



ANALISI DATI AI TIER-2 E AI TIER-3

Massimo Masera

Torino

Saranno descritte in modo critico il modello di analisi degli esperimenti con le strategie già adottate e quelle che si pensa di adottare, con analisi di pro e contro delle diverse soluzioni prese in considerazione e le proposte alla comunità dei servizi calcolo sullo sviluppo di soluzioni comuni. Dovranno essere descritte in modo più quantitativo le esigenze degli analisti (numero stimato di analisti, stime sulle richieste di storage e CPU per la fase off-grid, ...)



Analysis framework

- Il sistema di classi per l'analisi è stato sviluppato a partire dal 2006 ed è attualmente usato in ALICE con regolarità per **tutte** le analisi che hanno portato/stanno portando a risultati pubblicati
- Fa parte di AliRoot che è il framework generale per l'offline di ALICE
- Forte integrazione con *l'ALICE production Environment* (Alien)
- L'analysis framework è usato per:
 - ✓ analisi gestite centralmente (scheduled)
 - ✓ analisi caotica

Analysis macro / 1



- Ogni macro di analisi istanzia un `AliAnalysisManager` steering object
- Si possono selezionare 3 modalità di analisi
 - ✓ **local** → CPU locale, dati su GRID
 - ✓ **grid** → si registra un “plugin” (`AliAnalysisGrid`) nel manager, che permette di scrivere un jdl e sottomettere i job di analisi:
 - pluginmode:
 - test (verifica pre-start)
 - full (the real thing)
 - terminate (per terminare una sessione iniziata in precedenza)
 - ✓ **proof** → analisi su un cluster PROOF

Analysis macro / 2



- Vengono registrate nel manager uno o più oggetti derivati da **AliAnalysisTask**
 - ✓ Sono **TTask** di root → implementano l'analisi vera e propria
 - ✓ Definiscono Input e Output handlers
 - ✓ Output:
 - Può essere ESD → AOD + AOD specifici per filone di analisi
 - Analisi finale: tipicamente istogrammi. Ntuple con qualche caveat
- Gergo: macro = *Analysis Train* – task = vagone del treno
- Diverse modalità di input alla macro si usa:
 - ✓ XML collection di *Analysis Object Datasets* (AOD) (o di *Event Summary Data* - ESD) realizzata sul file catalogue
 - ✓ Dataset su CAF (PROOF mode)
 - ✓ “Directory” su catalogo
 - ✓ Lista di runs
- Selezione dati in input:
 - ✓ Definizione di tagli su *data TAGS* (poco usato per ora → in prospettiva sarà la norma)

Analysis macro / 3



- Analisi locale: chain di AOD (o ESD) a partire dalla collezione di input
- Analisi su GRID: il plugin lancia N job, ciascuno dei quali processa non più di M AOD di input. L'aggregazione è fatta in automatico
- In uscita (fine chain / fine dei job / fine proof session):
 - ✓ **Terminate** → opzionale: merge degli output - importante per analisi finale
 - ✓ Grid mode:
 - il file di output è salvato sullo SE Grid preselezionato nella stessa macro
 - output – file root- replicato su disco locale
- Un'unica macro per
 - ✓ Testare il codice localmente
 - ✓ Utilizzarlo su piccoli dataset con PROOF
 - ✓ Utilizzarlo su grande scala con la GRID

Modalità di analisi/1



- Analisi schedulata (MC e dati) :
 - ✓ Analysis Train lanciati centralmente
 - ✓ Tasks approvate dai Physics Working Groups
 - ✓ Produzione di AOD standard e “Delta” AOD (=AOD specifici per filone di analisi)

PRODUCTION CYCLES

Train Details » No filter

Manage »

Production info						Jobs status				Comment
Production	Description	Status	Completion rate	Config	Results	Total	Done	Running	Waiting	
QA4_LHC10c_Merging	QA4_LHC10c_Merging: PWG1 QA train	Running	61%			13	8			QA4_LHC10c_Merging: PWG1 QA train
QA4_LHC10c	QA4_LHC10c: PWG1 QA train	Running	89%			6520	5828	103		QA4_LHC10c: PWG1 QA train
QA3_LHC10c_Merging	QA3_LHC10c_Merging: PWG1 QA train (no TRD and FMD)	Completed	75%			57	43			QA3_LHC10c_Merging: PWG1 QA train (no TRD and FMD)
QA3_LHC10c	QA3_LHC10c: PWG1 QA train (no TRD and FMD)	Completed	92%			26190	24216			QA3_LHC10c: PWG1 QA train (no TRD and FMD)
QA2_LHC10c	QA2_LHC10c: PWG1 QA train (ESD friends enabled)	Completed	25%			1876	479			QA2_LHC10c: PWG1 QA train (ESD friends enabled)
QA2_LHC10b_Merging	QA2_LHC10b_Merging: PWG1 QA train	Completed	48%			188	91			QA2_LHC10b_Merging: PWG1 QA train
QA2_LHC10b	QA2_LHC10b: PWG1 QA train (ESD friends enabled)	Completed	49%			21554	10667			QA2_LHC10b: PWG1 QA train (ESD friends enabled)
QA1_LHC10b_Merging	QA1_LHC10b_Merging: PWG1 QA train	Completed	55%			142	79			PWG1 QA train - Merging
QA1_LHC10b	QA1_LHC10b: PWG1 QA train	Completed	70%			6702	4752			PWG1 QA train
TR023_LHC10b4	TR023: ESD+MC -> AODMC + delta AOD	Completed	71%	📁		238	170			TR023: ESD+MC -> AODMC + delta AOD
TR022_LHC10b3	TR022: ESD+MC -> AODMC + delta AOD	Completed	80%	📁		420	337			TR022: ESD+MC -> AODMC + delta AOD
TR021_LHC10b2	TR021: ESD+MC -> AODMC + delta AOD	Completed	97%	📁		864	839			TR021: ESD+MC -> AODMC + delta AOD
TR020_LHC10b1	TR020: ESD+MC -> AODMC + delta AOD	Completed	99%	📁		757	755			TR020: ESD+MC -> AODMC + delta AOD
TR019_PASS6	TR019: LHC09d-Pass6 ESD filtering w. PhysSelection -> AOD (including muon deltas)	Completed	100%			31	31			TR019: LHC09d-Pass6 ESD filtering w. PhysSelection -> AOD (including muon deltas)
QA003_PASS6	QA003: PWG1 QA train	Completed	96%			31	30			QA003: PWG1 QA train
TR018_LHC09a5AOD	TR018: AOD -> Vertexing deltas	Completed	100%	📁		368	368			TR018: AOD -> Vertexing deltas
TR017_LHC09a5ESD	TR017: ESD+MC -> AODMC + delta AOD	Completed	91%	📁		404	368			TR017: ESD+MC -> AODMC + delta AOD
QA002_PASS5	QA002: PWG1 QA train	Completed	100%			28	28			QA002: PWG1 QA train
QA001_PASS4	QA001: PWG1 QA train	Completed	100%			31	31			QA001: PWG1 QA train

Modalità di analisi / 2



- Il numero di AnalysisTask sta crescendo:
 - ✓ sta raddoppiando ogni 6 mesi circa
- Il treno più “lungo” consta di 67 task indipendenti
- I Physics Working Groups hanno inoltre degli Analysis Train che vengono gestiti da un responsabile del PWG:
 - ✓ Maggiore flessibilità
 - ✓ Modalità semi-caotica
- Esperienza acquisita finora:
 - ✓ Contenere durata dei job a qualche ora max. → limite al numero di vagoni
 - ✓ Dati in input replicati su 3-4 SE per aumentare la disponibilità
 - ✓ Input all’analisi: AOD più veloce di ESD
- Analisi caotica:
 - ✓ Effettuata da fisici singoli
 - ✓ Tipicamente di seconda generazione (a partire dall’output di un’analisi scheduled)
 - ✓ Stessa modalità: treno con al limite 1 vagone. Istogrammi come output

Esempio: analisi $D^+ \rightarrow K^- \pi^+ \pi^+$



```
class AliAnalysisGrid;
TString mySE="ALICE::TORINO::SE"; // Storage Element for output data

void RunAnalysisAODVertexingHF()
{
  //
  // Test macro for AliAnalysisTaskSE's for heavy-flavour candidates
  // It has the structure of a Analysis Train:
  // - in this macro, change things related to running mode
  // and input preparation
  // - add your task using a AddTaskXXX macro
  //
  // A.Dainese, andrea.dainese@lnl.infn.it
  // "grid" mode added by R.Bala, bala@to.infn.it
  //
  gSystem->Setenv("alien_CLOSE_SE",mySE.Data());

  gSystem->SetIncludePath("-I. -I$ROOTSYS/include -I$ALICE_ROOT -I$ALICE_ROOT/include -I$ALICE_ROOT/ITS -I$ALICE_ROOT/OT/TPC -I$ALICE_ROOT/CONTAINERS -I$ALICE_ROOT/STEER -I$ALICE_ROOT/TRD -I$ALICE_ROOT/macros -I$ALICE_ROOT/ANALYSIS -I$ALICE_ROOT/PWG3 -I$ALICE_ROOT/PWG3/vertexingHF -g");
  //
  TString trainName = "D2H";
  TString analysisMode = "grid"; // "local", "grid", or "proof"
  TString inputMode = "xml"; // "list", "xml", or "dataset"

  Bool_t useParFiles=kFALSE;
  Bool_t useAlienPlugin=kTRUE;
  TString pluginmode="full"; // it can be test for 1 file or terminate to merge all the jobs done
  // - otherwise is full
  TString loadMacroPath="$ALICE_ROOT/PWG3/vertexingHF/";
}
```

La macro viene fatta girare interattivamente a partire da un qualunque PC che abbia installati:

1. ROOT
2. ALIROOT
3. ALIEN

L'utente deve essere un utente della VO ALICE

Esempio: analisi $D^+ \rightarrow K^- \pi^+ \pi^+$



```
//-----  
//  
AliAnalysisGrid* CreateAlienHandler(TString pluginmode="test", Bool_t useParFiles=kFALSE)  
{  
  // Check if user has a valid token, otherwise make one. This has limitations.  
  // One can always follow the standard procedure of calling alien-token-init then  
  // source /tmp/gclient_env_$UID in the current shell.  
  if (!AliAnalysisGrid::CreateToken()) return NULL;  
  AliAnalysisAlien *plugin = new AliAnalysisAlien();  
  // Set the run mode (can be "full", "test", "offline", "submit" or "terminate")  
  plugin->SetRunMode(pluginmode.Data());  
  plugin->SetUser("masera");  
  plugin->SetNtestFiles(1);  
  // Set versions of used packages  
  plugin->SetAPIVersion("V1.1x");  
  plugin->SetROOTVersion("v5-26-00b-5");  
  plugin->SetAliROOTVersion("v4-19-12-AN");  
  // Declare input data to be processed.  
  // Method 1: Create automatically XML collections using alien 'find' command.  
  // Define production directory LFN  
  plugin->SetGridDataDir("/alice/cern.ch/user/r/rbala/data7TeV_114916to117222NewRun");  
  // Set data search pattern  
  plugin->SetDataPattern("AliAOD.root");  
  plugin->SetFriendChainName("./AliAOD.VertexingHF.root");  
}
```

Scelta della versione del codice

Dati input

File che verranno selezionati come input (AOD + Deta-AOD)

Esempio: analisi $D^+ \rightarrow K^- \pi^+ \pi^+$

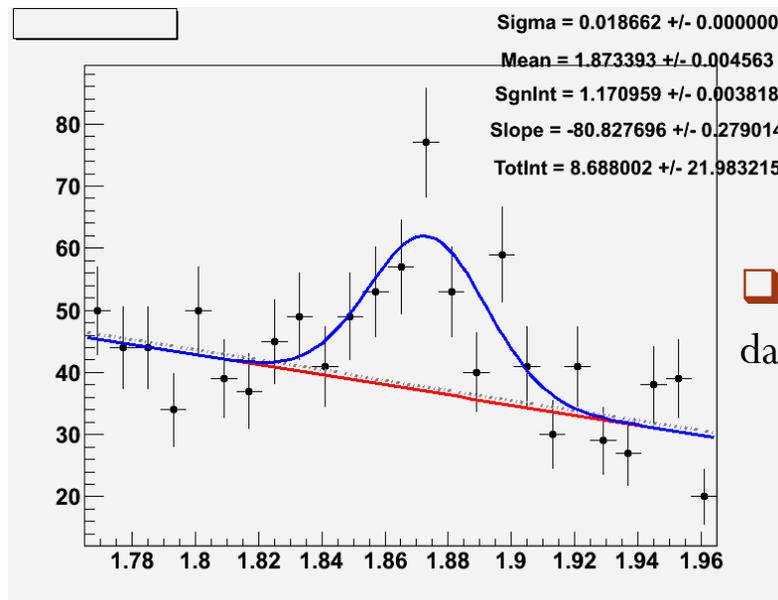


Masterjob 45644288 of masera, status : **DONE** (refresh | JDL | more details | update status)

DONE (78)

```
45644295 : trace | trace all | log files
45644297 : trace | trace all | log files
45644298 : trace | trace all | log files
45644299 : trace | trace all | log files
45644300 : trace | trace all | log files
45644301 : trace | trace all | log files
45644302 : trace | trace all | log files
45644303 : trace | trace all | log files
45644304 : trace | trace all | log files
45644305 : trace | trace all | log files
45644306 : trace | trace all | log files
45644307 : trace | trace all | log files
45644308 : trace | trace all | log files
45644309 : trace | trace all | log files
45644310 : trace | trace all | log files
45644311 : trace | trace all | log files
45644312 : trace | trace all | log files
45644313 : trace | trace all | log files
45644314 : trace | trace all | log files
```

- In questo esempio, l'analisi utente è stata condotta su un dataset p-p a 7 TeV di 30 M eventi.
- Il plugin ha lanciato 78 job che hanno girato sui CE su cui si trovano i dati
- Il risultato è stato copiato sul disco del laptop da cui l'analisi è partita



- Plot prodotto a partire dal file salvato localmente



Analisi: dove?

- L'esempio appena fatto mostra come sia possibile arrivare a un risultato finale usando un approccio GRID-oriented
- I job di analisi girano **dove sono i dati di input**
 - ✓ I dati di input sono il frutto dell'analysis train precedente
 - ✓ Nel caso in esempio sono replicati 4 volte
 - ✓ La collocazione delle repliche non è controllata di default da chi fa girare il primo treno di analisi
 - Quindi i job girano **un po' dappertutto**
 - Nell'esempio precedente: CNAF, Russia T2, Torino, CERN (non li ho controllati tutti)
 - ✓ Al momento non c'è distinzione tra Tier-1 e Tier-2
- L'utente **non** ha un centro di riferimento, se non per quanto riguarda l'output della sua analisi, specificato nella macro

Extra GRID nulla salus



- C'è spazio per attività di calcolo in modo locale?
 - ✓ In ALICE qualsiasi attività extra GRID è ufficialmente deprecata
 - ✓ Il management board dell'esperimento ha stabilito che non si può pubblicare nessun risultato che non sia disponibile su GRID
 - ✓ La risposta canonica alla domanda è quindi NO.
- Tuttavia un uso marginale di risorse non GRID c'è stato e in qualche misura c'è ancora:
 - ✓ Attività legate allo sviluppo e commissioning di detector hanno beneficiato di calcoli fatti extra GRID
 - ✓ Attività di sviluppo software → test con software non rilasciato e quindi non disponibile su GRID
- Il calcolo extra GRID sta comunque diminuendo con una eccezione:
 - ✓ L'analisi interattiva con PROOF
 - ✓ Questa attività è integrata nel modello di calcolo



Perché solo GRID?

- Gli utenti hanno sempre preferito il calcolo locale laddove possibile:
 - ✓ Perché è più facile.
 - ✓ Perché “la GRID non funziona”.
 - ✓ Perché è più rapido.
 - ✓ Perché si può usare software non ancora rilasciato o modificato.
- In realtà soltanto l’ultima motivazione è rimasta in piedi:
 - ✓ L’analysis framework è ben collaudato e ci sono esempi a non finire a cui ispirarsi: agli utenti, specie ai più giovani, piace.
 - ✓ I servizi di GRID e i siti sono sempre più affidabili e stabili.
 - ✓ Alla fine con GRID può essere più rapido, specie se la statistica cresce.
 - ✓ La policy di tagging è molto dinamica:
 - Tags sul release branch per la ricostruzione una o due volte al mese.
 - Tags molto frequenti (settimanali) sul trunk per l’analisi.
- Catalogo e Offline Conditions DataBase:
 - ✓ Per quasi tutte le attività di calcolo occorre qualche interfaccia o con il catalogo o con l’OCDB o con entrambi → questo limita l’uso di farm standalone
 - ✓ La facilità di installazione di AliEn su macchine desktop potenti consente la possibilità di fare test con codice non rilasciato e dati su GRID con molta facilità

Perché analysis facilities?



- L'uso della CAF via PROOF ha avuto un successo enorme in ALICE.
- Tutte le analisi codificate nell'ambito del framework possono essere usate con PROOF.
- PROOF serve quando i job di analisi sarebbero brevi e la latenza della GRID penalizzante.
- PROOF serve quando si vuole una risposta in tempo quasi reale, interattiva:
 - ✓ First Physics.
 - ✓ Offline Calibration.
 - ✓ Fine tuning di procedure di analisi su data set limitati.
- Finora la CAF è bastata, ma le Analysis Facility stanno sorgendo anche fuori dal CERN.
- Riteniamo importante poter avere qualcosa del genere anche in Italia (*cfr. talk di Dario Berzano*).

ALICE e i Tier-3



- Nel computing model di ALICE non è previsto un ruolo per centri di calcolo diversi da Tier-1 e Tier-2
- Il vero test del modello di calcolo sarà più avanti, in particolare con dati Pb-Pb
- Tuttavia al momento non c'è evidenza che ci sia necessità di farm gestite professionalmente per raccogliere gli output dell'analisi scheduled...
- ...Con l'eccezione di Facility per il calcolo interattivo con PROOF, che nel modello di ALICE giocano il ruolo di Tier-3

Il modello di analisi scala?



- Questi primi mesi di presa dati non hanno ancora fornito una risposta “sperimentale” a questa domanda
- Ci sono degli improvement in corso per l’analisi scheduled e per la selezione degli eventi:
 - ✓ automazione delle procedure di analisi dopo i passi di ricostruzione
 - ✓ uso intensivo delle TAG per la selezione degli eventi
- I fabbisogni di calcolo tengono conto dell’analisi per cui non si prevedono delle necessità che vadano al di là di quelle espresse finora
- Il numero di utenti attivi su GRID è in crescita ed è ora poco meno di 200 persone e gli italiani sono poco meno di 50

