Content

Soleil Expert System

Majid OUNSY

Synchrotron SOLEIL, Saint Aubin, France,
http://www.synchrotron-soleil.fr

Content

- **Context**
- *SOLEIL Expert System Building Blocks*
  - *Passerelle*
  - *Drools*
  - *CDMA*
- *Use Case*

Repetitive Accelerator Operation failure check tasks

⇨ In daily operation an operator has to perform many manual checks

- Are the control system services working fine (is the archiving system really logging data
- Is the alarm detected by a supervision application linked to a control system sub module or is it an equipment problem
- Is beam correctly delivered to beamlines

- Same kind of problems **analysis** to do again and again
- Many **different applications** to interact with
- operator **knowledge** dependent
- Very **error-prone**
- Why not **automate** all these tasks?

Repetitive Accelerator Operation analysis and diagnosis tasks

⇨ In case of abnormal operation (beam loss)

- Collect data for analysis (extract archived data, post-mortem data,…)
- Check the elog book or an accelerator expert to see what is the relevant recovery process to apply
- Perform in sequence the advised operation/rules for the given situation

- No uniform to collect data from different sources
- Lack of centralization accelerator operation expertise
- Non automated repetitive sequences of operations
- Very error-prone
- Why not use an expert system ?

Aim of the expert system

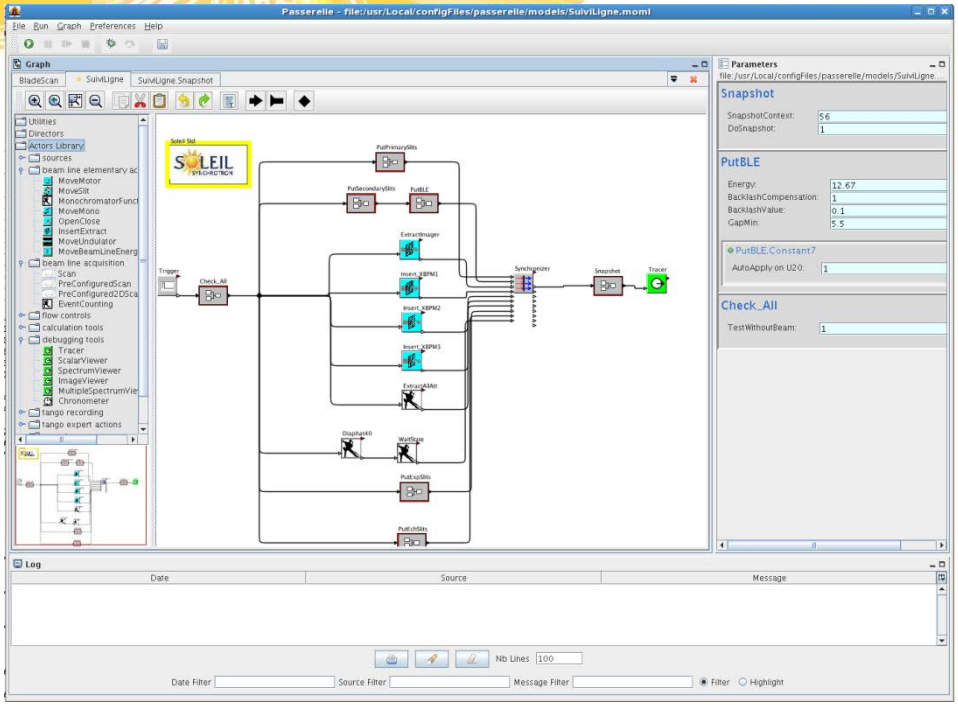
⇨ Automation of an analysis/diagnosis process workflow takes care of :

- Collecting and consulting the available data for analysis
- Formulating decisions/diagnosis rules
- Preparing advised actions to repair the problematic situation

- Need for a workflow modeling environment : **Passerelle**
- Need for a uniform data access layer : **CDMA**
- Need for a rules based environment : **Drools**

PASSERELLE

PASSERELLE allow to graphically design complex workflows and execute them

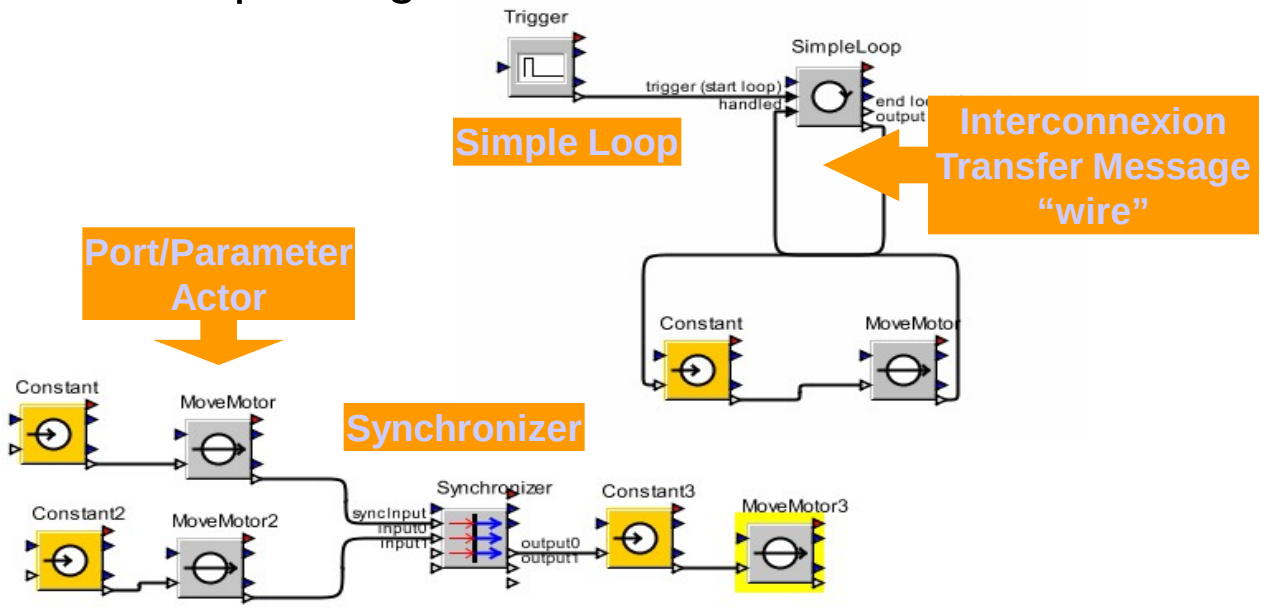


- **PASSERELLE is provided by a company called ISENCIA**
  - ✓ **Specialized in process driving for industrial companies.**
- **PASSERELLE is based on an environment for scientific modelisation and simulation: PTOLEMY (developed by the Berkeley University)**

http://ptolemy.eecs.berkeley.edu/ptolemyII/

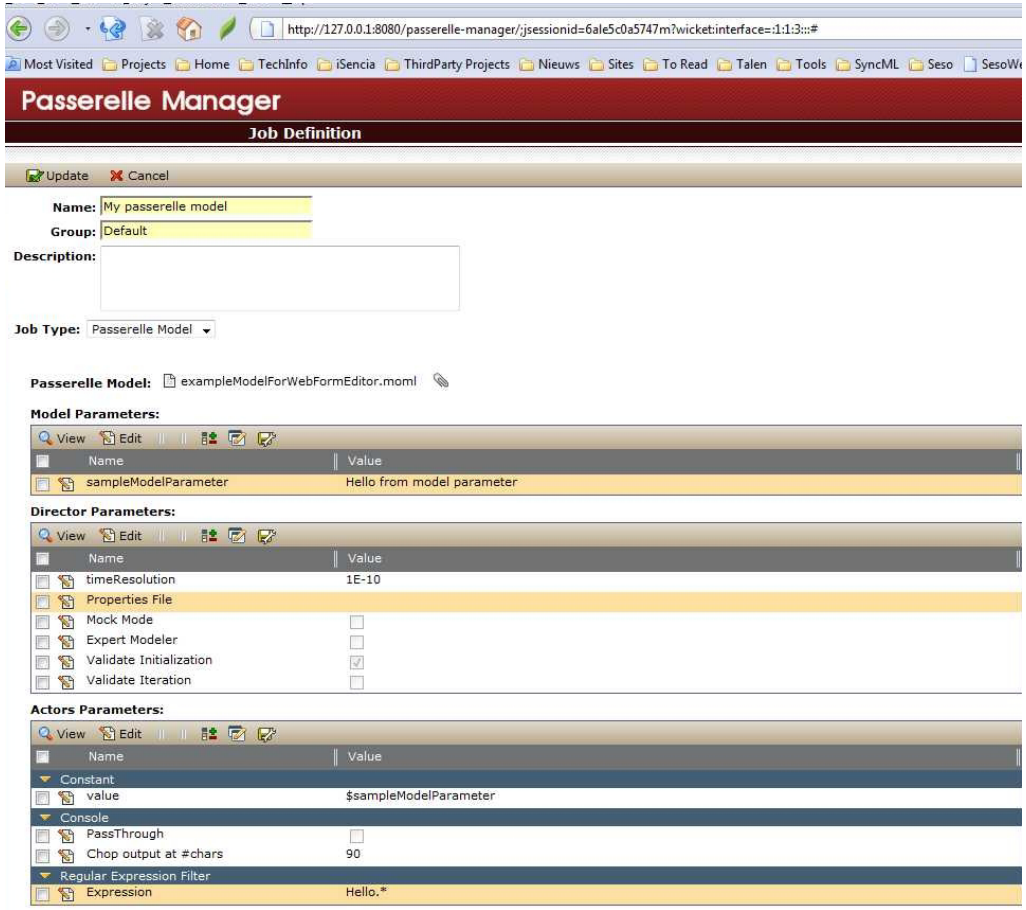
PASSERELLE : principle

- We connect « boxes » and « wires »:
  - The « boxes » are called **ACTORS**, they execute an action.
  - The « wires » are called **MESSAGES**, they transfer data.
- The graphic language for editing sequences provided all functionalities to build complex logics:



- Loops
- Comparison
- Error Management
- Parameterization

PASSERELLE : Remote Execution Environment



Full-featured remote access via std web technology

- ✓ **Collaborate on process design, execution, monitoring etc on a secure and robust server-based platform**
- ✓ **Automated scheduling of process executions**
- ✓ **Monitoring and diagnostics within same platform**
- ✓ **Base suite extensible "live" through OSGi bundle upload & integrated management console**
- ✓ **Process definitions, config data, analysis rules, traces & OSGi bundles all in relational DB**

Drools : The Core

- **Rules engine**
  - “advanced if-then”
  - Declarative programming model
  - Evaluate collection of registered “facts” in a “knowledge base”
  - New facts can be derived by the rules, and added to the knowledge base
  - This can trigger (other) rules again
  - Easy-to-use Java API to integrate Drools in a “normal” Java program

Passerelle & Drools integration

- **Specific actor library**
  - Data collection
    - Adaptations of existing Tango actors (AttributeReader etc)
    - SQL-DB-reader
    - File reader
    - Web-service adapters
  - Analysis
    - Request-based Drools Expert actor
    - Stream-based Drools Fusion actor
  - Diagnosis
    - Report generator (based on eclipse BIRT)
    - Reuse request-based Drools Expert actor

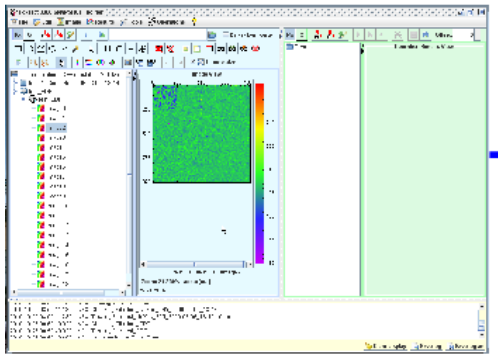
CDMA

A API aimed to provide to programmers a data access layer that is independent of the scientific datasets format.

Key points:

- A plug-in system that allows support of various data sources
- An abstract interface for navigation through data sets
  - Concrete classes are provided by data plug-ins
- A dictionary mechanism to retrieve measurements and technical values independently of the data structures

It is intended to be the data access layer of data reduction applications.



keywords Declaration file

```
<data-def name="Experiment name">
<!-- ex: EXAFS, SAXS,... →

  <item key="mono_energy">
  <item key="mono_type">

</data-def>
```

CDM

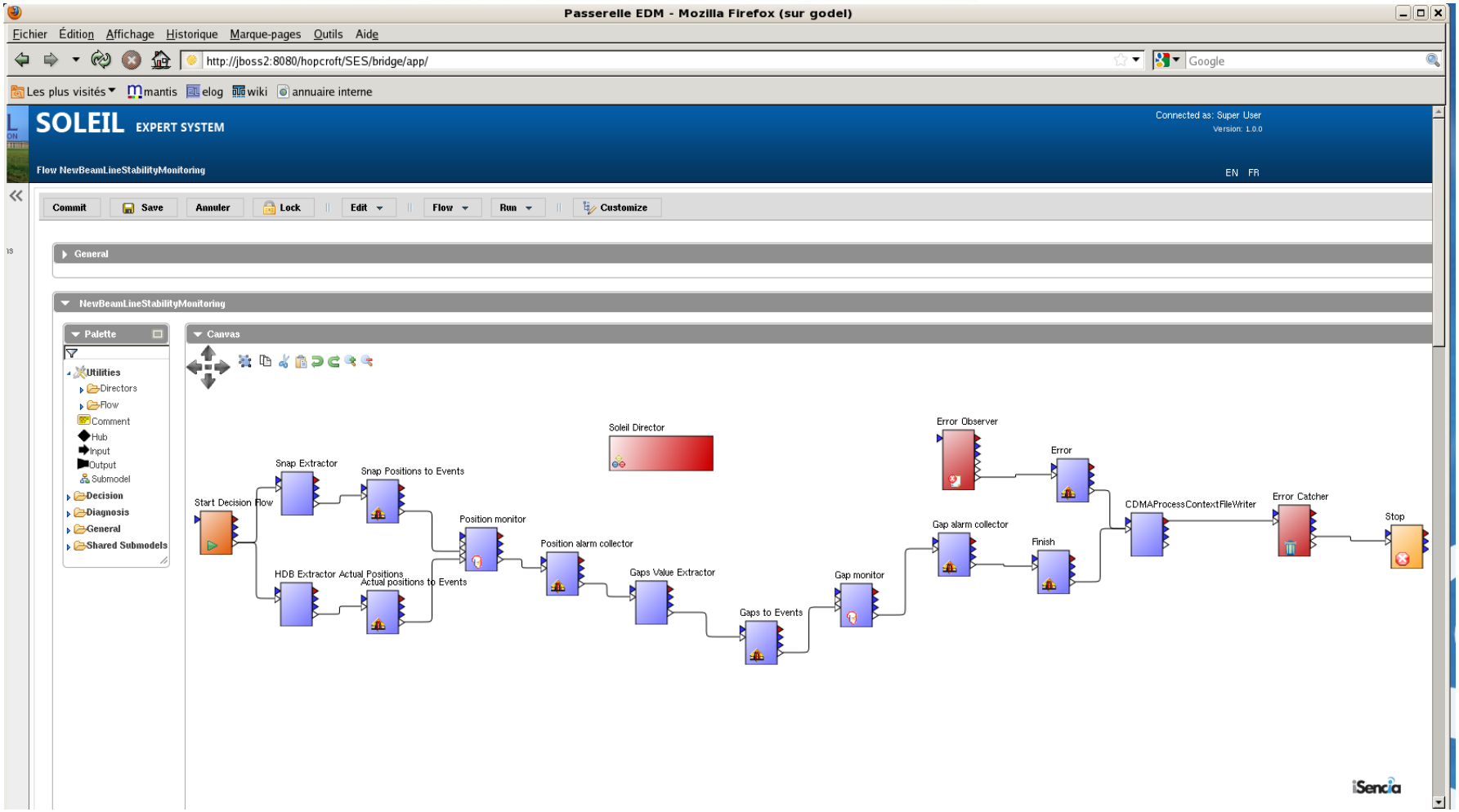
Files format plugin

keywords Mapping file

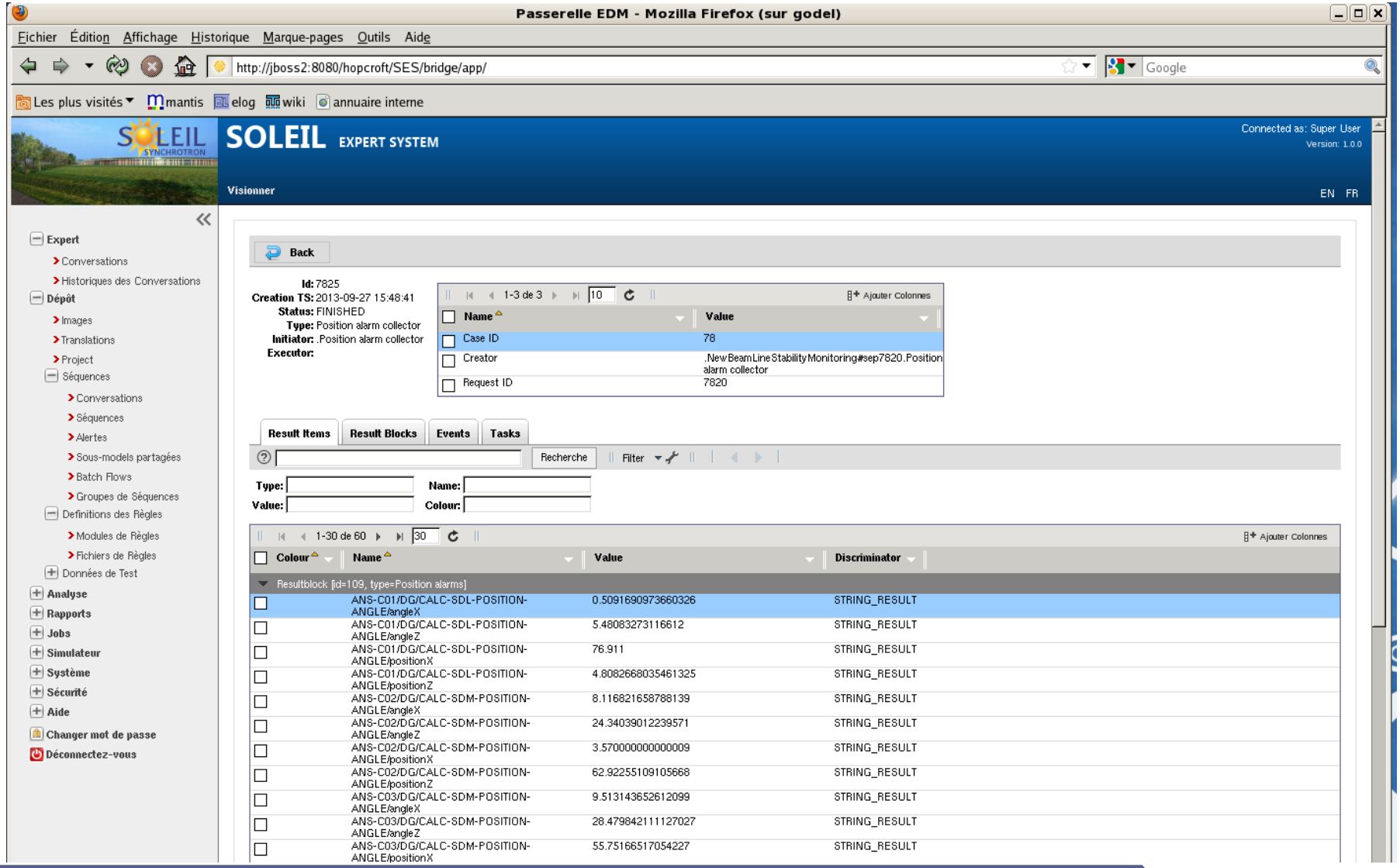
```
<map-def name="Experiment name">
<!-- ex: EXAFS, SAXS,... -->

  <item key="mono_energy">
    <path>path/to/user/mono/energy</path>
  </item>
  <item key="mono_type">
    <path>path/to/user/mono/type</path>
  </item>
  ...
</map-def>
```

0100110
1001110
0100110
11110...

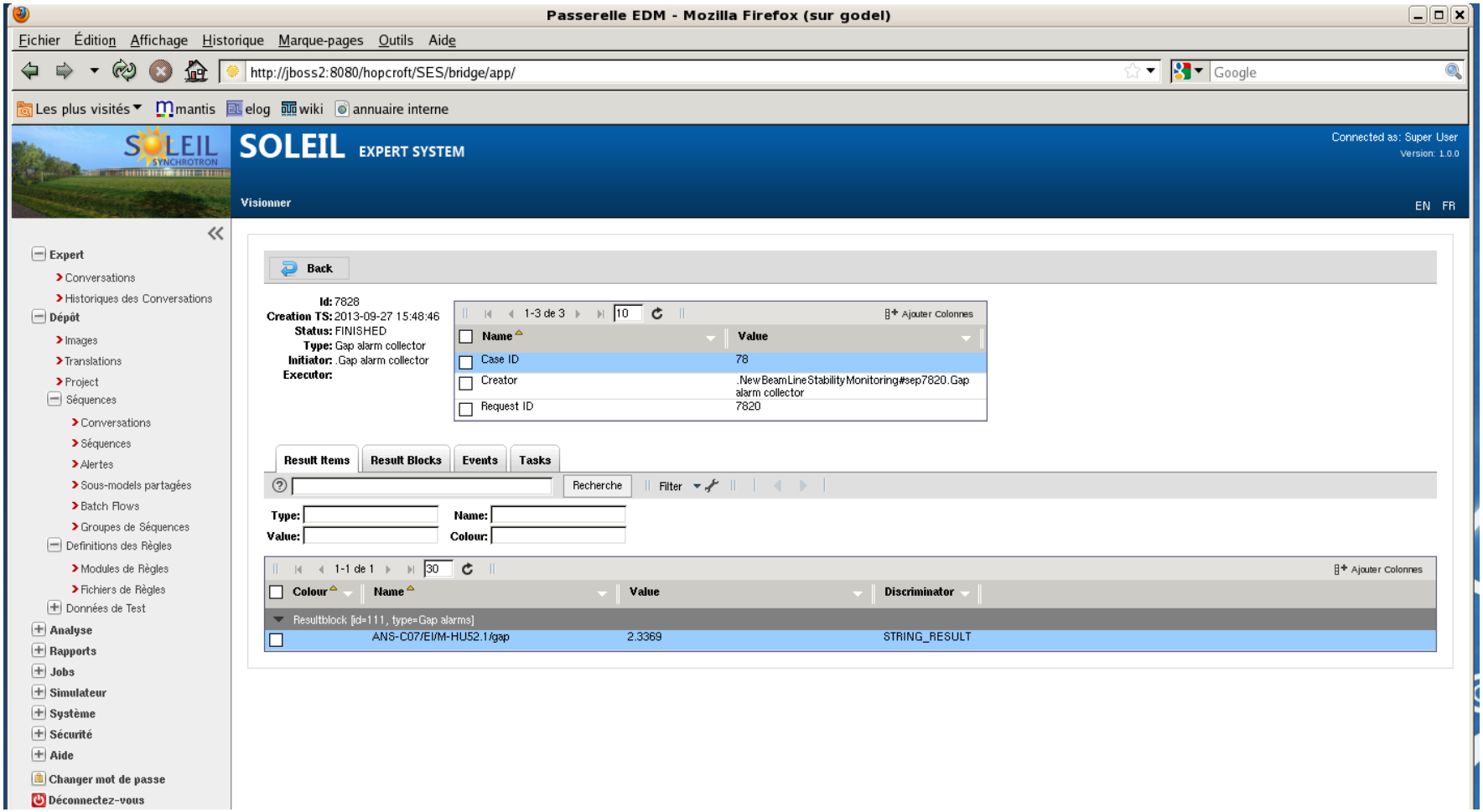
Simple Beam Stability Scenario



Beam Position Alarms



Possible Reason : Insertion Gap Positions



Thanks for your attention