

Evoluzione StoRM, 27/4/2018

Modified 2018-04-29 Created 2018-04-27

Presenti: Gaetano, Luca, Vladimir, Antonio, Marco, EnricoV, Francesco, Daniele, Andrea, Davide

Luca: c'è un costo crescente in WLCG per la gestione dello storage. Sempre in WLCG c'è l'idea di riorganizzare lo storage per minimizzare il numero di copie dei dati che stanno su una infrastruttura distribuita "data lake".

Lo stato attuale: il nostro storage sta su software commerciale (GPFS, TSM) + prodotti in-house, compreso StoRM. A breve vorremmo fare test con il CERN (non abbiamo detto su cosa nello specifico).

Architetturalmente vogliamo mantenere storage di basso livello più interfacce standard.

Non è previsto in WLCG che SRM nei T1 venga dismesso, piuttosto lasciato solo come protocollo di accesso ai nastri. Ad oggi SRM serve anche nel front-end. Tuttavia FTS sta venendo sviluppato per gestire i token (senza l'uso di GridFTP) e quindi potrebbe interfacciarsi direttamente a IAM. Questo va in parallelo a quanto viene discusso in WLCG AuthZ per definire che cosa deve fare "il nuovo VOMS".

Attualmente con http non abbiamo supporto per la third party copy, che invece c'è in GridFTP. DPM e dCache (per EOS non è chiaro) hanno implementato una interfaccia che definisce come effettuare la 3rd party copy.

Richieste di estensione di funzionalità per StoRM emerse in riunione:

- Nel breve / medio termine servirebbe un'evoluzione di StoRM per integrare es. IAM per evitare di utilizzare certificati etc.
 - Vedi sotto per l'integrazione in nginx.
- Inoltre vorremmo includere protocolli di accesso come WebDAV in modo nativo.
 - WebDAV viene richiesto tipicamente da VO non-LHC, es. per http federations. Ad esempio lo usa Belle. Vedi sotto per l'evoluzione dell'attuale storm-webdav.
- Avere un'estensione di third party copy è un requirement per StoRM.

Gestione dello StoRM attuale:

- lo StoRM front-end va in crash frequentemente.

Per i prossimi 3 anni siamo coperti con le licenze GPFS. In ogni caso i sistemi di storage che ora abbiamo acquistato avranno dati sotto GPFS per i prossimi 5 anni. La conclusione dunque è che abbiamo bisogno di un middleware di mediazione su GPFS. Questo

middleware si ritiene che debba essere StoRM, il cui mantenimento e sviluppo va dunque supportato.

StoRM 2 dovrebbe supportare back-end multipli, ad esempio CEPH e GPFS. Non è chiaro se da un unico servizio StoRM uno possa indirizzare alcuni file che stanno su GPFS e altri su CEPH. CEPH avrebbe il vantaggio di una facile integrazione con infrastrutture cloud, oltre al supporto facile sia di object storage sia di posix.

Nella figura di StoRM 2 mostrata da Andrea, c'è nginx davanti a tutto per la gestione dell'autenticazione. Questo rende molto facile integrare IAM, similmente a quanto fatto ad es. da Luca per CHnet. nginx consente anche di integrare VOMS (già fatto). Ciò permette di non scrivere/mantenere storm-webdav in Storm 2 ma usare (da verificare i dettagli) nginx-voms + nginx-webdav.

Ad oggi sono stati completati gli Sprint 0 e 1. Nello Sprint 2 si sta implementando srmPing.

Con l'effort allocato oggi a StoRM sono previsti 1.5 anni per arrivare a un prodotto che rimpiazzhi lo StoRM attuale.

Questo è un tempo molto lungo. Attualmente tutto SD contribuisce allo sviluppo. Una possibilità per velocizzare lo sviluppo è quella di coinvolgere Carmelo Pellegrino, che al momento è sotto contratto EEE (potrebbe eventualmente passare su un altro tipo di contratto). Un'altra possibilità è che si potrebbe chiedere contributi sullo sviluppo al nuovo A/R che dovrebbe arrivare a breve in user support (Federico Fornari).

Alla fine di ogni Sprint (una volta al mese) si farà un report sullo stato di StoRM 2 alla riunione settimanale del Tier-1.

- Come introdurre StoRM in XDC?

Vedi slide di Daniele (T4.2 architecture). Nella figura mostrata l'INFN dovrebbe già contribuire nella parte "cloud.tsys.de" (le nuvole "C" - su questo lavora Antonio Falabella). Qui potrebbe entrare in modo relativamente facile StoRM. L'INFN non è invece attualmente presente nell'ovale "Managed Buffer Zone" (la nuvola "B").

Sarebbe anche importante inserire StoRM anche nel task legato al QoS (T4.1).

Patrick sembra prevedere StoRM solo per QoS e caching.

In T4.2 abbiamo 12PM. A parte i PM, dobbiamo però trovare le persone che operativamente lavorano su questa parte.

Andrea/Enrico danno disponibilità a seguire e lavorare su T4.1 e T4.2 per quanto riguarda StoRM.

SD fornirà una timeline sintetica per lo sviluppo di StoRM 2 con milestone mensili da ora fino a conclusione del progetto.