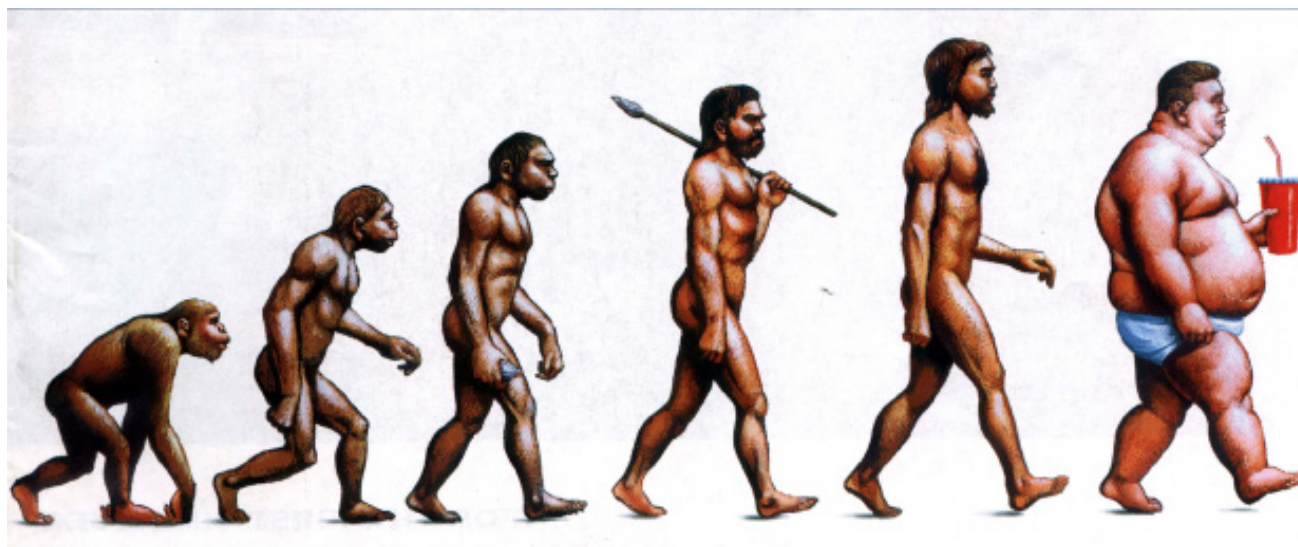


Evoluzione dello Storage?



Vincenzo Vagnoni

INFN Bologna

Workshop CCR e INFN Grid
20 Maggio 2010



Ma cos'è lo "Storage"?



◆ Low level fabric

- Dischi, controller
- Infrastruttura SAN
- Disk server, network

◆ Low level services

- Data access protocols and systems
- Tape management
- Databases

◆ Intermediate level services

- Storage Resource Manager (SRM)
- GridFTP

◆ High level services

- File Transfer Service (FTS)
- File catalog

Svariate componenti ad elevato contenuto di complessità

Svariati punti di fallimento strettamente interconnessi

Necessità di competenze molto variegata per la gestione ordinaria e la soluzione dei problemi

Architetture molto complesse, talvolta cervelotiche

Tant'è che... in WLCG lo storage ancora oggi ha grossi problemi



Che facciamo allora?



Evolution of WLCG Data & Storage Management

March 25, 2010

Evolution of WLCG Data & Storage Management

Summary of a first discussion on the strategy for evolution of the data and storage management in WLCG

- ◆ “The LHC experiment managements have expressed concern over the performance and scalability of access to data, particularly for their analysis use cases.”
- ◆ “It is anticipated that following this meeting a series of working groups to address particular technical areas would be set up, with the goal of producing incremental working prototypes to validate some of the ideas. These working prototypes should be of immediate use and help to resolve some of the shorter term concerns of performance and functionality. There must be incremental improvements to the current system.”
- ◆ “Note: It is not the intent to replace the working system that is currently in place during the first years of data taking, but to prototype solutions for the future, with a tentative timescale of 2013 for large scale use.”



Qualche (vaga) proto-idea...



- ◆ “A working model is to be far more network-centric idea than today. The initial hypothesis is to have a (few) large archival data repositories that are responsible for long term data curation, and a cloud of storage used as short term data caches with peer-peer technologies used to transfer data. Data may be accessed from the cache or across the network.”
- ◆ “A simplified model where the “tape” back ends are treated as truly archival [...] The disk storage at Tier 1 and 2 sites then should be seen as a cloud of disk caches with the assumption that data can be actively moved (and cached) so that work can run without the requirement that all the data that it needs be located in one place.”
- ◆ **La discussione vera comunque deve ancora iniziare, vediamo cosa uscirà fuori dal prossimo DM jamboree**



Che cosa vi racconto oggi?



- ◆ Non posso dirvi quale sarà il futuro, non ho la palla di vetro e non sono un docente
- ◆ Mi sono sforzato di identificare vari aspetti qualificanti che caratterizzano o potrebbero caratterizzare un sistema di storage evoluto ed efficace, mettendoli a confronto con l'intenzione di dare spunti di riflessione, e non soluzioni
- ◆ Procederò mettendo in ogni slide in risalto un singolo aspetto saliente
- ◆ Alla fine ci saranno comunque più domande che risposte

Basic storage models

DAS vs SAN

- ◆ **Fondamentalmente due approcci nei siti WLCG, ma vie ibride sono comunque possibili**
 - **Modello basato su Direct Attached Storage**
 - Server con disk array acceduti localmente che agiscono come “file server”
 - L’accesso ai file dai client avviene tramite network utilizzando protocolli custom
 - Il bilanciamento delle scritture dei file su diversi file server è effettuata mediante apposite macchine di frontend che agiscono da bilanciatori
 - Ogni file vive sui dischi un singolo server (a meno che non ci sia replicazione), cioè non c’è striping su server multipli
 - “nameserver” centralizzati tengono traccia su DB convenzionali di dove si trovano i file (o in alternativa si usano tecniche di discovery e caching)
 - **Modello basato su Storage Area Network**
 - I disk server accedono direttamente ai dischi attraverso la SAN
 - Modello particolarmente indicato per filesystem paralleli, ma può essere usato pur non sfruttandolo appieno anche con altri sistemi di accesso ai dati
 - Con filesystem paralleli è possibile fare striping dei file su più server, gestire failure dei singoli server in maniera trasparente, l’accesso è POSIX e non sono necessari DB convenzionali da mantenere



Approcci MSS

Tape-centric vs Disk-centric



- ◆ **Per quanto riguarda il mass storage, il modello tradizionale è tape-centrico**
 - La concezione dell'architettura del sistema ruota intorno alla gestione e agli accessi ai nastri
 - La parte disco è vista come una pura cache al tape
 - CASTOR è un classico esempio, ma anche dCache nasce originariamente con questo scopo, sebbene oggi sia usata anche come sistema puro disco
- ◆ **Il modello alternativo è invece disk-centrico**
 - Il building block fondamentale è il sistema di accesso al disco
 - Il tape è visto come un accessorio per l'archiviazione, possibilmente con accesso resilient ai file su tape
 - Il grosso dell'attenzione è sulle performance, stabilità e affidabilità del sistema disco
 - Sembra banale da dire ma, detto in parole povere, se un file è residente su disco la probabilità che un accesso prolungato nel tempo comporti un fallimento deve essere zero



Protocolli di accesso

Connected-POSIX vs unconnected-custom



- ◆ In sistemi quali CASTOR, dCache e DPM, la connettività con lo storage avviene direttamente a livello applicativo tramite librerie per protocolli custom come ad es. RFIO e DCAP
 - Sui worker node non va installato nulla (no kernel modules), i worker node sono neutri rispetto al sistema di accesso ai dati
 - Gli applicativi devono essere compilati con apposite librerie (problema minimale nel settore HEP dato che tutto è incapsulato in plugin di ROOT). Più importante è invece l'affidabilità delle connessioni per tempi lunghi
 - Ad es. mantenere un file aperto in lettura o scrittura per 24 ore è nei fatti problematico
 - Gli esperimenti cercano di ovviare copiando in locale i file di input all'avvio del job e scrivendo in locale l'output. Tuttavia questo è molto inefficiente per l'analisi dei dati, al limite addirittura impraticabile, ad es. quando si devono analizzare pochi eventi da un file che ne contiene molti di più mediante *event tag collections*
- ◆ Le soluzioni di tipo filesystem parallelo invertono pro e contro
 - In particolare, la permanenza della connessione ai server è garantita sotto allo strato applicativo, che non deve preoccuparsene



Struttura dei sistemi monolitica vs modulare



- ◆ **Soluzioni integrate come CASTOR e dCache hanno il vantaggio/svantaggio della monoliticità**
 - Integrano accesso ai dati, SRM, GridFTP
 - Installato il panettone, non occorre sapere nient'altro
 - Manutenibilità molto più difficoltosa, un unico team di sviluppo (tipicamente piccolo) deve mantenere e supportare tutto. Ma ancora più importante, la base di utenza di tutte le singole componenti rimane per forza di cose molto limitata
- ◆ **Soluzioni basate su componenti che vivono anche di vita propria hanno pro e contro opposti**
 - Tante cose diverse da installare, configurare, maggiore manualistica da leggere, maggiore probabilità di mis-configurazioni, necessità di conoscere più a fondo il prodotto che si usa
 - Base di utenza delle singole componenti più vasta, maggiore stabilità delle componenti
 - Possibilità di accedere a informazioni e supporto di comunità più vaste



Modelli di gerarchia hierarchical QoS vs plain



- ◆ Le tipologie di dati in uso possono essere di diversa natura e necessitare di differenti qualità di servizio
- ◆ Tipicamente oggi noi definiamo al massimo 2 QoS nei sistemi gerarchici
 - ONLINE (su disco) e NEARLINE (su nastro)
 - Nei Tier-2 poi c'è solo ONLINE
- ◆ In generale può essere molto utile avere una situazione più strutturata
 - Possono ad esempio esistere dati che hanno bisogno di essere acceduti con grande banda (leggi dati per job di analisi e di merging) e altri dati che non necessitano accessi ad alta banda
 - Regolare tutto il design del sistema di storage disco sulla fascia alta, oppure sulla fascia media può risultare nel primo caso in uno spreco di danaro, nel secondo in minore efficienza
 - Un modello gerarchico basato su disk pool a QoS differenti sulla base del dato, con meccanismi di migrazione resilient da un pool all'altro se necessario, porterebbe a maggiore efficienza e minori costi dell'infrastruttura
 - Es. banale: i file acceduti meno di frequente vengono migrati dal sistema sui pool costituiti dai dischi meno costosi e/o più vecchi mentre sui pool gold vengono mantenuti i file acceduti più di frequente



Licensing

Proprietary vs Open Source *et similia*



- ◆ Questo è un argomento molto delicato nel nostro mondo
 - Rispetto al settore privato siamo in generale molto più restii a puntare su soluzioni proprietarie
 - Ci sono buoni motivi ovviamente, ma anche uno cattivo, cioè che si tende a non dare il giusto valore al costo intrinseco del nostro lavoro
 - Siamo anche spesso portati a rifarci in casa cose che esistono già fuori
- ◆ Molto spesso il mondo HEP fa da apripista, perché si trova ad affrontare problemi di scala che la domanda privata ancora non ha raggiunto, ma la questione è: siamo proprio sicuri che nel settore dello storage sia ancora oggi così?
- ◆ Altra domanda, forse più importante: ha senso ed è sostenibile da un punto di vista economico effettuare una competizione con il privato quando offre soluzioni confrontabili se non migliori rispetto a quelle fatte in casa nostra?
- ◆ Infine: HEP è ancora o rimarrà ancora a lungo il caso d'uso più demanding in fatto di storage dei dati? Le soluzioni costruite in casa HEP sono esportabili in altri settori?

La questione SRM

- ◆ **SRM è forse uno dei pochi veri protocolli in uso in WLCG che sono stati disegnati in modo canonico**
- ◆ **Si discute spesso della sua complessità, ed in effetti il protocollo ha molte funzionalità con una logica non sempre elementare**
 - Questo può essere un difetto nel momento in cui si vuole fare adottare il protocollo al di fuori dell'ambito HEP
- ◆ **Tuttavia, in ambito WLCG, SRM funziona bene**
 - Visto che quello che c'è funziona, si dovrebbe forse ragionare su come poterla migliorare aggiungendo funzionalità nuove, piuttosto che cambiare strada
 - Comunque una cosa è certa: i problemi più gravi dello storage di WLCG non sono dovuti alla SRM, ma all'affidabilità dei sistemi sottostanti

Il problema dei file catalog

- ◆ I file catalog, in particolare LFC, sono usati dagli esperimenti per varie questioni, la più importante è la localizzazione delle repliche di un certo file logico esistenti negli storage element dei vari siti
- ◆ Il problema serio dell'attuale modello è legato alla perdita di sincronizzazione tra il contenuto degli storage element e il catalogo
 - Può avvenire e avviene alla grande, per tanti motivi, che repliche di un file logico esistenti nel catalogo sono state in realtà cancellate o sono andate perse sullo storage element, causando fallimenti degli applicativi che vanno a colpo sicuro a cercare un file che non esiste più
 - Può avvenire anche l'opposto, e cioè che repliche zombie rimangono sugli storage element, non esistendo però più nei cataloghi e quindi non potendo più essere accedute, continuando ad occupare spazio in modo indebito
- ◆ La sincronia tra catalogo e storage element è al momento completamente demandata agli applicativi di esperimento, non esistendo nessun vincolo di integrità intrinseco che garantisca in ogni istante la consistenza delle informazioni



Dove andiamo a parare?



- ◆ **Non lo so...**
- ◆ **WLCG comunque non sta iniziando a ragionare al super-mega-storage-galattico, ma ad un modello che dopo il 2013 si possa mettere concretamente in campo e a regime, ma...**
 - non è il caso secondo me di imbarcarsi in rivoluzioni, e non è proprio il caso di sbagliare
 - non manca molto tempo, ciò che esiste di buono va identificato ora e intorno a quello si deve costruire
- ◆ **Non sono eccessivamente ottimista sul fatto che si possa giungere a soluzioni largamente condivise, nel senso che i soggetti in gioco sono tanti, e le spinte centrifughe altrettante**
- ◆ **Secondo me l'INFN ha buone carte in mano**
 - Penso in particolare alle soluzioni di storage sviluppate e adottate al Tier-1, perché è il centro con l'infrastruttura più grande e che conosco meglio, ma anche ad alcuni Tier-2 e proto-Tier-2 e a diverse sezioni
 - Non c'è comunque bisogno di nascondere problemi e difficoltà, che esistono

Building blocks of the new system

- Disk-centric system** with five fundamental components

1. **GPFS**: disk-storage software infrastructure

2. **TSM**: tape management system

3. **StoRM**: SRM service

4. **GEMSS**: StoRM-GPFS-TSM interface

5. **Globus GridFTP**: WAN data transfers

