Trigger Commissioning Meeting

Atlas Meeting of
DESY / HU Group
9. June 2006 at DESY Zeuthen

Report on the Trigger Commissioning Meeting at CERN, 2. – 4. May 2006

Martin zur Nedden
Humboldt-Universität zu Berlin

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issues** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

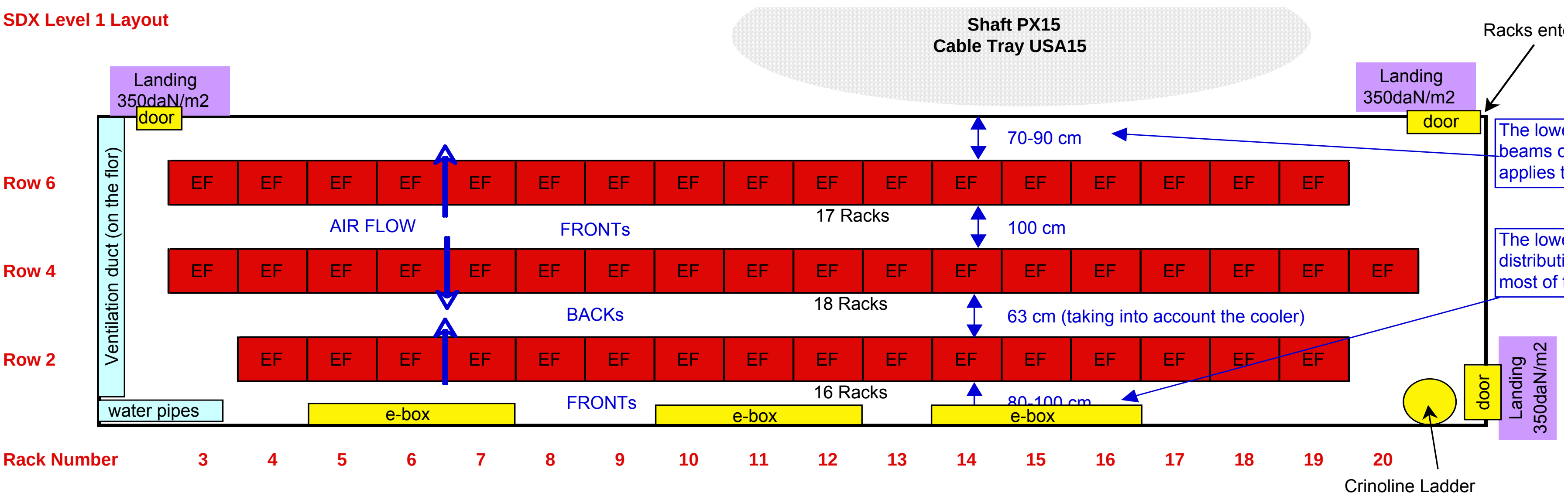
  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Hardware Issues

- Review the issues that can affect the algorithms and the performance of the HLT trigger related to the hardware we can buy:

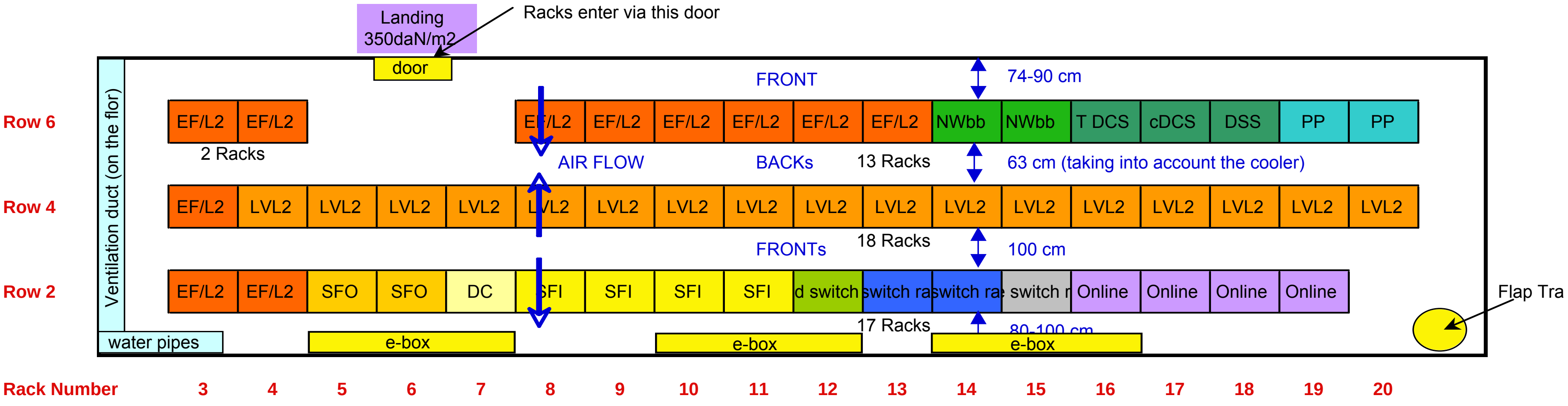
  - The use of multicore processors
  - The availability of hardware for 2006/2007
  - When to buy and how much?
  - Make sure the correct input from the algorithms is given in order to plan ahead

Rack Allocation in SDX



SDX Level 1 Layout

Rack Number	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Row 6	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	
Row 4	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF
Row 2		EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	EF	



SDX Level 2 Layout

Rack Number	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Row 6	EF/L2	EF/L2				EF/L2	EF/L2	EF/L2	EF/L2	EF/L2	EF/L2	NWbb	NWbb	T DCS	cDCS	DSS	PP	PP
Row 4	EF/L2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2
Row 2	EF/L2	EF/L2	SFO	SFO	DC	SFI	SFI	SFI	SFI	d switch	switch ra	switch ra	switch r	Online	Online	Online	Online	

HLT Racks

- **Rack weight limit ~ 800 kg**
- **Weight limit for HLT procs/rack ~600 kg**
- **Power limit per rack ~10 kW**
- **Plan for each 1U HLT PC to weight < 20 kg and use < 300 W => 30 HLT PC's per rack**
- **Rack allocations for final SDX system (2009 ?)**

Rack Type	Number		Procs/Rack	Total # Procs	
	2007	2009		2007	2009
L2 Racks	9	17	30	270	510
L2/EF Racks	11	11	30	330	330
EF Racks	16	51	30	480	1530

HLT PC’s

- **2 sockets**
- **Each socket with either**
  - Dual-core 3.2 GHz Intel / 2.2 GHz AMD - available now
  - Quad-core 2.8 GHz Intel / 1.8 GHz AMD - available during 2007
    - Remember TDR assumed 8GHz single-core, but scaling not ideal - ~6 kSI2000 per box
- **RAM - at least 1 GB/core**
  - (I.e. per parallel event - worker thread or application)
- **20 kg**
- **300 Watts - the hardest constraint to meet!**
- **Two GbE ports (1 for data + 1 for control)**
- **IPMI v2 (perhaps with additional ethernet port) for farm monitoring/management**
- **USB port for system management**
- **Small disk (e.g. 80 GB SATA drive)**

CPU’s

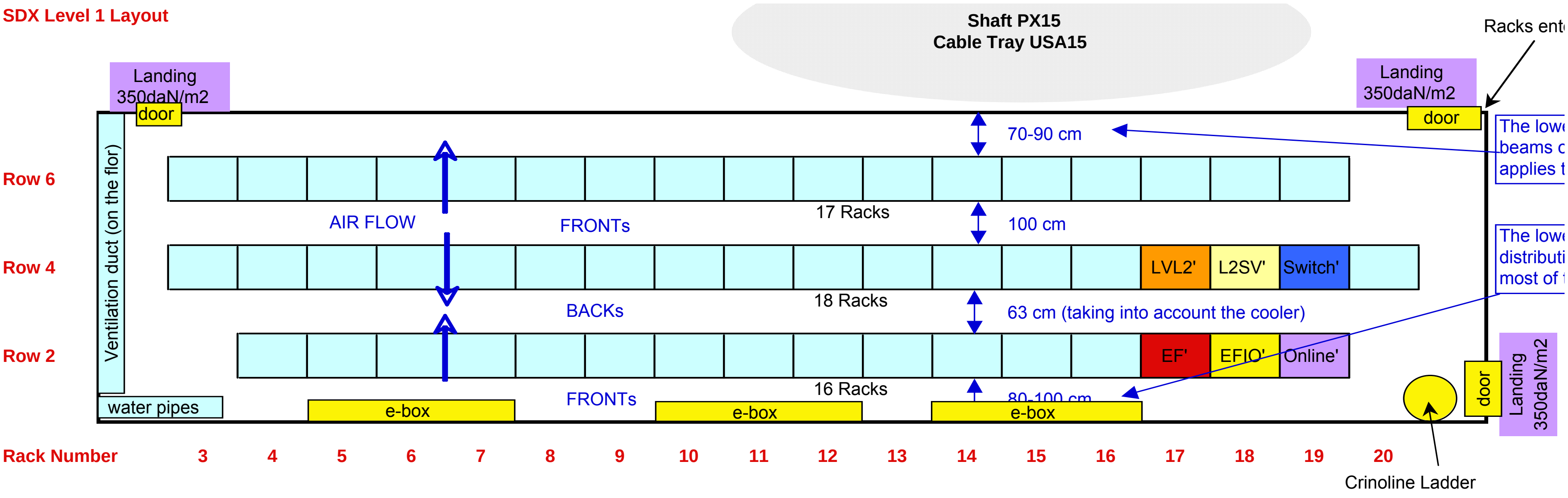
- “Brick wall” at ~4 GHz (Power used becomes excessive)
- Opterons have 95W envelope
  - 2 Sockets + all the rest ~300W
  - Dual-cores available for last 12 months in 90 nm technology
  - Latest versions higher clock speeds (2.4+ GHz) + DDR2
  - Quad-cores move to 65 nm - launch in 2007
- Intel Xeons have 135W envelope (but some even higher)
  - 2 Sockets + all the rest ~500W
  - Dual-cores (Dempsey) about to be launched in 65nm technology
  - Clock speeds up to 3.46 GHz (3.6 GHz possible)
  - Quad-cores (Clovertown) promised for 2007 (45 nm)
- Low power versions promised by both manufacturers
  - Opteron HE - 55W envelope, but slower clocks (2 steps back)
  - Intel (Woodcrest) - 95W (possibly more initially, but then lower)
    - Launch in 3Q06

So what?

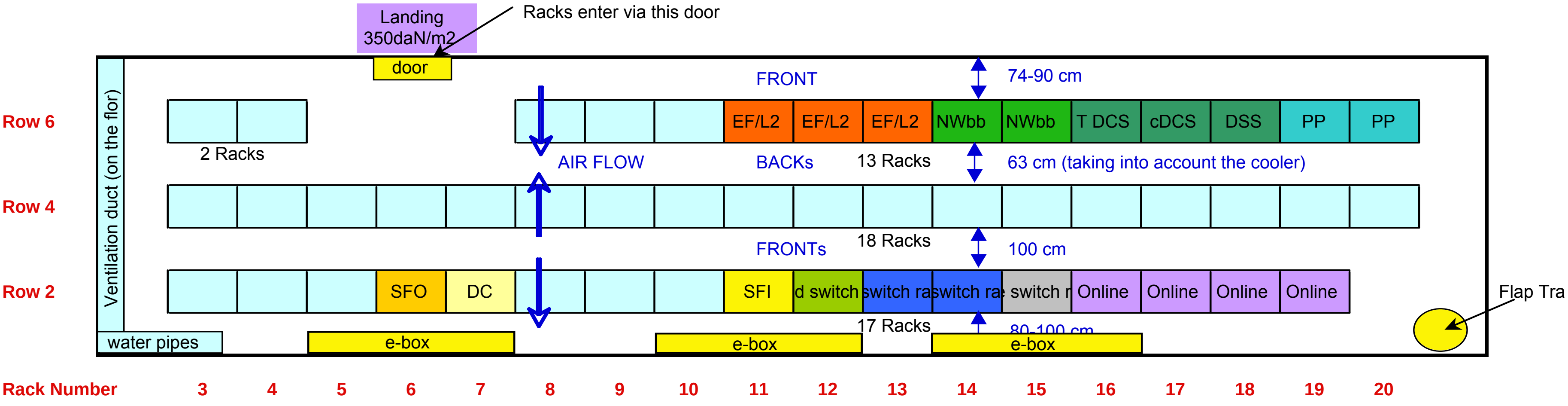
- Need multi-cores to achieve the processing power
- To obtain this processing need at least 1 thread/application per core
- Thus dual-socket quad-core has to run 8 threads/application (L2PU or PT) per box
- Many more applications to configure and control
  - ~5000 L2PU’s and ~15000 PT’s
  - Even in 1 rack have to configure and control up to ~250 applications
    - If each application needs 100 MB data then a total of 25 GB per rack - if done serially via a single GbE link takes > 5 mins
    - Do we need 2 servers per rack ?
- Latency of each event is longer, but run more events in parallel
  - 10 ms for L2PU assumed 8GHz => ~30 ms for quad-core 3GHz
  - 1 s for PT assumed 8 GHz => ~3 s for quad-core 3GHz

Rack Allocation in SDX - 2006

SDX Level 1 Layout

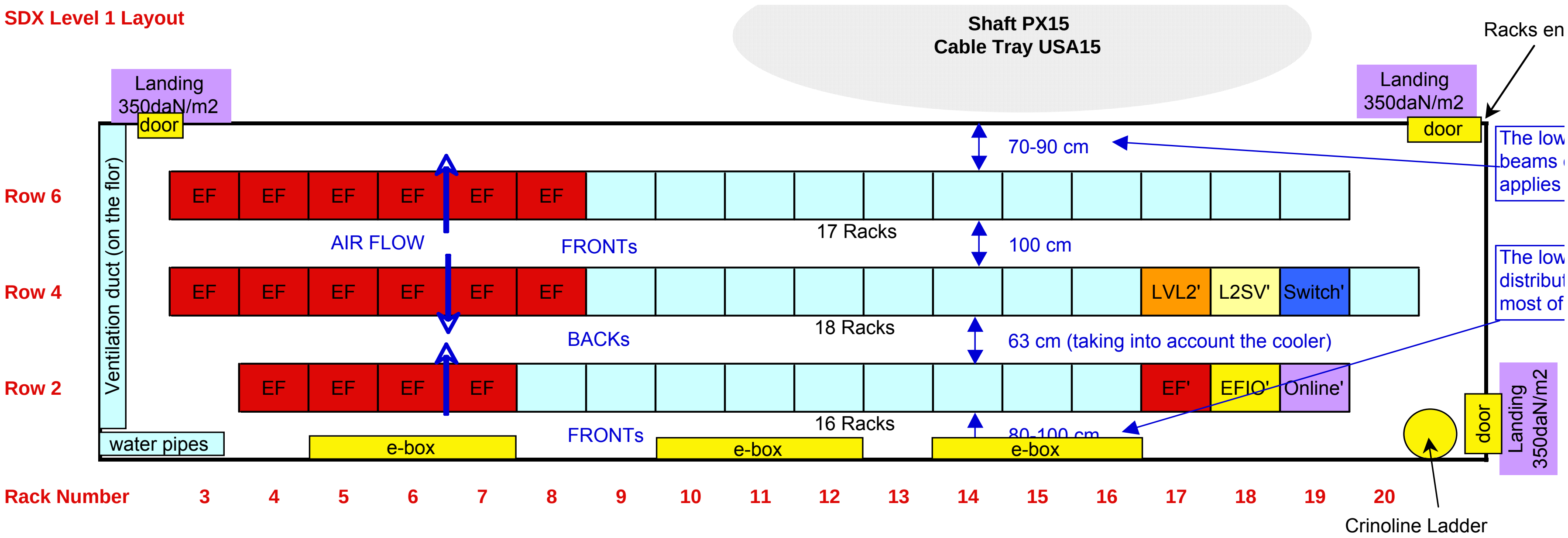


SDX Level 2 Layout



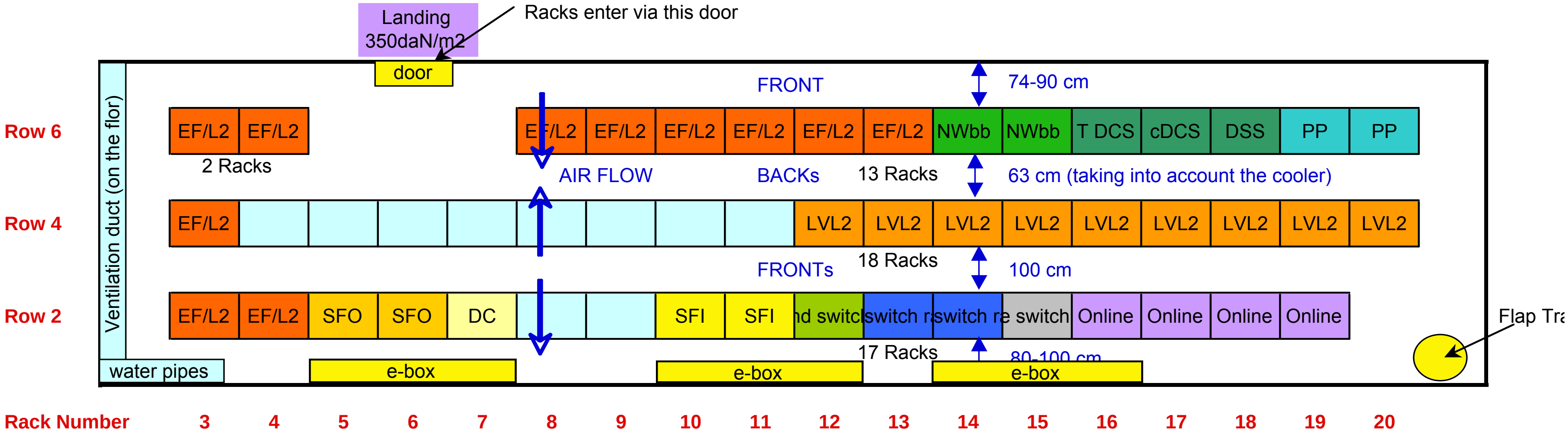
Rack Allocation in SDX - 2007

SDX Level 1 Layout



Rack Number	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Row 6	EF	EF	EF	EF	EF	EF												
Row 4	EF	EF	EF	EF	EF	EF									LVL2'	L2SV'	Switch'	
Row 2		EF	EF	EF	EF										EF'	EFIO'	Online'	

SDX Level 2 Layout



Rack Number	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Row 6	EF/L2	EF/L2				EF/L2	EF/L2	EF/L2	EF/L2	EF/L2	EF/L2	NWbb	NWbb	T DCS	cDCS	DSS	PP	PP
Row 4	EF/L2									LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2	LVL2
Row 2	EF/L2	EF/L2	SFO	SFO	DC			SFI	SFI	hd switc	switch ra	switch ra	e switch	Online	Online	Online	Online	

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issuses** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Renitializtion issues

- Review and plan the introduction of reinitialization and reconfiguration of the HLT system:

  - What can and cannot be changed between runs without reinitializing the complete DAQ system?
  - What are the recommendations to the algorithm developers to comply with the requirements of a reconfigurable system?
  - What are the tools existing to check the implementation is correct and how to use them?
  - Agree on a plan on how to start the implementation of these issues

Introduction Reinitialisation

- Normal Athena jobs follow simple state transitions

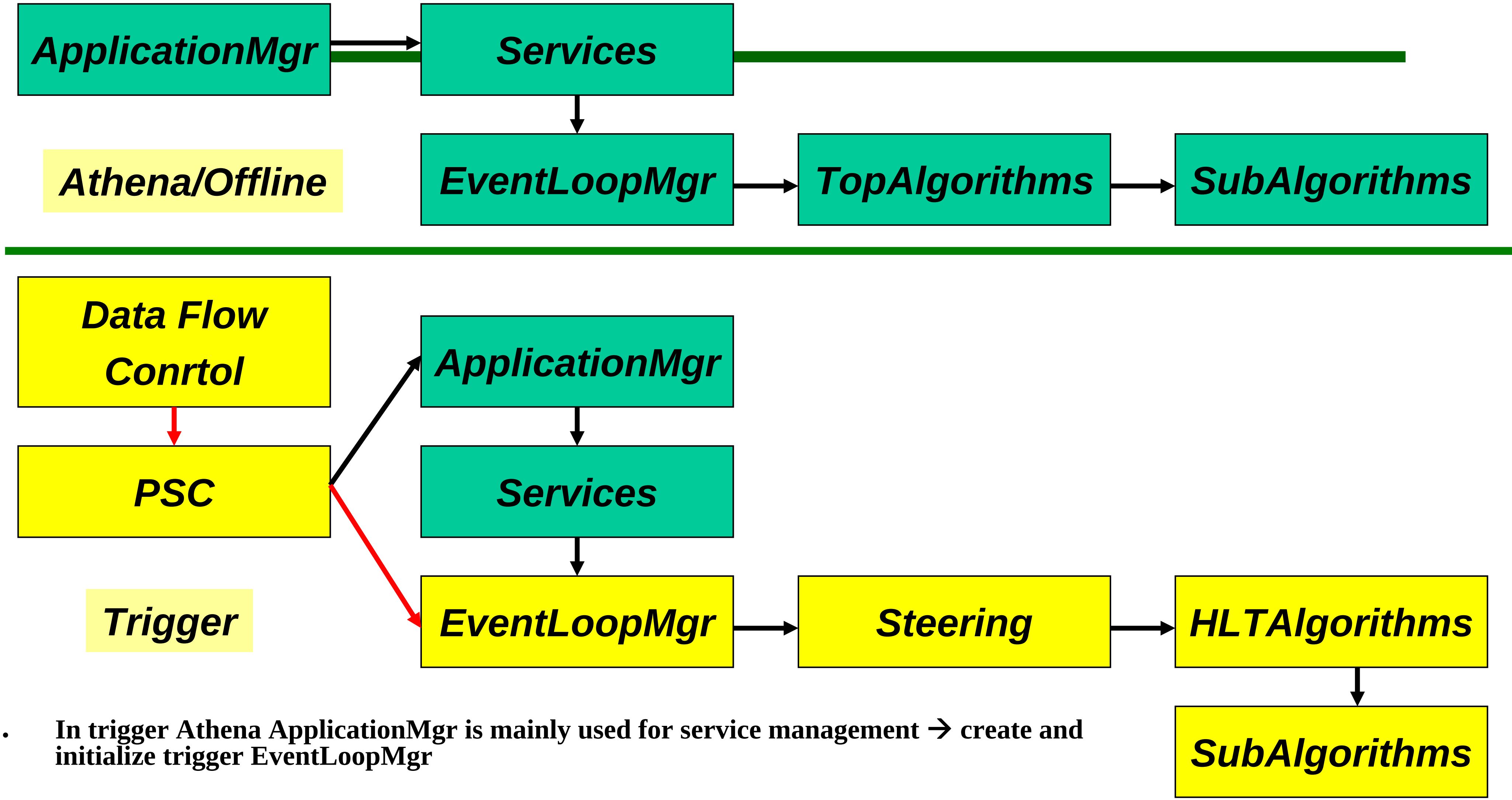
  (initialize ⇨ execute(=process all events) ⇨ finalize)

  - No cycling between states
- Trigger follows a more complex state machine with cycling through states, e.g. load ⬄ configure ⬄ prepareForRun ⬄ process events ⬄ Stop
- Developers have to take this into account in their algorithms and services
  - reinitialization of properties and internal structures for a reconfiguration of the system
  - Treatment of run dependent information
  - Control of internal algorithm/service states
- Athena provides
  - Methods for specific state transitions in some base classes
  - Incident mechanism to trigger updates
- Need to map this on trigger operation

“Reinitialization”

- Simple case: no configuration/conditions data change
  - Need only to implement run dependent methods → e.g. reset counters, reinitialize/recreate histograms
  - Internal algorithm states “initialized”/”finalized” remain valid
  - Present framework software can do this already
- Update configuration/conditions data → difficult case
  - Inform all clients who are concerned: how ?
    - call back ? → need to create more specific incidents
    - forced reinitialize for everybody ? (probably too time consuming)
    - forced reinitialize for list of “allowed” clients ?
  - Check/update internal client state → eventually recalculate internal data for all affected clients
  - Needs input from trigger operation model
    - when data are allowed to be updated
    - what data are updated
    - update frequency
  - Present framework software has only partial support for this mode
    - Allows already use of IOVSvc
    - Updates triggered with “Incident Service”

Control Flow



- **In trigger Athena ApplicationMgr is mainly used for service management → create and initialize trigger EventLoopMgr**
- **Trigger has special EventLoopMgr adapted to trigger FSM**
- **PSC can directly access EventLoopMgr in state transitions**
- **In trigger “Steering” needs to correctly forward transitions**

Test Tools: athenaMT/PT

- Documentation at http://atlas.web.cern.ch/Atlas/GROUPS/DAQTRIG/HLT/html/AthenaMT.htm
- athenaMT interactive mode → cycle through state transitions
  - *E.g. : athenaMT –i -n 2 –f <data file> <jobOptions.py>*
- Allows to write test script for state cycling (→ e.g. for nightly tests). *E.g. :*

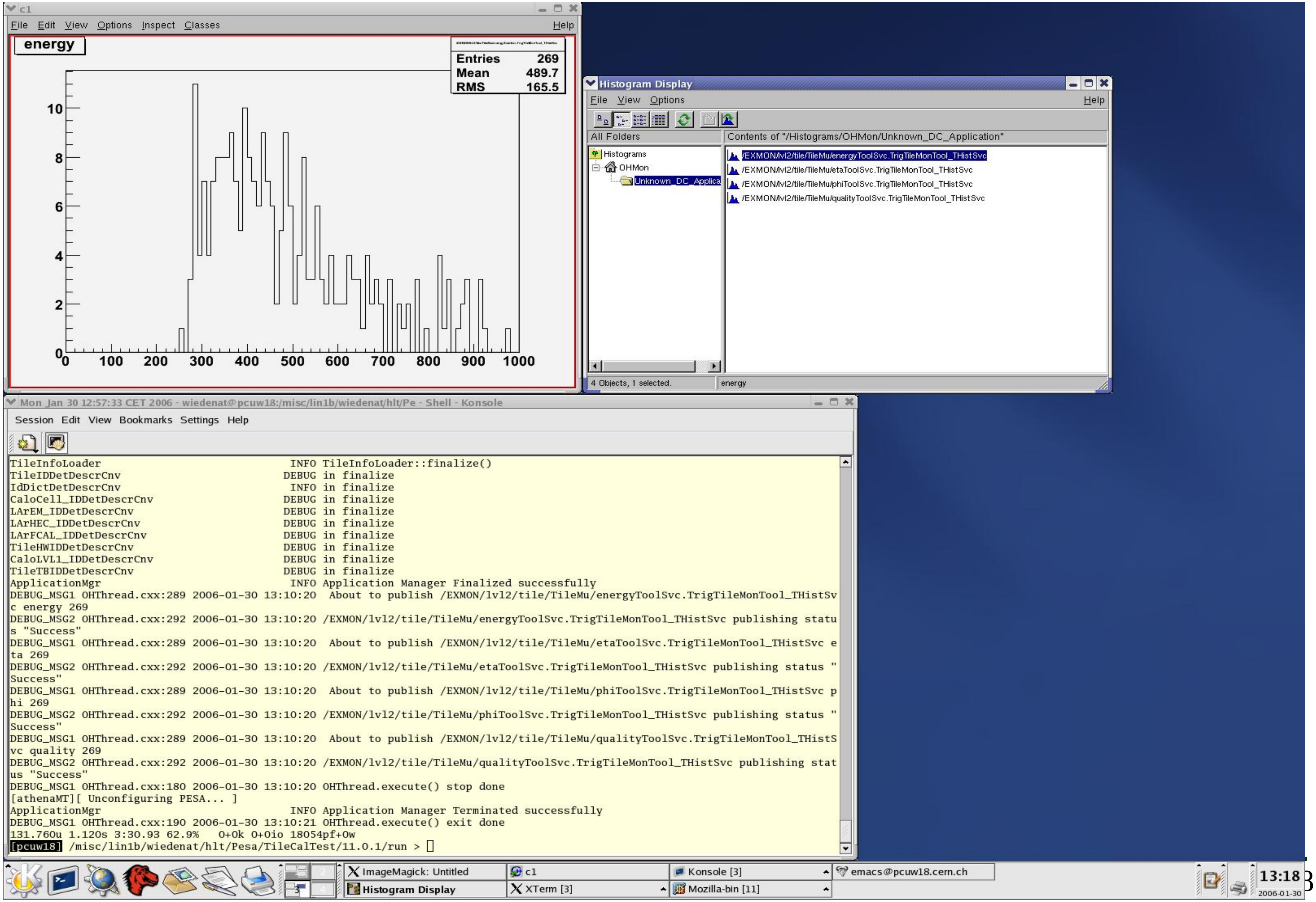
```
athenaMT –i  -n 2 –f <data file> <jobOptions.py> <<EOI
f     f     b     b     b
EOI
```

- It is possible to specify on the command line a run number and change this run number later in the “prepareForRun” transition. This should allow to simulate the retrieval of time dependent conditions data (tagged by run number).
- athenaPT: need still to implement interactive mode

Test Histograms/Monitoring in Trigger FSM

- Test monitoring histograms with athenaMT with state cycling
- athenaMT allows to create ROOT based histograms and Ntuples
  - Compliant with offline THistSvc
  - Histograms can be send to a “remote display”
  - Ntuples can only be stored on ROOT files
  - Introduction of a monitoring thread, which handles asynchronously histogram update and sending → special implementation of THistSvc for online use (integration with OH monitoring)
- Setup monitoring with athenaMT (bash)
  - *$(athenaMon –env)   (set environment for remote display)*
  - *athenaMon –display   (start display)*
  - *athenaMT –i –M –n 20 –f <input data file> <jobOption.py> (run athenaMT with monitoring thread)*

athenaMT: Histograms/Monitoring



Contents of "/Histograms/OHMon/Unknown_DC_Application"

/EXMON/lvl2/tile/TileMu/energyToolSvc.TrigTileMonTool_THistSvc
/EXMON/lvl2/tile/TileMu/etaToolSvc.TrigTileMonTool_THistSvc
/EXMON/lvl2/tile/TileMu/phiToolSvc.TrigTileMonTool_THistSvc
/EXMON/lvl2/tile/TileMu/qualityToolSvc.TrigTileMonTool_THistSvc

```
TileInfoLoader                                     INFO TileInfoLoader::finalize()
TileIDDetDescrCnv                                 DEBUG in finalize
IdDictDetDescrCnv                                  INFO in finalize
CaloCell_IDDetDescrCnv                            DEBUG in finalize
LArEM_IDDetDescrCnv                               DEBUG in finalize
LArHEC_IDDetDescrCnv                              DEBUG in finalize
LArFCAL_IDDetDescrCnv                             DEBUG in finalize
TileHWIDDetDescrCnv                               DEBUG in finalize
CaloLVL1_IDDetDescrCnv                            DEBUG in finalize
TileTBIDDetDescrCnv                               DEBUG in finalize
ApplicationMgr                                     INFO Application Manager Finalized successfully
DEBUG_MSG1 OHThread.cxx:289 2006-01-30 13:10:20  About to publish /EXMON/lvl2/tile/TileMu/energyToolSvc.TrigTileMonTool_THistSv
c energy 269
DEBUG_MSG2 OHThread.cxx:292 2006-01-30 13:10:20 /EXMON/lvl2/tile/TileMu/energyToolSvc.TrigTileMonTool_THistSvc publishing statu
s "Success"
DEBUG_MSG1 OHThread.cxx:289 2006-01-30 13:10:20  About to publish /EXMON/lvl2/tile/TileMu/etaToolSvc.TrigTileMonTool_THistSvc e
ta 269
DEBUG_MSG2 OHThread.cxx:292 2006-01-30 13:10:20 /EXMON/lvl2/tile/TileMu/etaToolSvc.TrigTileMonTool_THistSvc publishing status "
Success"
DEBUG_MSG1 OHThread.cxx:289 2006-01-30 13:10:20  About to publish /EXMON/lvl2/tile/TileMu/phiToolSvc.TrigTileMonTool_THistSvc p
hi 269
DEBUG_MSG2 OHThread.cxx:292 2006-01-30 13:10:20 /EXMON/lvl2/tile/TileMu/phiToolSvc.TrigTileMonTool_THistSvc publishing status "
Success"
DEBUG_MSG1 OHThread.cxx:289 2006-01-30 13:10:20  About to publish /EXMON/lvl2/tile/TileMu/qualityToolSvc.TrigTileMonTool_THistS
vc quality 269
DEBUG_MSG2 OHThread.cxx:292 2006-01-30 13:10:20 /EXMON/lvl2/tile/TileMu/qualityToolSvc.TrigTileMonTool_THistSvc publishing stat
us "Success"
DEBUG_MSG1 OHThread.cxx:180 2006-01-30 13:10:20 OHThread.execute() stop done
[athenaMT][ Unconfiguring PESA... ]
ApplicationMgr                                     INFO Application Manager Terminated successfully
DEBUG_MSG1 OHThread.cxx:190 2006-01-30 13:10:21 OHThread.execute() exit done
131.760u 1.120s 3:30.93 62.9%   0+0k 0+0io 18054pf+0w
[pcuw18] /misc/lin1b/wiedenat/hlt/Pesa/TileCalTest/11.0.1/run >
```

For Discussion: What to do with histograms ?

- Proposal (from Tomasz) for algorithms
  - stopRun()
    - Histograms are send with annotation about full statistics
    - All histograms are deleted (→ can be controlled by trigger THistSvc)
    - Nothing to be done in algorithm
  - startRun()
    - New creation & booking of histograms according new configuration
    - Algorithm has to create the histograms in “beginRun()” method and not in “initialize()” as now
    - Algorithm has to obey directory naming conventions for histograms
- Sending/suppression of new/old groups of histograms is done via OH configuration
  - specify histogram directories
  - specify update frequencies

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issuses** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Interface between LVL2 and EF

- What is existing L2Result content?

  Review the different cases that an EF algorithm can encounter
  - No LVL2 result
  - Corrupted LVL2 result
  - LVL2 result with no ROI (from force accept in LVL2/LVL1?)

  Review whether it is clear where the information to know what happened is correctly stored and decoders exist

  Agree how the system has to react to everyone of the above mentioned circumstances and whether changes have to be done at the algorithm (EF) level or in the steering??

Points for Discussion

- **We have never had a detailed discussion on how EF has to react to special cases**
  - In fact, EF algorithms react differently to special situations
    - TrigMoore can handled a corrupted LVL2 result (just ignores it and runs in non-seeded mode)
    - TrigCaloRec does not run in the same situation
- **In real running we will have to cope with cases like:**
  - LVL2 Result without RoI information
    - Force accepted event
    - pROS not working properly?
  - Dummy LVL2 Result
- **There might be different ways of treating these special cases**
  - The Steering invents a LVL2Result to seed the EF?
  - All the first algorithms in the slices are required to handled the special cases?
- **In this discussion, we should aim at having a robust and coherent solution for these special situations**

Basic Operation Scheme (LVL2/EF)

- For every event the HLT software provides in StoreGate a L2Result object, which summarizes the selection process and provides the event decision.
- The Steering Controller (SC) gets from this L2Result object the event decision and returns it as a Boolean to the dataflow software indicating if an event is accepted/rejected
- If the event is rejected → nothing further from L2 SC
- In case the event is accepted:
  - The SC receives from the dataflow software a buffer of given length in which the SC puts a L2Result ROB fragment.
  - The payload of the ROB fragment is provided by the HLT steering in the L2Result object in StoreGate. The SC gets the L2Result object from StoreGate and serializes it into a vector of integers for the ROB fragment.
  - The ROB fragment is send to the pROS where it gets picked up by EF/event building
- In case of an L2 accept always an accompanying L2Result is expected by the data flow software.

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issuses** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

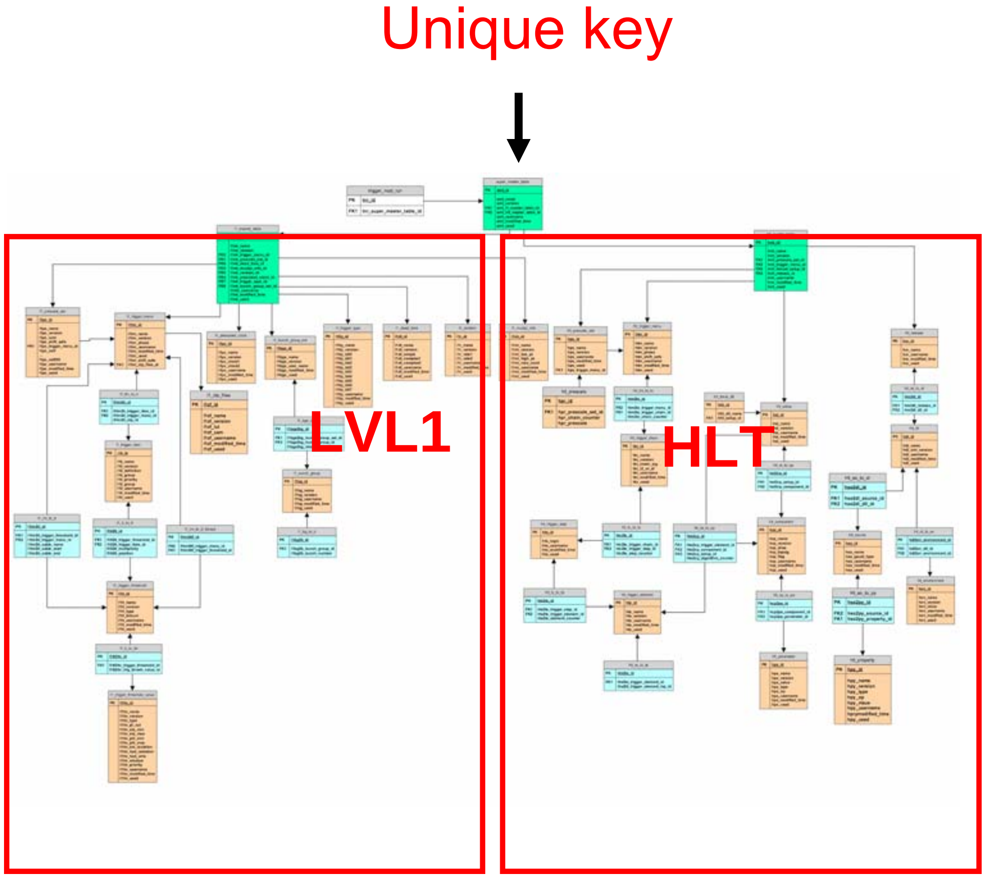
  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Configuration, Steering and DBS issues

- Review the DBS configuration and how is it going to affect algorithm development / integration:
- Scheme for development configuration and porting of tools from offline to online
- Changes needed in the algorithms themselves
- Plan for integration on the online running, specially those chagnes that affect the offline code
- Review & discuss the changes planned on the steering package (specially the new ones) and how are they going to affect algorithms and integration

Configuration of TriggerDB

- Foreseen “modus operandi” for configuration from TriggerDB.
  - What are the methods to change the online event selection?
  - Where are replication of TriggerDB available?
  - Who has read/write access to replications of TriggerDB?
  - How can developers interact with the online configuration?
  - How can they use the TriggerDB?
- Status and Timeline



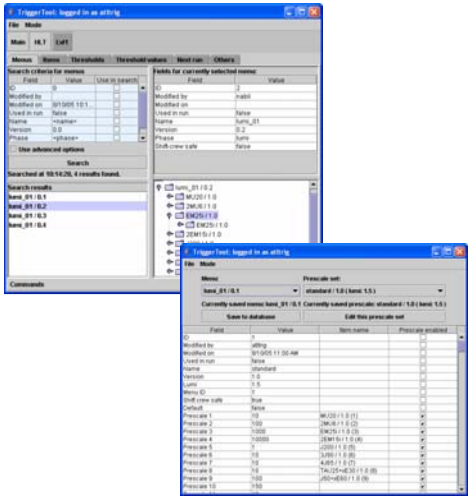
Configuration online

- The master TriggerDB will be part of the ATLAS online DB system (Oracle).
- Only authorized people (administrators) will have write access to master TriggerDB.
- Replications will be available at other places (see below).
- From the master TriggerDB the online clients will be configured.
- All configuration changes that affect the online selection are first stored in the master TriggerDB and then loaded by the online clients.
- two use cases:
  - smaller “on-the-fly” changes
  - Normal Trigger production cycle

(all changes need of course agreement of ATLAS level, not discussed here, only the technical process)

“On-the-fly” changes

- Intended for a small set of changes: e.g.
  - inclusion of already tested trigger chains in a run
  - changes of prescale factors, random rates, …
  - changes of cuts in HYPO algorithms, …
- “technical“ process to follow:
  1) Changes in the TriggerDB are done via Trigger online GUI (TriggerTool).
  2) Any change triggers update of all necessary tables in TriggerDB, esp. new configuration key.
  3) Check: configuration of Athena for validation with new key (depending on the actual change)
  4) New configuration is flagged “safe”
  5) New configuration can be used online



Field	Value	Use in search
ID	0	□
Modified by		□
Modified on	8/10/05 10:1	□
Used in run	false	□
Name	<name>	□
Version	0.0	□
Phase	<phase>	□

Searched at 10:14:29, 4 results found.

lumi_01 / 0.1
lumi_01 / 0.2
lumi_01 / 0.3
lumi_01 / 0.4

Fields for currently selected menu:

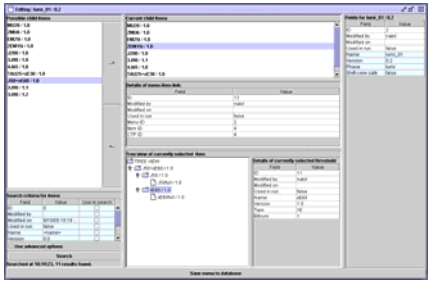
Field	Value
ID	2
Modified by	nabil
Modified on	
Used in run	false
Name	lumi_01
Version	0.2
Phase	lumi
Shift crew safe	false

lumi_01 / 0.2
MU20 / 1.0
2MU6 / 1.0
EM25i / 1.0
EM25i / 1.0
2EM15i / 1.0

lumi_01 / 0.1 · standard / 1.0 (lumi: 1.5)

Currently saved menu: lumi_01 / 0.1 Currently saved prescale: standard / 1.0 (lumi: 1.5)

Field	Value	Item name	Prescale enabled
ID	1		□
Modified by	atltrig		□
Modified on	8/10/05 11:00 AM		□
Used in run	false		□
Name	standard		□
Version	1.0		□
Lumi	1.5		□
Menu ID	1		□
Shift crew safe	true		□
Default	false		□
Prescale 1	10	MU20 / 1.0 (1)	☑
Prescale 2	100	2MU6 / 1.0 (2)	☑
Prescale 3	1000	EM25i / 1.0 (3)	☑
Prescale 4	10000	2EM15i / 1.0 (4)	☑
Prescale 5	1	J200 / 1.0 (5)	☑
Prescale 6	10	3J90 / 1.0 (6)	☑
Prescale 7	10	4J65 / 1.0 (7)	☑
Prescale 8	10	TAU25+xE30 / 1.0 (8)	☑
Prescale 9	100	J50+xE60 / 1.0 (9)	☑
Prescale 10	150		☑


Editing: lumi_01 / 0.2
Possible child items
Current child items
Details of menu item link
Tree view of currently selected item
TREE VIEW
Details of currently selected threshold
Search criteria for items
Use advanced options
Search
Save menu to database

Normal trigger production cycles

- Intended for more severe changes:
  - e.g. new trigger chains, new sequences, new menu, new release…
- “technical” process to follow:

1) Developers:
   a) develop new code/ configuration
   b) test configuration/code offline (following normal HLT dev. process)
   c) submit configuration in interchange format (XML, python, not yet fully defined) to Trigger Configuration Administrator for inclusion in TriggerDB.
2) Trigger Configuration Administrators:
   a) test new code/configuration locally
   b) insert the job in TriggerDB properly (reuse of existing parts)
   c) stamp new configuration “safe”
3) new configuration can be used online

- Later, dev can retrieve the job back, in interchange format or directly from the TriggerDB, and modify/test/submit it again.

Summary Trigger DB

- **TriggerDB is a production system that must guarantee that online data-taking is safe.**
- Jobs inserted by Trigger Configuration Administrators.
- Developers submit new configurations in an interchange format (not yet fully defined, XML python).
- Various replications available (read-only).
- Offline development to be done with other techniques (today: XML (menu), python (JO); interchange format) or using local, private copy of the TriggerDB.

Steering: Longer term developments

- Requirements and design document: Revisions following review are complete, document available
- General code tidy up
- Integration of new configuration
  - New configuration Service
  - Integration in step controller underway
  - Configuration verification tools
  - Problem of dependencies and projects
- StepDecision
  - Prescales and forced accepts
- LVL1 conversion
  - Expand LVL1 RoIs into TriggerElements for each threshold
  - take account of active chains
- Algorithm interface
  - Different types of algorithms and their interfaces, including navigation
  - Add support for topological triggers, try some topological algorithms
  - Prevent multiple FEX algorithm execution on the same RoI
- Monitoring

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issuses** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Monitoring issues

- **Monitoring Sources:**

  Review the different monitoring sources and how the monitoring information is going to be publised from them

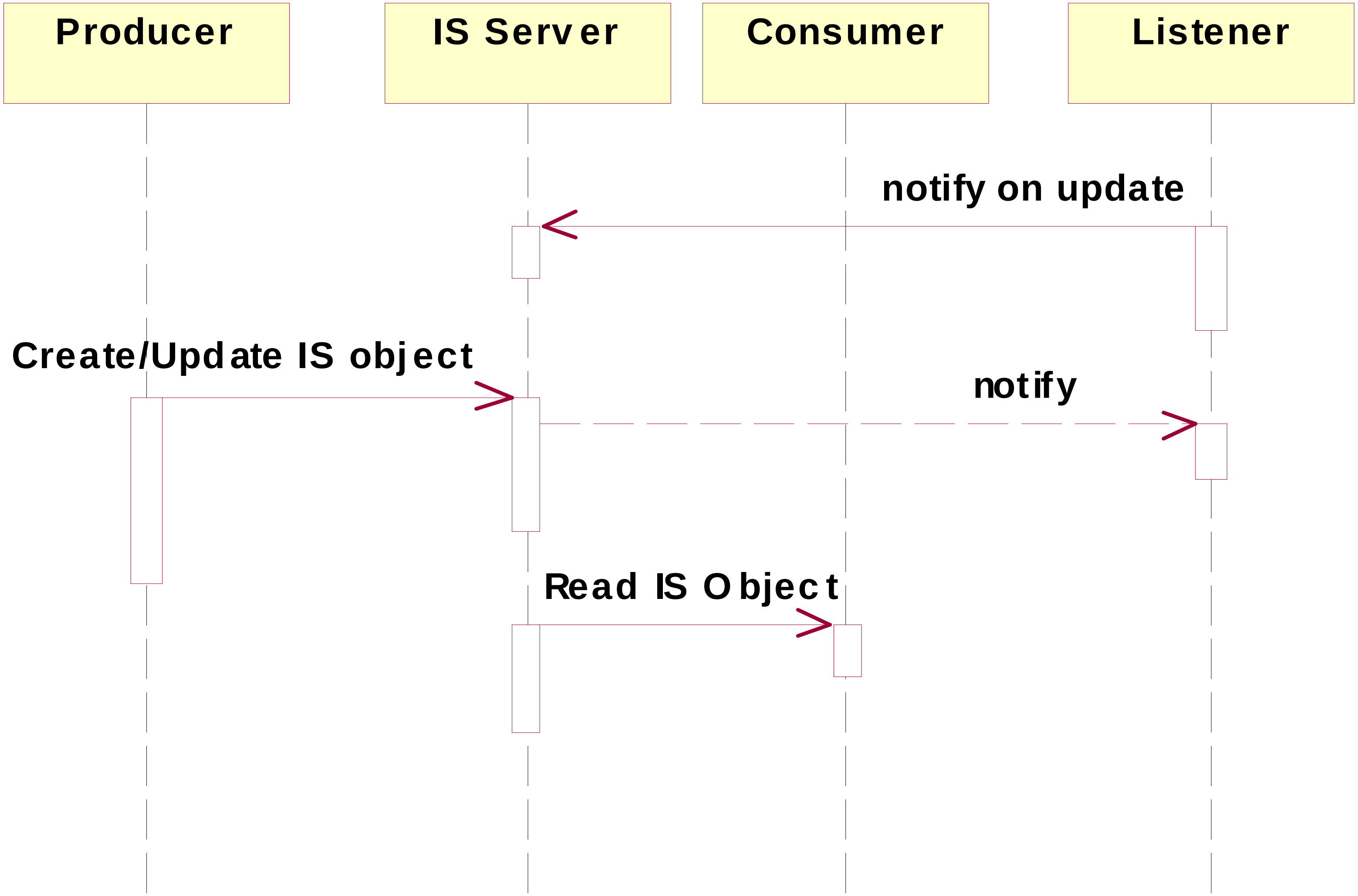
- **Histogramming inside Algorithms**
- **Counters/Timers on the Steering**
- **HLT Infrastructure**
- **Monitoring Output and post-processing:**

  Review the tools needed for display of monitoring information and brainstorm on the needed tools for commssioning / debugging / operation

The Online Information Service

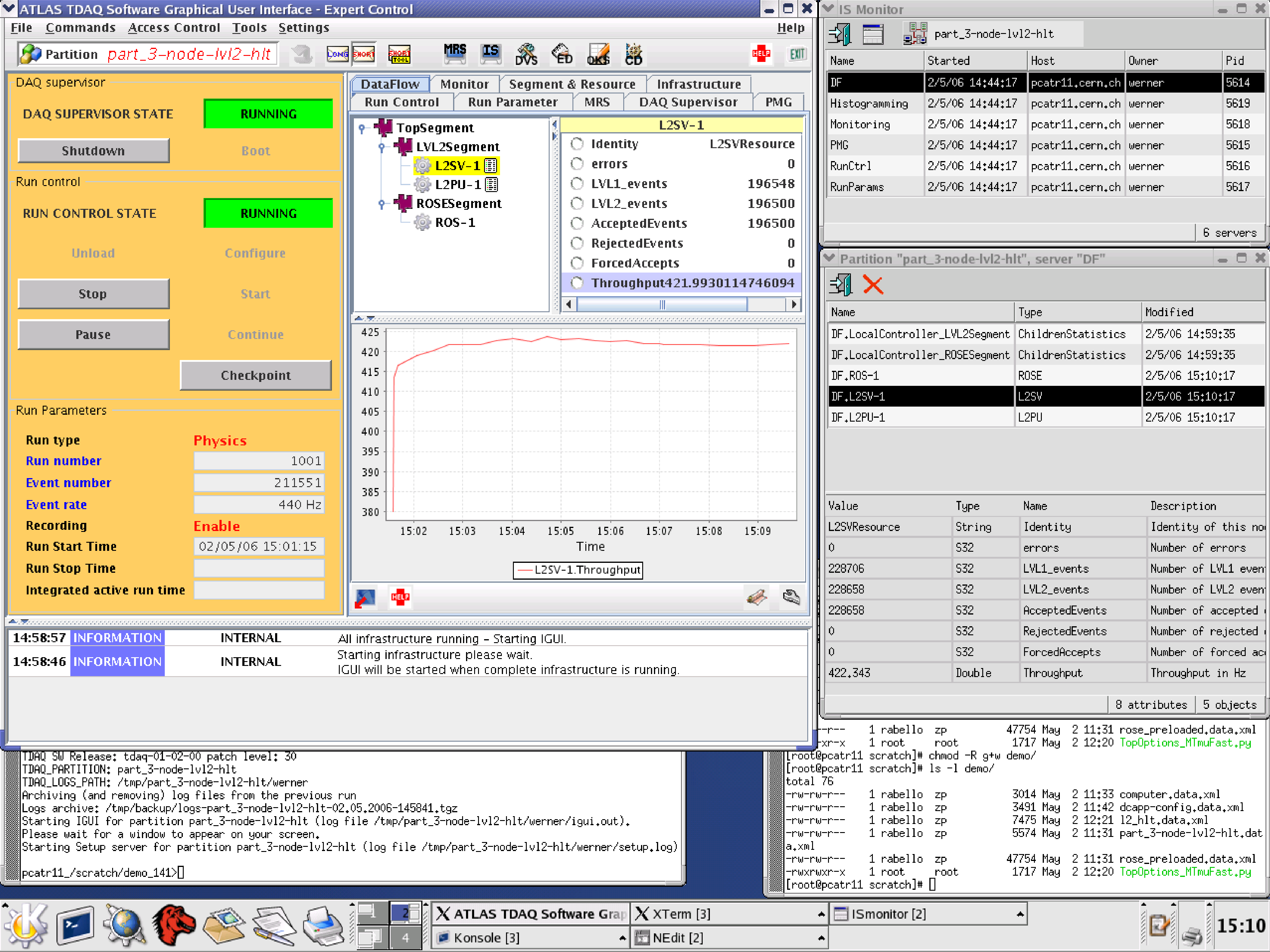
- *The Information Service, IS, is provided by the DAQ subsystem*
  - *Producers can store typed objects in an IS server*
  - *consumers can retrieve these objects from an IS server*
  - *applications can be notified (by a call-back) when an IS object is modified*

IS - Simplified view



IS Producers, Consumers and Listeners

- **HLT-I applications are producers of operational monitoring information**
- **The Online Information Service, IS, is used to store(and retrieve) this information, e.g.,**
  - Event and data rates, payloads, errors, etc.
  - From HLT3 and tdaq-01-06-00, histograms (T.Bold)
- **producers updates information with a configurable interval, e.g., 20 seconds for the L2PU information**
- **Consumers are, e.g.,**
  - Online tools: IS monitor and IGUI.DataFlowPanel
  - Auto-test scripts collecting IS data (for later analysis)
  - Any application accessing IS data



Partition part_3-node-lvl2-hlt

DAQ supervisor

DAQ SUPERVISOR STATE	RUNNING
Shutdown	Boot

Run control

RUN CONTROL STATE	RUNNING
Unload	Configure
Stop	Start
Pause	Continue
	Checkpoint

Run Parameters

Run type	Physics
Run number	1001
Event number	211551
Event rate	440 Hz
Recording	Enable
Run Start Time	02/05/06 15:01:15
Run Stop Time	
Integrated active run time	

L2SV-1

○ Identity	L2SVResource
○ errors	0
○ LVL1_events	196548
○ LVL2_events	196500
○ AcceptedEvents	196500
○ RejectedEvents	0
○ ForcedAccepts	0
○ Throughput421.9930114746094	

14:58:57	INFORMATION	INTERNAL	All infrastructure running - Starting IGUI.
14:58:46	INFORMATION	INTERNAL	Starting infrastructure please wait. IGUI will be started when complete infrastructure is running.

part_3-node-lvl2-hlt

Name	Started	Host	Owner	Pid
DF	2/5/06 14:44:17	pcatr11.cern.ch	werner	5614
Histogramming	2/5/06 14:44:17	pcatr11.cern.ch	werner	5619
Monitoring	2/5/06 14:44:17	pcatr11.cern.ch	werner	5618
PMG	2/5/06 14:44:17	pcatr11.cern.ch	werner	5615
RunCtrl	2/5/06 14:44:17	pcatr11.cern.ch	werner	5616
RunParams	2/5/06 14:44:17	pcatr11.cern.ch	werner	5617

6 servers

Partition "part_3-node-lvl2-hlt", server "DF"

Name	Type	Modified
DF.LocalController_LVL2Segment	ChildrenStatistics	2/5/06 14:59:35
DF.LocalController_ROSESegment	ChildrenStatistics	2/5/06 14:59:35
DF.ROS-1	ROSE	2/5/06 15:10:17
DF.L2SV-1	L2SV	2/5/06 15:10:17
DF.L2PU-1	L2PU	2/5/06 15:10:17

Value	Type	Name	Description
L2SVResource	String	Identity	Identity of this no
0	S32	errors	Number of errors
228706	S32	LVL1_events	Number of LVL1 even
228658	S32	LVL2_events	Number of LVL2 even
228658	S32	AcceptedEvents	Number of accepted
0	S32	RejectedEvents	Number of rejected
0	S32	ForcedAccepts	Number of forced ac
422.343	Double	Throughput	Throughput in Hz

8 attributes 5 objects

```
TDAQ SW Release: tdaq-01-02-00 patch level: 30
TDAQ_PARTITION: part_3-node-lvl2-hlt
TDAQ_LOGS_PATH: /tmp/part_3-node-lvl2-hlt/werner
Archiving (and removing) log files from the previous run
Logs archive: /tmp/backup/logs-part_3-node-lvl2-hlt-02.05.2006-145841.tgz
Starting IGUI for partition part_3-node-lvl2-hlt (log file /tmp/part_3-node-lvl2-hlt/werner/igui.out).
Please wait for a window to appear on your screen.
Starting Setup server for partition part_3-node-lvl2-hlt (log file /tmp/part_3-node-lvl2-hlt/werner/setup.log)

pcatr11_/scratch/demo_141>
```

```
-r--     1 rabello  zp         47754 May  2 11:31 rose_preloaded.data.xml
xr-x     1 root     root        1717 May  2 12:20 TopOptions_MTmuFast.py
[root@pcatr11 scratch]# chmod -R g+w demo/
[root@pcatr11 scratch]# ls -l demo/
total 76
-rw-rw-r--    1 rabello  zp          3014 May  2 11:33 computer.data.xml
-rw-rw-r--    1 rabello  zp          3491 May  2 11:42 dcapp-config.data.xml
-rw-rw-r--    1 rabello  zp          7475 May  2 12:21 l2_hlt.data.xml
-rw-rw-r--    1 rabello  zp          5574 May  2 11:31 part_3-node-lvl2-hlt.dat
a.xml
-rw-rw-r--    1 rabello  zp         47754 May  2 11:31 rose_preloaded.data.xml
-rwxrwxr-x    1 root     root        1717 May  2 12:20 TopOptions_MTmuFast.py
[root@pcatr11 scratch]#
```

Existing and Missing features

- Existing
  - counters, rates etc. are exported by applications to IS
  - from HLT3 and tdaq-01-06-00 thread safe histograms will be available (implemented by T.Bold)
  - **The Gatherer can summerise information from many sources, e.g., for all L2SVs.**
- **Missing**
  - To operating large systems we need '**Active Monitoring**', e.g., warn when a threshold is passed for
    - individual applications
    - a group of applications
  - IS information, e.g., Histograms, that can be **enabled/disabled during a RUN**

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issuses** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Analysis of BS, ESD and AOD data

- Review how are we going to analyze and check the BS data from the HLT point of view.
- Also consider the content of the ESD/AOD specially in view of the HLT analysis.
- What trigger data is available offline and what data should be available

Trigger Data Analysis: Motivation

1) Trigger signature/menu study and optimisation:

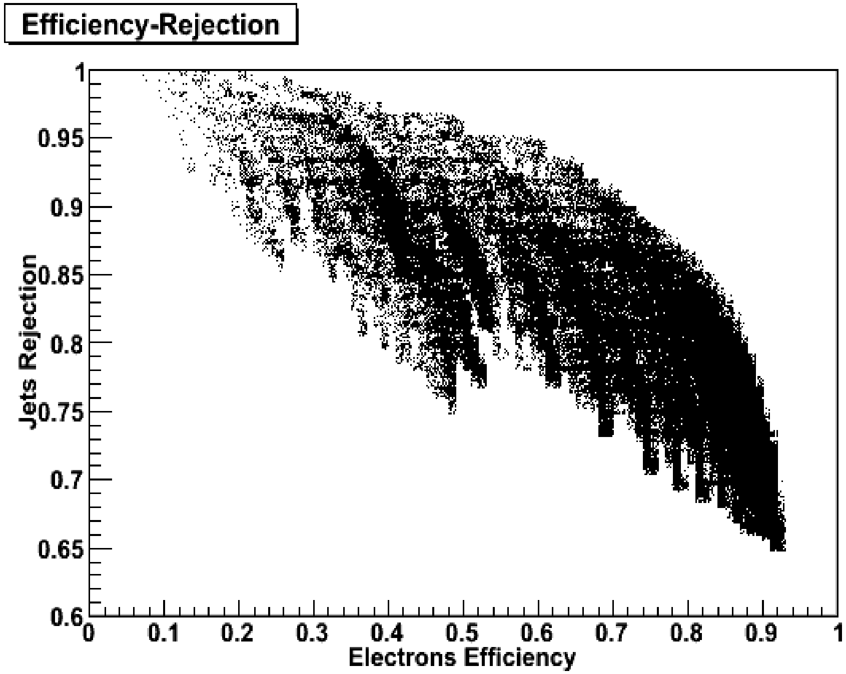
- ❑ Should be done before including signatures in online menu (i.e. signature requests should normally be justified by studies)
- ❑ Check trigger decision/objects against truth information
- ❑ Can be validated by re-running trigger on force-accepts and prescaled loose-threshold events

2) Check trigger operation:

- ❑ Offline: compare with truth to get efficiency/purity, improve reconstruction, estimate bias
- ❑ Testbed studies: trigger behaviour must be the same “online” and offline ⇒ compare Bytestream and ESD for the same generated events
- ❑ Online: study trigger operation in force-accepted events and prescaled signatures; debug aborted events (diagnostic stream); pathological events…

Signature/Menu studies

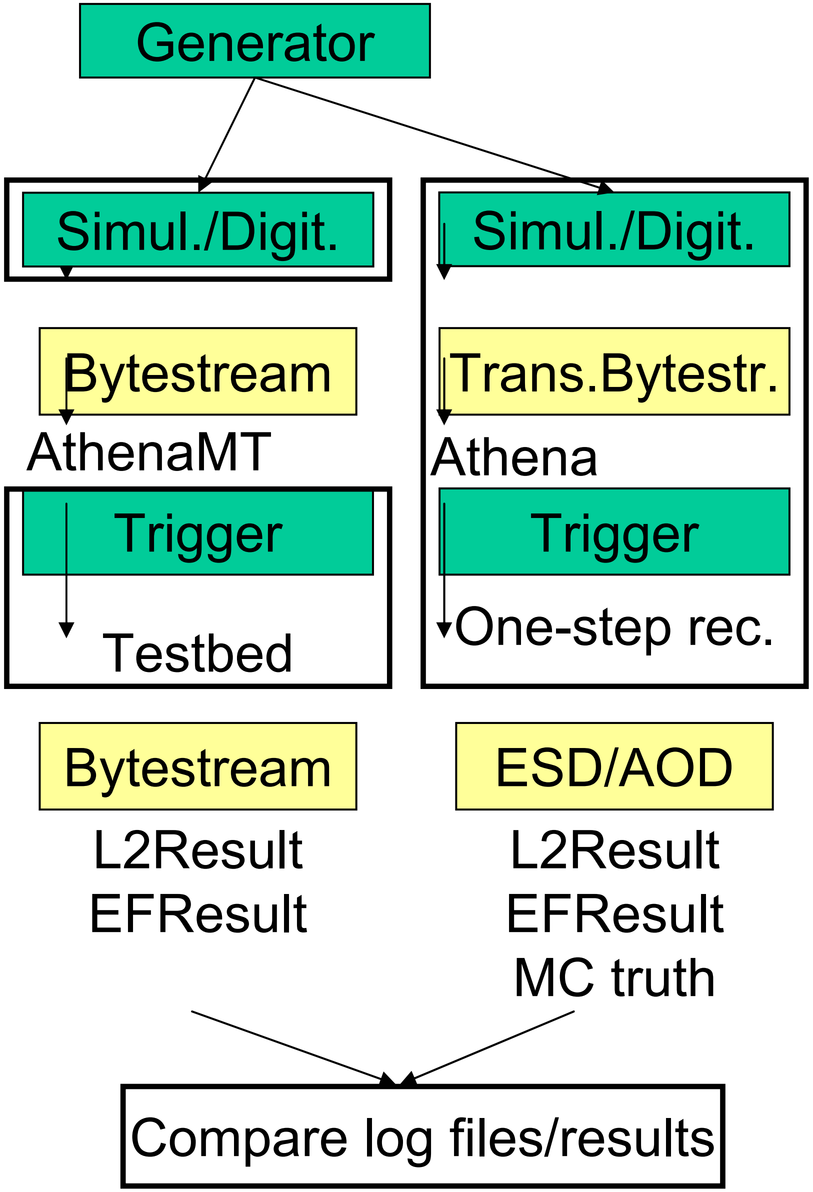
- Few optimisation studies have been performed and these took a lot of effort;
- These required relevant data to be available offline;
- A large menu, with O(100) signatures x3 levels must be optimised for commissioning;
- Further studies will be needed when background rates are better known;
- Bytestream files from prescaled low-threshold signatures and force-accepted events may be used to generate data for these studies.



Trigger operation debugging

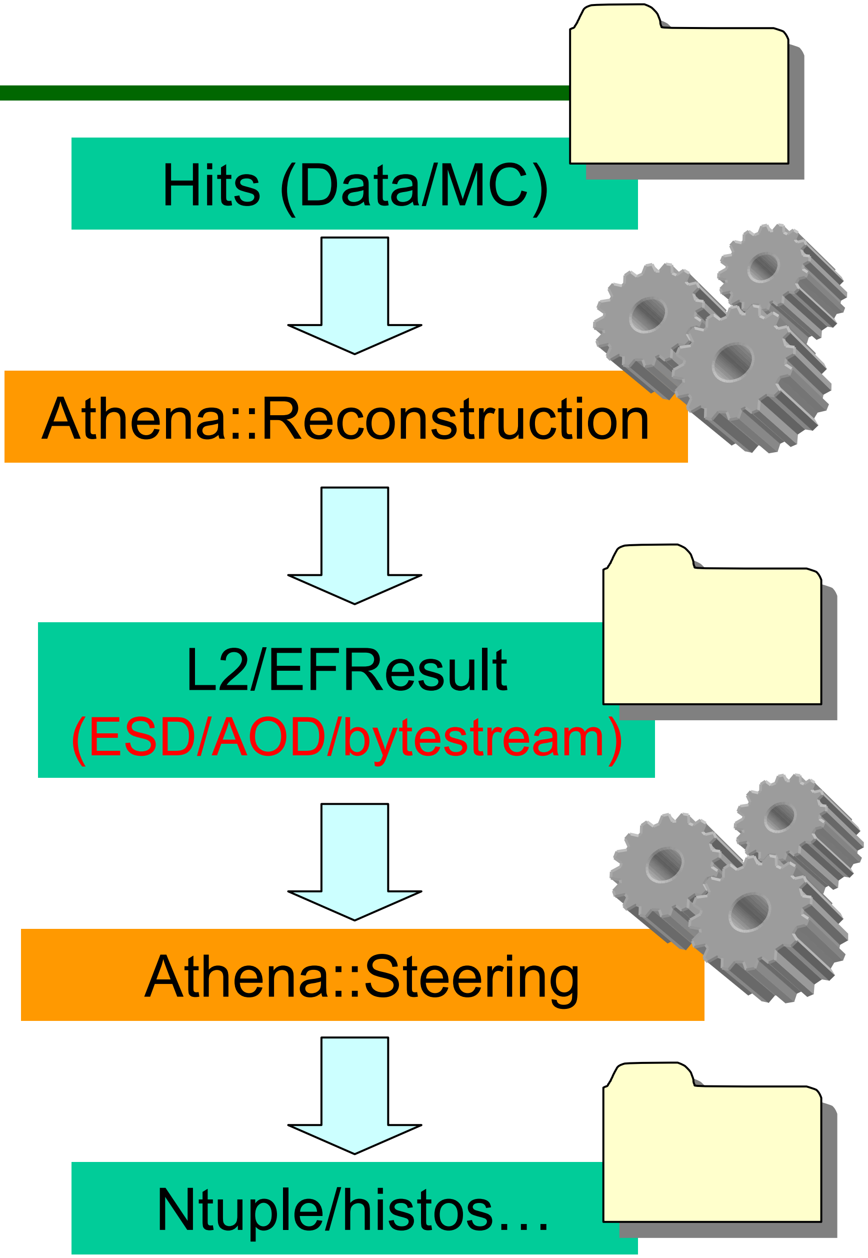
- Statistical: monitoring histograms
  - No need to store any data offline
  - Probably not useful if we're looking for small effects or unusual occurrences

- Event by event:
  - Only way to make HLT event data available is through L2/EFResult ⇒ Serializer
  - Regression-type test, running on bytestream data and comparing to ESD: in both cases, retrieve data from L2/EFResult and run hypothesis algorithms on it
  - MC truth information also available in ESD/AOD: compare reconstruction and truth



How to analyse ESD/AOD/BS

- To produce ESD/AOD/BS:
  1. The Steering serializes all relevant data objects into L2/EFResult (not at present)
  2. TriggerDecision object filled after Steering has run and put in ESD/AOD
  3. L2Result and EFResult attached to event (BS) or stored in ESD/AOD
- To analyse ESD/AOD/BS:
  1. The Steering de-serializes all data objects
  2. Then either look at reconstructed features and TriggerDecision or
  3. Run hypothesis algorithms only on reconstructed features
- **Note**:
  - In this way, the hypothesis algos can be run many times over the same data objects as if running online, and the cuts **optimized**
  - The cuts can only be tightened wrt the original cuts



Hits (Data/MC)

Athena::Reconstruction

L2/EFResult (ESD/AOD/bytestream)

Athena::Steering

Ntuple/histos…

Contence of the Workshop

- **Hardware issues** (Fred Wickens)
- **Reinitialization issues** (Tomaz Bold)
- **Interface between LVL2 and EF** (Werner Wiedenmann)
- **Configuration and DBS issuses** (Johannes Haller)
- **Steering issues** (Andreas Hoecker/Johnannes Haller)
- **Monitoring issues** (Martin zur Nedden)
- **Analysis of BS, ESD ana AOD data** (Ricardo Concalo)
- **Cosmic Ray run** (Jamie Boyd)
- **Software Tutorials**
- **find all talks at:**

  http://agenda.cern.ch/fullAgenda.php?ida=a061959

Cosmic Ray Runs: Introduction

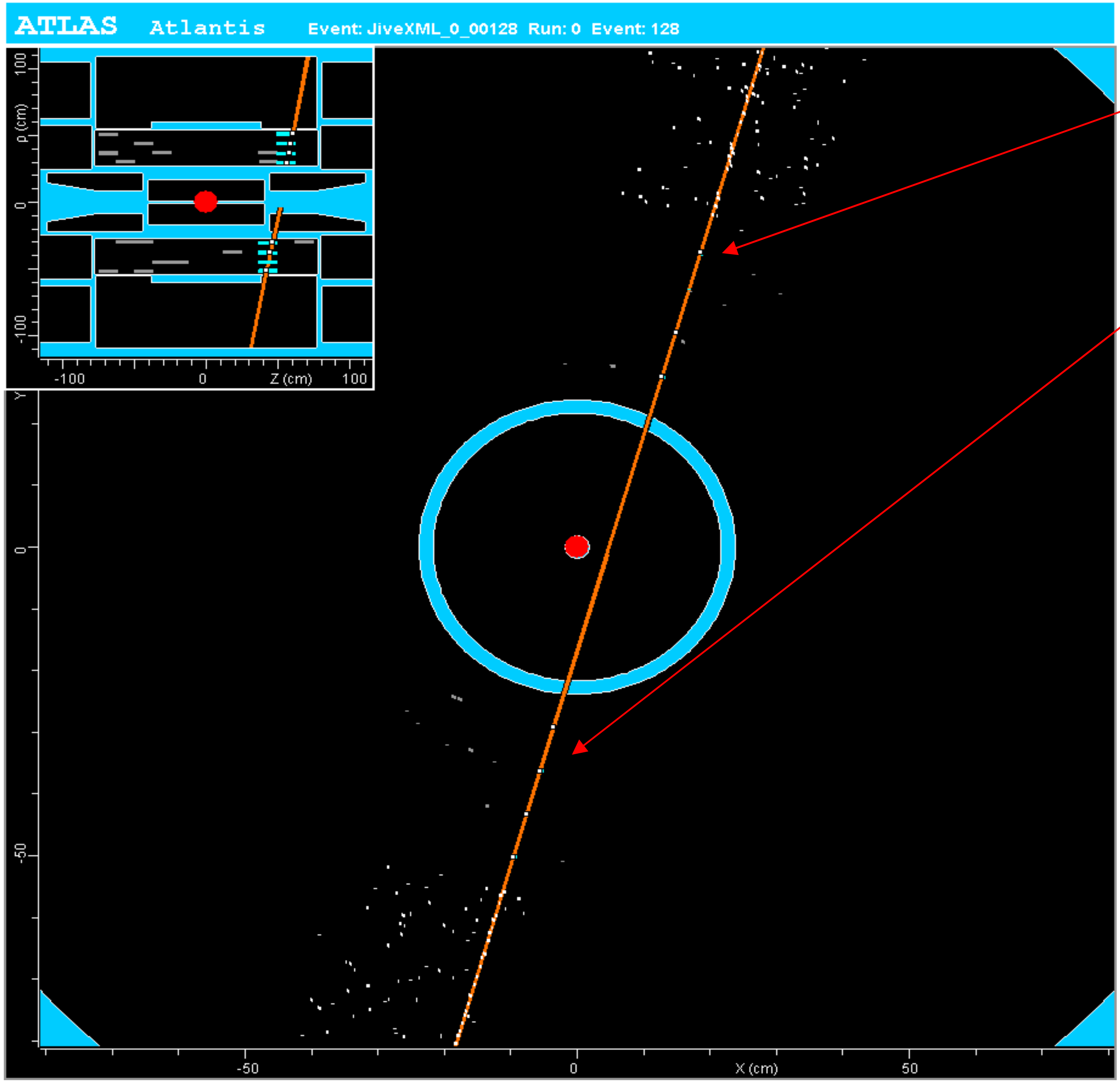
- Status of the LVL2 algorithm (muon,ID) development and plans.
- Ideas on using cosmic runs for HLT commissioning.
- Cosmic running and HLT
  - Cosmic run for commissioning and calibrating the detectors.
    - MIP signal.
    - Timing.
  - First opportunity to run the full trigger chain with real data.
    - Integration of LVL1, LVL2, EF and DAQ.
    - Verify the trigger decisions.
    - Performance evaluation of HLT algorithms.

LVL2 cosmic muon trigger: overview

(Main issue here is to reconstruct non-pointing muons)

1. Scan all RPC hits and find pairs in the doublets.
   - □ Currently, use RPC2 hit as the seed.
   - □ In future, use RoI from the LVL1.
2. Find hits around the seed hit. → RPC track
   - □ Find hits in the same pad regions. (>=2 layers)
   - □ Straight line fit of RPC hits to find associated MDT hits.
3. MDT pattern recognition → MDT track segments
   - □ Find MDT hits in the region around the extrapolated RPC track. (Now using $\eta \times \varphi = 0.4 \times 0.4$ for the RegionSelector and ±30 cm road)
   - □ Local straight line fit of MDT hits in each chamber. (Drift time information not used.)
   - □ Check the number of MDT hits and the compatibility of track direction to the RPC track.

LVL2 ID trigger



2 trigger tracks found
1 with 3 space points
and the other with 4.

Lines up very well with
the offline reco track.

Based on IDSCAN:

- Shift x-coordinate to make the track coming from the IP.
- Optimize the binning in the histogramming method to find the Z vertex.
- Scanning the full ID, not using the RegionSelector.

Some ideas on what we can gain from cosmic running

- Interesting topic to aim for during the commissioning.
  - For detectors, calibration and alignment.
  - For trigger, make the 3 trigger levels work online.
  - For DAQ, get real data from various detectors to the storage.
- HLT algorithm development is only a part of what we need to do. Needs a wider community to get involved.
  - Detectors must be operational
  - LVL1, LVL2, EF and DAQ must be available.