
- la gestione dinamica dello spazio, e gestione di file per un utilizzo efficiente della risorsa.

# Perche' una interfaccia?

- Tanti tipi di storage e soluzioni differenti.
  - Necessita' di un sistema per nascondere le differenze e presentare lo storage per la sua vera, e forse unica, funzionalita': **memorizzare dati**.
- Per permettere interoperabilita' di servizi.
  - Esempio di Service Oriented Architecture (SOA)
  - Sotto questo punto di vista, probabilmente un successo.
- *Side effect*: Ha generato una competizione tra le differenti soluzioni. (bene, male?)





# L'evoluzione: SRM v1.1

## SRM v1 e SRM v1.1. [2001]

- Principali funzioni:
  - Get, getRequestStatus, pin, unpin
  - Put, setFileStatus, Copy
  - getProtocols, AdvisoryDelete, FileMetaData
- Principali problemi:
  - mancanza di una esplicita gestione dello spazio. Lo spazio era assegnato con politiche di default.
  - Mancanza di una gestione delle directory e del namespace.
  - Assenza di funzionalita' come l'Abort di richieste



# L'evoluzione: SRM 2.0/2.1

## SRM v2 e SRM v2.1 [2003]

- Introdotto il namespace con le funzionalita' per la sua gestione.
- Introdotto un sistema per la gestione dell'autorizzazione dei file (ACL).

Nessuna implementazione di SRMv2.1 e' stata mai implementata/utilizzata.

- Conseguenza: nessun feedback.

# L'evoluzione: WLCG Baseline Service e MoU

## Baseline Service Workgroup document [**Jun., 2005**]

- Nel giugno 2005 il Baseline Service Working Group ha pubblicato un documento dove:
  - "A Storage Element Service is mandatory and high priority".
  - Vengono raccolti i requirements degli esperimenti (termine nel febbraio 2006)
- CHEP 2006, Mumbai [**Feb., 2006**]
  - I rappresentanti apprendono nuovi requirements, e si delinea un MoU.
- MoU siglato nel **maggio 2006** a FNAL.



# L'evoluzione: il MoU di WLCG

La sintesi del MoU:

- Supporto del solo "static space reservation"
- Supporto dei soli "permanent files"
- Introduzione dei concetti di Storage Class: T0D1, T1D1, T1D0
- "No tape transition"

L'interpretazione di molte funzionalita' sono riviste in funzione del modello concordato.

## L'evoluzione: SRM v2.2

### Conseguenze dell'MoU, [**Apr., 2007**]

- Nascita di SRM v2.2 per accogliere le richieste di WLCG
- lo space diventa un concetto ortogonale al namespace, ovvero un attributo dei files.
- lungo processo di discussioni e di chiarimenti per appianare le tante mis-interpretazioni.
  - **6 release** delle specifiche SRM v2.2!!
- Interoperability
  - grande lavoro per rendere interoperabili le varie implementazioni

Finalmente SRM e' uno standard OGF. [**May, 2008**], tuttavia..

# L'evoluzione: Addendum all'MoU

Viene definito un Addendum all'MoU. [Jun., 2008]

- Perché l'Addendum?
  - ci sono alcuni casi d'uso che non sono ben gestiti.
  - In particolare l'Addendum è incentrato su:
    - **Space Protection** : Ovvero si vuole far sì che lo spazio sia utilizzabile solo dagli autorizzati
    - **Space Selection** : Un utente deve in qualche modo poter scegliere quale spazio utilizzare
    - **Space utilization** : Le VO vogliono poter sapere lo stato delle varie Storage Area

# L'evoluzione: Oggi

- Critiche al modello SRM derivanti da problemi legati alla complessita' del modello.
- Le implementazioni sono concentrate per migliorare l'affidabilita' e l'efficienza dei servizi.
  - Le richieste dell'Addendum hanno una prioritá' minore.

# Un sunto dell'evoluzione

- SRM e' nato per una gestione dinamica dello spazio:
  - File lifetime, gestione dinamica dello spazio
- Introduzione dei concetti di:
  - static space reservation, (size e qualita' predefinita) [MoU]
  - solo permanent file (!) non si vuole lasciare il controllo all'SRM. [MoU]
  - Ortogonalita' dello spazio rispetto al namespace. [Introdotta in SRM v2.2]
  - Space protection e Space utilization [Addendum]
  - Space selection [Addendum]

Queste complicazioni erano davvero necessarie?



## Prospettive di SRM (1/3)

Sul fronte SRM **l'INFN e' in prima linea**. Prossimo incontro (prossima settimana) (ri-)discuteremo alcuni argomenti, fra cui:

- Il layer security.
  - GSI e' un bottleneck. La lunghezza della chiave e' passata da 512bit a 1024bit.
  - I servizi per gestire questo cambiamento degradano di un fattore 4.
  - Un servizio web con GSI deve essere replicato per poter gestire lo stesso workload.
  - E il risparmio energetico di cui si parlava lunedì? Pannelli solari o 512bit?



## Prospettive di SRM (2/3)

Possibili soluzioni se non si vuole rinunciare alla "security" potrebbero essere:

- Caching dei certificati e calcolo finger print per il riconoscimento (https)
- Diminuire il nr. di richieste SRM:
  - Bulk operations.
  - Servizio di notifica piuttosto che di polling con le richieste di getStatus.

## Prospettive di SRM (3/3)

Rivisitazione del concetto di Space. [*Opinione personale*].

- Troppi gradi di flessibilita' genera troppa complessita'.
- "*keep it simple*":
  - Nessun utente utilizza pienamente l'ortogonalita',
  - anche se molti promettono che prima o poi potrebbero anche utilizzarla.
  - Per cosa? Quale e' la reale necessita'?
- Concetto di Spazio per avere:
  - space accounting, quota enforcement e protection.
  - Storage gerarchico: possibilita' di gestire flussi staging/recall su aree diverse.



# Conclusioni

- SRM ha ormai una lunga storia.
- L'INFN ne ha fatto parte ed ha contribuito molto.
- Commessi errori.
  - L'INFN non ne ha colpa. :)
- SRM vuole crescere per rispondere alle esigenze degli utenti.
  - non e' facile capire i loro requirements.
  - Possiamo sempre fare un altro Addendum.
  - Possiamo semplificare SRM.

## Verso un nuovo Addendum?