

Lo sviluppo del middleware per il calcolo distribuito dopo EMI

Valerio Venturi

Workshop CCR, Genova, 29 Maggio 2013

Il progetto EMI

- In EMI sono confluite le attività di supporto, manutenzione e sviluppo del middleware di Grid che erano state fatte nei progetti EGEE
 - Middleware gLite Argus, LCAS/LCMAPS glExec, VOMS, StoRM, DPM, LFC, FTS, WMS, CREAM, APEL etc, sviluppati da CERN, INFN, NIKHEF, CESNET, SWITCH etc
- In EMI sono entrati anche altri middleware
 - ARC sviluppato dalla collaborazione Nordugrid
 - dCache sviluppato da DESY
 - UNICORE
- L'ambizione di EMI, e l'auspicio della commissione quando l'ha finanziato, era di armonizzare e semplificare manutenzione e sviluppo del middleware di Grid

Il valore di EMI per utenti e provider

- Coordinamento delle attività sul middleware
- Procedure di supporto uniforme per tutti i prodotti
 - Canale unico GGUS, processo unico, livelli di servizio concordati con EGI
- Processo di manutenzione codificato per tutti i prodotti
 - Tempi per le request for changes
- Release unica per tutto il middleware WLCG/EGI
 - Annunci e scadenze unici
 - Repository unici
- Coordinamento della raccolta di requirements
 - Attraverso la partecipazione al TCB di EGI ed al GDB WLCG
 - Unico canale
 - Implementazione uniforme sui prodotti

Il valore di EMI per gli sviluppatori

- EMI aveva oltre 10FTE in attività di supporto al lavoro degli sviluppatori
 - Coordinamento di supporto e manutenzione, release management, sviluppo e gestione di tools, gestione del testbed, quality assurance, coordinamento dello sviluppo
- Integrazione
 - Assicurare che i cambiamenti in software inter dipendenti non rompano il sistema
 - Riunioni settimanali dell'EMT, gestione dell'integration testbed
- Processo
 - Ciclo di rilascio
- Tools
 - Continuous integration, repositories, issue tracking

Il valore di EMI per gli sviluppatori

- Coordinamento dello sviluppo
 - Per aree, compute, data, security, infrastructure
- Armonizzazione delle attività di sviluppo
 - Interfacce comuni
 - Librerie comuni

Attività INFN in EMI

- Attività di progetto
- Coordinamento del work-package SA1
- Release management
- Coordinamento dell'area compute

- Importante perché averlo fatto per tutti ci mette nella posizione migliore adesso per farlo per noi

Lo scenario post EMI

- EMI è finito il 30 Aprile 2013 e non ci sono al momento attività per il proseguimento del finanziamento di manutenzione e sviluppo del middleware di Grid
- INFN manterrà (con i fondi del progetto speciale IGI) i contratti delle persone che lavorano al middleware fino ad Aprile 2014
- Non c'è un contenitore per le attività sul middleware come c'è stato nei passati 12 anni (DataGrid/DataTAG, EGEE*, EMI)
 - Almeno fino alle prime call di Horizon 2020, dove comunque è improbabile che vengano ancora finanziate attività sul middleware Grid

Attori

- WLCG
 - Gradisce l'idea di della fine di una collaborazione coordinata, vuole coordinarsi direttamente con i product team
 - Indica il GDB come luogo di discussione tra vo, siti e product team
 - Ritiene che puntare sul rilascio in EPEL per la gestione dell'integrazione sia sufficiente
- EGI
 - Sondato i product team sulle modalità del supporto, se continuerà con i livelli attuali o verrà rilassato
 - Disponibile a prendere in carico le attività di coordinamento della release
 - Costituito un gruppo di lavoro (Unified Release Team, URT) che dovrebbe svolgere le attività che svolgeva l'EMT
 - Cambierà la composizione del Technology Coordination Board per sostituire al technical director di EMI i rappresentanti dei prodotti

Continuazione della collaborazione EMI

- SciencePAD
 - Repository per il software scientifico
- MeDIA
 - Forum per attività sul middleware per calcolo distribuito
 - Impegno stabile minimo, non ancora chiarito, partecipazione best effort dei team di sviluppo
 - Piu' chiaro cosa non farà di quello che farà: no heavyweight coordination, no release management

Cosa cambia per utenti e siti

- EGI dovrebbe assicurare che le procedure di **supporto** continuino ad essere uniformi e che sia chiaro il livello di servizio che ciascun product team si impegna a rispettare e che lo rispetti
- WLCG ed EGI dovrebbero assicurare l'uniformità del processo, delle decisioni delle priorità, e dei tempi della **manutenzione**
- EGI dovrebbe assicurare l'unicità della **release** del middleware
 - Nel breve periodo sarà assicurata da un accordo per la continuazione delle attività di release di EMI, CERN mette i repository, INFN mette il release manager
- WLCG, EGI (e MeDIA) dovrebbero assicurare i canali di raccolta **requirements** tra utenti, siti e product team
- “Dovrebbe” perché EMI era finanziato per queste attività, EGI non lo è, WLCG non ha problemi di risorse ma interesse solo verso le cose WLCG

Cosa cambia per gli sviluppatori

- Nessuno metterà a disposizione i **tools** necessari allo sviluppo
 - CERN garantirà per un periodo limitato la manutenzione dei repository
 - ETICS non è piu' supportato
- Nessuno farà attività di **integrazione**
 - Non ci sarà piu' un testbed dove vengono dispiegate le versioni candidate al rilascio e testato tutto il middleware in integration
- Nessuno farà la definizione ed il monitoraggio del processo di **manutenzione**
- Forse in MeDIA si potrà fare il **coordinamento dello sviluppo**
 - MeDIA è al momento molto indefinito e sarà comunque un attività spontanea e best effort

Lo sviluppo middleware INFN dopo EMI

- Dobbiamo attrezzarci per colmare le lacune lasciate dalla fine di EMI
- Erano attività finanziate in EMI, sono attività per cui dobbiamo reperire effort adesso
- Abbiamo avuto un ruolo primario in molte di queste attività in EMI ed abbiamo le capacità per farle
- Di come erano fatte alcune cose (l'issue tracking con Savannah, la continuous integration con ETICS) non siamo mai stati contenti e avevamo aderito per obblighi di progetto
- Ci sono servizi INFN che ci possono dare supporto in questo
 - e noi possiamo rendere a chi fa sviluppo nell'INFN

Issue tracking

- Savannah era un servizio molto vecchio che abbiamo continuato ad usare perché conteneva tutto il middleware gLite
 - In realtà per VOMS e StoRM abbiamo già usato altro fin da metà del terzo anno
- L'INFN ha un servizio nazionale per l'issue tracking basato su Atlassian JIRA, che è la migliore scelta disponibile al momento
- Tutto il middleware Grid INFN sarà tracciato con JIRA
 - VOMS e StoRM quasi completamente migrati, CREAM e WMS seguiranno
- Oltre il semplice bug tracking che facevamo in Savannah
 - Supporto per la gestione dei contenuti delle release
 - Supporto per la gestione dello sviluppo

Repositories

- Per alcuni mesi saranno ancora disponibili i repository di EMI
- Fino a che c'è EGI saranno disponibili i repository UMD
- Noi avremo i nostri repository, per tutto il middleware sviluppato da INFN e per ciascun prodotto
 - Del prodotto contenente la edge version, per chi vuole testare gli ultimi sviluppi o dispiegare istantaneamente un bug fix
 - Del prodotto contenente la release candidate, per il passaggio alla certificazione
 - Del prodotto contenente la release ufficiale, per gli utenti finali (staged roll-out)
 - Gli ultimi due saranno disponibili anche per tutto il middleware INFN

Continuous integration

- ETICS lo abbiamo sempre digerito perché unica soluzione disponibile al momento
- Adesso ci sono soluzioni mature e molto migliori, Jenkins
- Abbiamo cominciato il setup di una nostra infrastruttura di continuous integration e la migrazione dei prodotti
 - Cominciato con un deployment minimale di Jenkins su un pugno di macchine, adesso costruendo un setup molto più stabile
 - VOMS e STORM già migrati, WMS e CREAM in corso
- Questa attività è costata e costerà ancora molto effort in setup, ma ci porterà molto più avanti di quello che ci dava EMI
- L'obiettivo è di avere una infrastruttura di continuous delivery che ci permetta di abbassare drasticamente il tempo di rilascio dei prodotti

Continuous delivery

- Ogni cambiamento del codice (commit) triggera il build del software
 - Il fallimento del build viene notificato immediatamente a chi ha fatto il commit ed agli altri specificati nella configurazione
- Il build triggera la costruzione di un repository yum
 - E' sempre disponibile (link permanente) un repository yum da cui installare l'edge di un prodotto
- La costruzione del repository triggera i test
 - Deployment test, da macchina pulita (clean deployment test) e da ciascuna precedente versione supportata (upgrade deployment test)
 - Functional test
 - Stress test
- Ogni cambiamento del codice è disponibile per uno yum update poche ore dopo
 - Il build richiede pochi minuti ma i test girano per alcune ore

Continuous delivery

- Chi vuole potrebbe fare update continuo all'edge di un prodotto con la sicurezza che ha passato tutti i test
- La maggior parte di chi installa continuerà ad aspettare le versioni ufficiali, che abbiano passato anche la certificazione che comporta alcuni controlli non automatizzabili
- Automatizziamo anche la creazione del repository per le release candidate, quello contenente la versione che va in certificazione
 - Il meccanismo di promozione disponibile in Jenkins, con un bottone
- E dopo la certificazione sarà automatizzata anche la promozione al repository delle release ufficiali

Integrazione

- Punto delicato, chi assicura che un cambiamento in un software da cui dipendiamo non ci rompa qualcosa?
- Molto del nostro middleware utilizza meccanismi di protezione
 - Il software Java si porta le dipendenze dietro e le dipendenze vengono cambiate dagli sviluppatori
- Per quello che non usa questi meccanismi quasi tutto il software da cui dipende è in epel e possono essere usati i meccanismi di protezione di epel
 - Ma vanno usati bene, vedi incidenti con Globus di qualche mese fa
- Ci sono potenziali debolezze che dobbiamo ancora coprire

Coordinamento dello sviluppo

- Vedremo cosa sarà di MeDIA
- Comunque le attività di coordinamento le dovremo portare avanti best-effort
- Non ci saranno imposizioni di attività fatte perché sono nel DoW del progetto
 - e che magari erano nel DoW del progetto solo per alzare il finanziamento
- Sceglieremo accuratamente le collaborazioni da seguire, con l'occhio a quelle che ci mettano nella posizione migliore per Horizon 2020

Riepilogo

- Per la prima volta dopo oltre dieci anni lo sviluppo del middleware di Grid sarà senza un contesto collaborativo a livello europeo
- L'impegno di WLCG, EGI e dei product team dovrebbe assicurare una transizione indolore per utenti e siti
- Sarà richiesto uno sforzo non banale ai gruppi di sviluppatori per colmare le lacune lasciate da alcune attività finanziate in EMI
- I gruppi di sviluppo INFN sono nella posizione di fare questo sforzo e di cogliere l'occasione per migliorare i propri processi