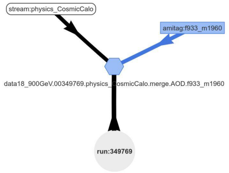
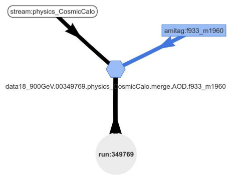

stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

Status of EI Core & UI

- Status
- MC Dataset Overlaps
- New Aux Accessors
- Graph DB
  - Database
  - Web Service
  - Migration plans

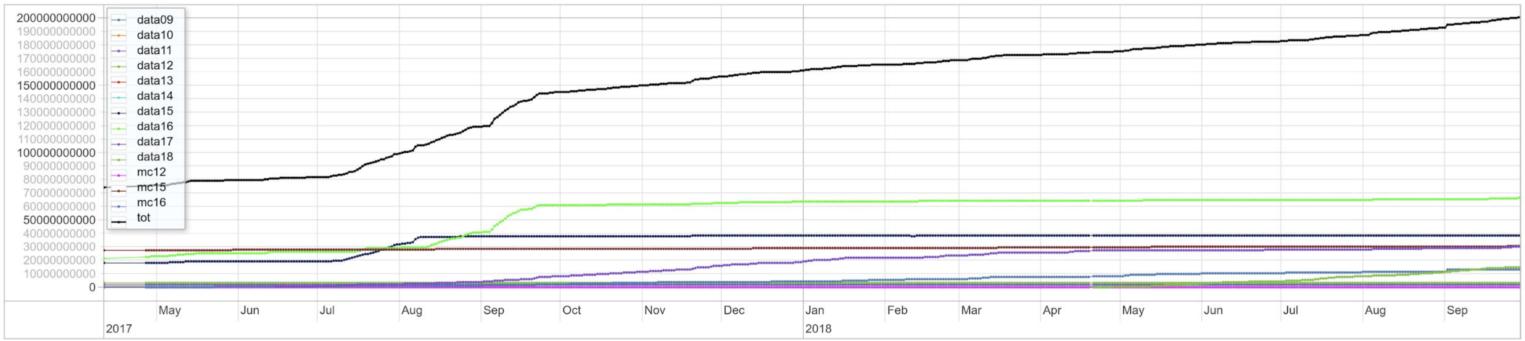
Julius Hrivnac, LAL
EI WS, 1 Oct 2018, CERN


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

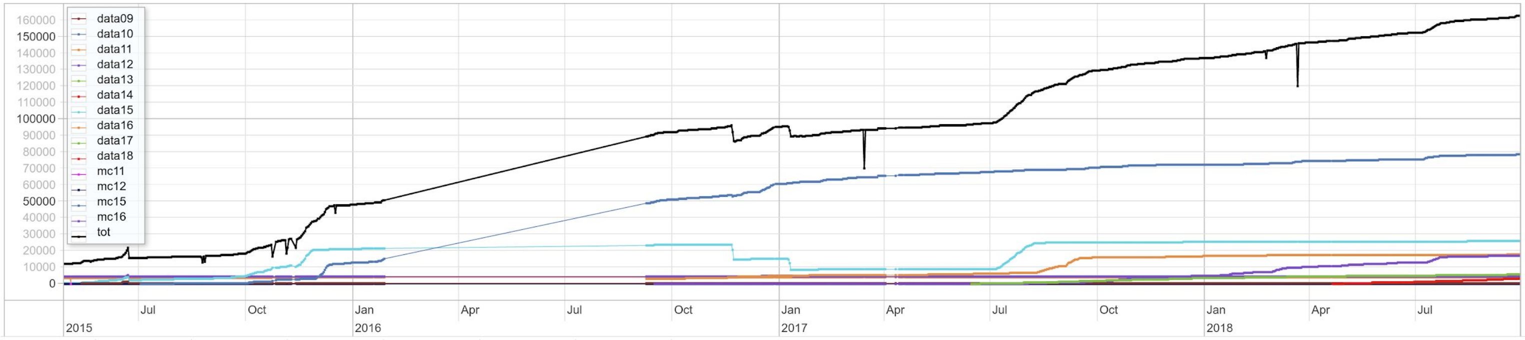
Status - Import

More than 200 000 000 000 events available !

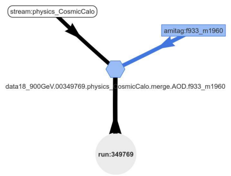
Events (200819704088 @ 2018-09-30)



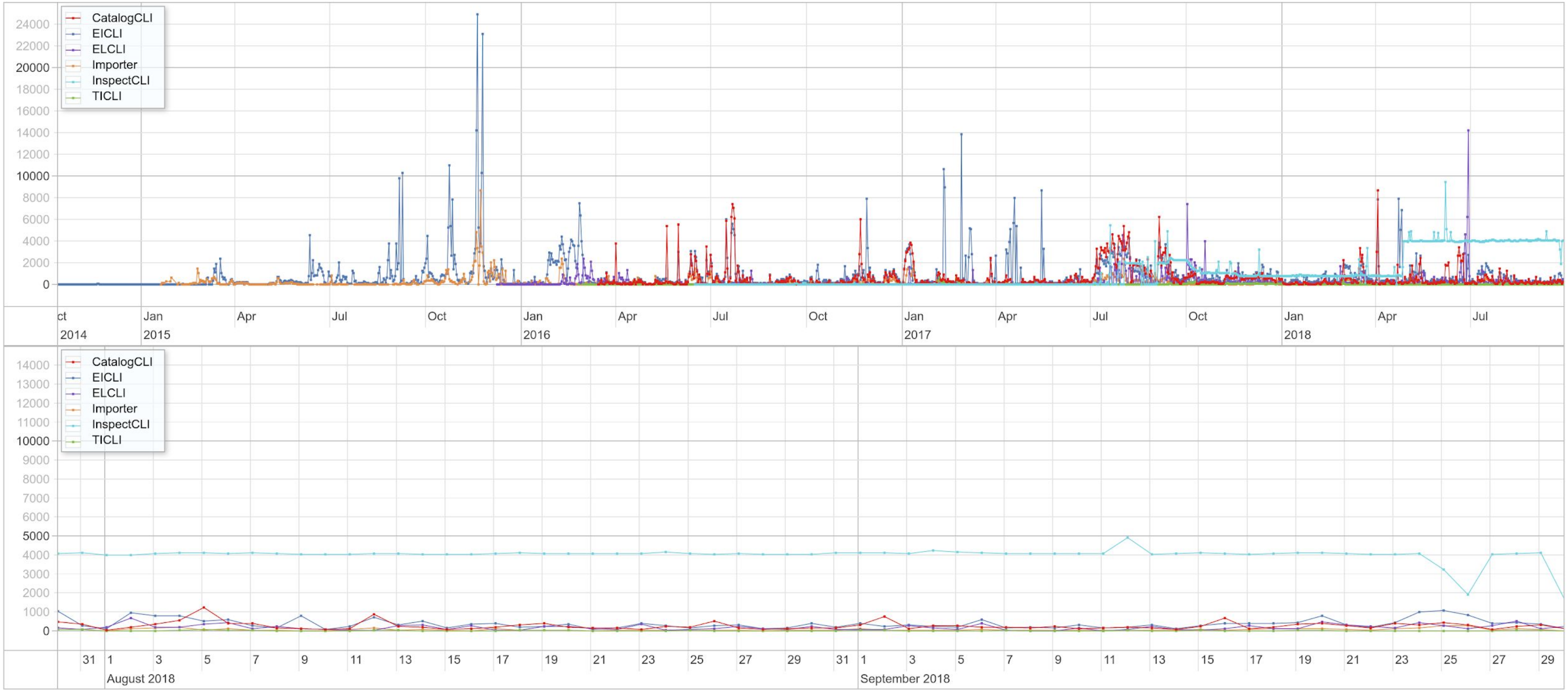
Datasets (163033 @ 2018-09-30)



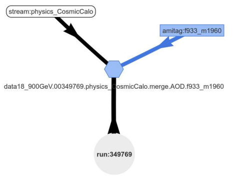
Status - Access


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

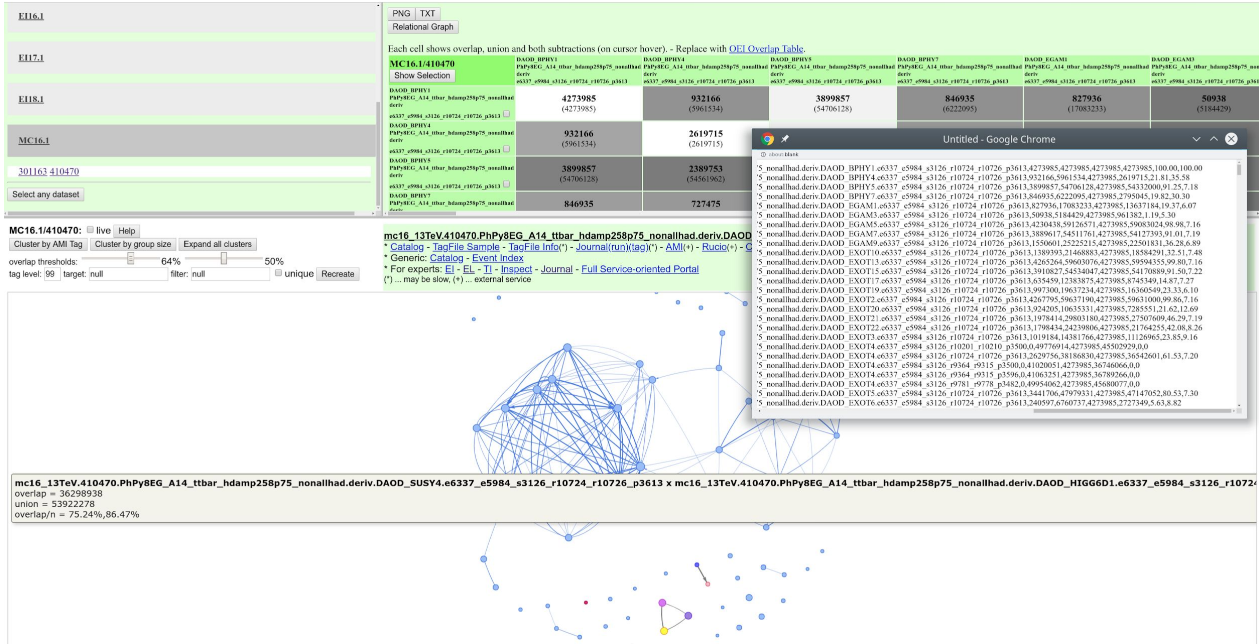
Daily Access statistics (3232909 @ 2018-09-30)



MC Dataset Overlaps


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- Implemented on request for run MC16.1/410470
    - Different handling of RunNumbers
- Triggered some improvements to general interface
- New possibility: TXT export from WS



Each cell shows overlap, union and both subtractions (on cursor hover). - Replace with OEI Overlap Table.

MC16.1/410470	DAOD_BPHY1 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613	DAOD_BPHY4 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613	DAOD_BPHY5 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613	DAOD_BPHY7 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613	DAOD_EGAM1 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613	DAOD_EGAM3 PhPy8EG_A14_ttbar_hdamp258p75_non deriv e6337_e5984_s3126_r10724_r10726_p361
DAOD_BPHY1 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613 ☐	**4273985** (4273985)	**932166** (5961534)	**3899857** (54706128)	**846935** (6222095)	**827936** (17083233)	**50938** (5184429)
DAOD_BPHY4 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613 ☐	**932166** (5961534)	**2619715** (2619715)				
DAOD_BPHY5 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad deriv e6337_e5984_s3126_r10724_r10726_p3613 ☐	**3899857** (54706128)	**2389753** (54561962)				
DAOD_BPHY7 PhPy8EG_A14_ttbar_hdamp258p75_nonallhad	**846935**	**727475**				

```
'5_nonallhad.deriv.DAOD_BPHY1.e6337_e5984_s3126_r10724_r10726_p3613,4273985,4273985,4273985,4273985,100.00,100.00
'5_nonallhad.deriv.DAOD_BPHY4.e6337_e5984_s3126_r10724_r10726_p3613,932166,5961534,4273985,2619715,21.81,35.58
'5_nonallhad.deriv.DAOD_BPHY5.e6337_e5984_s3126_r10724_r10726_p3613,3899857,54706128,4273985,54332000,91.25,7.18
'5_nonallhad.deriv.DAOD_BPHY7.e6337_e5984_s3126_r10724_r10726_p3613,846935,6222095,4273985,2795045,19.82,30.30
'5_nonallhad.deriv.DAOD_EGAM1.e6337_e5984_s3126_r10724_r10726_p3613,827936,17083233,4273985,13637184,19.37,6.07
'5_nonallhad.deriv.DAOD_EGAM3.e6337_e5984_s3126_r10724_r10726_p3613,50938,5184429,4273985,961382,1.19,5.30
'5_nonallhad.deriv.DAOD_EGAM5.e6337_e5984_s3126_r10724_r10726_p3613,4230438,59126571,4273985,59083024,98.98,7.16
'5_nonallhad.deriv.DAOD_EGAM7.e6337_e5984_s3126_r10724_r10726_p3613,3889617,54511761,4273985,54127393,91.01,7.19
'5_nonallhad.deriv.DAOD_EGAM9.e6337_e5984_s3126_r10724_r10726_p3613,1550601,25225215,4273985,22501831,36.28,6.89
'5_nonallhad.deriv.DAOD_EXOT10.e6337_e5984_s3126_r10724_r10726_p3613,1389393,21468883,4273985,18584291,32.51,7.48
'5_nonallhad.deriv.DAOD_EXOT13.e6337_e5984_s3126_r10724_r10726_p3613,4265264,59603076,4273985,59594355,99.80,7.16
'5_nonallhad.deriv.DAOD_EXOT15.e6337_e5984_s3126_r10724_r10726_p3613,3910827,54534047,4273985,54170889,91.50,7.22
'5_nonallhad.deriv.DAOD_EXOT17.e6337_e5984_s3126_r10724_r10726_p3613,635459,12383875,4273985,8745349,14.87,7.27
'5_nonallhad.deriv.DAOD_EXOT19.e6337_e5984_s3126_r10724_r10726_p3613,997300,19637234,4273985,16360549,23.33,6.10
'5_nonallhad.deriv.DAOD_EXOT2.e6337_e5984_s3126_r10724_r10726_p3613,4267795,59637190,4273985,59631000,99.86,7.16
'5_nonallhad.deriv.DAOD_EXOT20.e6337_e5984_s3126_r10724_r10726_p3613,924205,10635331,4273985,7285551,21.62,12.69
'5_nonallhad.deriv.DAOD_EXOT21.e6337_e5984_s3126_r10724_r10726_p3613,1978414,29803180,4273985,27507609,46.29,7.19
'5_nonallhad.deriv.DAOD_EXOT22.e6337_e5984_s3126_r10724_r10726_p3613,1798434,24239806,4273985,21764255,42.08,8.26
'5_nonallhad.deriv.DAOD_EXOT3.e6337_e5984_s3126_r10724_r10726_p3613,1019184,14381766,4273985,11126965,23.85,9.16
'5_nonallhad.deriv.DAOD_EXOT4.e6337_e5984_s3126_r10201_r10210_p3500,0,49776914,4273985,45502929,0,0
'5_nonallhad.deriv.DAOD_EXOT4.e6337_e5984_s3126_r10724_r10726_p3613,2629756,38186830,4273985,36542601,61.53,7.20
'5_nonallhad.deriv.DAOD_EXOT4.e6337_e5984_s3126_r9364_r9315_p3500,0,41020051,4273985,36746066,0,0
'5_nonallhad.deriv.DAOD_EXOT4.e6337_e5984_s3126_r9364_r9315_p3596,0,41063251,4273985,36789266,0,0
'5_nonallhad.deriv.DAOD_EXOT4.e6337_e5984_s3126_r9781_r9778_p3482,0,49954062,4273985,45680077,0,0
'5_nonallhad.deriv.DAOD_EXOT5.e6337_e5984_s3126_r10724_r10726_p3613,3441706,47979331,4273985,47147052,80.53,7.30
'5_nonallhad.deriv.DAOD_EXOT6.e6337_e5984_s3126_r10724_r10726_p3613,240597,6760737,4273985,2727349,5.63,8.82
```

MC16.1/410470: ☐ live

overlap thresholds: 64% 50%

tag level: 99 target: null filter: null ☐ unique

mc16_13TeV.410470.PhPy8EG_A14_ttbar_hdamp258p75_nonallhad.deriv.DAOD

* Catalog - TagFile Sample - TagFile Info(*) - Journal(run)(tag)(*) - AMI(+) - Rucio(+) - C

* Generic: Catalog - Event Index

* For experts: EI - EL - TI - Inspect - Journal - Full Service-oriented Portal

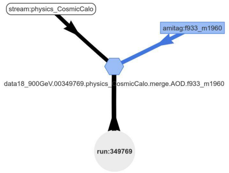
(*) ... may be slow, (+) ... external service

mc16_13TeV.410470.PhPy8EG_A14_ttbar_hdamp258p75_nonallhad.deriv.DAOD_SUSY4.e6337_e5984_s3126_r10724_r10726_p3613 x mc16_13TeV.410470.PhPy8EG_A14_ttbar_hdamp258p75_nonallhad.deriv.DAOD_HIGG6D1.e6337_e5984_s3126_r1072

overlap = 36298938

union = 53922278

overlap/n = 75.24%,86.47%

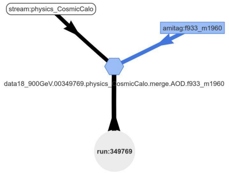

stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

New Aux Accessors

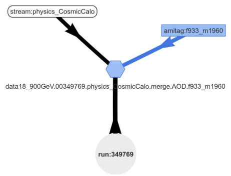
- `net.hep.atlas.Database.EIHadoop.Accessor.Aux.`**`Filler`**
  - Fill TagFile with missing columns from another TagFile
  - For example: fill trigger info from AOD to DAOD
- `net.hep.atlas.Database.EIHadoop.Accessor.Aux.`**`Unique`**
  - Extract only unique events from a set of TagFiles

```
$ ei -query … -mr … -aux ...
```

Graph Databases


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- Traditional data structures in HEP:
  - tuples (tables)
  - trees
  - nested tuples (trees of tuples)
  - relational (SQL-like)
- Schemafull or schemaless
- **But many of our data are graph-like** & schemaless
  - **Entities with relations**
  - G = (V, E) # *graph = (vectors, edges)*
- Not well covered by relational (SQL) databases
  - We don’t need only a possibility to add new data with pre-defined relations
  - We need to add new relations
- Graph databases exist since a long time
  - Matured only recently thanks to Big Data & AI
  - Very good implementations & (de-facto) standards available

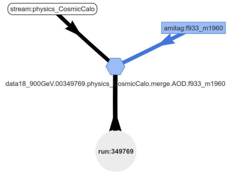

stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

Graph Database for EI

- **Large parts of EI data are graph-like**
  - Catalog:
    - Various relations between dataset
    - Dataset overlaps
  - Events:
    - Versions of the same event in different places

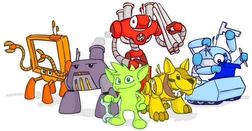
Graph Databases

Standards & Choices


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- De-facto standard language/api: **Gremlin**
  - Gremlin is a functional, data-flow language to traverse a property graph. Every Gremlin traversal is composed of a sequence of (potentially nested) steps. A step performs an atomic operation on the data stream. Every step is either a *map*-step (transforming the objects in the stream), a *filter*-step (removing objects from the stream), or a *sideEffect*-step (computing statistics about the stream).
  - Gremlin supports transactional & non-transactional processing in declarative or imperative manner.
  - Gremlin can be expressed in all languages supporting function composition & nesting.
  - Supported languages: Java, Groovy, Scala, Python, Ruby, Go, …
- Commonly used framework: **TinkerPop**
- Leading implementation: **JanusGraph**
  - Supported storage backends: Cassandra, HBase, Google Cloud, Oracle BerkeleyDB
  - Supported graph data analytics: Spark, Giraph, Hadoop
  - Supported searches: Elastic Search, Solr, Lucene
- Chosen visualisation: **visj.org**

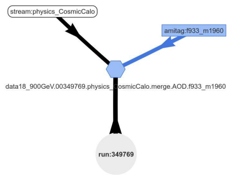
Gremlin $G = (V, E)$



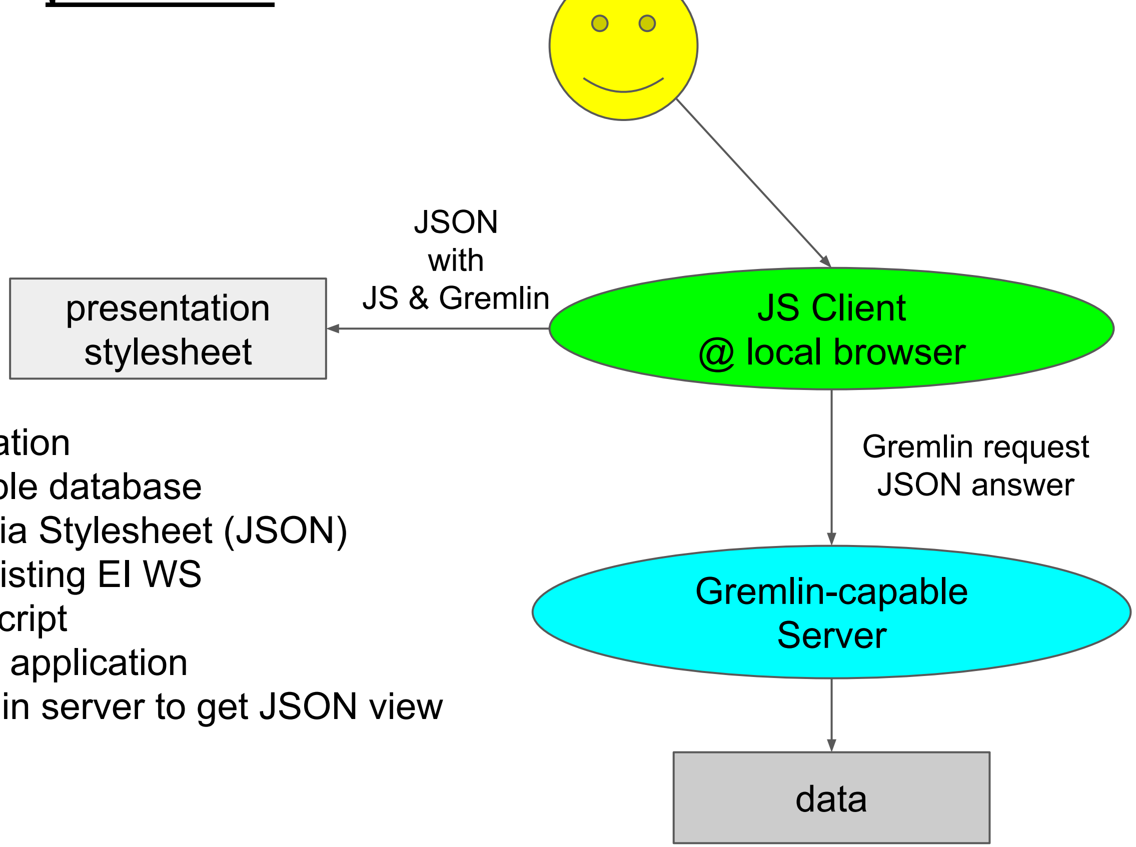
JanusGraph

Graph Database for EI

Architecture


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- Using standard Graph DB
  - Importing data from EI
- Generic Web Service graphical visualisation
  - Can display any Gremlin-compatible database
  - Visualisation can be customised via Stylesheet (JSON)
    - To give the same L&F as existing EI WS
  - Implemented completely in JavaScript
    - So doesn't need server-side application
    - Connects to standard Gremlin server to get JSON view of data



presentation stylesheet

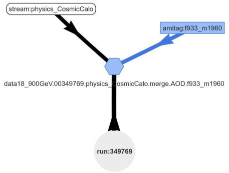
JSON with JS & Gremlin

JS Client @ local browser

Gremlin request JSON answer

Gremlin-capable Server

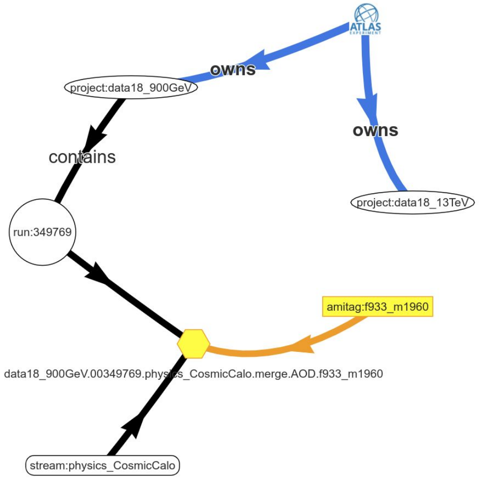
data

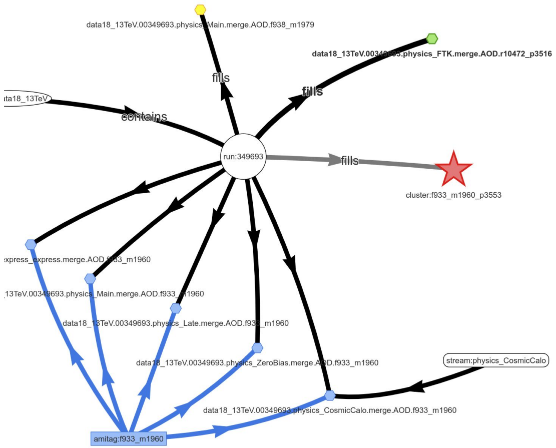

stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

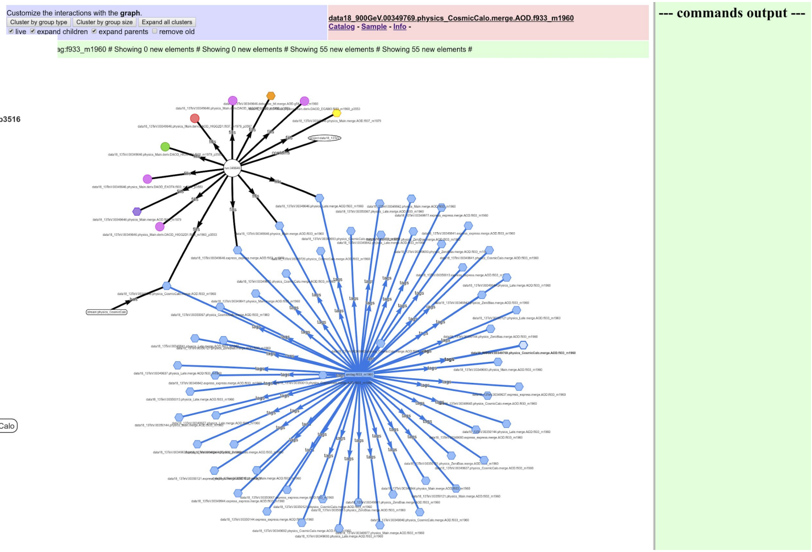
Graph Database for EI

Status

- ➢ Currently using standard installation (with Oracle BerkeleyDB backend)
  - ○ Will migrate to HBase backend
- ➢ Snapshot of EI18 Catalog with a subset of overlap tables imported
  - ○ All Catalog & all overlap tables can be easily imported
- ➢ Most of the graphical part implemented
  - ○ By standalone JS implementation


owns
owns
project:data18_900GeV
contains
project:data18_13TeV
run:349769
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
stream:physics_CosmicCalo


data18_13TeV.00349693.physics_Main.merge.AOD.f938_m1979
data18_13TeV.00349693.physics_FTK.merge.AOD.r10472_p3516
fills
fills
ata18_13TeV
contains
run:349693
fills
cluster:f933_m1960_p3553
xpress_express.merge.AOD.f933_m1960
13TeV.00349693.physics_Main.merge.AOD.f933_m1960
data18_13TeV.00349693.physics_Late.merge.AOD.f933_m1960
data18_13TeV.00349693.physics_ZeroBias.merge.AOD.f933_m1960
data18_13TeV.00349693.physics_CosmicCalo.merge.AOD.f933_m1960
stream:physics_CosmicCalo
amitag:f933_m1960


Customize the interactions with the **graph**.

Cluster by group type | Cluster by group size | Expand all clusters

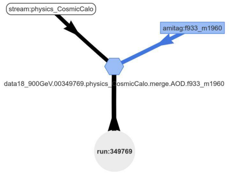
☑ live ☑ expand children ☑ expand parents □ remove old

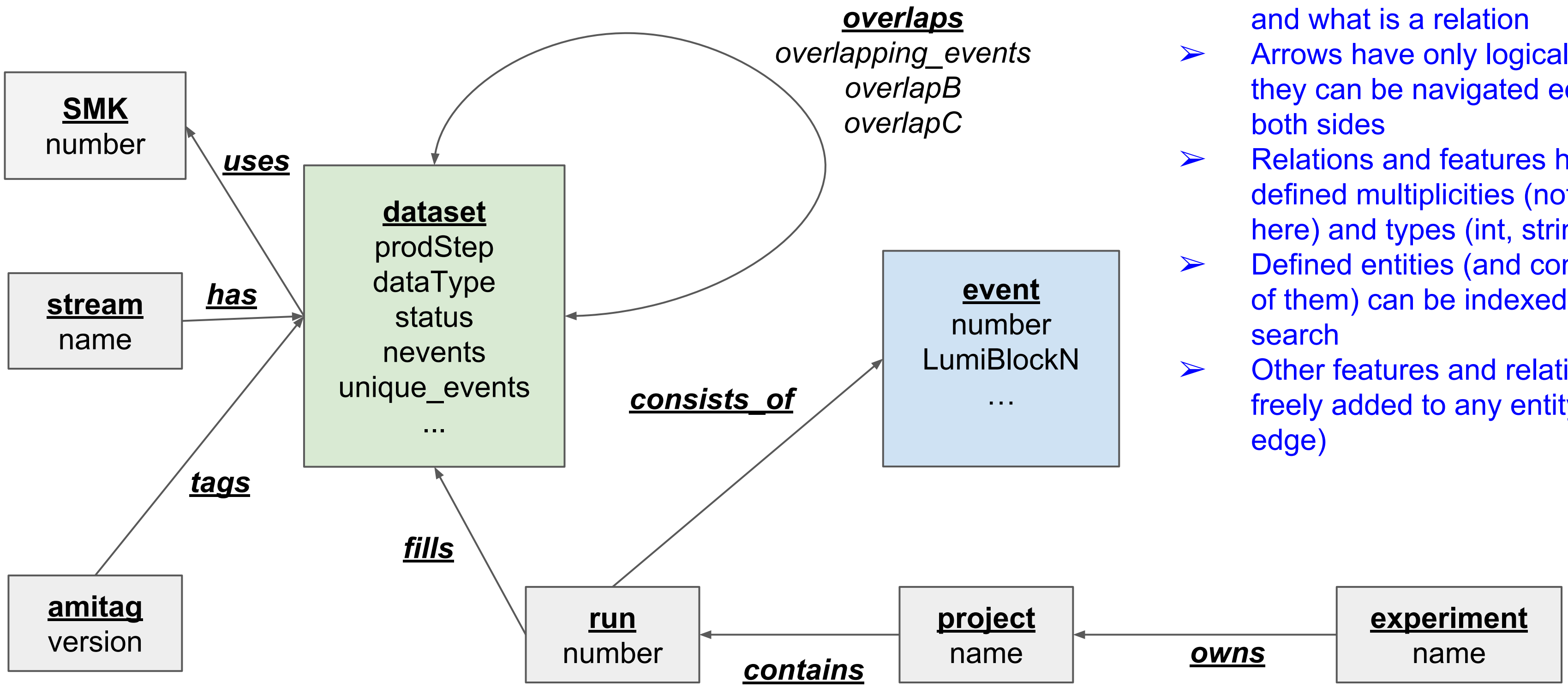
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
Catalog - Sample - Info -

--- commands output ---

ag:f933_m1960 # Showing 0 new elements # Showing 0 new elements # Showing 55 new elements # Showing 55 new elements #

GraphDB Schema


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769



SMK
number

stream
name

amitag
version

uses

has

tags

dataset
prodStep
dataType
status
nevents
unique_events
...

overlaps
overlapping_events
overlapB
overlapC

event
number
LumiBlockN
…

consists_of

fills

run
number

contains

project
name

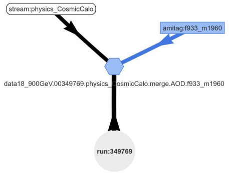
owns

experiment
name

- ➢ Should decide what is a feature and what is a relation
- ➢ Arrows have only logical meaning, they can be navigated equally from both sides
- ➢ Relations and features have defined multiplicities (not shown here) and types (int, string, date,...)
- ➢ Defined entities (and combinations of them) can be indexed for fast search
- ➢ Other features and relations can be freely added to any entity (vertex or edge)

Graph Database for EI

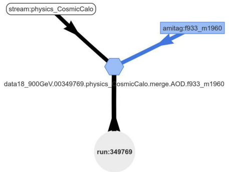
Plan - Datasets


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- Install JanusGraph on Hadoop server (aiatlas016,54)
  - With data stored in HBase
  - [J.G. is already installed and functional on aiatlas016, using BerkeleyDB]
- Replicate all Catalog & Overlap data
  - Scripts exist, just to run them
- Migrate Catalog WS, CLI, API (the same API, new implementation)
  - `Catalog` & `SimpleCatalog` classes

Graph Database for EI

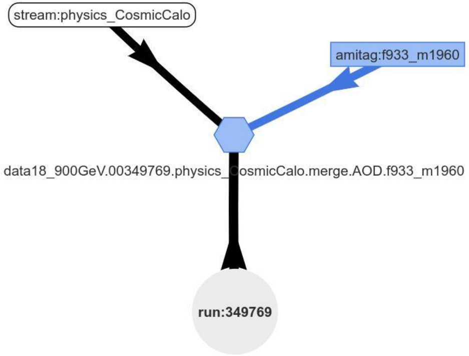
Plan - Events


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- **Structure of HBase backend of JanusGraph is very similar to our EventLookup HBase table (designed by Rainer)**
  - **So we can expect similar performance (i.e excellent performance)**
  - **Much simpler API: standard Gremlin instead of Rainer’ script & quite complex access code**
- Design & Create JanusGraph HBase for events
  - Including data, which are currently only in HDFS files
    - Also trigger
  - While GraphDB doesn’t require schema, we should define a reasonable structure
    - And formal schema/constraints (for some entities) can speed up searching
- Replicate all data
  - Connected to Catalog (already in JanusGraph)
- Migrate WS, CLI, API (the same API, new implementation)
- Setup new workflow
- Switch to new implementation
  - With existing CLI, WS

Graph Database for EI

Gremlin examples (Groovy style)

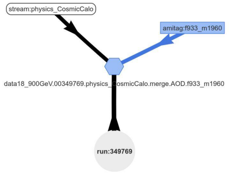

stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

```
# add a vertex ‘experiment’ with the name ‘ATLAS’
g.addV('experiment').property('name', 'ATLAS')
# add edges ‘owns’ from all vertices ‘project’ to vertex ‘experiment’ ‘ATLAS
g.V().hasLabel('project').addE('owns').from(g.V().hasLabel('experiment').has('name', 'ATLAS'))
# a function deriving a dataset name (which is not stored as such)
# from existing dataset relations by traversing the graph
def datasetName(d) {
  return d.sideEffect(values("prodStep").store("4"))
          .sideEffect(values("dataType").store("5")).in()
          .sideEffect(hasLabel("run").values("name").store("2"))
          .sideEffect(hasLabel("amitag").values("version").store("6"))
          .sideEffect(hasLabel("stream").values("name").store("3"))
          .sideEffect(hasLabel("run").in().hasLabel("project").values("name").store("1"))
          .cap("1", "2", "3", "4", "5", "6").next().values().join().toString()
  }
# in a similar way, one can navigate from an event to its dataset of certain dataType and amitag
```

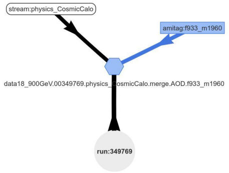
There is also SQL API to Gremlin

Graph Database for EI

Summary


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

- ➢ Existing implementation provides
  - ○ Excellent performance
  - ○ Flexible architecture - new requirements are implemented within hours
- ➢ Most data (Catalog + EL) are stored in HBase tables
  - ○ Using de-facto graph structure (entities with relations)
- ➢ Will migrate storage to GraphDB on HBase
  - ○ i.e. keep storage & migrate structure into standard implementation
- ➢ Will keep (and partially re-implement) API and UI (CLI+WS)
- ➢ Adiabatic change
  - ○ All clients keep working with the same interface
  - ○ Minimal perturbation
- ➢ Will lose possibility to (directly) use Hadoop M/R
  - ○ Those tasks should be re-developped
- ➢ Generic browser for all GraphDB data
  - ○ Atlas has many domains with graph-like data
  - ○ Graph databases are often used in AI (machine learning)


stream:physics_CosmicCalo
amitag:f933_m1960
data18_900GeV.00349769.physics_CosmicCalo.merge.AOD.f933_m1960
run:349769

Info

Service: https://atlas-event-index.cern.ch/EIHadoop
Home: https://atlas-event-index.cern.ch
FAQ: https://atlas-event-index.cern.ch/doc/faq
GIT: https://gitlab.cern.ch/atlas-event-index/Atlas-Event-Index-Core.git
GraphDB: https://gitlab.cern.ch/atlas-event-index/GraphDB.git
Questions & Comments: mailto:Julius.Hrivnac