Nightly Tests for the Trigger Configuration

Sylvie Brunet

Plan

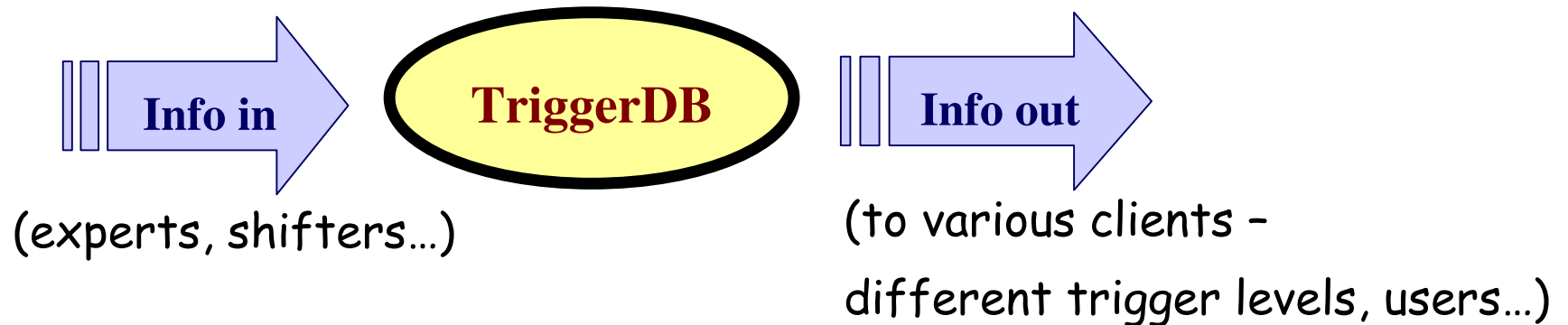**Nightly (RTT) Tests for the Trigger Configuration:**

- **What, why and how?**

- **What is already in place?**

- **What needs to be done?**

What,why and how? (I)

● Trigger Configuration

◉ "The Trigger Configuration system takes care of the consistent configuration of all trigger levels for online event selection & offline trigger simulation. *(It also guarantees the archival of the configuration information used and provides tools easily modify and change configurations)"*

Info in → **TriggerDB** → **Info out**

(experts, shifters…)

(to various clients –

different trigger levels, users…)

◉ Important for the whole trigger

◉ **We want to make sure that the developed software & tools are behaving properly** (offline and online)

◉ To do that we need a tool…

What,why and how? (II)

Run Time Tester (RTT)

"The **Run Time Tester** is a Python-coded testing framework that can set up and run jobs in an automated manner, perform post-job actions and then report results to a web-accessible location."

> Not only nightly builts (code compiling or not)
>
> RTT really runs code and perform different kinds of tests
>
> Already used for many packages in ATLAS

A « typical » RTT job is :
- Run python option file with athena
    (run on some events, produce plots, output files…)
- Compare/perform tests on the outputs
    (ROOT macros, file grepper etc)

What,why and how? (III)

Creating a RTT test for your package - the recipe -

- Have some code to run in your package

- Prepare a .xml file (also in your package) which gives instructions to RTT
  - Code to be run
  - Tests to be performed

- Commit and Tag your Code

- Add your tag into the nightlies (Tag Collector)

- Tell the RTT folks that a brand new test has been created (they have to update their libraries once)

- Watch the results on the web interface. Understand what worked/what didn't...

- Modify your tests consequently & redo the steps as many times as you want!

What,why and how? (IV)

A first "*simple*" RTT test for the Trigger Configuration

Run HLTtest_standalone.py (taken from TriggerRelease package, which is already widely used by people trying to integrate the offline trigger sw to the online running environment)

- run on events with a given trigger configuration (from options files)
- (by adding some lines) gets the trigger conf. info as output .txt file
- do that for HLT: LVL2 and Event Filter

Convert .txt files into .xml files

Load the test database with these .xml files

Re-run HLTtest_standalone.py **from the database** (LVL2 & EF)

Post-job actions: check/compare outputs, etc…

What,why and how? (V)

"simple" but not sooo simple…

● SW is evolving very very fast. New packages to add, new setups, new versions of compilers etc… Just running locally (with success) the steps mentioned in the previous page is a challenge by itself… -also means that these sw tests are important!

● In the TrigConfiguration package, many « standalone » applications, which is not supported for the moment by the RTT framework. Have to hack a bit… Running an application from ROOT macros or directly using « system » in a Python script…

● Tests require the use of AthenaPT/MT (based on Athena but « mimic » the online environment) which were not supported by the RTT framework until last week. Now it seems to work!

● Does not correspond (in general) to typical RTT jobs…

● Needs adaptation…

What is already in place?

Already worked out:

- have a dedicated oracle DB @ CERN for these tests

- have a new package dedicated for these tests (TrigConfiguration/TrigConfTest/)

- Some code has been written and locally tested for a part of the « simple » test. (now waiting for appropriate permissions to add everything in the nightlies and see how it goes.)

What needs to be done?

Medium-short term:

- Complete, understand and debug the « simple test » in the ATLAS & RTT framework.

The best is still to come:

- Decide (within the Trigger Configuration group) on other useful tests for the Trigger Configuration and setting that up!

- Then, just have to keep an eye on the nightly results...

Summary

● **We want to test the trigger configuration software (offline and pseudo-online) in the Run Time Tester framework**

● **Beside understanding how it works, some concrete work has started. Need to complete & validate the "not so simple" test and eventually set up more tests.**

● **Will be really useful to track bugs and strange behaviors related to the trigger configuration – which is very important for the whole trigger…**