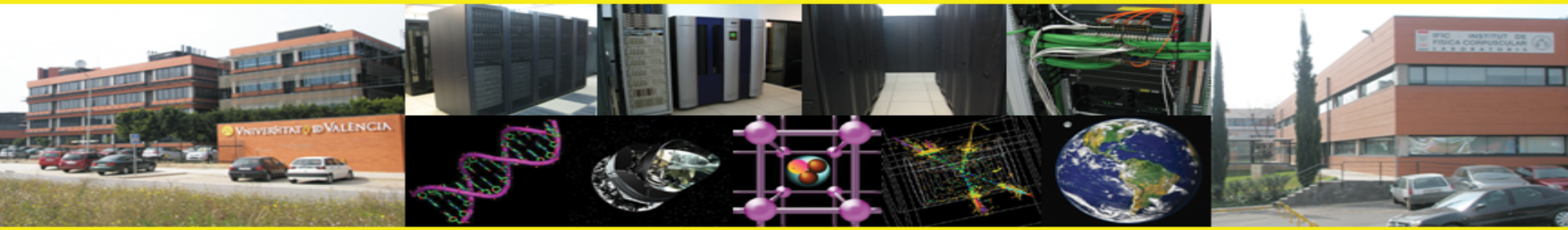
ATLAS EventIndex

Álvaro Fernández Casaní

IFIC computing

Training Course I-COOP+2016 project: COOPB20247

Valencia. July 2017



Outline

ATLAS experiment at CERN

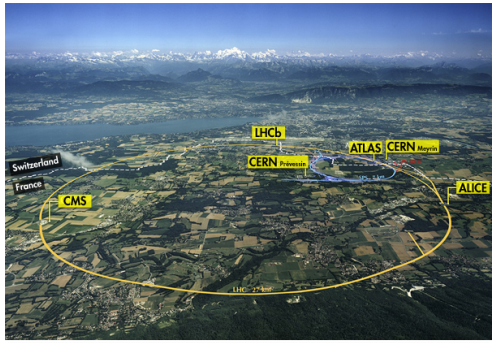
EventIndex project

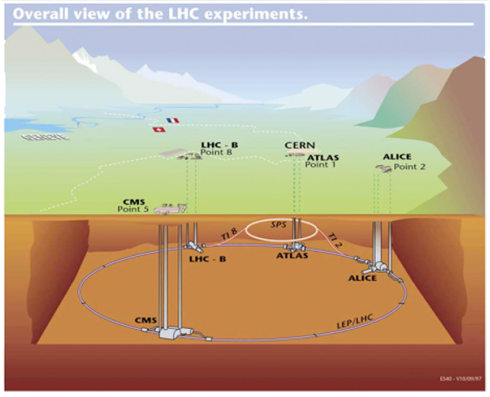
Architecture

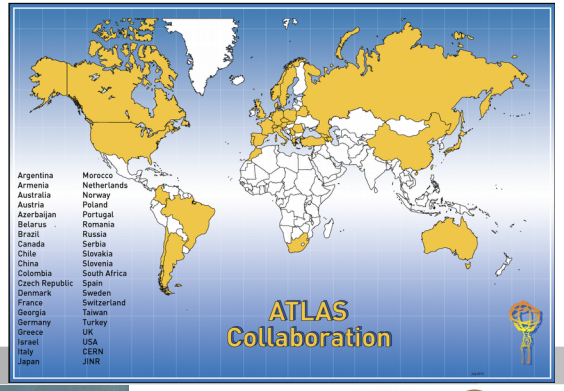
- Data Production
- Data Collection
- Data Storage

Summary

ATLAS experiment at CERN





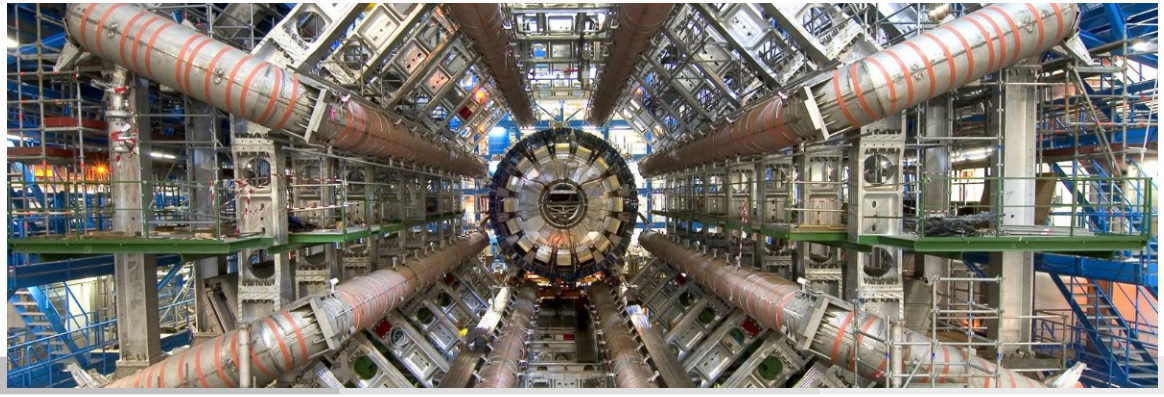


Large Hadron Collider (LHC) is a particle accelerator located at CERN in the border of Switzerland and France.

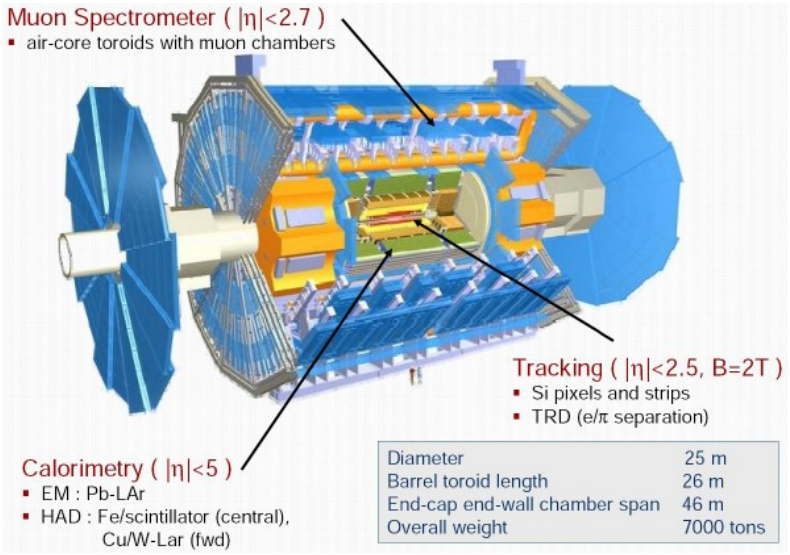
The circular tunnel has a length of 27 km, and is 175 meters below ground.

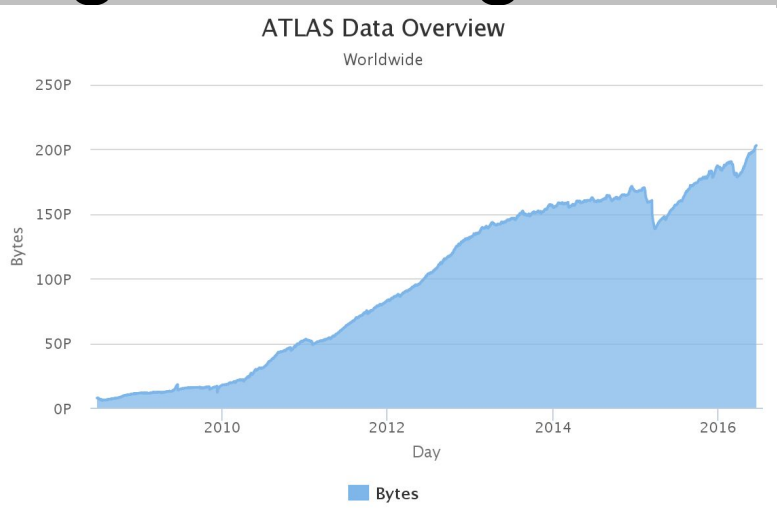
ATLAS is one the 4 big detectors, devoted to test the predictions of the Standard Model, that lead to the discovery Higgs boson in 2012, and to physics beyond the Standard Model and the development of new theories to better describe our universe.

ATLAS experiment is a **collaboration** of 5000 scientists from about 180 institutions around the world, representing 38 countries.



ATLAS Computing Challenges





•<u>The offline computing</u>:

- 2016: 1 Khz real data taking.10^9 events/year
- Average event size (raw data): 0.8 MB/event

<u>Processing</u>:

- >150 centres with thousands of cores

<u>Storage</u>:

- raw data recording rate 440 MB/sec
- Accumulating at 5-8 PB/year

<u>Data Access:</u>

- Physicist spread around the world. Grid technologies to access computing and data.

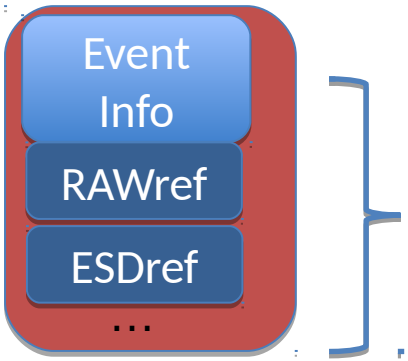
2016 milestone (run 2): 200 Petabytes

- GRID is used to solve problems of data simulation, storage, reprocessing and analysis.
- Data per year: ≈ Petabytes
  - event generation
  - simulation of what happens in the detector
  - reconstruction of an event from what happened in the detector

Predicted run 4 (2024): Exabyte scale

EventIndex: an event catalog

- **A catalog of data** (all events in all processing stages) is needed to meet multiple use cases and search criteria. A small quantity of data per event is indexed.



Events stored in files(identified by GUID)
GUIDs are grouped into DATASETS
Wanted Event Index information ~= 300bytes to 1Kbyte per event:

- Event identifiers (run and event numbers, trigger stream, luminosity block, BCID)
- Online trigger decissions
- References (pointers) to the events at each processing stage: Guid of the file that contains + pointer (for Event picking)
- **[RAW], [ESD], AOD, (DAOD) for real events**

EVNT, [RDO], [ESD], AOD, (DAOD) for simulated events

Use Cases

1) Event picking: users able to select single events depending on constraints. Order of hundreds of concurrent users, with requests ranging from 1 event (common case) to 30k events (occasional).

2) Count and select events based on Trigger decisions

3) Production completeness consistency checks

Duplicate event checkings: events with same Id appearing in same or different files/datasets.

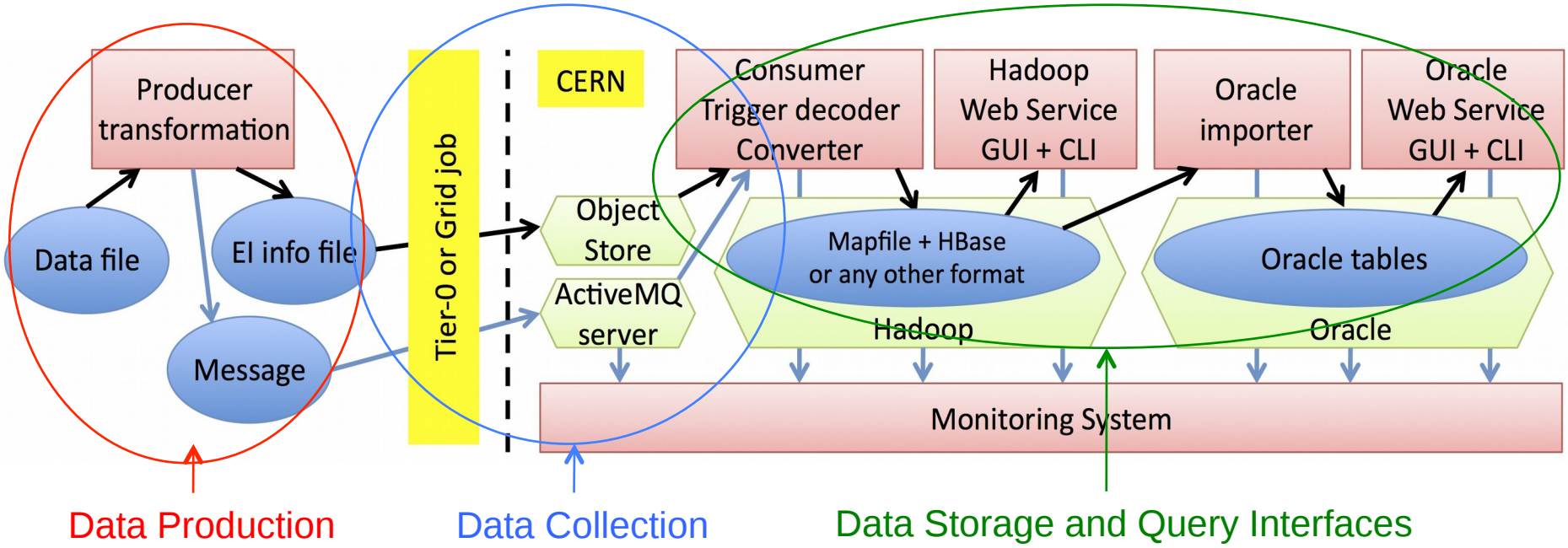
4) (Derivation) Overlap detection in derivation framework: construct the overlap matrix identifying common events across the different files.

5) Trigger chain overlap counting: number of events in a real data Run/Stream satisfying trigger X which also satisfies trigger Y.

https://twiki.cern.ch/twiki/bin/view/AtlasComputing/EventIndexUseCases

EventIndex Architecture

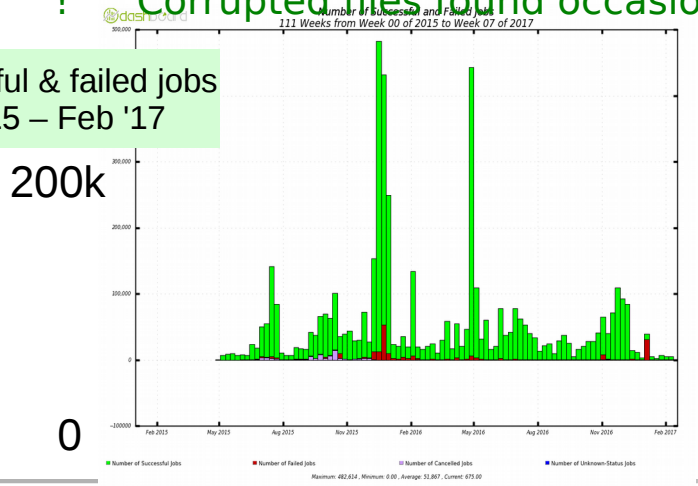
- **Data Production:** extract event metadata at Tier-0 and grid-sites
- **Data Collection:** reliably transfer this info to CERN
- **Data Storage:** permanent storage of all info (HADOOP) and a subset of info at ORACLE (only real data, no trigger).
- **Monitoring:** health of all services



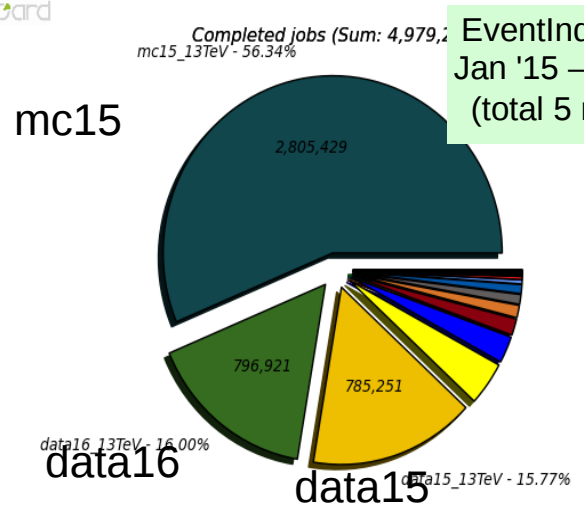
Data Production

- Tier-0 jobs index merged physics AODs, collecting also references to RAW and (if existing) ESD files
- Similarly, Grid jobs collect info from EVNT and AOD datasets as soon as they are produced and marked "ALL EVENTS AVAILABLE" in AMI
  - Other data formats (HITS, DAOD etc.) can be (and are) indexed on demand
  - Continuous operation since spring 2015
- System now in routine operation
  - Very low number of failures:
    - ! Site problems (fixed by retries)
    - ! Corrupted files found occasionally

Successful & failed jobs
Jan '15 – Feb '17



EventIndex jobs
Jan '15 – Feb '17
(total 5 million)



mc15_13TeV - 56.34% (2,805,429)
data16_hip8TeV - 4.00% (198,932)
data15_cos - 1.13% (56,232)
data15_comm - 0.36% (18,093)
mc15_5TeV - 0.10% (5,000)
mc15_14TeV - 0.03% (1,645)
mc15_valid - 0.02% (898.00)
group - 0.00% (31.00)
mc15 - 0.00% (1.00)
data16_13TeV - 16.00% (796,921)
data16_hip5TeV - 2.44% (121,374)
data16_cos - 0.88% (43,894)
data12_8TeV - 0.27% (13,486)
unknown - 0.09% (4,260)
mc14_13TeV - 0.03% (1,330)
mc15_7TeV - 0.00% (57.00)
valid3 - 0.00% (10.00)
mc12_valid - 0.00% (1.00)
data15_13TeV - 15.77% (785,251)
data15_hi - 1.50% (74,560)
data15_5TeV - 0.82% (40,695)
data16_comm - 0.16% (7,802)
mc16_13TeV - 0.05% (2,339)
mc15_8TeV - 0.02% (959.00)
data15_900GeV - 0.00% (50.00)
mc15_900GeV - 0.00% (9.00)

Dario Barberis: ATLAS EventIndex

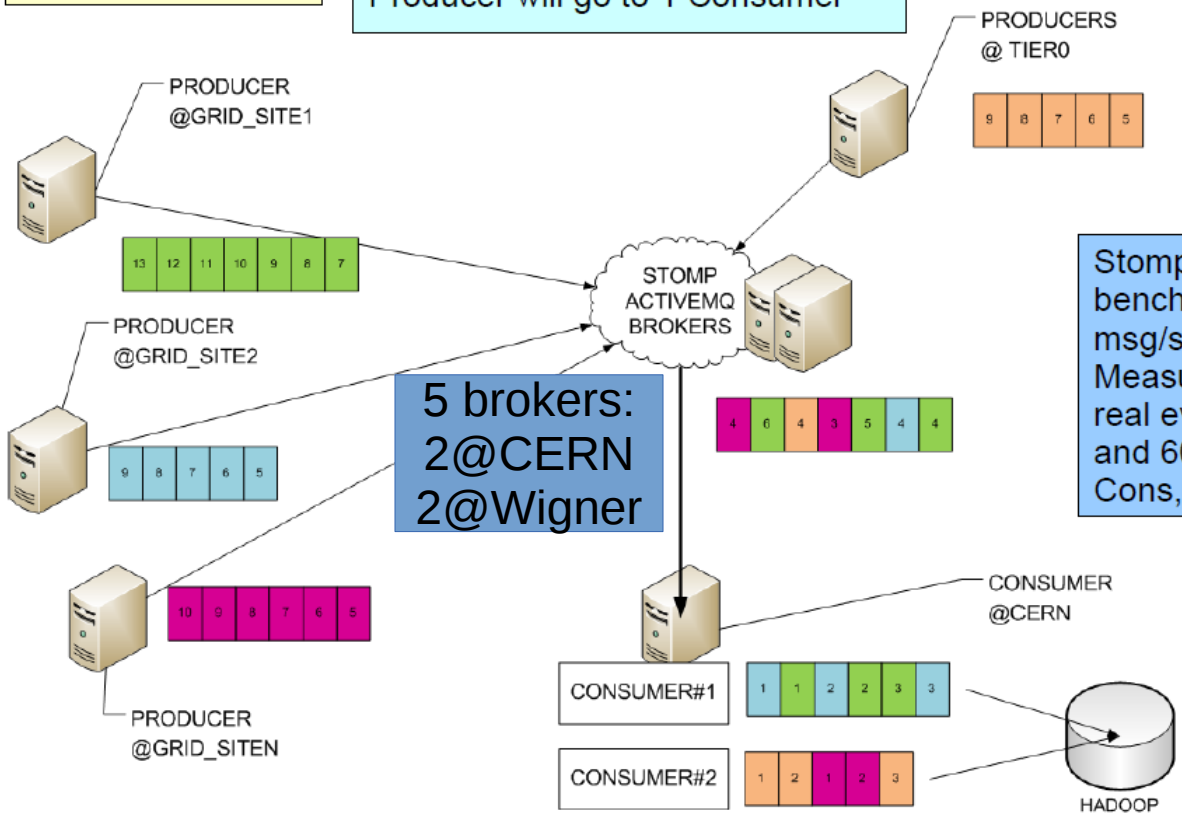
Message Flow

Message size is set small 1-10kB to keep broker queues agile.

Producers tag messages by group (JMSXGroupID)

Ensures that all messages from 1 Producer will go to 1 Consumer

Atomic transactions on Producers:
if connections breaks no partial processing occurs.

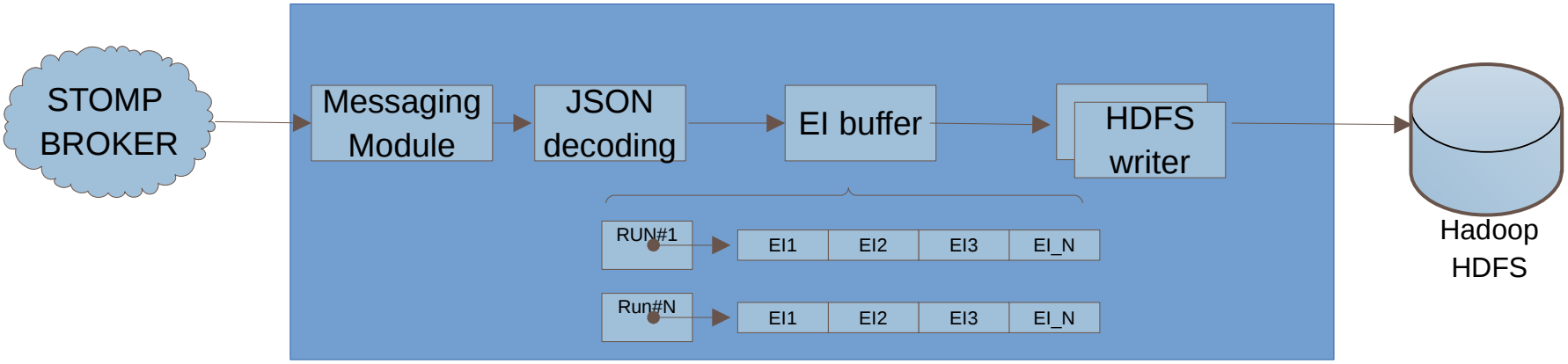


Status messages are sent from producers and consumers to an alternate queue.

Stomp performance in our benchmarks reaches over 350 msg/s and 10Mb/s per Producer. Measured performance for sending real events reached 200K event/s and 60Mb/s (1Broker, 6 Prod/s, 4 Cons, 50K events/job)

Files stored on Mapfile format usable by Hadoop Core Services

Consumer (Hadoop Backend storage)



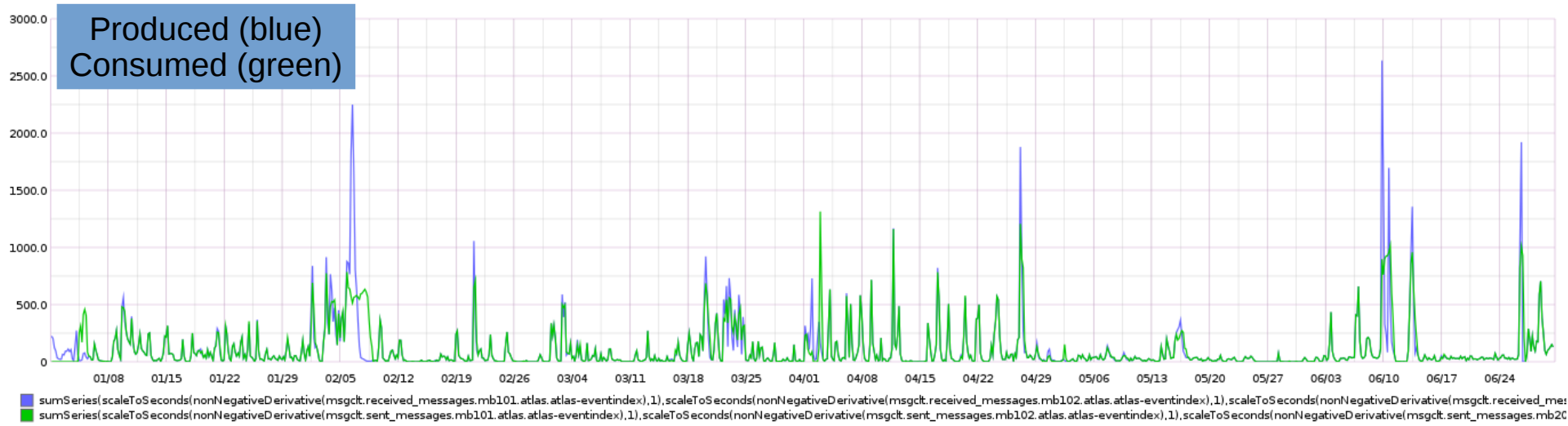
Multithreaded Java Consumer.

Consumer connects only to 1 broker. Messages are received, Json decoded, and queued by GUID.

Writer module writes data in HDFS. Current approach is to write a Mapfile per GUID, grouped by dataset directory.

Consumer sends back statistic messages (# messages, # files, # events) to broker to report status. Agents subscribe to these to monitor progress and healthy working.

Data Collection Performance (2016)

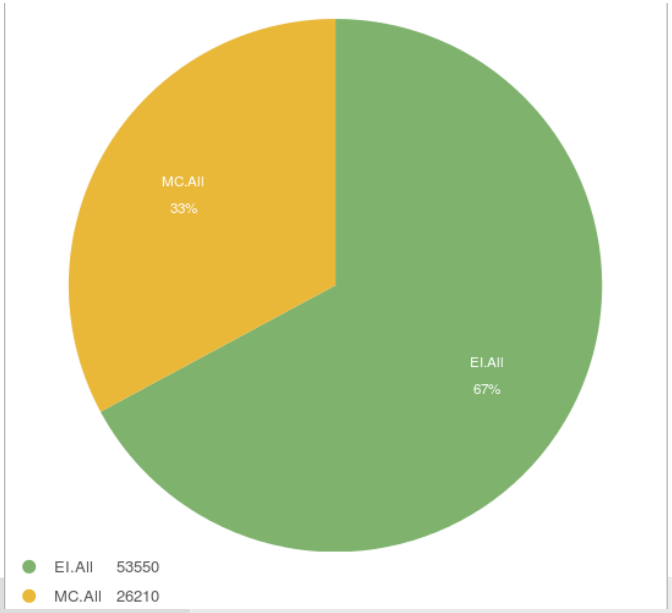


Messaging Statistics

- 4 Brokers. 3 long-lived Consumers per broker. Varying number of simultaneous short-lived Producers.
- More than 10^{12} messages handled.
- Usual rate of 100 msg/s produced. Peaks of >3500 msg/s, not consumed instantaneously (possible future congestion problems)

Current EventIndex Data in Hadoop:

- 122 TB of events data (89 TB real data, and 33 TB MonteCarlo simulated data)

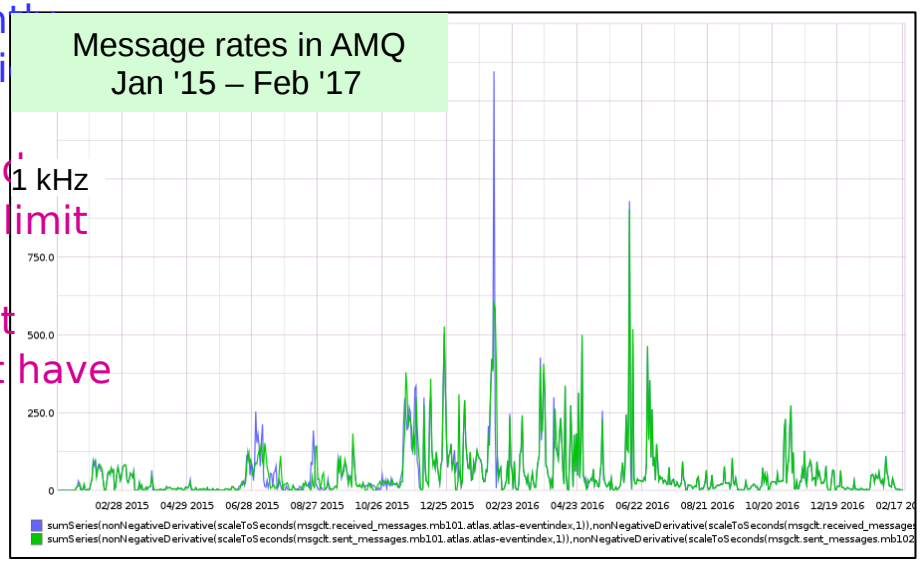


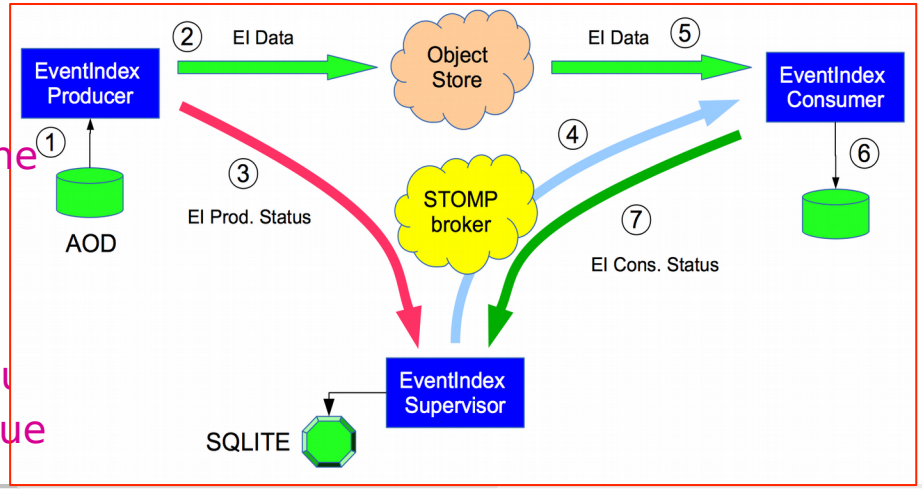
Data Collection: Transport

- The ActiveMQ message brokers are current
  used to transfer EI info from Tier-0 and Gri
  jobs to CERN
  - All OK with 5 brokers (3 in Geneva and
    2 in Wigner), but getting close to the limit
    of scalability
  - ActiveMQ forces splitting the info sent
    by each job into many messages that have
    to be recombined at destination

NEW! New development: replace ActiveMQ as transport mechanism with sending EI info through an Object Store

  - Sending a single file with EI info from
    Grid job can save some headache
  - Testing now with sending the file to the
    S3 Object Store at CERN
    - ! Retrial and failover policy being discussed
  - Needs changes to Producer and Consu
  - Notifications and statistics will continue to use the ActiveMQ servers





Data Collection: Validation+Storage

- **Now:**
  - Consumers read data from AMQ (Object Store in near future)
  - Validation is done at file and dataset level (number of events, duplicates etc.)
  - Trigger is decoded for each event to store trigger chains instead of bit patter
  - Data are imported to Hadoop (organised by dataset)

NEW! **Under development:**

- Merge all these steps into one
  - ! Avoid the unnecessary intermediate "many small files" in HDSF and speed up the data flow
- Make it more automatic than it is now
  - ! Including fault reporting, duplicate events found etc.

NEW! **New data flow monitoring tool and browser under development (EventIndex Supervisor):**

Datasets Tier 0

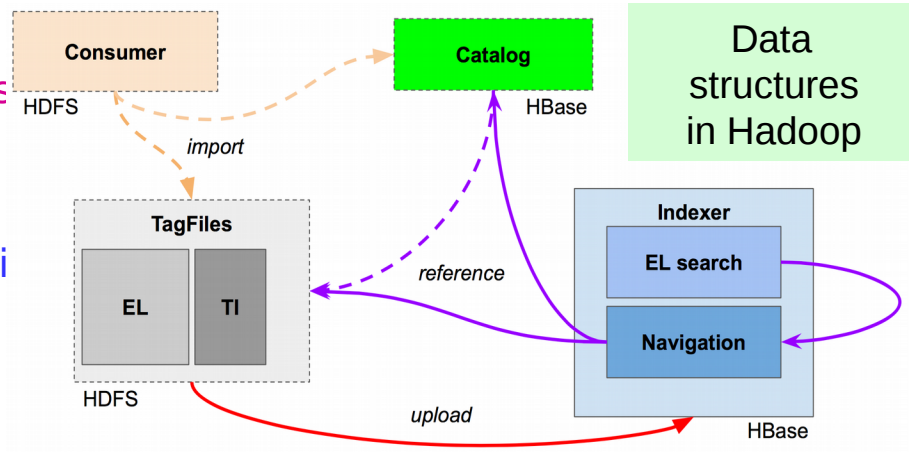
Filters

Show All entries Search within results

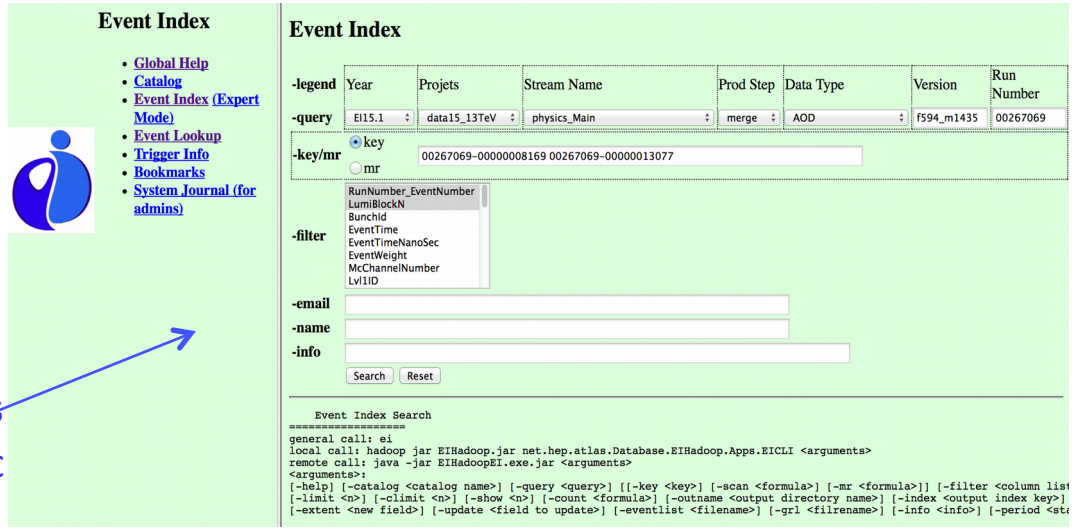
Dataset	TaskId	Creation	#Jobs	#Finished	#Produced	#Consumed
data16_13TeV.00296938.physics_CosmicCalo.merge.AOD.f684_m1576	2092083	Sun, 24 Apr 2016 10:03:00 GMT	2	2	2	2
data16_13TeV.00296939.express_express.merge.AOD.f685_m1576	2092584	Mon, 25 Apr 2016 19:40:00 GMT	38	38	38	38
data16_13TeV.00296939.express_express.merge.DAOD_SCTVALID.f685_m1579	2092583	Mon, 25 Apr 2016 19:40:00 GMT	12	12	12	12
data16_13TeV.00296939.physics_CosmicCalo.merge.AOD.f685_m1576	2092571	Mon, 25 Apr 2016 19:11:00 GMT	103	103	103	103
data16_13TeV.00296939.physics_L1Calo.merge.AOD.f685_m1576	2092630	Tue, 26 Apr 2016 09:40:00 GMT	26	26	26	26

Hadoop storage and queries

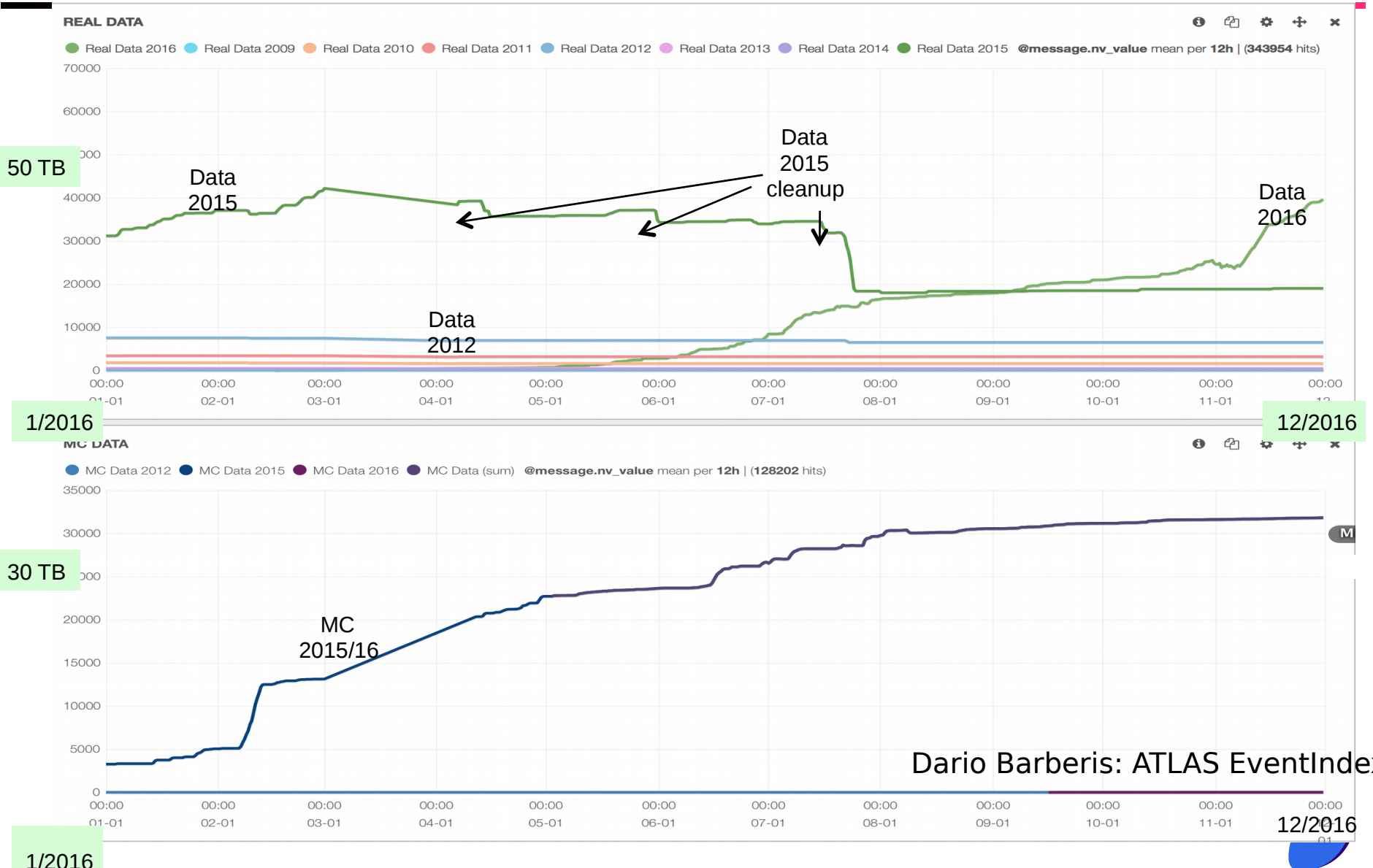
- Hadoop is the baseline storage technology
  - It can store large numbers (10s of billions) of simply-structured records and search/retrieve them in reasonable times
- Hadoop "MapFiles" (indexed sequential fi are used as data format
  - One MapFile per dataset
  - Internal catalogue in HBase (the Hadoop database) keeps track of what is where and dataset-level metadata (status flags)
  - Event Lookup index in HBase
- Data volumes:
  - Real 2009-2016: 89 TB
  - Simul mc15-mc16: 33 TB
  - Other (incl. backup): 126 TB
- CLI, RESTful API and GUI interfaces available for data inspection, searc and retrieval



Dario Barberis: ATLAS EventIndex



EI Hadoop data volume



50 TB

1/2016

12/2016

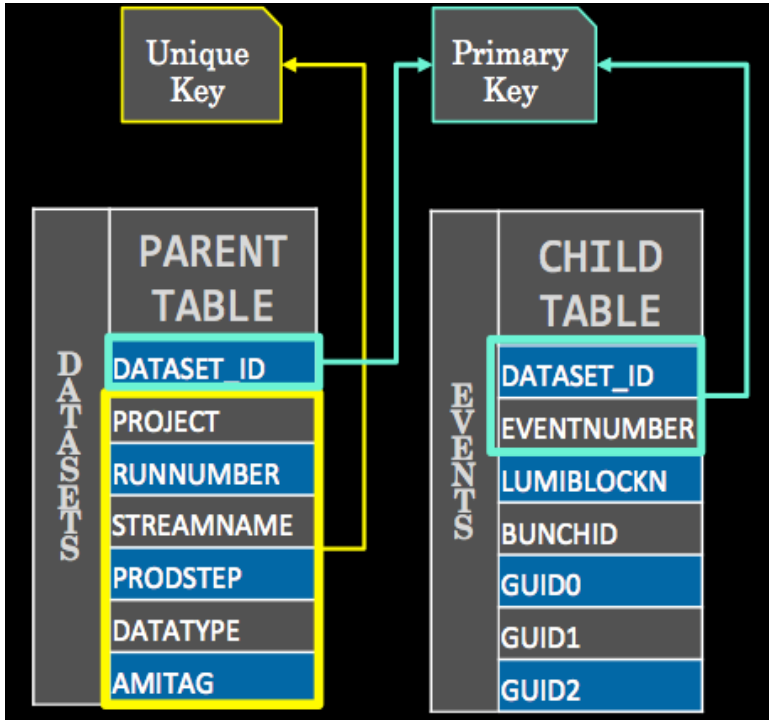
30 TB

1/2016

12/2016

Dario Barberis: ATLAS EventIndex

EI Oracle storage



- Simple schema with dataset and event tables
  - Exploiting the relational features of Oracle
- Filled with all real data, only event identification and pointers to event locations
  - Optimised for event picking
  - Very good performance also for event counting by attributes (lumi block and bunch ID)

- Connection to the RunQuery and AMI databases to check dataset processing completeness and detect duplicates
- Easy calculation of dataset overlaps
- GUI derived from COMA database browser to search and retrieve info (next slide)

Currently:

- 48.2 billion event records
- stored in table of 925 GB
- plus 838 GB index

EIO data browser

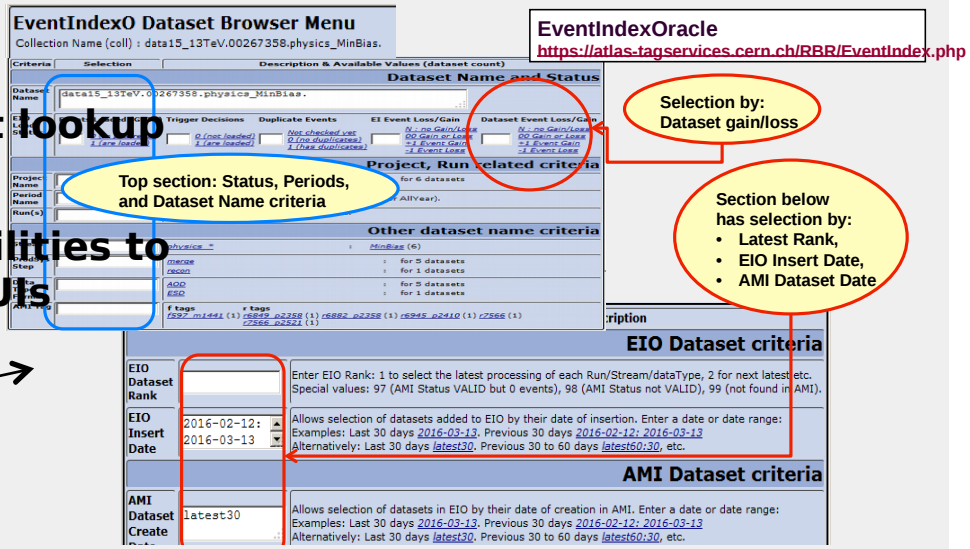
https://atlas-tagservices.cern.ch/tagservices/RunBrowser/runBrowserReport/EventIndex.php

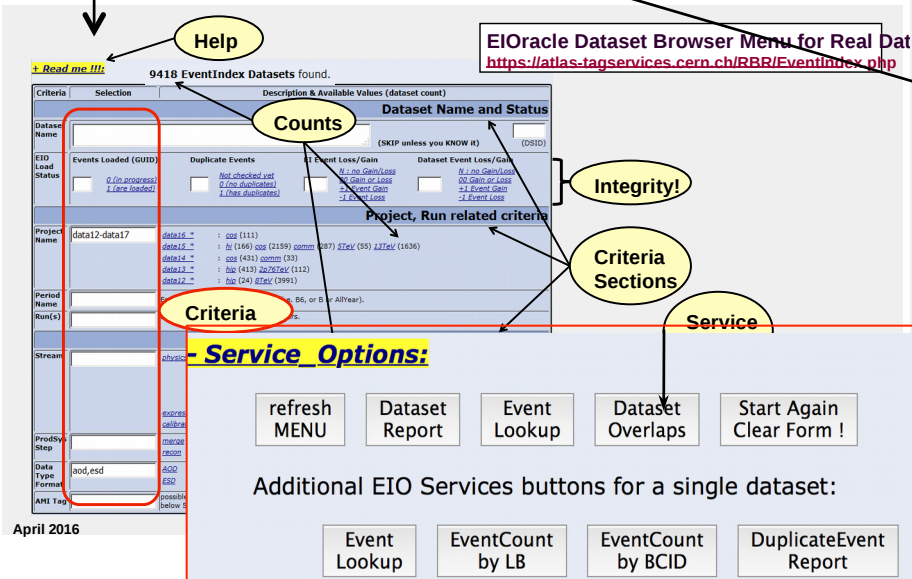
Entry point for dataset search and event lookup

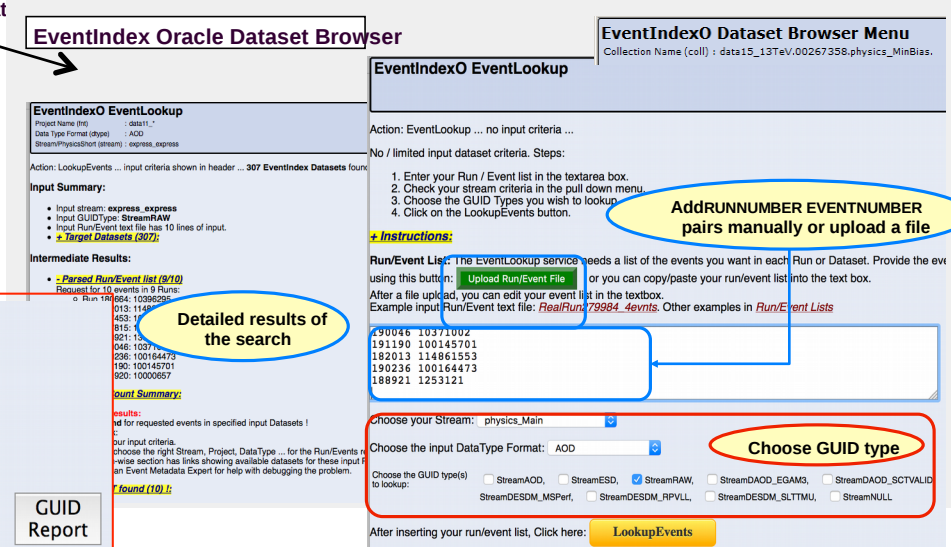
Real data only

Extensive use of the relational DB capabilities to implement the search and retrieve GUIs

1) **Search dataset**
2) **Refine search**
3) **Get results**







April 2016

Summary

ATLAS EventIndex project was presented.

System is currently running at production level.

- Indexed billions of events from thousand of grid jobs running in distributed manner worldwide.
- More information in: “The ATLAS EventIndex: Full chain deployment and first operation”. D. Barberis, J. Cranshaw, A. Favareto, A. Fernández Casaní, et al. Nuclear and Particle Physics Proceedings (2016), pp. 913-918. DOI information: 10.1016/j.nuclphysbps.2015.09.141

Currently work in the areas of

- Distributed Data Collection: improve scalability for future runs. (IFIC)

BACKUP

New EI format description I

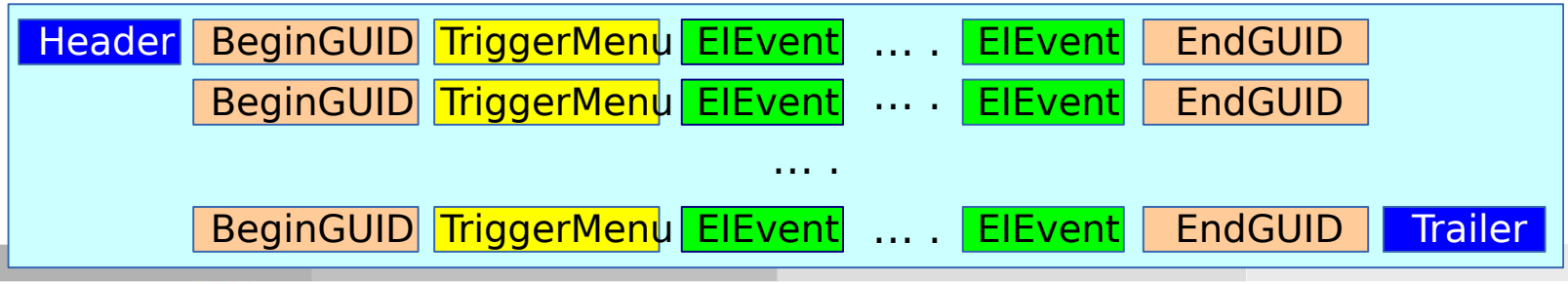
New EI file contains a stream of Google Protocol Buffer messages (SPB). In addition file is compressed using gzip library (on the fly).

PB messages does not have type information, so it is necessary to prepend type and length before the message itself.

- type: byte
- len: varint

6 different message types:

- H: Header.
- T: Trailer
- B: BeginGUID
- E: EndGUID
- T: TriggerMenu
- X: EIEvent



Event sizes with new format

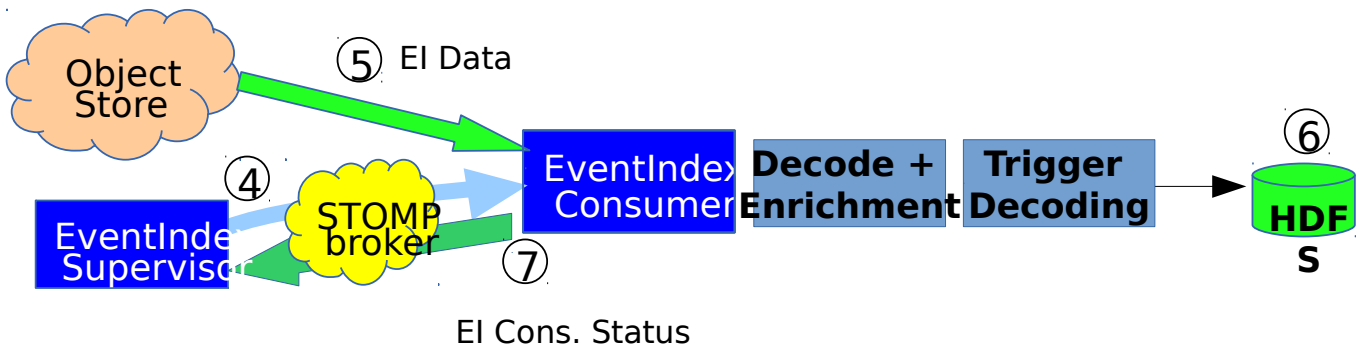
data15_13TeV:data15_13TeV.00279515.physics_Main.merge.AOD.r7562_p2521

taksid	jobid	#files	usize	size	# events	# uevents	ubytes/evt	bytes/evt	
7800581.G	2776051765.0	9	58,927,969	2,809,591	99594	99594	592	28	no trigger
7803056.G	2776553110.0	9	91,502,438	9,546,191	99594	99594	919	96	
			45,759,437		99594	99594	459		stomp

mc15_13TeV:mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.

taksid	jobid	#files	usize	size	# events	# uevents	ubytes/evt	bytes/evt	
7800586.G	2776057428.0	10	171,587,135	15,299,286	100000	99600	1716	153	
7800586.G	2776057435.0	10	171,509,364	15,324,929	100000	99000	1715	153	
7800586.G	2776057438.0	10	171,423,769	15,316,530	100000	100000	1714	153	
7800586.G	2776057443.0	10	171,520,830	14,912,468	100000	99600	1715	149	
7800586.G	2776057448.0	10	171,640,128	15,270,769	100000	100000	1716	153	
7800586.G	2776057457.0	10	171,495,661	14,637,193	100000	99600	1715	146	
7800586.G	2776057499.0	10	171,528,968	15,273,445	100000	100000	1715	153	
7800586.G	2776057542.0	10	171,382,043	14,877,061	100000	99800	1714	149	
7800586.G	2776057608.0	10	171,588,475	15,309,369	100000	100000	1716	153	
7800586.G	2776173262.0	10	171,424,648	14,964,659	100000	100000	1714	150	
			130,693,810		100000	100000	1307		stomp

ObjectStore Consumer



Step 4 Validation Messages use ActiveMQ Broker. Json formatted, each message represents a valid dataset, and contains a number of urls tuples: [dafile_url, data_index]. Example:

s3://atlas_eventindex/valid/panda/2016/04/08ba84d75f744c90841ca8d5e1f9254e.valid

{"**nfiles**":2000,"**dsname**":"mc15_13TeV:mc15_13TeV.305387.Pythia8EvtGen_A14NNPDF23LO_CI_plusLL_Lambda15TeV_JZ6W.evgen.EVNT.e5012","**events**":1000000,"**urls**":

[{"**url**":"s3://cs3.cern.ch:443/atlas_eventindex/panda/8331606/8331606.G_2845544335.0_0c50ce5e2a6f4ee1a252adbfbc3cccc7.ei.spb","**files**":[0]},

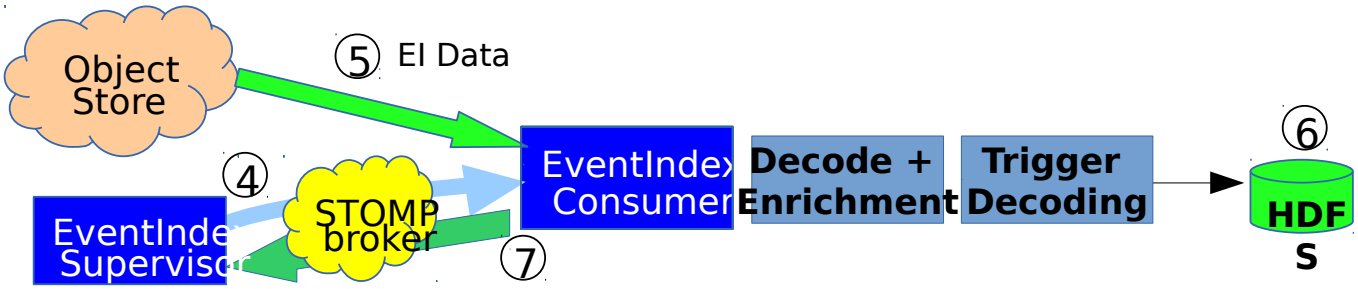
{"url":"s3://cs3.cern.ch:443/atlas_eventindex/panda/8331606/8331606.G_2845572042.0_e9d6d3e9e5974cf9b7fd69814a5db78e.ei.spb","**files**":[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19]},

…

],"**uevents**":1000000}

Potential validation per complete dataset, or tid: don't wait to complete datasets (still reducing the previous number of received not validated GUIDs with messaging)

ObjectStore Consumer II



For each tuple, consumer retrieves from Object store the data (step5):

- Using **amazonaws s3** java packages. Done
- Streams data directly (no storage on local disk). Decompressed on the fly. Done
- Processing only the events for the indicated files. (according to validation info sent in step4). Done

Data decoding and enrichment chain

- Decoding of all new Protocol buffer format provides more information. Done
- Data enrichment: simplifies cataloguing by HadoopCore task. Done (to be tested)
- Trigger decoding: enriched protobuf information able to decode triger. In development

HDFS Writing

- New Protocol buffer format includes new fields. Currently development version writes in HDFS Mapfiles same information as production and 1 mapfile per GUID (backward compatible). Done
- Consolidation phase to group files. Done (to be tested)
- Working in reducing number of HDFS files, grouping GUIDs, and not necessarily ordered (SeqFiles, or future Parquet or other format). With later consolidation phase if needed. In development

Currently testing using test-broker (mb099.cern.ch) and HDFS Test Consumer area (ConsumerData/test)

Review of USE CASES

a - event picking for many events in RAW and AOD format

b - trigger counting

c - duplicate event checking for EVNT and AOD

A) event picking for many events in RAW and AOD format

Insert rate of 1 kHz (real data from Tier-0)
plus 2 kHz during reprocessing campaigns,
plus >0.5 kHz (the average simulated AOD rate over last 4 months); let's say 5 kHz including some contingency, with a few 10s datasets active at any point in time.
Requests for event picking of a few events almost continuously plus maxi-requests up to 30k events occasionally.

B) trigger counting

Count for each real data run (approximately one a day) all triggers (including some combinations?); return the results in a finite and not-too-long time (Elizabeth to quantify better). Only datasets produced by Tier-0 are used for this, as the reprocessed data won't contain trigger info any longer.

c) duplicate event checking for EVNT and AOD

Checking duplicate events within a dataset must accept input rates of 5 kHz for AODs (see above) and over 20 kHz for EVNT. The structure used for these checks, at least for EVNT data, are not necessarily the same as we can use for event picking as nobody cares about picking EVNT.

So we could have a more table-like structure only for EVNT duplicate detection and a more storage-like structure for info coming from AODs, including trigger and provenance.

Duplicates checking

Was being done on demand with a map-reduce job per dataset:

- Added extra job on Hadoop
- Not clear reporting schema defined.

Time consuming job, had to be integrated into framework:

- Now integrated in Julius' Hadoop Importing (still map-reduce job per dataset granularity)
- Users could check collections of datasets.

Future will be integrated in the Consumer chain.

Discover overlaps

An event is reprocessed and stored in several formats (several output files along the time).

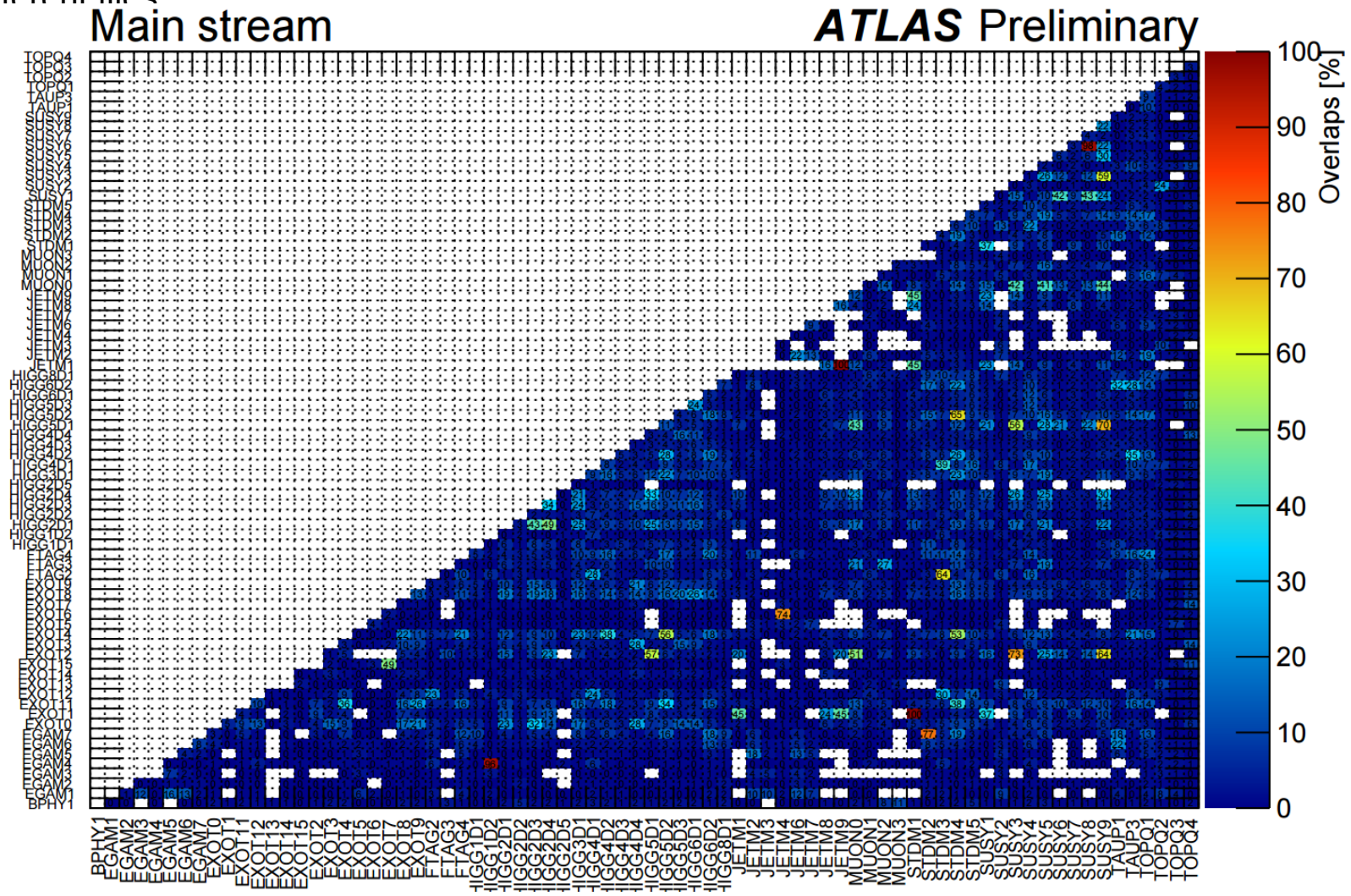
For derivation framework, currently there are n streams being produced which will be spread among several trains (processing jobs) and will end in n files.

So 1 input file, n output files: The event overlap between these needs to be monitored.

Wish to find out how many, and which, events end up in each stream. For a number of datasets(input files)

Overlap matrix

For Event Index, means construct the overlap matrix identifying common events across the different files



data15_13TeV.00267639.physics_Main.merge.DAOD_XXXX.f598_m1441_p2361

73 streams – 8369 files – 85Gb compress / 160 Gb .csv data - Aprox 10 hours computation in a single machine. 5.5 hours in Hadoop cluster

New EI format description II

Header			
	uint64	startProcTime	Job start time stamp (ms)
	string	taskID	Task ID + 'G' or 'T'
	string	jobID	Job ID + Attempt
	string	inputDsName	Input dataset name
	bool	provenanceRef	Is provenance included ?
	bool	triggerInfo	Is trigger included ?
Tailer			
	uint64	endProcTime	Job end time stamp (ms)
	uint32	nentries	total number of events
	uint32	nfiles	number of files processed
beginGUID			
	uint64	startProcTime	guid start processing time stamp (ms)
	string	AMITag	AMI Tag
	string	trigStream	Trigger stream
	string	projName	Project Name
	string	guid	guid
endGUID			
	unit64	endProcTime	guid end processing time stamp (ms)
	uint32	nentries	number of events in this guid

New EI format description III

EIEvent			
	uint32	runNumber	EvenInfo EventID
	uint64	eventNumber	EvenInfo EventID
	uint32	lumiBlock	EvenInfo EventID
	uint32	timeStamp	EvenInfo EventID
	uint32	timeStampNSOffset	EvenInfo EventID
	uint32	bcid	EvenInfo EventID
	uint32	extendedLevel1ID	EventInfo TriggerInfo
	bool	isSimulation	EventInfo EventType
	bool	isCalibration	EventInfo EventType
	bool	isTestBeam	EventInfo EventType
	string	L1PassedTrigMask	TrigDecisionTool
	string	L2PassedTrigMask	TrigDecisionTool
	string	EFPassedTrigMask	TrigDecisionTool
	uint32	SMK	Metadata
	uint32	HLTPSK	Metadata
	uint32	L1PSK	Metadata
	float	mcEventWeight	EventInfo EventType
	uint64	mcEventNumber	EventInfo EventType
	uint32	mcChannelNumber	EventInfo EventType
	*EItoken	eitoken	eitoken
EItoken			
	string	name	stream name
	string	token	pool/root token

New EI format description IV

TriggerMenu			
	uint32	SMK	Master Key
	uint32	L1PSK	L1 Prescaler Key
	uint32	HLTPSK	L2 Prescaler Key
	string	L1Menu	L1 Menu
	string	L2Menu	L2 Menu (if exists)
	string	EFMenu	EF Menu (if exists)
	string	HLTMenu	HLT Menu (if exists)