



GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA



COSMONAT - Cloud Object Storage and Metadata for Nuclear Astrophysics

Salvatore Messina

Tutor Emidio Giorgio



Il nostro use case

- ▶ Nei nostri esperimenti (astrofisica nucleare) si produce una grande quantità di dati
 - ▶ dati estratti durante l'acquisizione tramite l'apparato sperimentale, formato da targhette di uno specifico elemento, fasci di particelle, rivelatori ed elettronica di supporto.
 - ▶ File piccoli (~2MB) ma in elevata quantità (200 per esperimento)
 - ▶ il campo di indagine è molto vasto, serve conservare i dati secondo criteri che ne permettano efficientemente recupero e riutilizzo.
 - ▶ Ciascuno dei parametri dell'apparato può essere visto come un parametro di ricerca
 - ▶ Un'opportuna catalogazione permette un accesso veloce ai dati acquisiti, scartando quelli non necessari, ed evita ripetute acquisizioni di eventi analoghi, velocizzando i processi decisionali



Object Storage & Metadata

- ▶ I metadati sono uno strumento molto efficace di ricerca e catalogazione di file, specialmente se in gran numero.
- ▶ **OpenStack Swift** supporta l'inserimento di metadati associati agli oggetti, ma non la ricerca basata su metadati.
- ▶ **Soluzione:** Interfacciare Swift (lato server) con un motore di ricerca dedicato ai metadati, **ElasticSearch**. Entrambi i servizi sono dotati di interfacce API REST, che hanno permesso di:
 - ▶ **Associare** ad ogni oggetto su Swift una entry su ElasticSearch (avente come chiavi i metadati precedentemente associati all'oggetto su Swift). Il mapping è fatto lato server
 - ▶ **Interfacciare** ElasticSearch con Openstack Keystone, garantendo l'uniformità di autenticazione con il resto dell'infrastruttura OpenStack.
 - ▶ Sviluppare con semplicità **un'interfaccia utente** per ricerca e gestione di file e metadati.

Attività svolta nel I trimestre

Setup Infrastruttura

Nell'infrastruttura OpenStack implementata nel progetto **CatAq** tra LNS (Laboratori Nazionali del Sud) e LNGS (Laboratori Nazionali del Gran Sasso), è stato creato un *tenant* per l'astrofisica nucleare.



Successivamente sono stati implementati degli script in python per estendere le funzionalità fornite dal client command line di OpenStack, ad esempio aggiungere più metadati insieme, senza cancellare quelli inseriti in precedenza.

Infine è stato implementato, lato server, un filtro Swift che ad ogni oggetto inserito lo mappa ad una entry ElasticSearch. (vedi slide #5)

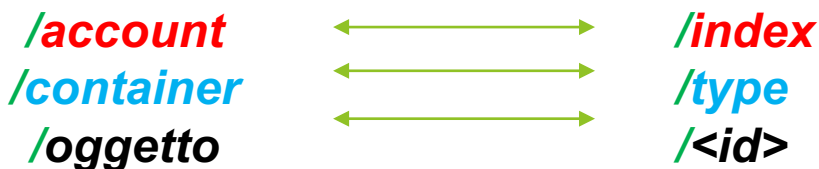


Primo inserimento di dati test

Attività svolta nel I trimestre

Mapping OpenStack / ElasticSearch

Gli oggetti dentro Openstack sono caratterizzati da (account,container,oggetto), mentre su ElasticSearch le entries sono sempre in 3 livelli secondo la terna (index,document,id):



Il mapping tra gli oggetti è stato implementato in modo da fornire un'associazione 1-1 tra oggetti su Swift ed entry su ElasticSearch.


https://swiftlabs.lns.infn.it:8080/v1/AUTH_acd54e43d5ca4b26935b1099837bc65b/Immagini/2MeV.png

<http://90.147.166.53:9200/cosmonat/Immagini/2MeV.png> 



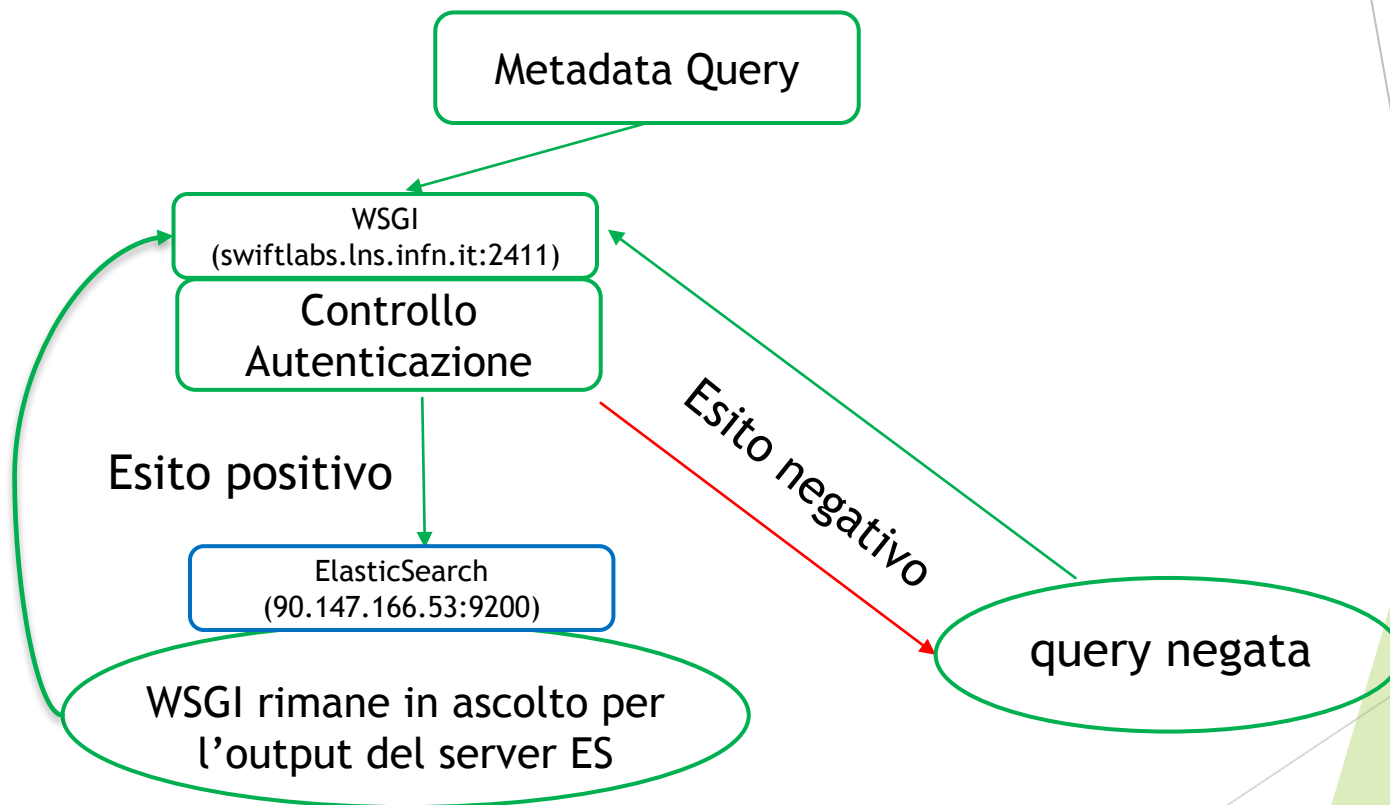
Attività svolta nel II trimestre

Autenticazione chiamate a Elasticsearch

- Elasticsearch non fornisce alcuna protezione sulla API, che potrebbe essere contattata da chiunque. Per preservare la confidenzialità dei dati, abbiamo inserito l'endpoint nel dominio di autenticazione Openstack.
 - le query utente non vengono rivolte direttamente al metadata engine, ma sono filtrate da un [WSGI](#) in esecuzione su un web server Apache.
 - Il [WSGI](#) verifica la presenza di un token OpenStack valido nella richiesta http, ed eventualmente inoltra la chiamata al server Elasticsearch.
 - Verifica anche l'autorizzazione
 - Il metadata engine è accessibile direttamente solo dal WSGI

Attività svolta nel II trimestre

Autenticazione chiamate a Elasticsearch





Attività svolta nel III trimestre

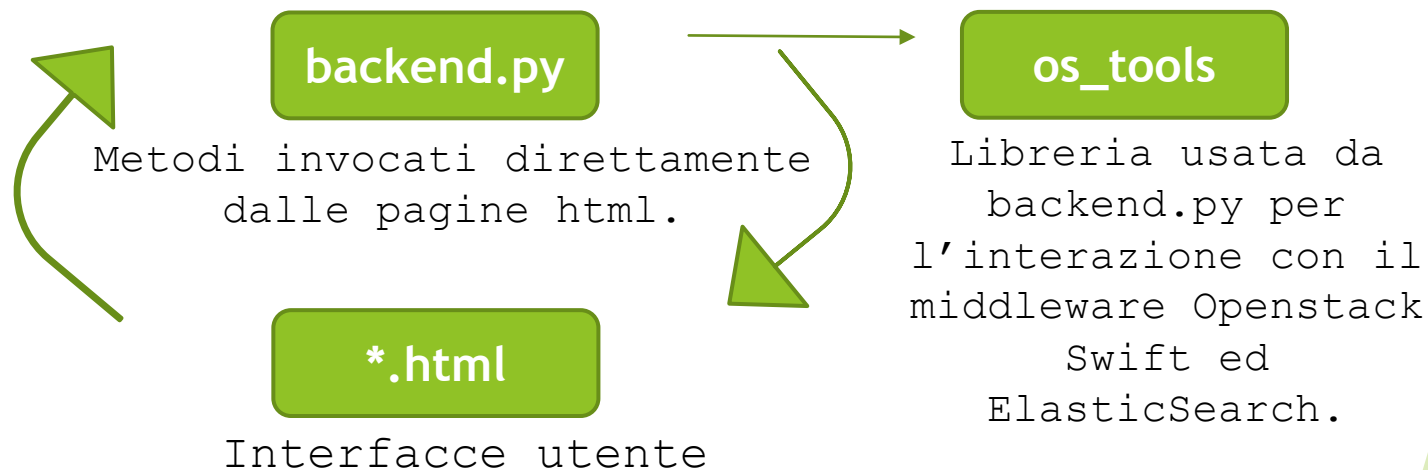
Sviluppo dell'interfaccia grafica

- ▶ Un'interfaccia grafica è essenziale per la manipolazione di elevate quantità di file e metadati
- ▶ Il framework scelto è [htmlpy](#), che si basa su interfacce grafiche per utente definite in html, le quali richiamano delle funzioni python.
 - ▶ Wrapper a QT WebKit
 - ▶ Permette il riuso di codice python
 - ▶ Interscambiabile con interfacce web
- ▶ L'html è reso dinamico attraverso template jinja2, i cui valori vengono passati dal codice Python attraverso dizionari.

Attività svolta nel III trimestre

Sviluppo dell'interfaccia grafica

Per una maggiore pulizia del codice, il backend Python è stato a sua volta suddiviso in:





Attività svolta nel III trimestre

Le principali funzioni di **backend.py** sono:

- **searchByMdVal**: implementa la ricerca basata sui metadati. Riceve da un form il nome dell'attributo e il valore cercato.
- **addObj**: permette di aggiungere uno o più file locali ad un container;
- **download**: scarica l'oggetto desiderato in un path locale predefinito;
- **newcontainer**: crea un nuovo container (avvisa se già esistente);
- **showContainer**: mostra la lista dei container;
- **showContainersObject**: visualizza gli oggetti di un container;
- **showObjMD**: visualizza il nome e il valore di ogni metadato associato ad un oggetto selezionato.

Ognuna di queste funzioni invoca un metodo di **os_tools** che gestisce l'interazione con il middleware OS/ELK

Attività svolta nel III trimestre

```
@htmlPy.Slot(str)
def salvaNuovoCont(self, jsonFormData):
    #creo il form con il nome del container da inserire
    formData=json.loads(jsonFormData)
    cName=formData['nomeContainer']
    result=self.swiftManager.createContainer(cName)
    if result == False:
        message='Inserimento non valido'
    else:
        message='Inserimento riuscito'
    self.app_gui.template = (".newcontainer.html", {'Messaggio':message,
                                                    'ctx':self.userCtx}
    )
```

Codice python
backend.py

Pagina html

```
{% extends "base.html" %}
{% block inncontent %}
<hr>
Container name : {{Messaggio}}
<tr>
<form action="OpenstackBackEnd.salvaNuovoCont" data-bind="true">
<input type='text' name='nomeContainer' size='20' maxlength='30'>
<input type="reset" value="Reset">
<input type="submit" value="Add+">
</tr>
</table>
<hr>
{%endblock%}
```

Interazione tra la funzione in python *salvaNuovoCont* e la pagina *newcontainer.html* che ne specifica i parametri utente (nome del container da creare).



GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Attività svolta nel III trimestre

Sviluppo dell'interfaccia grafica

L'interfaccia grafica all'avvio si presenta in questo modo:

The screenshot shows a web browser window titled "Cosmonat app". The header displays "Username : smessina - Endpoint : https://swiftlabs.lns.infn.it:5000 - Region : LNS" and a "Home" link. The main content area is divided into two sections: "Object management" and "Metadata manager". The "Object management" section contains links for "Show containers" and "Add a new container+". The "Metadata manager" section includes a search form with "Attribute" and "Value" input fields, a "Submit" button, and links for "Edit metadata schema" and "Credits".

Cosmonat app

Username : smessina - Endpoint : https://swiftlabs.lns.infn.it:5000 - Region : LNS [Home](#)

Object management

[Show containers](#)
[Add a new container+](#)

Metadata manager

Search object by metadata value

Attribute Value

[Edit metadata schema](#)

[Credits](#)



Attività svolta nel III trimestre

Con varie funzionalità:

Username : smessina - Endpoint : https://swiftlabs.ins.infn.it:5000 - Region : LNS

These are your containers

Name	Number of objects
File_root	85
He3Be9	32
Immagini	26
Li6Li6_2017	48
macro	52
prova2	525

Submit

1

Objects within container Li6Li6_2017

Add Object+ Choose file Upload

Name	Byte	Last Modified	View MD	Download
zag38.root 216		2017-08-02T14:42:40.567530	View	Downl
zag39.root 103933		2017-08-02T14:43:15.721470	View	Downl
zag40.root 125796		2017-08-02T14:44:05.393080	View	Downl
zag41.root 18958		2017-08-02T14:44:41.901830	View	Downl
zag42.root 67874		2017-08-02T14:45:10.149130	View	Downl
zag44.root 26216		2017-08-02T14:46:01.348050	View	Downl
zag45.root 9965		2017-09-29T13:40:15.813320	View	Downl
zag46.root 8371		2017-08-02T14:47:08.165120	View	Downl

2

Lista Oggetti

3

Upload Oggetto/i

4

Username : smessina - Endpoint : https://swiftlabs.ins.infn.it:5000 - Region : LNS

Objects within container Li6Li6_2017

Add Object+ Choose

Inserimenti corretti = 1 Inserimenti errati = 0

Name	Byte	Last Modified
r1.root	27303611	2017-11-08T15:57:28.030
zag38.root 216		2017-08-02T14:42:40.567
zag39.root 103933		2017-08-02T14:43:15.721

Upload Oggetto/i



GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Piano per il IV trimestre

- ▶ completamento interfaccia grafica:
 - ▶ implementare la funzione che permette la creazione di uno schema di metadati
 - ▶ un meta-form da associare poi ai vari container
 - ▶ rispetto dello schema agli oggetti Swift inseriti dentro questo container
 - ▶ questo schema permette di fornire non solo metadati come stringhe, ma anche come tipi ordinabili (si pensi a valori di angoli, energie, anno dell'esperimento, ecc...), fondamentali in fisica.
- ▶ armonizzazione dell'interfaccia grafica
 - ▶ rinominare le funzionalità esposte all'utente in modo più accurato e significativo
- ▶ Completata la parte di sviluppo, si procederà con uno stress test, per misurare la capacità di carico con dataset costituiti da file sia di grosse dimensioni che in elevata quantità.





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Conclusioni

L'attività proposta si prefiggeva l'uso di un'infrastruttura cloud per il salvataggio dei dati acquisiti da esperimenti di astrofisica nucleare, e l'estensione della stessa infrastruttura cloud con un motore di ricerca per metadati per la catalogazione dei dati memorizzati.





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Conclusioni

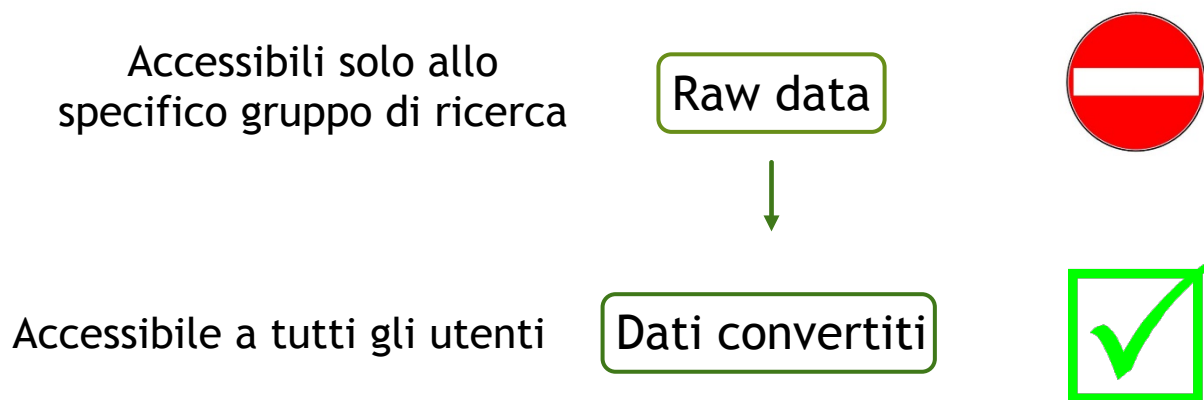
- ▶ Gli obiettivi prefissati sono stati quasi tutti raggiunti
 - ▶ Interazioni tra openstack e elasticsearch
 - ▶ L'idea dell'interfaccia grafica, avuta in corso d'opera, ha dato a questo progetto maggior consistenza
- ▶ Il beta testing ha fatto sorgere nuove possibili funzionalità
 - ▶ Acl per i dati
 - ▶ Supporto a federazioni di identità
 - ▶ Rappresentazione visuale statistiche a supporto dei processi decisionali



Prospettive future

Un possibile miglioramento sarebbe il supporto alle Access Control List. L'idea di potenziamento nasce dopo l'uso dell'applicazione, in modo pratico

► A esempio, suddivisione in:





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Prospettive future

- ▶ Integrazione con tools di rappresentazione visuale e statistica, di grande utilità non solo quando si ha a che fare con una grande mole di oggetti, ma anche quando si vogliono consultare i risultati di ogni singolo esperimento e/o pubblicazione.
- ▶ Estendere il numero di parametri per la ricerca, per renderla più rigorosa e aumentare la manipolabilità degli oggetti.
 - ▶ Combinazione di diversi criteri





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Prospettive future

- ▶ Supporto a identity provider federati
 - ▶ Fondamentale per estendere la potenziale base utenti del progetto.





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Prospettive future

- Possibili altre funzionalità dettate dall'estensione del progetto ad altre comunità (non necessariamente di Astrofisica Nucleare), vedi MAGNEX





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Riferimenti

- ▶ Repository codice
 - ▶ <https://baltig.infn.it/giorgioe/Cosmonat>
 - ▶ https://baltig.infn.it/giorgioe/os_utils





GIORNATA DI INCONTRO BORSE DI STUDIO GARR "ORIO CARLINI"
MARTEDI' 12 DICEMBRE 2017 - ROMA

Demo

