



Deliverable N°: D1.21

Deliverable Title: Development of an analysis tool for reliability studies at PITZ

Task: DS-1

Author: G. Asova et al.

Contract N°: 011935

**Project funded by the European Community
under the “Structuring the European Research Area” Specific Programme
Research Infrastructures action**

DAQ representation improvements

G. Asova

Outline

- News in DAQ since PITZ1
- DAQ data analysis
 - Environment and hosts
 - Libraries for analysis
 - daqbr
 - what's new
 - examples

News in DAQ

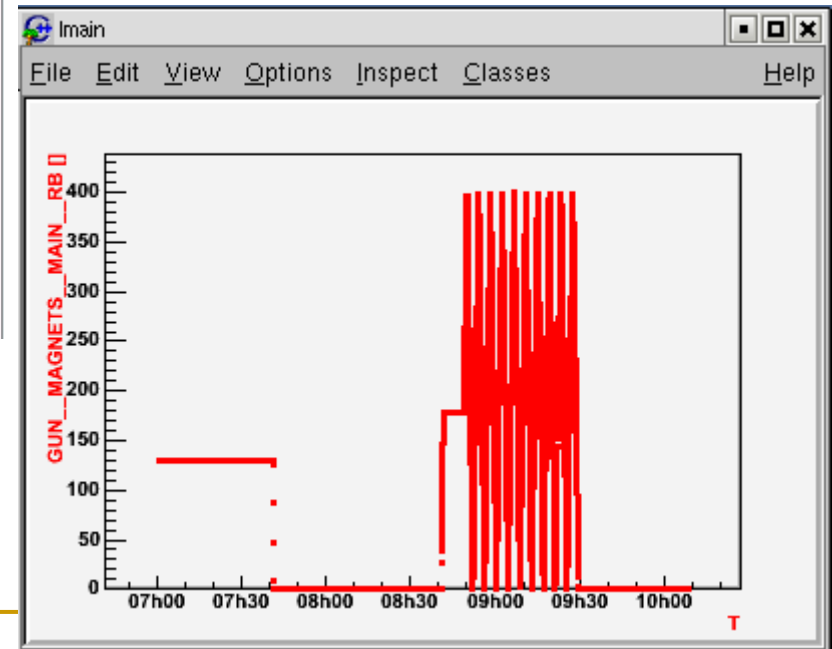
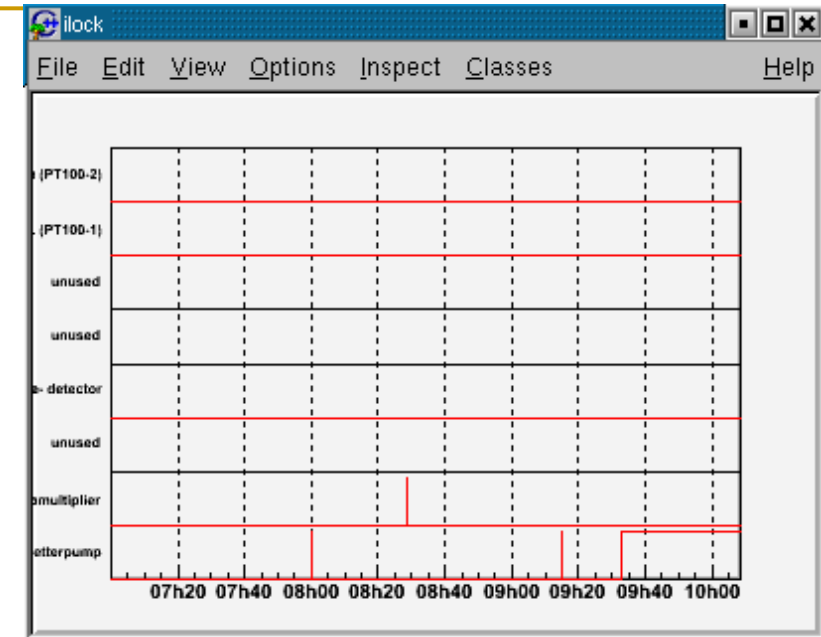
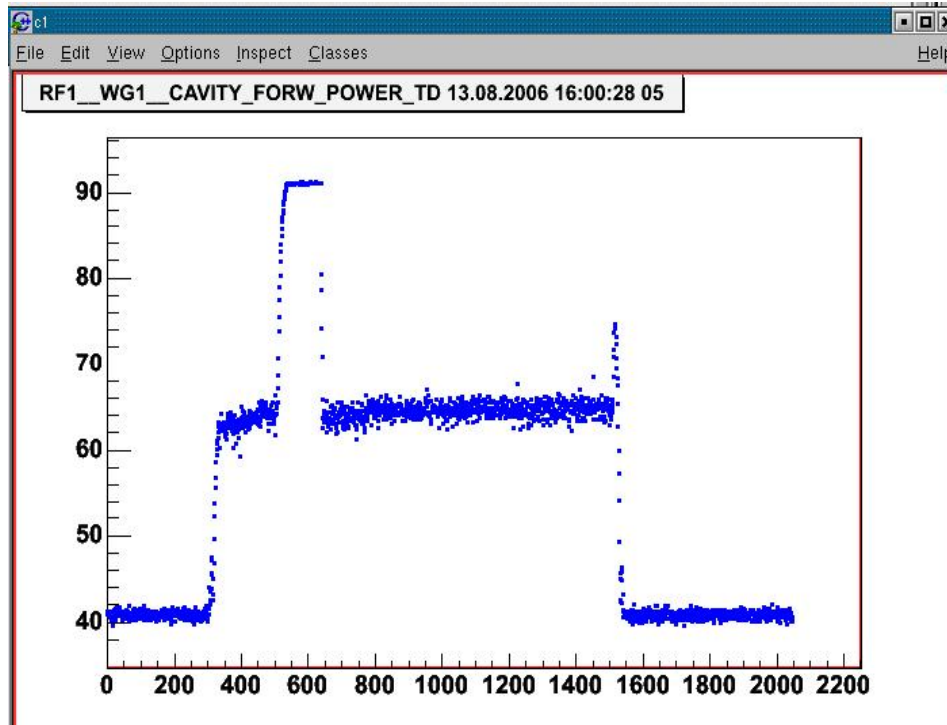
- Data are stored on dCache read pools immediately after closing a file
 - Online analysis will be bad idea
 - 405 channels and more to come (< 250 before)
 - class TTFFloatData {
private:
 TArrayF *data;
 Bool_t _delete_data;
 TTFFData::Status data_status;
 TTFFEventTime eventtime;
 Int_t data_from;
 Int_t data_step;
 Int_t data_type;
 ...
};
-

Why it is necessary?

- **data_type**
 - 2 -> single float value (e.g. GUN__MAGNETS__MAIN__SP)
 - 4 -> status (bit – 0/1) (e. g. I_LOCK__BIS1__DI_0_TO_7)
 - 19 -> float array (e.g. RF1__WG1__CAVITY_REFL_POWER_TD)

- **3 different data types and the analysis techniques for each of them differ**
 - Single value can be used as such in any analysis code
 - Status bits should be extracted properly from a “binary array”
 - Vector data includes noise which usually is subtracted

Like this



Environment and hosts

- DAQ is running on *picus1* & *picus2*
 - SL3 with ROOT 4.00.08

- DAQ analysis is to be done on *picus4*
 - 64-bit SL3 machine
 - ROOT 4.00.08
 - Qt 3.2.1

http://www-zeuthen.desy.de/pitz/control_doc/pitz_hosts/index.shtml#Anchor-Specific-47857

Libs for analysis -> http://www-zeuthen.desy.de/pitz/control_doc/load_daq_libs/index.shtml

daqbr

- Reads data directly from dCache
 - A bit slower, but still fast enough since almost all files are on a read pool
 - All data from the beginning of PITZ1.6 is there

Disk Space Usage

CellName	DomainName	Total Space/MB	Free Space/MB	Precious Space/MB	Layout (precious/used free)
grid-zyklop19-0	zyklop19Domain	1356800	1210091	145754	
grid-zyklop20-0	zyklop20Domain	1356800	1210546	146253	
pitz-hugin-0	pitz-hugin-0Domain	512000	63366	0	
pitz-hugin-1	pitz-hugin-1Domain	512000	63500	0	
pitz-hugin-2	pitz-hugin-2Domain	512000	63746	0	
pitz-hugin-3	pitz-hugin-3Domain	512000	63762	0	
pitz-hugin-4	pitz-hugin-4Domain	194560	194560	0	
pitz-hugin-5	pitz-hugin-5Domain	256000	256000	0	
pitz-kunin-0	pitz-kunin-0Domain	512000	508778	0	
pitz-kunin-1	pitz-kunin-1Domain	512000	509262	0	
pitz-kunin-2	pitz-kunin-2Domain	537600	508987	0	
pitz-kunin-3	pitz-kunin-3Domain	537600	508511	0	

daqbr...

- Docu http://www-zeuthen.desy.de/pitz/control_doc/DAQbrowser/index.shtml
- Data types for analysis
 - DaqFloat, DaqFloatArray, DaqBits {
 cVarName,
 daqName,
 doocsName,
 units,
 firstIndex, lastIndex,
 frequency,
 device_type,
 bit_titles
};
 - Why?
dq_GUN__MAGNETS__MAIN__RB
dq_LASER__PHOTO_MULT_VIRT_CATH1_TD[1234]

Example

- ❑ Extract data with the acquisition rate of a chosen property (leading) – default is the value of RepRate
 - Simple plot and output of single, vector and status data
 - Add simple filters
 - Add status filters
 - Set C++ encoded filters/plots/output to remind the meaning of the 3 types for analysis

Small hints

Time | Import | Leading Property | Machine Status | Filter | Plots + Histos | Output

Property Selector	DAQ Name	F / D / L / P
<ul style="list-style-type: none"> ⊕ BOOST <ul style="list-style-type: none"> ... BSTZ_STATE ... BSTZ_VALUE ⊕ CATH ⊕ CTS ⊕ DISP1 ⊕ DISD3 	<ul style="list-style-type: none"> BSTZ_STATE BSTZ_VALUE 	<ul style="list-style-type: none"> PITZ.UTIL/MEMORY/BST PITZ.UTIL/MEMORY/BST

Standard checks (AND combined) | Additional checks (C++ code)

	DAQ Name	[units]	[1st.. / mask	..last]	>=	<=
1	I_LOCK_GIS1_DI_0_TO_7	bit	xxxx	xxx0	n/a	n/a

Small hints...

The screenshot shows a software interface with a menu bar at the top containing: Time, Import, Leading Property, Machine Status, Filter, Plots + Histos, and Output. Below the menu bar is a table with three columns: Property Selector, DAQ Name, and F/D/L/P. The table lists several properties, some expanded with a plus sign icon.

Property Selector	DAQ Name	F/D/L/P
BOOST		
... BSTZ_STATE	BSTZ_STATE	PITZ.UTIL/MEMORY/BSTZ/INTSTS
... BSTZ_VALUE	BSTZ_VALUE	PITZ.UTIL/MEMORY/BSTZ/BSZ
... CATH		
... CTS		
... DISP1		
... nisp1		

Below the table is a horizontal scrollbar. Underneath is another menu bar with: Y(t) Plots, Y(x) Plots, 1-dim Histos, 2-dim Histos, and C++ coded Plots + Histos. Below this menu bar are several buttons: 180 : 1, Plot(), Plot2(), Histo(), Histo2(), CheckBits(), and CheckValue(). At the bottom is a code editor with the following code:

```
if ( CheckBits(l_LOCK_GIS1_DI_0_TO_7, "xxxx xxx0") )  
    Plot(l_LOCK_GIS1_DI_0_TO_7, "11xx 1x11", l_LOCK_GIS1_DI_0_TO_7 .GetBitTitles(), "canvas/pad");
```