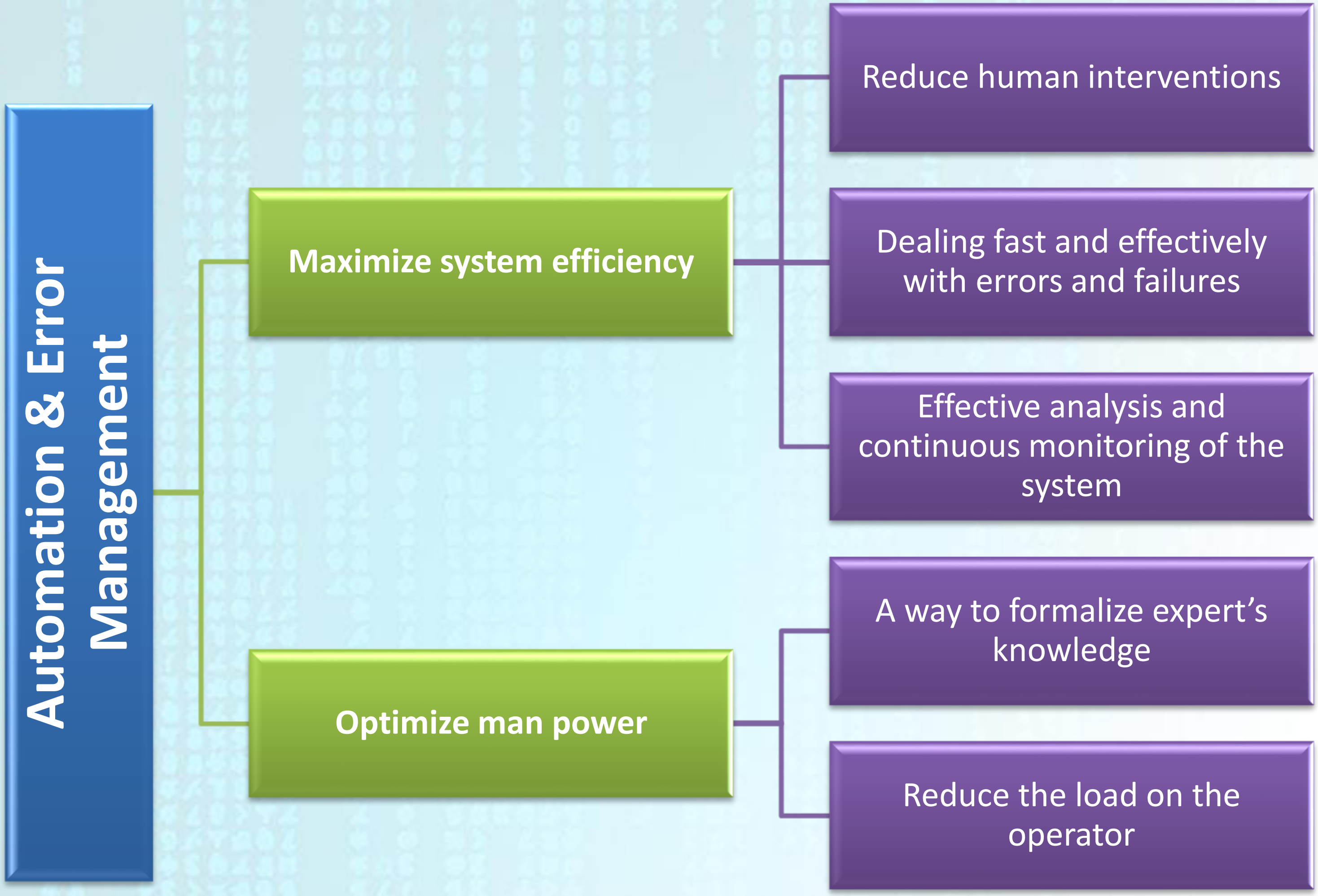
The ATLAS Run 2 Expert System

Giuseppe Avolio - CERN

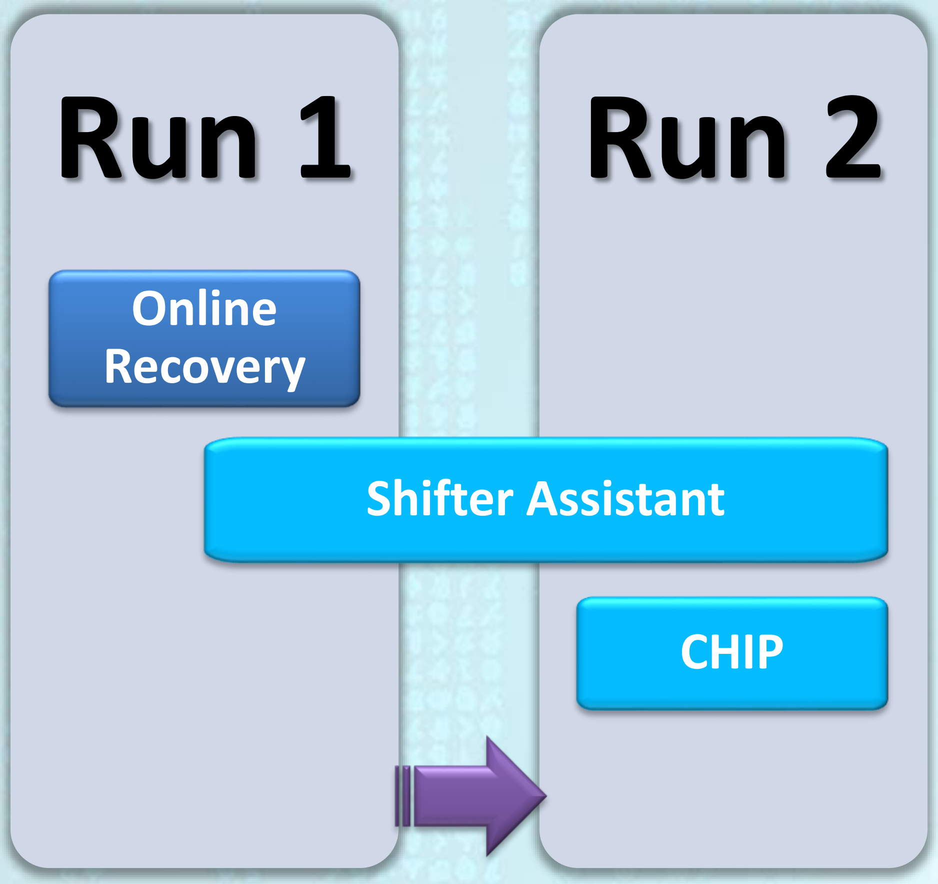
Outline

- **Expert Systems in ATLAS TDAQ**
  - Why?
  - From Run 1 to Run 2
- **Introducing a Complex Event Processing engine**
  - ESPER from EsperTech
- **Intelligent systems in Run 2**
  - The Central Hint and Information Processor (CHIP)
  - The Shifter Assistant
- **A different point of view**
  - The Shifter Assistant Replay (SAReplay)
- **Conclusions and outlook**

Why?


Automation & Error Management
Maximize system efficiency
Optimize man power
Reduce human interventions
Dealing fast and effectively with errors and failures
Effective analysis and continuous monitoring of the system
A way to formalize expert's knowledge
Reduce the load on the operator

From Run 1 to Run 2

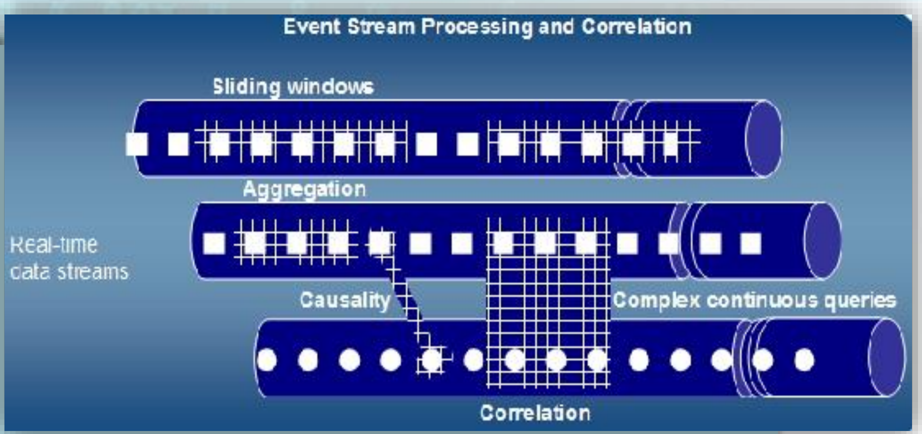


- **Online Recovery**
  - Based on CLIPS
- **Shifter Assistant (SA)**
  - Tool to assist the shifter in accomplishing his/her duties
  - Based on a Complex Event Processing (CEP) engine, ESPER
- **Central Hint and Information Processor (CHIP)**
  - Replaces Online Recovery
  - Same CEP engine as the SA

Complex Event Processing

- **A set of technologies to process events and discover complex patterns among streams of events**
  - Used in financial analysis, wireless sensor networks, business process management
- **A cross between Data Base Management System and Rule Engines**
- **Main characteristics**
  - Continuous stream processing
  - Support for time/size windows, aggregation and grouping events
  - SQL-like pattern languages
    - Augmented with constructs to express event relationships (time, cause and aggregation)
    - Streams replacing tables in a continuous evaluation model

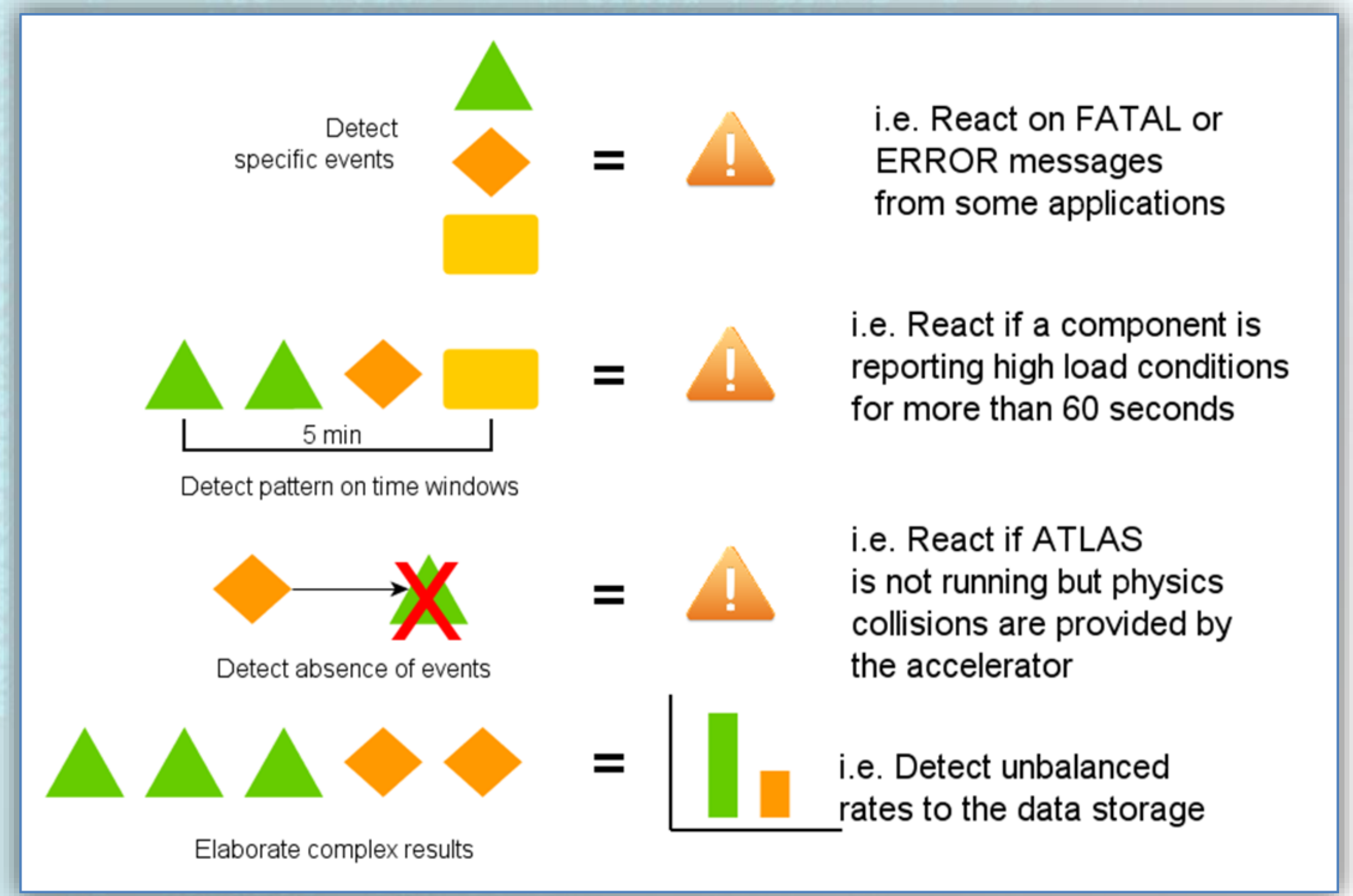
A CEP Engine - ESPER



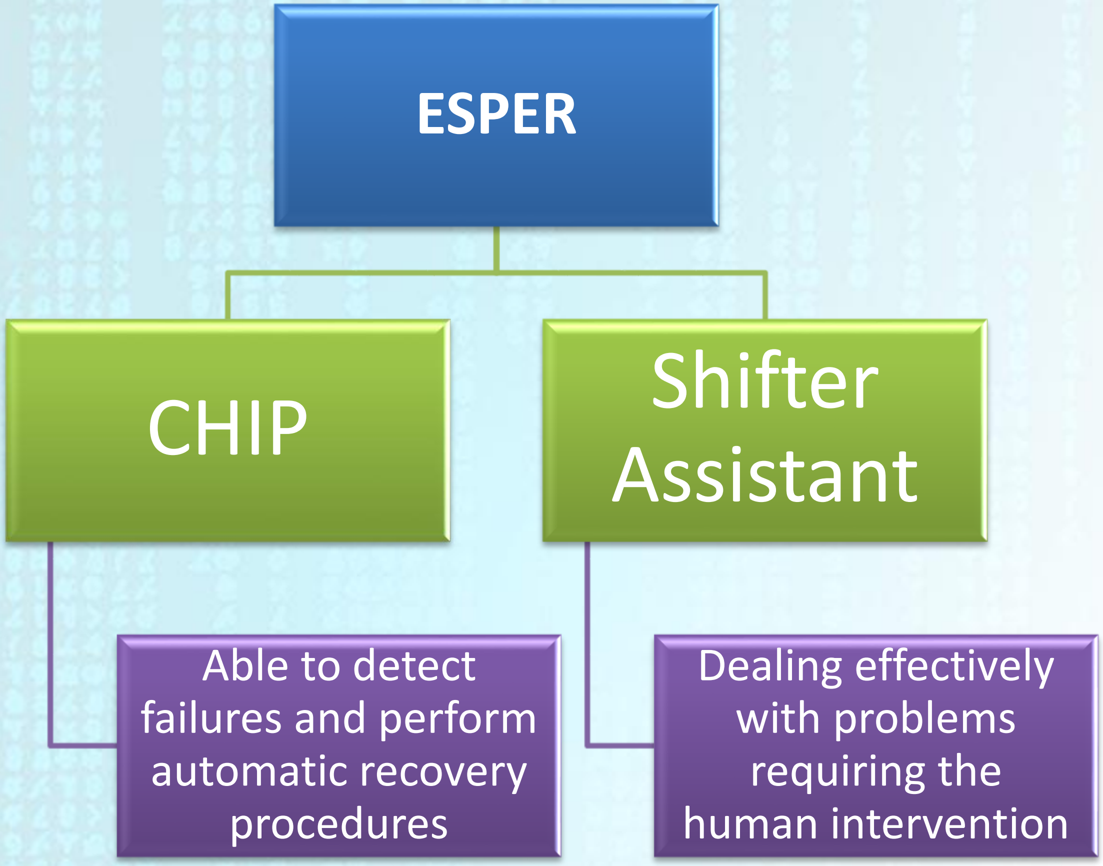
- **Java based**
  - Events as Java beans, XML documents, classes or key-value pairs
  - Fully embeddable
  - Open source
- **Support for advanced stream analysis**
  - Correlation, aggregation, sliding windows, temporal patterns
- **Knowledge base expressed in the Event Processing Language (EPL)**
  - Rich SQL-like language to express complex queries
- **Natively high-configurable multi-threaded architecture**
  - Inbound and outbound thread pools, timers
- **Support for historical data**
  - Full control over time!
- **Built-in advanced metrics**

Not available in CLIPS

Detecting Patterns

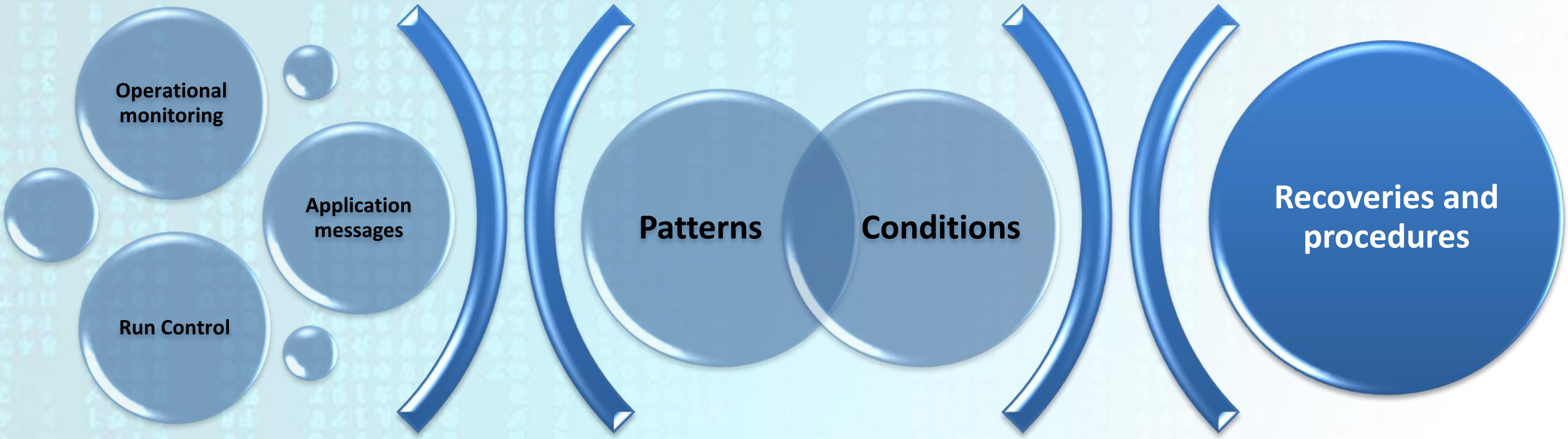


CEP in ATLAS DAQ



CEP in ATLAS DAQ

How it works



Information gathering

Information processing (Knowledge Base)

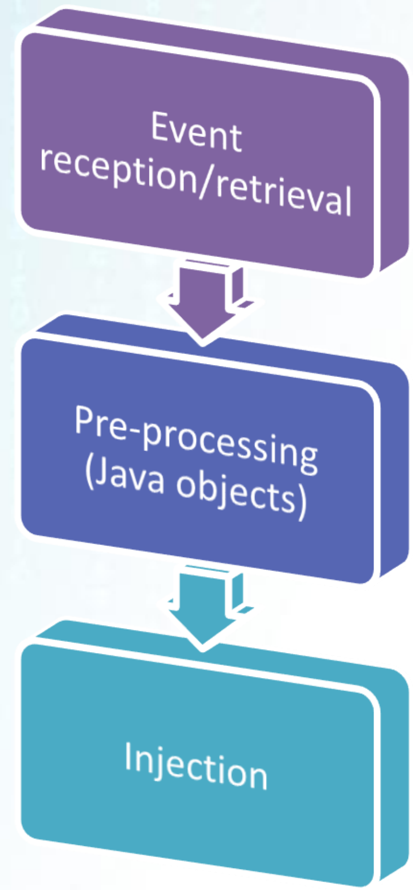
Actions

TDAQ system largely deterministic → Possible to identify "signatures" and react properly

Data Sources

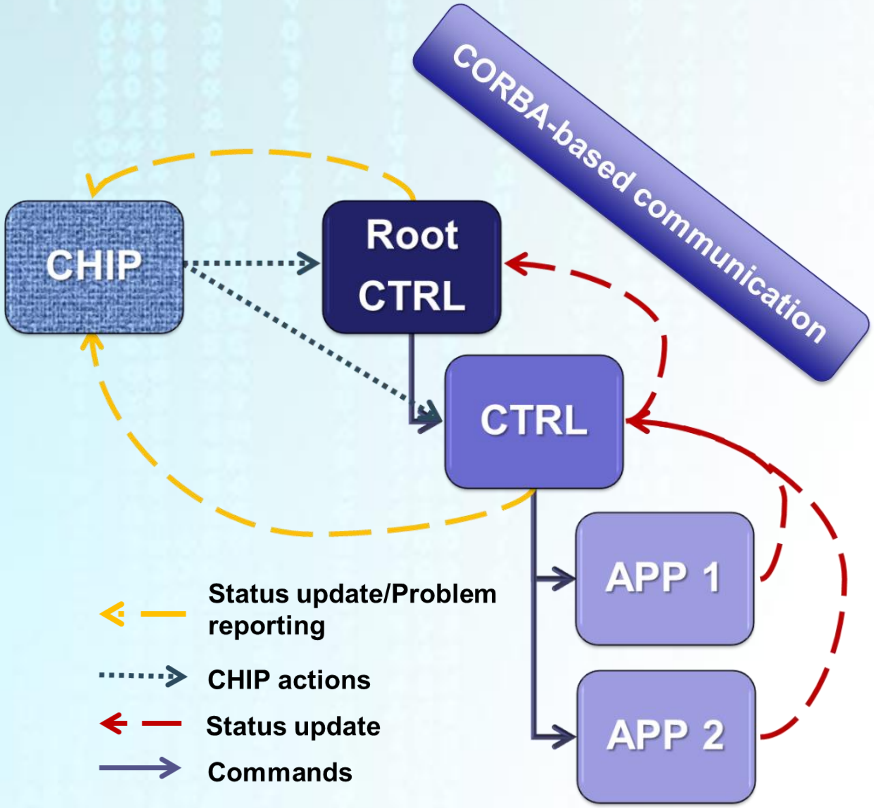
Typical information sources

Run Control	Process status
	Executed Commands
	FSM states
Application Messages	Different severities
	Reporting anomalies
	Can trigger on-demand actions
Operational Data	LHC status
	Detector working parameters
	Run parameters
System Configuration	Enabled detectors
	Application parameters
	Application hierarchy



CHIP and the Run Control

- **Applications in the Run Control (RC) are organized as a hierarchical tree**
  - "*Controllers*" are responsible of "*leaf applications*"
- **CHIP is the "brain" of the RC system**
  - Error management
  - Anomaly detection
- **Controllers and CHIP communicate through a well defined interface**



CHIP: Recoveries and Automation

Core

- Run Control error management
- Dealing with application failures

(Stop-less) Recovery

- (Re)synchronization of detectors
- Removal and re-insertion of busy channels
- Full reconfiguration of detectors

Automation

- Setting ATLAS reference clock
- Moving to physics mode with stable beams
- Detector specific procedures

Not The Full List!

More than 300 rules in 26 different contexts

CHIP & EPL: An Example

Application	Issued	Severity	Msg Id	Message
DCM:HLT-12:tpu-rack-1...	13 Mar 2016 18:02:09 CET	Fatal	rc:TransitionFailed	The transition "CONNECT" has not been properly co...

Problem detection - Application sending a FATAL error

```
@Name('INSERT_INTO_Problem_ERSFatal_New')
on ERSEvent(severity in ("FATAL", "Fatal")) as ers
insert into Problem
select
rcAppTable.controller,
ers.applicationName,
Problem$TYPE.ERS_FATAL,
Problem$STATUS._NEW,
Problem$ACTION.NONE
from RCApplicationTable as rcAppTable
where ers.applicationName = rcAppTable.name
and rcAppTable.isController = true;
```

Action decision - Set error state

```
@Name('INSERT_INTO_Problem_ERSFatal_New')
on Problem(type=Problem$TYPE.ERS_FATAL, status=Problem$STATUS._NEW, action=Problem$ACTION.NONE) as p
insert into Problem(controller, application, type, status, action)
select p.application as controller, p.application as application, p.type, p.status, Problem$ACTION.SET_ERROR;
```

Action execution - Call executor to send command

```
@Name('SUBSCRIBER_ProblemExecutor2')
@Subscriber(className='chip.subscriber.core.ProblemExecutorSynch')
select * from ProblemExecutor(action in (Problem$ACTION.SET_ERROR, Problem$ACTION.REMOVE_ERROR));
```

CHIP & EPL: More Complex Example

```
@Name('INSERT_INTO_Problem_BadHost_Resolved')
@Hint('PREFER_MERGE_JOIN')
context SegmentedByProblemPerApplication
insert into Problem
select
p.controller,
p.application,
Problem$TYPE.BAD_HOST,
Problem$STATUS.RESOLVED,
Problem$ACTION.NONE
from
pattern [ every p = Problem(type = Problem$TYPE.BAD_HOST, status = Problem$STATUS.WAIT_FOR_RESOLVED) ->
          (
            (every t = TestFollowUp(applicationName = p.application,
                                   globalTestResult = TestResult.PASSED,
                                   compTestResult.testResult = TestResult.PASSED,
                                   status = TestFollowUp$STATUS._NEW,
                                   action = TestFollowUp$ACTION.NONE))
             or
            (every app = RCApplication(name = p.application))
          )
          and not Problem(application = p.application,
                          type = Problem$TYPE.BAD_HOST,
                          status in (Problem$STATUS.RESOLVED, Problem$STATUS.DONE)) ] as pat
where
pat.t.component is (select runningHost from RCApplicationTable where name = pat.p.application)
or app.status is not STATUS.UP;
```

Temporal pattern

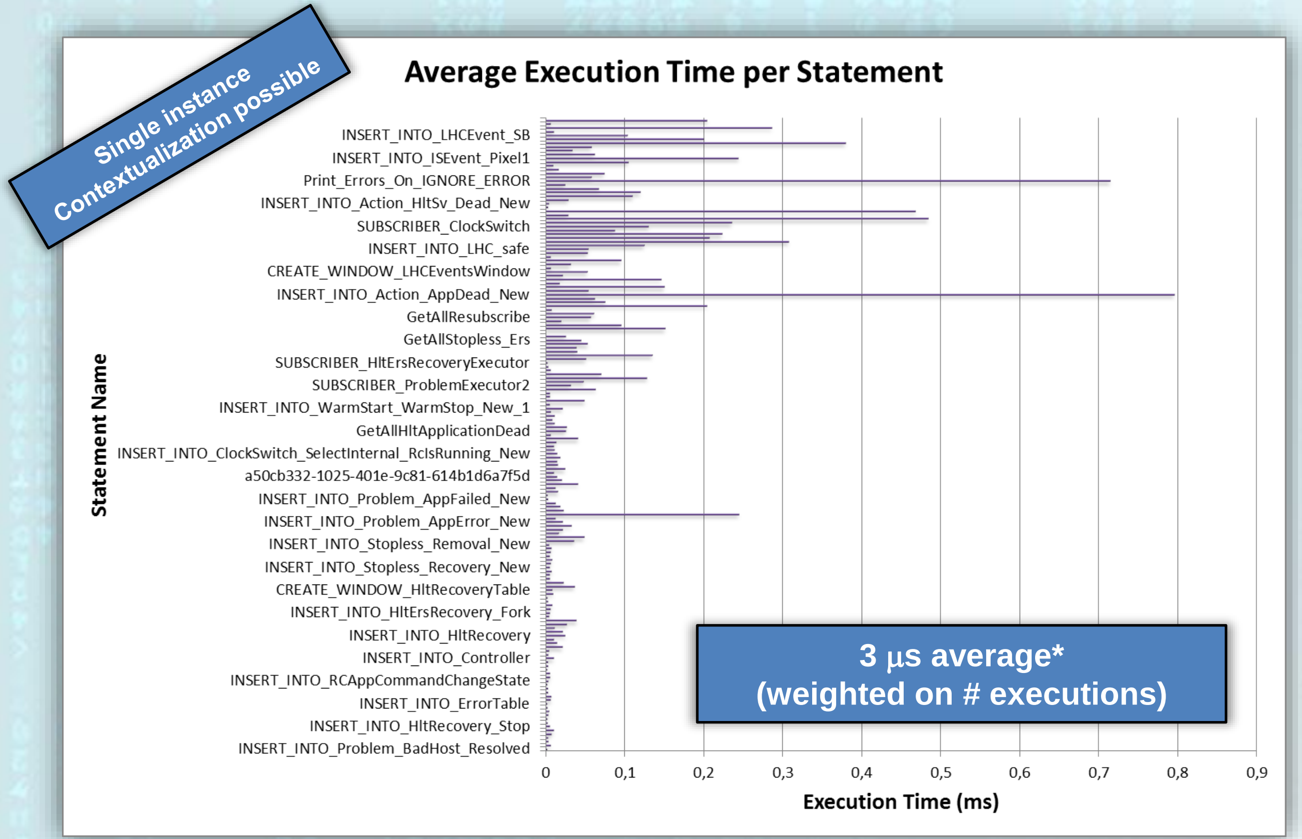
Different streams

Sub-query

CHIP: Metrics Analysis

- **Exploiting ESPER's built-in metrics**
  - Detailed information for every single rule in the knowledge base
- **Real-time and historical data**
  - Leveraging the flexibility of P-Beast (see Igor's talk on Thursday) and Grafana
- **"Live" enabling/disabling and configuration**

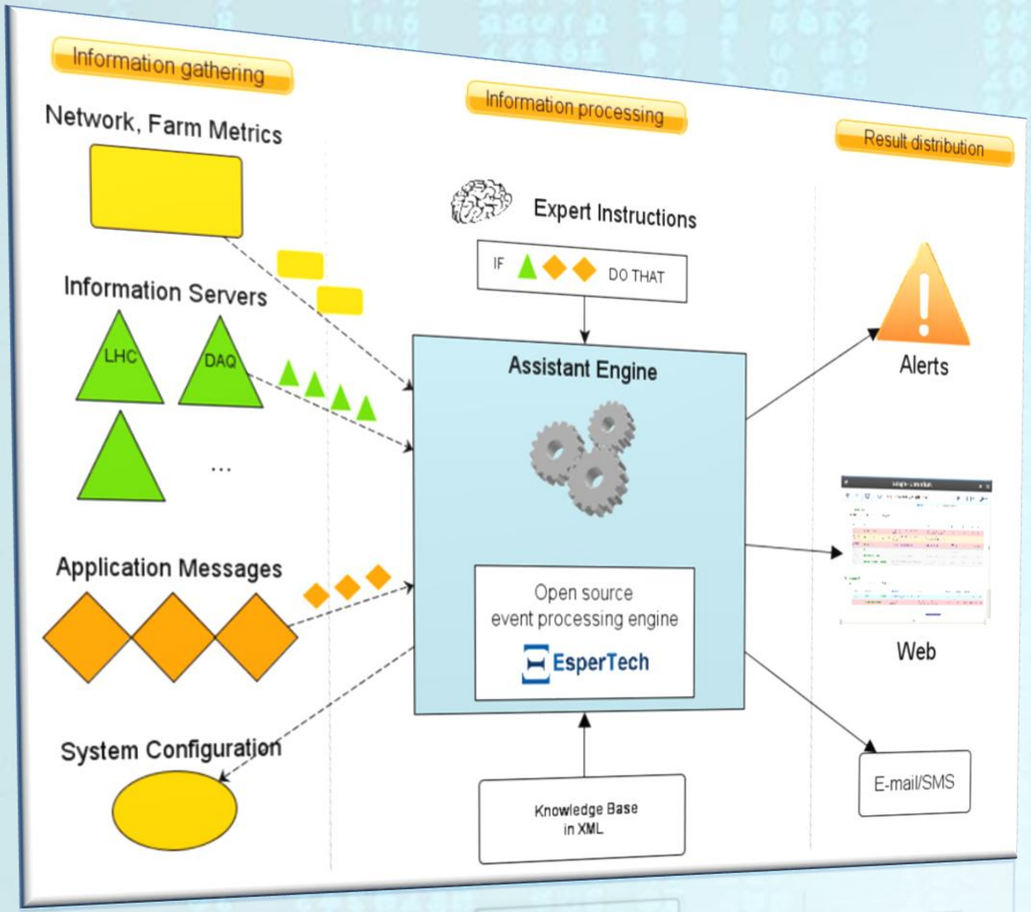
CHIP: Performances



* Running on Intel Xeon E5-2680 V3 - 64 GB RAM

- **Standard ATLAS configuration**
- **DAQ system cycled through various states**
- **Provoked various failures**

The Shifter Assistant



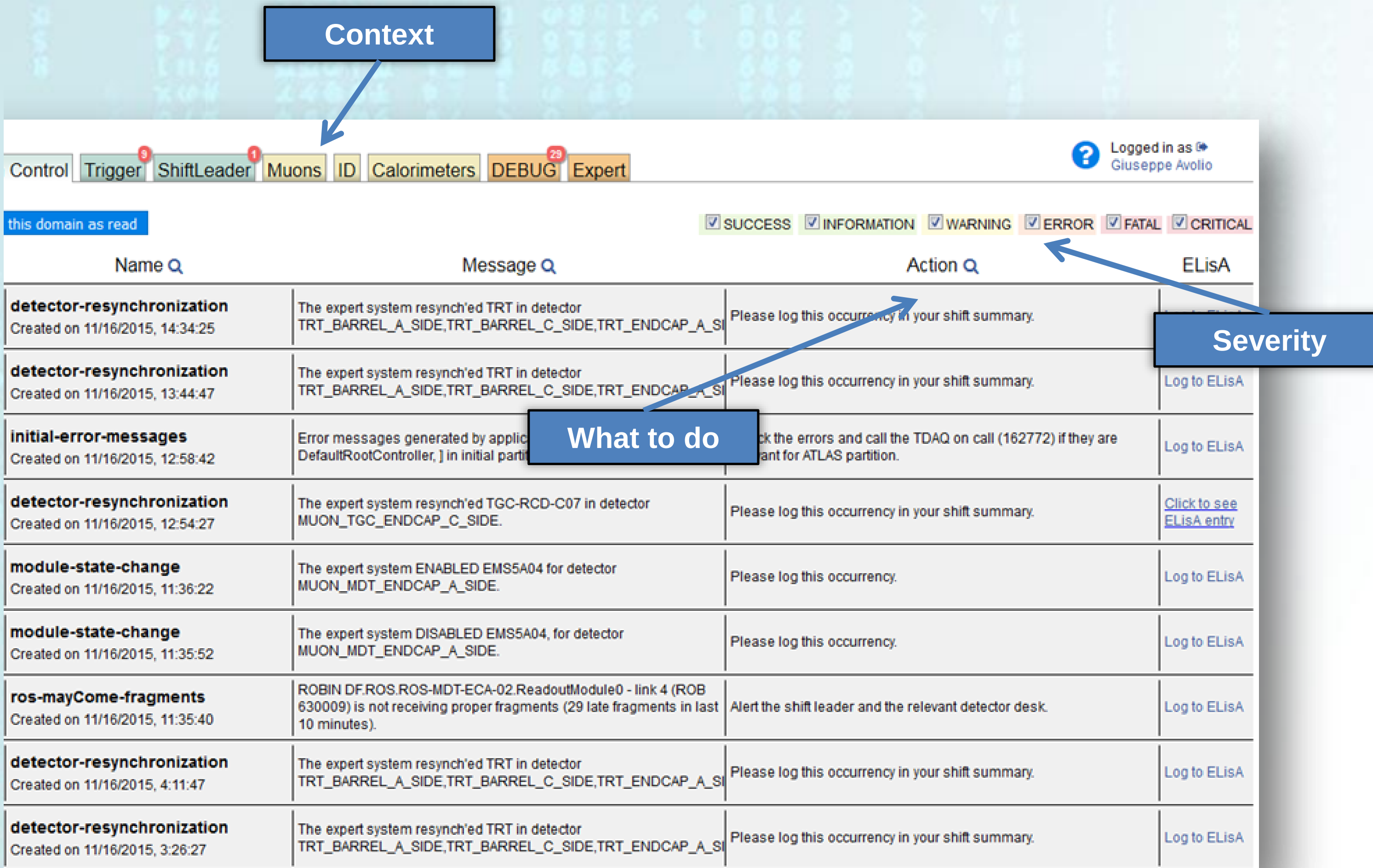
What

- A tool meant at guiding the operator in his daily work
  - Diagnosing problematic situations and suggesting action to take
  - Remind the operator he should (not) do something

Aim

- Reduce and simplify shifter tasks
- Help shifters with more detailed and pertinent information
- Be more efficient, avoid repetition
- Promptly notify operators of problems and failures

The Shifter Assistant: Web Interface

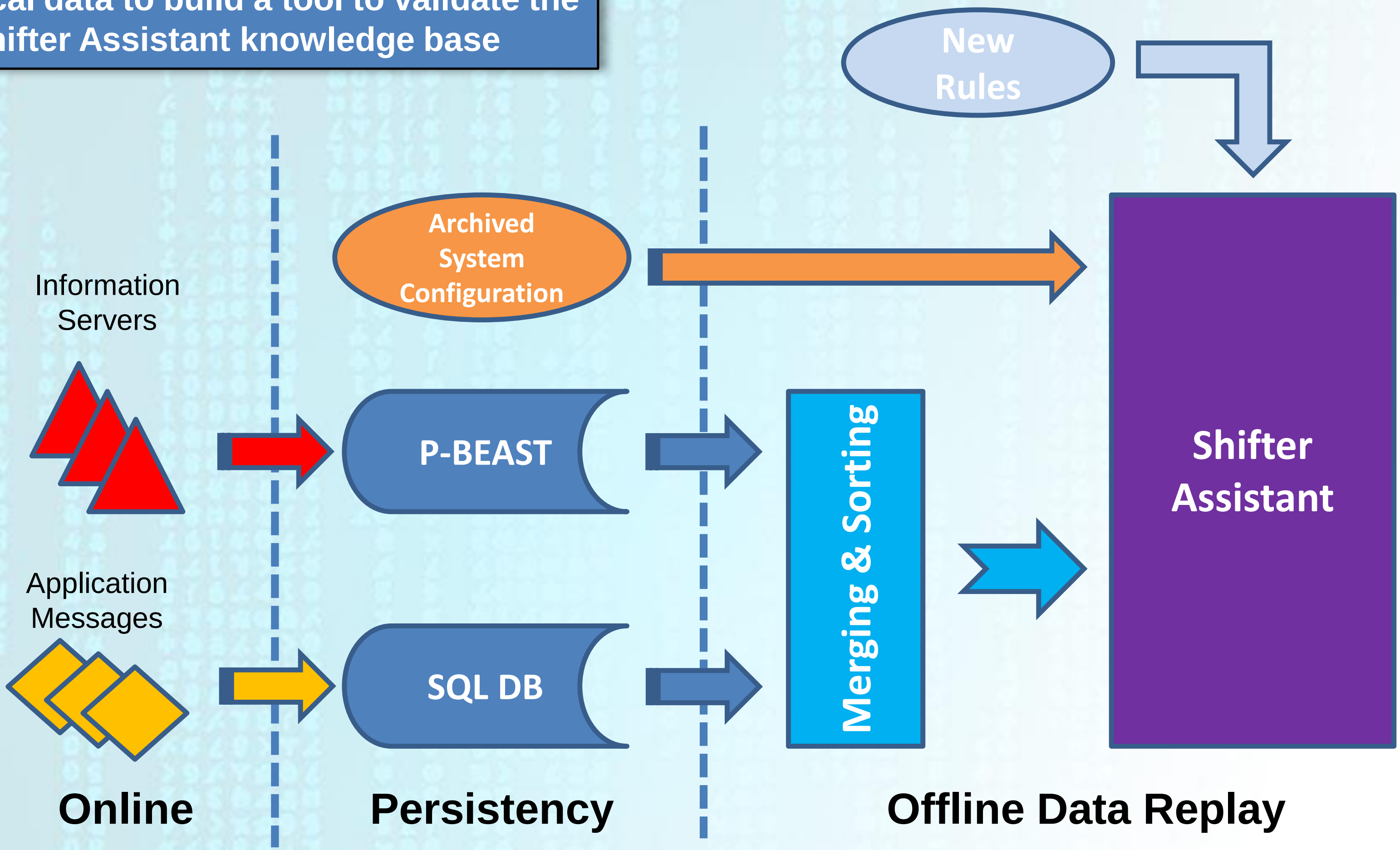


Name	Message	Action	ELisA
detector-resynchronization Created on 11/16/2015, 14:34:25	The expert system resynch'ed TRT in detector TRT_BARREL_A_SIDE,TRT_BARREL_C_SIDE,TRT_ENDCAP_A_SI	Please log this occurrency in your shift summary.	
detector-resynchronization Created on 11/16/2015, 13:44:47	The expert system resynch'ed TRT in detector TRT_BARREL_A_SIDE,TRT_BARREL_C_SIDE,TRT_ENDCAP_A_SI	Please log this occurrency in your shift summary.	Log to ELisA
initial-error-messages Created on 11/16/2015, 12:58:42	Error messages generated by applic… DefaultRootController,] in initial parti…	…k the errors and call the TDAQ on call (162772) if they are …ant for ATLAS partition.	Log to ELisA
detector-resynchronization Created on 11/16/2015, 12:54:27	The expert system resynch'ed TGC-RCD-C07 in detector MUON_TGC_ENDCAP_C_SIDE.	Please log this occurrency in your shift summary.	Click to see ELisA entry
module-state-change Created on 11/16/2015, 11:36:22	The expert system ENABLED EMS5A04 for detector MUON_MDT_ENDCAP_A_SIDE.	Please log this occurrency.	Log to ELisA
module-state-change Created on 11/16/2015, 11:35:52	The expert system DISABLED EMS5A04, for detector MUON_MDT_ENDCAP_A_SIDE.	Please log this occurrency.	Log to ELisA
ros-mayCome-fragments Created on 11/16/2015, 11:35:40	ROBIN DF.ROS.ROS-MDT-ECA-02.ReadoutModule0 - link 4 (ROB 630009) is not receiving proper fragments (29 late fragments in last 10 minutes).	Alert the shift leader and the relevant detector desk.	Log to ELisA
detector-resynchronization Created on 11/16/2015, 4:11:47	The expert system resynch'ed TRT in detector TRT_BARREL_A_SIDE,TRT_BARREL_C_SIDE,TRT_ENDCAP_A_SI	Please log this occurrency in your shift summary.	Log to ELisA
detector-resynchronization Created on 11/16/2015, 3:26:27	The expert system resynch'ed TRT in detector TRT_BARREL_A_SIDE,TRT_BARREL_C_SIDE,TRT_ENDCAP_A_SI	Please log this occurrency in your shift summary.	Log to ELisA

The Shifter Assistant Replay

Leveraging ESPER's support for import of historical data to build a tool to validate the Shifter Assistant knowledge base

Facilitate sharing of knowledge, safer and easier integration of new rules



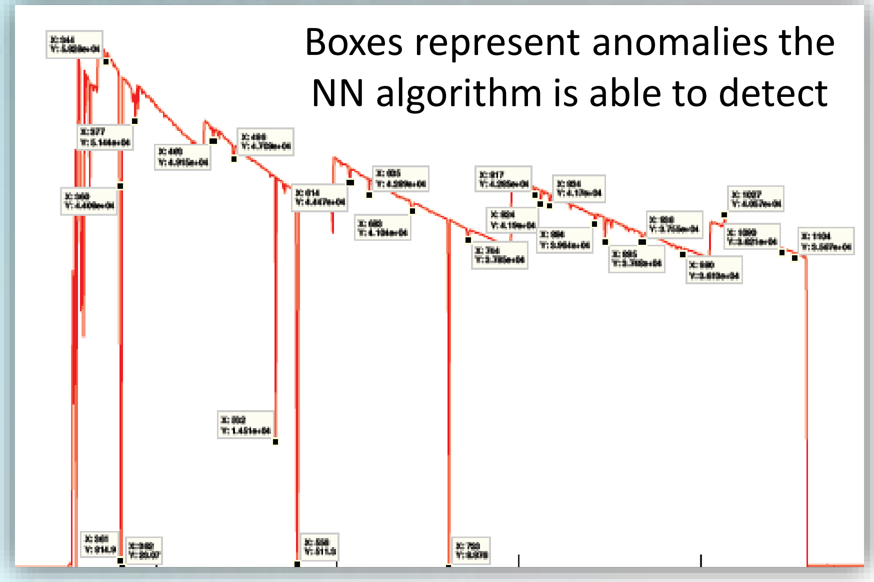
Going Further: Neural Networks

Beyond static definitions of rules

A Different Approach

- Teach the NN what a “normal” system behavior is
- Identify abnormal situations independently of the parameter space
  - Classification of anomalies

A Typical Fill Profile



Conclusions & Outlook

- **In both Run 1 and Run 2 the use of Expert Systems for automation and recovery proved to be a valuable asset**
  – Reduce probability of mistakes
  – Improve latency (computers are faster than humans…)
- **The introduction of a CEP engine has added flexibility and simplification**
  – Improved anomaly detection
  – New complex patterns
  – Re-use of historical data
  – Advanced configuration
- **Looking forward for more advanced automation and anomaly detection**