Vacuum controls and interlocks

P. Strubin
CERN, Geneva, Switzerland

Abstract
The vacuum control system is, in most cases, a subset of the general control system of an accelerator. As such, it shares the architecture and communication infrastructure of the main control system. Considered as a 'slow process' to control in the frame of accelerators, the vacuum control system can be built using commercial industrial controllers (PLCs). A data driven approach allows for changes in configuration without changing the software code but at the expense of a solid database. Modelling the equipment allows for easy adaptation of a variety of control units with the same functionality but different physical interfaces. It also allows for a uniform display of the available data and status values. Interlocks are required to protect the vacuum equipment itself against abnormal conditions, but also to protect other systems, like RF, which need a good vacuum to operate. They are an integral part of any vacuum control system.

1 Introduction

The vacuum control system is, in most cases, a subset of the general control system of an accelerator. As such, it shares the architecture and the communication infrastructure of the main control system, but often requires specific solutions at the process level to accommodate a large variety of equipment (pumps, gauges, valves, interlocks, bakeout controllers, etc.). In an environment like CERN, running many accelerators (Fig. 1) over many years, an additional challenge is to follow the evolution of the control techniques while minimizing the modifications to the hardware.

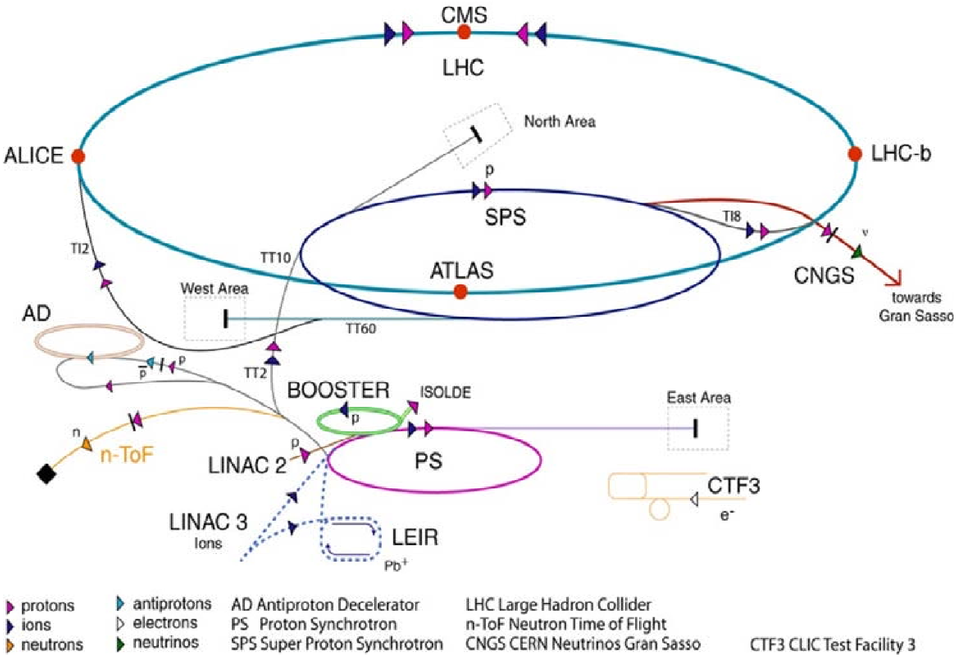


Fig. 1: The CERN accelerator chain

The format of presentation of the acquired data varies, depending on the final end-user and must allow for easy operation by machine operators as well as detailed diagnostic and fault hunting for the vacuum specialist.

Interlock strategy and implementation is also an important part of the vacuum control system to ensure the safe operation of a vacuum system.

Finally, unlike the control systems for most accelerator sub-systems, the vacuum control system must be up and running at all times, including shutdown periods.

2 Physical data to process and control

There is a large variety of equipment to control in a typical vacuum system, ranging from pumping and measuring devices (mechanical or static) to valves (separation or gate valves, roughing valves, venting valves, etc.). The equipment is most often defined by its functionality (e.g., a pumping station) and consists of an assembly of a number of individual components or sub-assemblies, including a control unit or power supply. From a vacuum user’s point of view, it is enough to know the global status of the equipment during normal operation (e.g., off, pumping, etc.). Figure 2 gives an example of the possible sequence of the states of a pumping station [1]. But it may also be necessary to obtain more detailed information about the status of one component of an assembly for troubleshooting. The designer of the controls equipment must have access to every available signal and the control system, in particular the software, must allow for both cases.

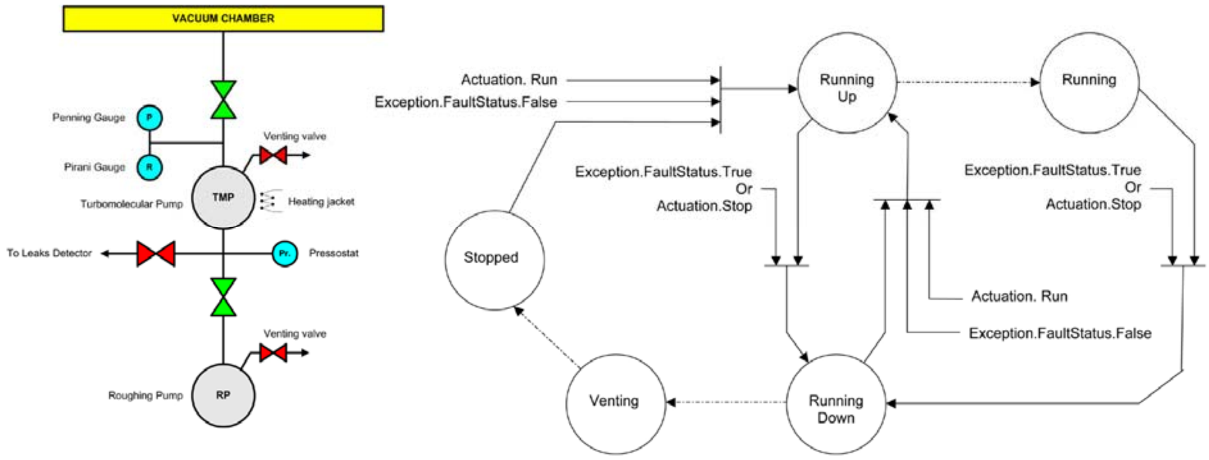


Fig. 2: Possible states of a pumping station

What is illustrated here is that the state of this particular equipment depends on different stimuli: an external action like a command (here referred to as ‘actuation’), the status of subcomponents or faults (here referred to as ‘exception’), but also external conditions and time. The latter is the case of the transition from the ‘Running Up’ state to the ‘Running’ state, illustrated by a turbo-molecular pump reaching its nominal speed.

On the measurement front, the main datum to be acquired is obviously pressure and, for more demanding applications, the composition of the residual gas. In many cases, the pressure is evaluated by the measurement of another signal (ion current, temperature, etc.), via a control unit. In this case also, the vacuum user must receive a meaningful end value, whereas the controls specialist must be able to read all signals. Table 1 shows an example of the required variables.

Table 1: Example of the required variables

Variables	Type	Values
Actuation	Discrete	Run, Stop
Status	Discrete	Stopped, RunningUp, Running, RunningDown, Venting
Exception	Structure	
FaultStatus	Boolean	False, True
FaultList	Set	TMOverheating, RPOverheating, ValveStuck, etc.
WarningStatus	Boolean	False, True
WarningList	Set	LowTMSpeed, etc.

3 Architecture

3.1 Definitions

The most common architecture in modern control systems is the so-called ‘three tiers’ architecture. The main feature of this architecture is a layered approach, allowing for a more generic approach for common services, like the communication layers or the application program interface. Figure 3 shows the way CERN implements this architecture for cryogenics controls [2], and Fig. 4 shows an example taken from DESY [3].

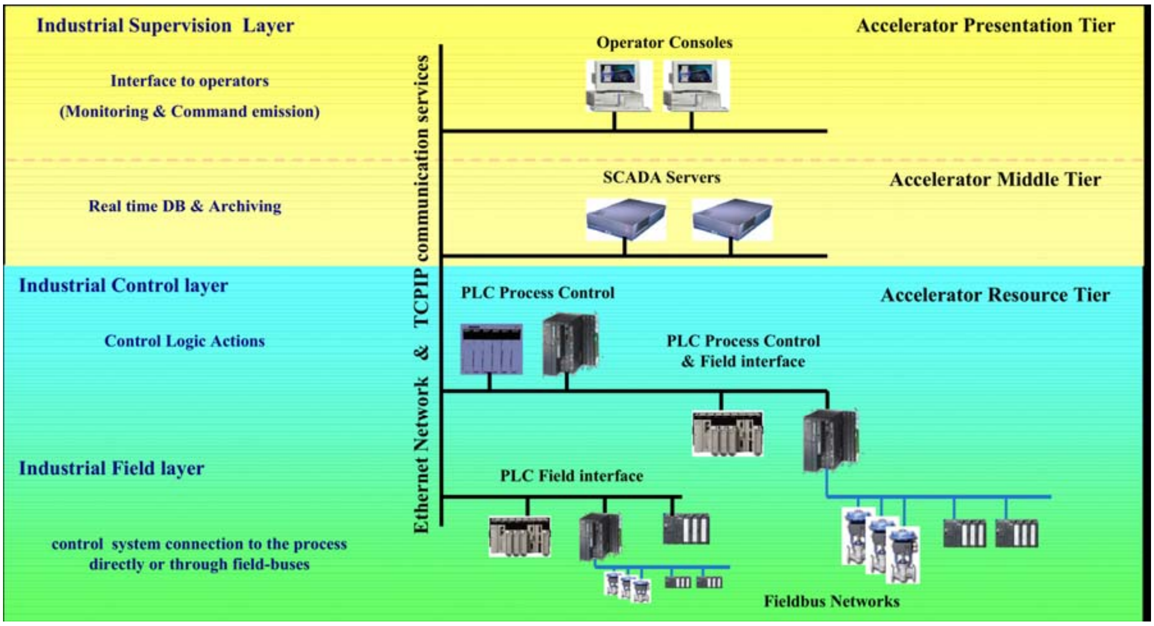


Fig. 3: Cryogenics controls architecture

The first example emphasizes the hardware layers, whereas the second one is more software oriented. It is not always evident to map the various software layers to physical ones. In general, however, it is more important to have clearly defined software layers and interfaces. The actual implementation of these layers matters less and, in any case, should not be the concern of someone developing vacuum controls (or any other process control, like cryogenics) who will concentrate on the lowest layer, close to the equipment.

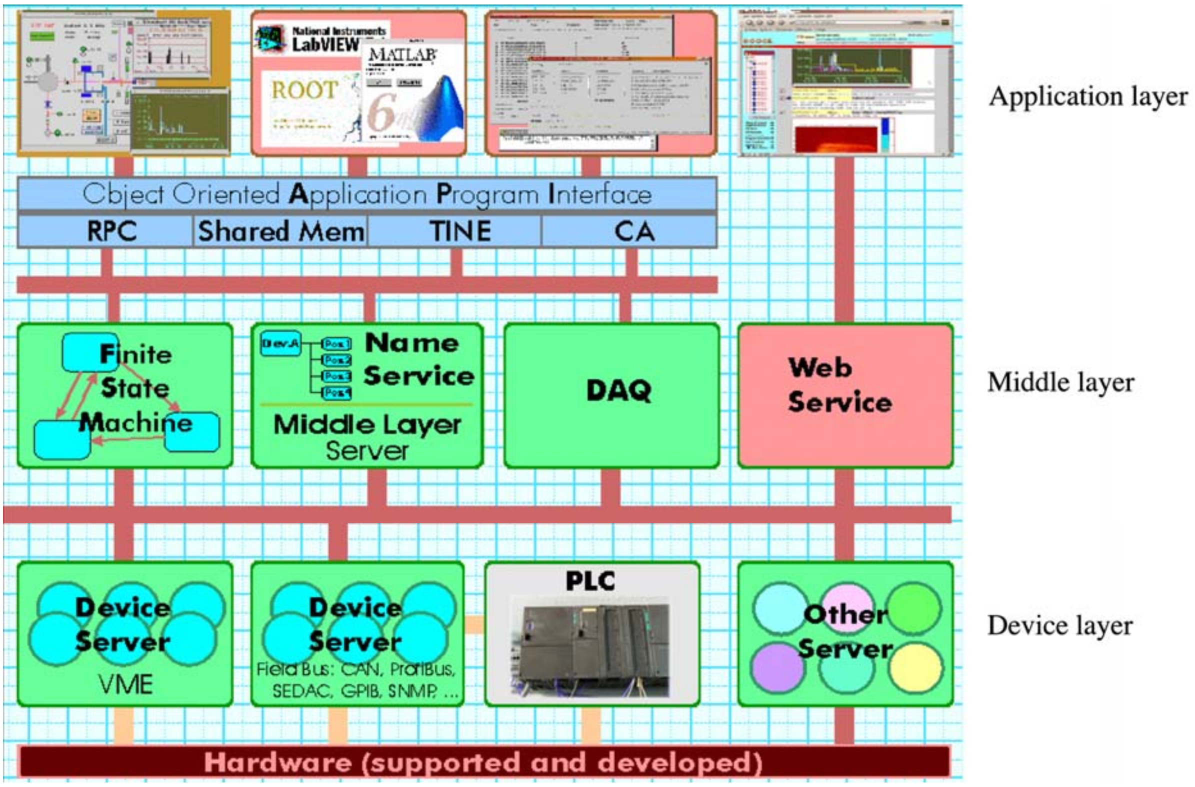


Fig. 4: Example taken from DESY

3.2 Standards

The driving force in modern control systems is the use of industrial standards (e.g. TCP/IP for communication protocols or Web-based servers for graphical interfaces) and industrial equipment (e.g., programmable logic controllers or PLCs for process control and data acquisition). The advantage is of course less expense, multi-vendor solutions as compared to custom-made development. The maintainability can also be easily outsourced. These solutions may not, however, always cover the needs, in particular for fast controls (e.g. beam diagnostic and beam steering), but they are suitable for vacuum controls. On the negative side, it is important to realize that using widely used and documented communication standards makes the control system more vulnerable to hackers. It already happened at CERN that the PLC programs controlling the cryogenic plant used to test the super-conducting magnets for the LHC were blocked by an outside attack. Similarly, it was found that malicious persons were remotely using the computing resources of a digital oscilloscope (essentially an industrial PC). In both cases, the equipment was connected to the main CERN backbone Ethernet network, using TCP/IP communication protocols. It therefore becomes very important to protect the networks. Using proprietary protocols like Profibus™ at the equipment level also helps.

3.3 Models

Another common feature of modern control systems is to develop an ‘object-oriented’ or ‘model-based’ approach, where the actual hardware is hidden behind a virtual representation which describes the data provided or requested by the equipment as well as the dynamic behaviour of the equipment, independently from the hardware. There exist many different implementations of the same concept, often motivated by the necessity to stay compatible with previous implementations of the control system and allow for a smooth transition. The layered approach allows for this and also takes into account that the lifetime of the lower layer (typically at least 10 to 15 years) by far exceeds the lifetime of the upper layer (often less than 5 years).

3.4 Architecture for LHC vacuum controls

CERN vacuum controls have been modernized many times during the 50 years of existence of the laboratory. Since 2000 and following the general trend, there is a strong push to get away from custom-made solutions towards industrial controllers (PLCs). The first vacuum control system to be converted to PLC was for the SPS, which then became the prototype for the future LHC [4]. The architecture follows the standards described above. Figure 5 shows the upper two layers of the 'three tiers' approach. The main part of the middle layer is the central data server (here a PVSS server). It allows for a common data representation to all users, from the vacuum specialist to the database and logging programs. This server communicates with so-called 'master PLCs' over Ethernet.

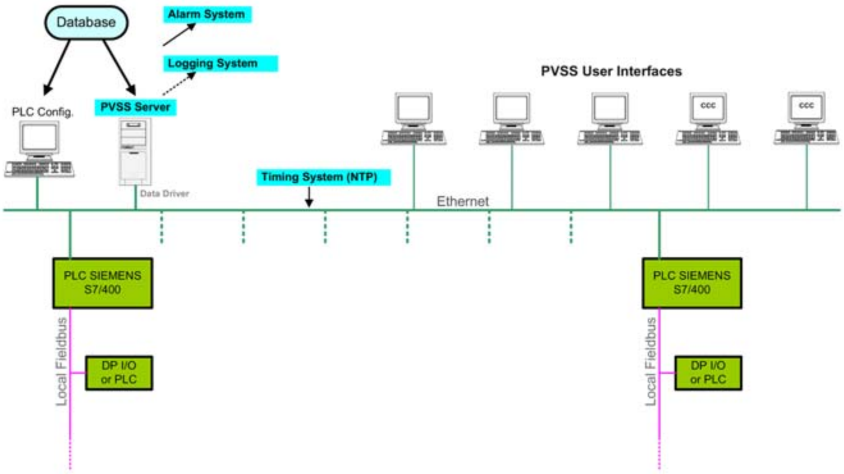


Fig. 5: Upper layers of the LHC vacuum control system

Figure 6 shows the way the lower layer, or equipment layer, has been implemented. The S7 PLC family from Siemens has been chosen in this case, together with Profibus™ and the Siemens proprietary communication protocol MPI. Not only is there a large diversity of equipment to connect, but some equipment with identical functionality, like Pirani or cold cathode gauges, comes with different hardware configurations. This is required in the LHC to minimize costs by using active gauges (sensitive to radiation) where the radiation level is low and passive gauges with a remote controller elsewhere. Slave PLCs are used to connect to the equipment which require a local control process (like a pumping station).

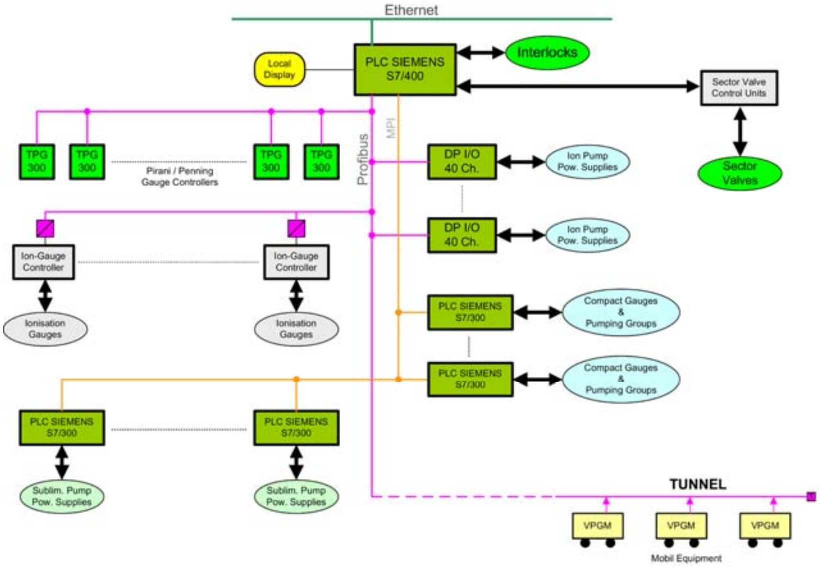


Fig. 6: Lower or equipment layer of the LHC vacuum system

Other equipment, in general self-protected, is connected to the master PLC using deported input–output modules. Finally, some equipment controllers come with a compatible interface and can be connected to the master using Profibus™.

One specificity of large modern accelerators is the use of mobile pumping and diagnostic means to reduce the cost. This requires that the control system be dynamically configured when mobile equipment is connected to it.

4 Describing the equipment

In order to minimize the variants of the applications driving or reading the vacuum equipment, it is necessary to have a uniform description of the properties of the equipment, independent of the actual hardware, as suggested in the ‘object-oriented’ approach. This technique was successfully implemented in LEP, where most of the equipment was described in generic control models. These models describe the data type, the data flow and the dynamic behaviour resulting from either actuation or an internal or external fault or protection. The models can be translated into ‘classes’ in the controls environment, allowing for inheritance of attributes and methods. The attributes describe the data flowing in or out of the model (commands, set values, raw and processed measurements, etc.). For example, an ion gauge is a sub-class of a generic class ‘vacuum gauge’ (a device which has at least the attribute ‘pressure’) [5]. Figure 7 shows the example of an ion gauge. The important feature of this approach is that the object to read or to control is the gauge, but this happens via the power supply or control unit. This is transparent to the user.

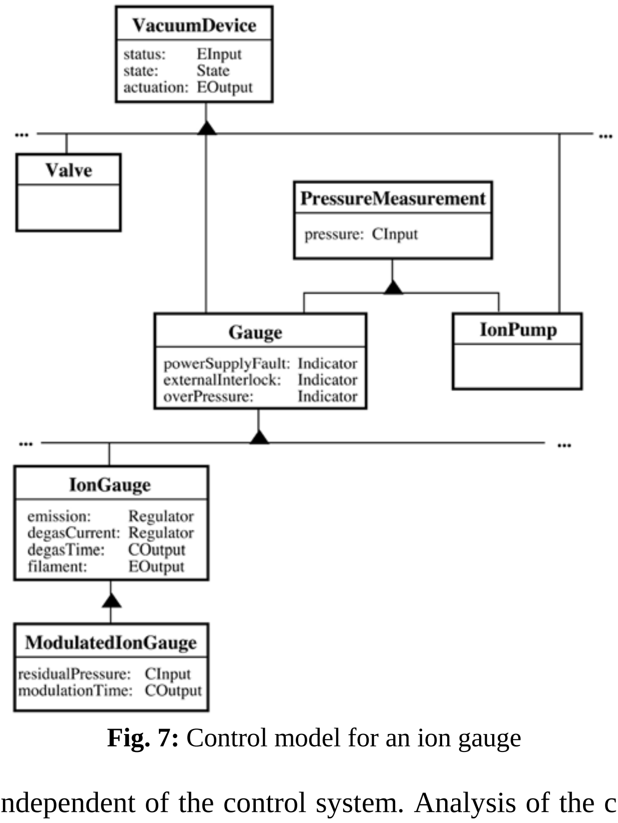


Fig. 7: Control model for an ion gauge

This approach is independent of the control system. Analysis of the control models reveals the common functionality and capabilities between vacuum devices. This leads to the definition of a class hierarchy in which the common features are placed at the top. Specialization of these generic classes provides the interface to more specific vacuum equipment, like gauges. Common functionality, like pressure measurement, can be added as another abstract class, from which ‘gauges’ would inherit. Finally, further specialization will lead to the classes representing the real equipment.

5 Mapping the model to the PLC

Once the attributes and the behaviour of an equipment have been defined in an abstract way, one needs to map these models to the real hardware. In the case of PLCs, the attributes are mapped to so-called 'data blocks' which contain the data at the level of the memory of the PLC. The way this has been implemented for the future LHC (and is already in use in the SPS and LEIR) is completely data driven, that is, the mapping between defined values and the location to access them is described in a database and can be automatically downloaded to the target PLC. The behaviour of the equipment, or how the equipment reacts to commands or faults, is controlled by small program loops. Figure 8 describes the global logical structure used. The structure is organized by the direction of flow of the data. Commands (actuations or changes of internal parameters, like the sensitivity of a gauge) are grouped in one type of data block ('Write Data Block'), acquired or computed values, as well as internal settings and timestamps are grouped in a second type of data block ('Read Data Block'). The 'Function Block' controls the behaviour of the equipment and how the physical values of interest (e.g. pressure) are computed. The differences between different physical devices of the same type (e.g., a cold cathode gauges) are described in a type of additional data block ('DEV_DB').

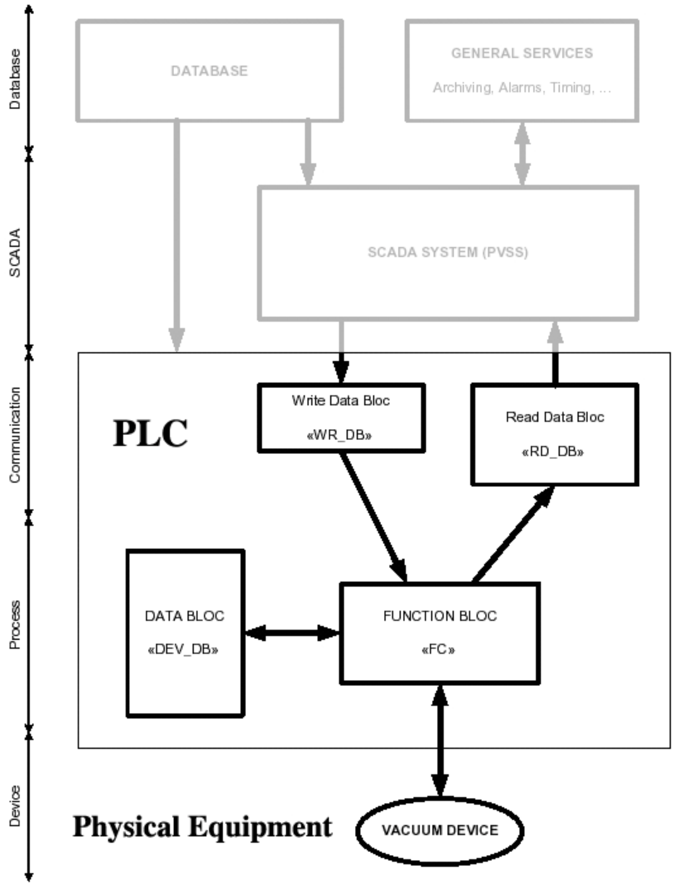
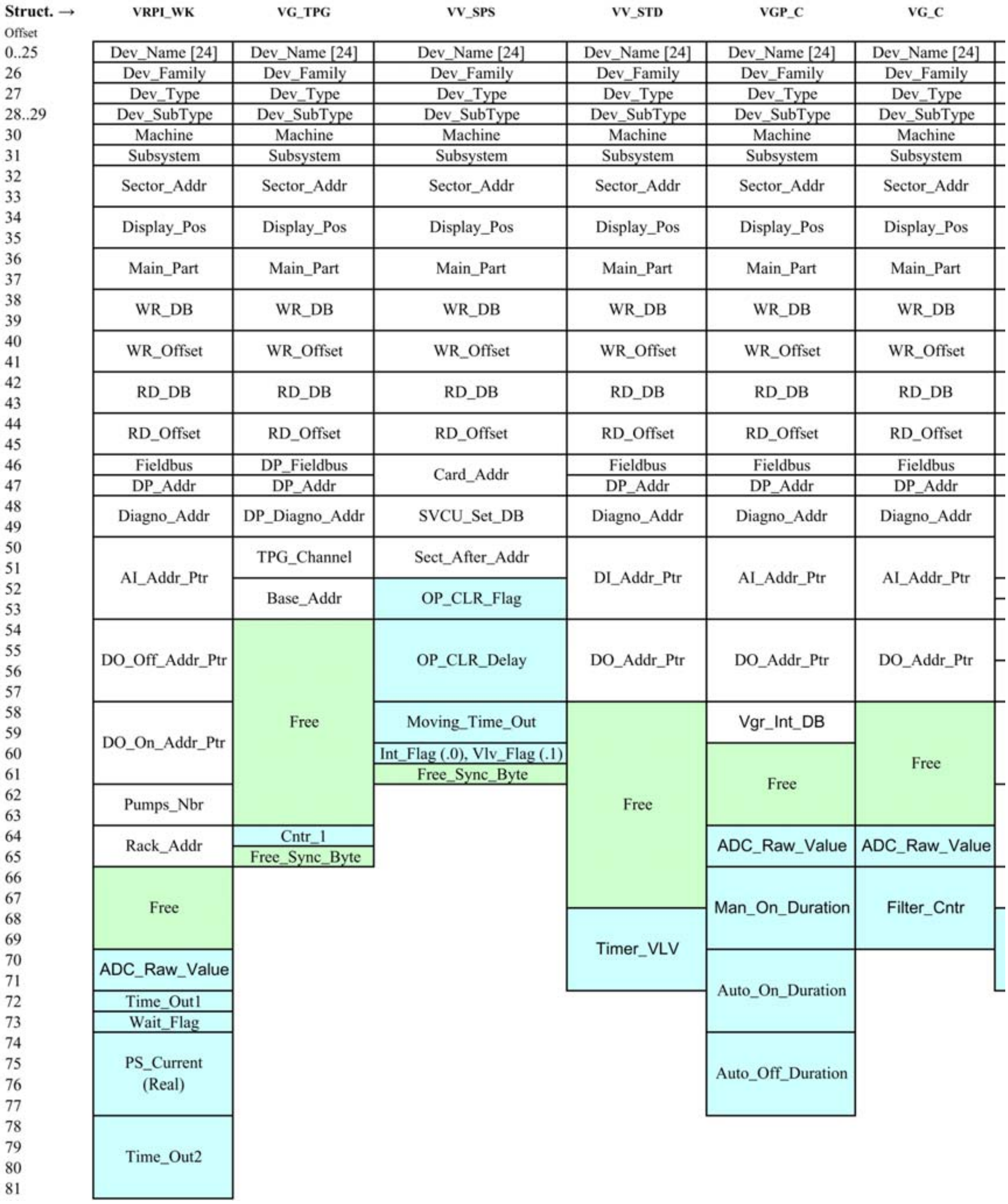


Fig. 8: Logical structure of the vacuum controls at CERN

The above description covers 'classes' of equipment. The PLC controller will manage a number of instances of every class. Hence, for each instance of equipment, a set of data blocks has to be defined and created in the memory of the PLC. Therefore additional information is required to define the physical location of the data blocks for every single instance of a piece of equipment. This is also done via the 'DEV_DB' data block. Figure 9 shows the structure and mapping of the attributes the 'DEV_DB' data blocks for some typical devices.



Struct. →	VRPI_WK	VG_TPG	VV_SPS	VV_STD	VGP_C	VG_C
Offset						
0..25	Dev_Name [24]	Dev_Name [24]	Dev_Name [24]	Dev_Name [24]	Dev_Name [24]	Dev_Name [24]
26	Dev_Family	Dev_Family	Dev_Family	Dev_Family	Dev_Family	Dev_Family
27	Dev_Type	Dev_Type	Dev_Type	Dev_Type	Dev_Type	Dev_Type
28..29	Dev_SubType	Dev_SubType	Dev_SubType	Dev_SubType	Dev_SubType	Dev_SubType
30	Machine	Machine	Machine	Machine	Machine	Machine
31	Subsystem	Subsystem	Subsystem	Subsystem	Subsystem	Subsystem
32	Sector_Addr	Sector_Addr	Sector_Addr	Sector_Addr	Sector_Addr	Sector_Addr
33						
34	Display_Pos	Display_Pos	Display_Pos	Display_Pos	Display_Pos	Display_Pos
35						
36	Main_Part	Main_Part	Main_Part	Main_Part	Main_Part	Main_Part
37						
38	WR_DB	WR_DB	WR_DB	WR_DB	WR_DB	WR_DB
39						
40	WR_Offset	WR_Offset	WR_Offset	WR_Offset	WR_Offset	WR_Offset
41						
42	RD_DB	RD_DB	RD_DB	RD_DB	RD_DB	RD_DB
43						
44	RD_Offset	RD_Offset	RD_Offset	RD_Offset	RD_Offset	RD_Offset
45						
46	Fieldbus	DP_Fieldbus	Card_Addr	Fieldbus	Fieldbus	Fieldbus
47	DP_Addr	DP_Addr		DP_Addr	DP_Addr	DP_Addr
48	Diagno_Addr	DP_Diagno_Addr	SVCU_Set_DB	Diagno_Addr	Diagno_Addr	Diagno_Addr
49						
50	AI_Addr_Ptr	TPG_Channel	Sect_After_Addr	DI_Addr_Ptr	AI_Addr_Ptr	AI_Addr_Ptr
51						
52		Base_Addr	OP_CLR_Flag			
53						
54	DO_Off_Addr_Ptr	Free	OP_CLR_Delay	DO_Addr_Ptr	DO_Addr_Ptr	DO_Addr_Ptr
55						
56						
57						
58	DO_On_Addr_Ptr		Moving_Time_Out	Free	Vgr_Int_DB	Free
59						
60			Int_Flag (.0), Vlv_Flag (.1)		Free	
61			Free_Sync_Byte			
62	Pumps_Nbr					
63						
64	Rack_Addr	Cntr_1			ADC_Raw_Value	ADC_Raw_Value
65		Free_Sync_Byte				
66	Free				Man_On_Duration	Filter_Cntr
67						
68				Timer_VLV		
69						
70	ADC_Raw_Value				Auto_On_Duration	
71						
72	Time_Out1					
73	Wait_Flag					
74	PS_Current (Real)				Auto_Off_Duration	
75						
76						
77						
78	Time_Out2					
79						
80						
81						

Fig. 9: Mapping of device attributes to physical memory blocks

6 The SCADA server

6.1 Functionality

The previous section describes one way to implement the lower or equipment layer of the ‘three tiers’ approach. At this point, the representation of the data is still quite close to the physical equipment, calling for an additional layer to make a homogeneous interface to the applications (graphical interface, logging, alarm services, etc.). At the same time, as the processing power of the PLC controllers is limited and best used to control the process, it makes sense to have a single access point which is materialized by one or more servers, sometimes called a SCADA server (Supervisory Control And Data Acquisition). The standard SCADA functionality can be summarized as follows:

- data acquisition;
- data logging and archiving;
- alarm handling;
- access control mechanism;
- human–machine interface including many standard features e.g., alarm display or trending.

As such, the SCADA server can replace the two upper layers of the ‘three tiers’ architecture. In general, however, a SCADA server also includes a flexible Application Program Interface (API) allowing access to all its features from an external application that would typically be on the upper layer of the system.

A specific SCADA product has been chosen by CERN, PVSS from the Austrian company ETM [6]. The system, illustrated in Fig. 10 is built around an ‘Event Manager’ which dispatches the data to and from the possible users. Towards the lower part of the ‘three tiers’ architecture, PVSS connects to the hardware (typically PLCs) via a set of drivers. Towards the upper layer, PVSS provides an Application Programming Interface (API) and a possibility for processing data in the background (this is where the control over a distributed process would happen). PVSS also provides the tools to build so-called ‘User Interface Managers’, overlapping in this case with the ‘Presentation Layer’ of the ‘three tiers’ architecture. Finally, PVSS maintains a database which contains all data from all connected devices, be they physical or virtual (e.g., a software process).

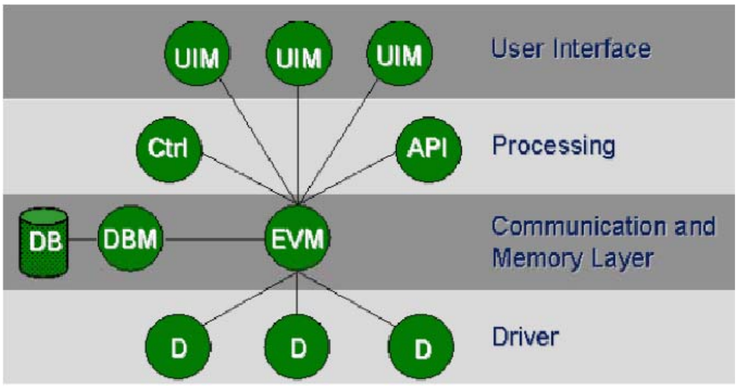


Fig. 10: Architecture of a PVSS system

As for the PLCs, the data to and from the equipment is collected into a data structure, in this case labelled ‘data points’ (roughly the equivalent of the data blocks in the PLC). The description of these data points is done via a ‘data point type’ structure, which is similar to a ‘class’ in object-oriented terminology. A data point contains the information related to a particular instance of an equipment defined by a data point type and a ‘data point element’ represent one particular value in a data point.

7 Databases

An important aspect of a modern control system is to avoid hard-coding device-specific features in the control programs as much as possible. This goal has been achieved by using the hardware and software structure described earlier. The data blocks describing the equipment in the PLCs are prepared in a central ORACLE based database. The same database also stores the description of the PVSS data points. Figure 11 shows the global database scheme used. The ‘survey database’ provides static information about the layout (typically the geographical locations of the equipment), the ‘LHC equipment database’ describes how the vacuum equipment is distributed (e.g., to which vacuum sector it belongs). The native export facilities of the database management system are used to move the data from one database to the other, via intermediate text files.

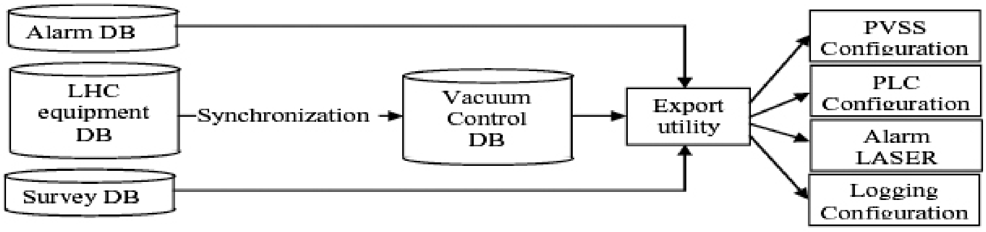


Fig. 11: Configuration database scheme

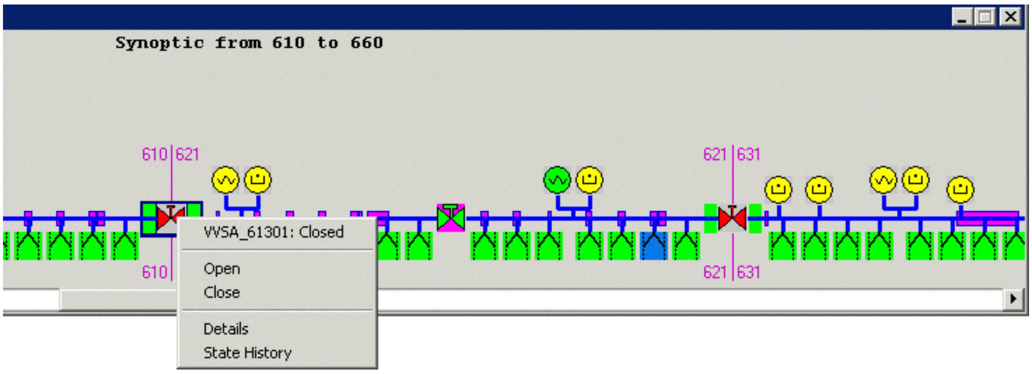
In an environment like CERN, it is desirable that a single database structure be used for all accelerators. This is achieved by having a single set of tables describing all families and types of equipment to control and a set of tables for every accelerator which is closely linked to the layout (includes, for instance, geographical information taken from the surveyors database). In this way it is possible to use the database not only to configure the controls equipment, but also to configure the way data is displayed on the user interface.

8 Data presentation and storage

Which data has to be displayed very much depends on the users. A control room operator is normally happy with a green light telling him or her that all pressures are below a given value and all sector valves are open to allow for the beam to circulate. As soon, however, as these conditions are no longer satisfied, it becomes important to have access to more detailed information.

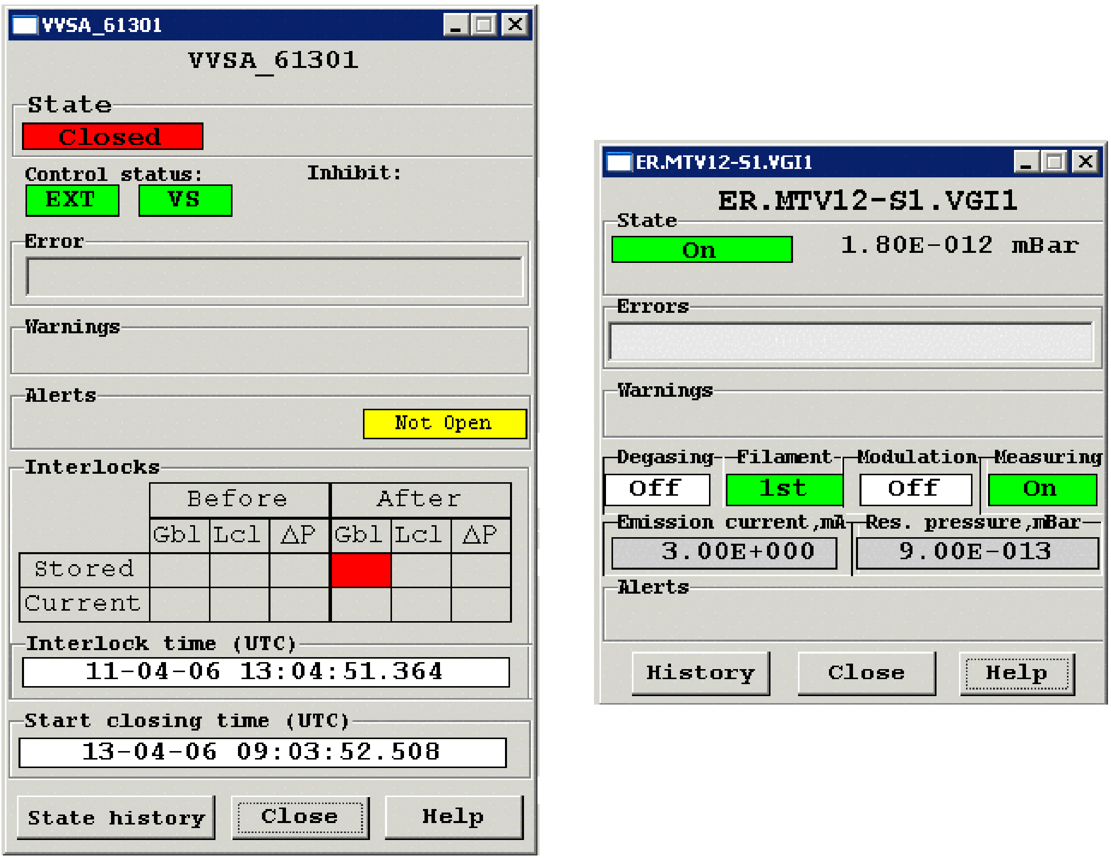
8.1 State of the equipment

Although not strictly necessary, the states of the vacuum equipment are most often displayed on a synoptic showing the sequence of elements and using a colour coding for the state. Although it is recommended to use normalized coding for states, the colour scheme is a recurrent subject of discussion, as many operators are more responsive to, say, a red symbol than to its shape. Figure 12 shows the pumps and the valves of a part of the SPS. Some other accelerator equipment is also sketched, so as to allow for a better localization. A ‘point and click’ approach is chosen to select a piece of equipment and show more details or enable the control panel. Simply pointing with the cursor to the device will return its main data (e.g. pressure for a gauge, state for a valve) and its name. Figure 13 shows the examples of a valve control panel and an ion-gauge control panel. It is from these panels that actions on individual pieces of equipment will be launched.


Synoptic from 610 to 660

VVSA_61301: Closed

Fig. 12: Synoptic view of part of the SPS



VVSA_61301

State: Closed

Control status: EXT VS

Inhibit:

Error

Warnings

Alerts: Not Open

Interlocks

	Before			After		
	Gbl	Lcl	ΔP	Gbl	Lcl	ΔP
Stored				🟥		
Current						

Interlock time (UTC): 11-04-06 13:04:51.364

Start closing time (UTC): 13-04-06 09:03:52.508

ER.MTV12-S1.VGI1

State: On 1.80E-012 mBar

Errors

Warnings

Degasing: Off

Filament: 1st

Modulation: Off

Measuring: On

Emission current,mA: 3.00E+000

Res. pressure,mBar: 9.00E-013

Alerts

Fig. 13: Example of control panels for a valve and an ion-gauge

Grouped actions are very often required in a vacuum system. For example, switch on all pumps in one vacuum sector or close all sector valves. These are normally implemented as a menu item of the synoptic view (for sector wide commands) or as a menu item of the global schematic view.

8.2 Displaying pressures

Displaying the pressure values is most often done in two ways: a profile over a given part of the accelerator (e.g., one vacuum sector) or the evolution over time of one or more pressure readings.

8.3 Pressure profiles

There are many different vacuum devices able to return a pressure value (different type of gauges, and sputter-ion pumps), but their useful ranges do not overlap. This presents a difficulty for the display of the pressure profile, which can be illustrated in Fig. 14. On the right part of the figure, all devices returning a pressure are shown and one can see that the readings from the Pirani gauges, although valid, do not reflect the actual pressure. On the left part of the figure, the Pirani gauges have been removed from the items to display, leading to a more useful representation. At the same time, the scales of the graph have been adjusted. The choice of the active elements as well as the scales should be dynamically chosen, but the rules for the choice may not always be obvious. In the present implementation at CERN, these choices have to be done manually.

Every value is plotted as a variable length bar, with a colour coding to draw the attention of the operator in case of abnormal values. As for the synoptic, pointing and clicking on a bar will allow access to more details about the equipment returning the pressure.

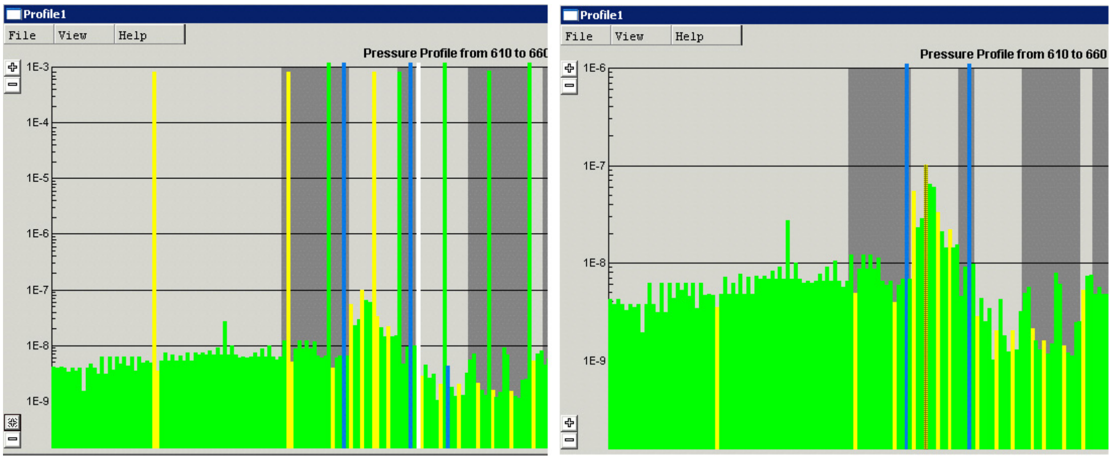
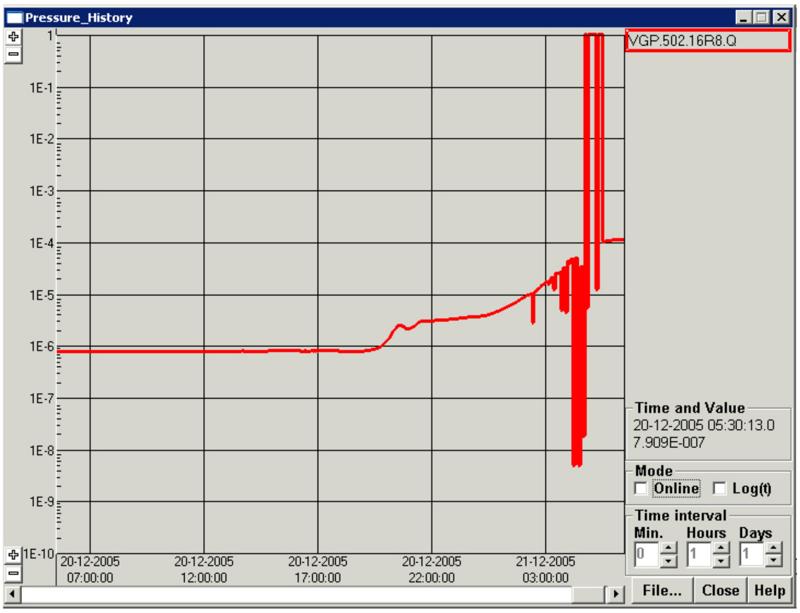


Fig. 14: Pressure profiles over on SPS sextant

8.4 Pressure history

Plotting one or more pressure values against time (histogram) has two different approaches: real-time data added to a graph or archived data over a given time span. The data available in the SCADA server is used for the first case. All data is stored in parallel in a long-term archive database from which it can also be retrieved and displayed. The user selects the equipment to be displayed (usually by selecting it on a pressure profile) and defines the time span of interest. Logarithmic scales are used by default for the pressure scale, linear and logarithmic scales must be available for the timescale, the logarithmic scale being most useful to follow up the pump-down of large volumes.

The histogram is a very powerful diagnostic tool. From the vacuum point of view it allows one to follow trends and give an early warning of a degradation of the vacuum conditions. Figure 15 shows the evolution of the pressure in the insulation vacuum when a part of the LHC helium distribution line was warmed up, showing that gases trapped onto the cold surfaces start to be released at higher temperatures. What is missing here, is the direct correlation between pressure and temperature. This must be done at the upper level of the architecture of the control system and emphasizes the need for a common presentation and access method to the data from all accelerator components.



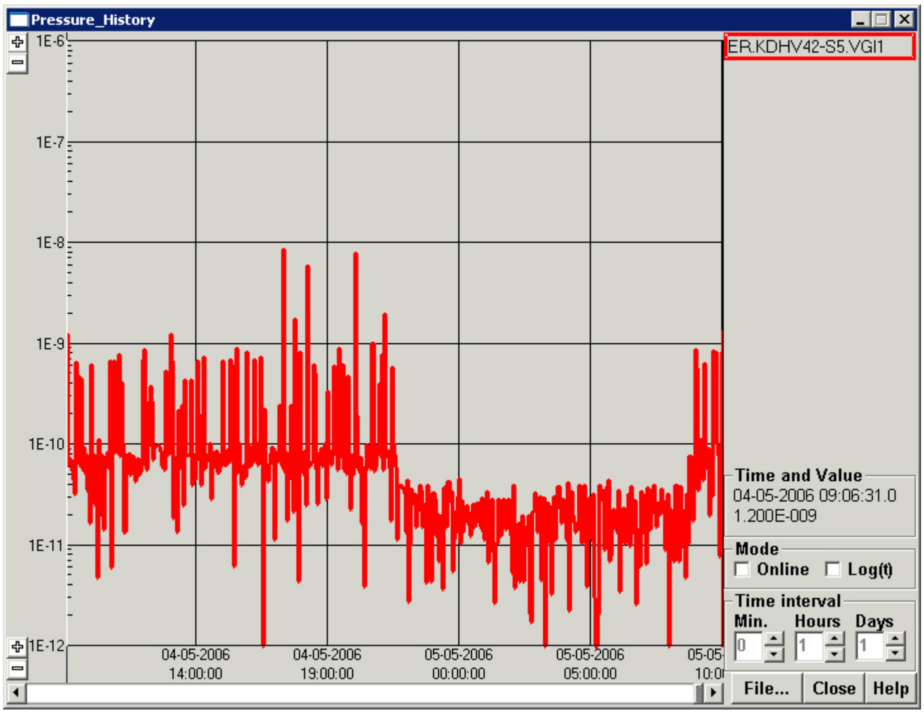
Time and Value
20-12-2005 05:30:13.0
7.909E-007

Mode
☐ Online ☐ Log(t)

Time interval
Min. 0 Hours 1 Days 1

Fig. 15: Histogram of the pressure during the warm-up of the LHC helium distribution line

A histogram can also help in identifying beam-related effects (losses, electrically induced noise, etc.). Figure 16 represents the values returned by an ion-gauge in LEIR while an ion beam was injected into the accelerator. The equipment used is very new and still quite some debugging is required to understand the reason for these very unstable readings.



ER.KDHV42-S5.VGI1

Time and Value
04-05-2006 09:06:31.0
1.200E-009

Mode
☐ Online ☐ Log(t)

Time interval
Min. 0
Hours 1
Days 1

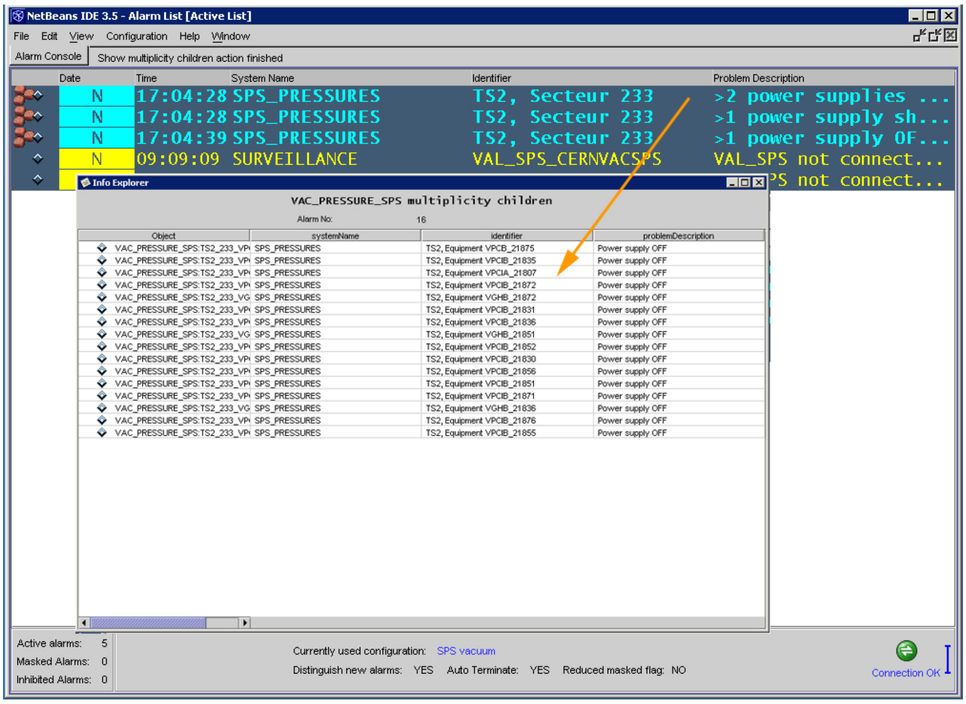
Fig. 16: Histogram of an ion-gauge reading in LEIR

8.5 Alarms

A typical vacuum system includes much controlled equipment, and hence may generate a considerable number of warnings and alarms. These range from pressure limits being reached to faulty power supplies or inappropriate valve states. The SCADA server is in charge of generating the alarms, usually based on a set of rules defined in a database. The SCADA server itself can display and log alarms, but they are most often also sent to a central alarm server which collates the alarms from all systems and presents them in an uniform way to the operators in the control room.

Reduction algorithms are used to minimize the number of messages an operator must cope with. They group similar alarms by vacuum sectors or sometime larger machine areas, like an octant. Figure 17 shows an example of an alarm screen, where some alarms are indeed reduced. It is possible though for the operator to display the details of the alarm sources.

Very often, the origin of the fault comes from an external event, like a power failure or a communication network problem. In this case, the reduction algorithm should make a global analysis on all systems and hierarchize the display of alarms. For instance, it does not make much sense to display a list of faulty power supplies if there is no power on the grid in the racks where the supplies are installed. So far, at least at CERN, the central alarm server does not allow for such overall analysis.



Show multiplicity children action finished

Date	Time	System Name	Identifier	Problem Description
N	17:04:28	SPS_PRESSURES	TS2, Secteur 233	>2 power supplies ...
N	17:04:28	SPS_PRESSURES	TS2, Secteur 233	>1 power supply sh...
N	17:04:39	SPS_PRESSURES	TS2, Secteur 233	>1 power supply OF...
N	09:09:09	SURVEILLANCE	VAL_SPS_CERNVACSPS	VAL_SPS not connect...
				PS not connect...

VAC_PRESSURE_SPS multiplicity children

Alarm No: 16

Object	systemName	identifier	problemDescription
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCB_21875	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21835	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIA_21807	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21872	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VG	SPS_PRESSURES	TS2, Equipment VGHB_21872	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21831	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21836	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VG	SPS_PRESSURES	TS2, Equipment VGHB_21851	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21852	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21830	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21856	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21851	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21871	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VG	SPS_PRESSURES	TS2, Equipment VGHB_21836	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21876	Power supply OFF
VAC_PRESSURE_SPS:TS2_233_VP	SPS_PRESSURES	TS2, Equipment VPCIB_21855	Power supply OFF

Active alarms: 5
Masked Alarms: 0
Inhibited Alarms: 0

Currently used configuration: SPS vacuum
Distinguish new alarms: YES Auto Terminate: YES Reduced masked flag: NO

Connection OK

Fig. 17: Example of an alarm screen and details of a reduced alarm

9 Special cases

9.1 Residual gas analysers

If the most common types of vacuum equipment are connected to the main control system using the techniques described above, it sometimes also happens that a solution provided by a supplier has to be integrated. A typical example is the operation of sophisticated residual gas analysers (RGAs) which come from the manufacturer with a suite of programs to operate them. Figure 18 shows how it has been done at DESY [7], where an interface with an Ethernet connection has been used to make the RGA hardware controllable from a remote PC. In a way, it also follows the layered approach, but does not make use of the common vacuum SCADA server. With even more recent RGAs, the control unit has its own Ethernet interface and uses standard communication protocols (with the inherent vulnerability mentioned previously).

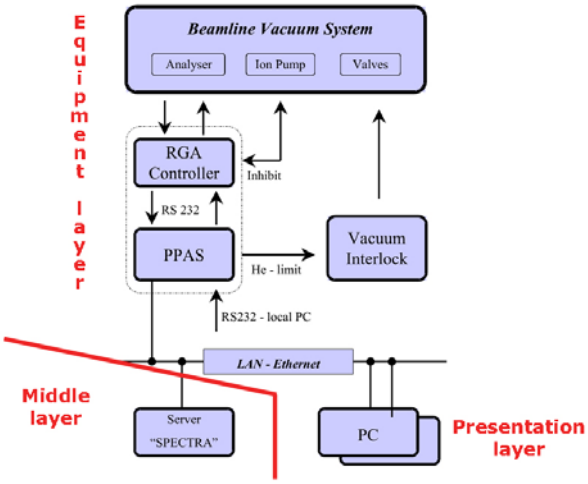


Fig. 18: Example of the remote control of a residual gas analyser

9.2 Control of bakeout process

Traditionally the bakeout process was controlled at CERN using one industrial PID controller per regulated channel. Easily available from industry, these controllers were, however, not very easy to integrate in a global system, at least if cost was a concern. With the development of PLC-based controls, it became attractive to use them for temperature regulation during the bakeout cycle. Among the advantages which were tested on prototypes at CERN, one can mention:

- built-in possibility to connect to the main control system;
- acquisition of external parameters, like power to the heating elements or pressure;
- selective recovery procedure from failures, based on external parameters;
- central alarm management and data logging.

Regulation and protection can be significantly improved using a PLC-based system. The program can automatically detect the time constants of the heated equipment. Measuring the power to the heating elements, allows for detection of faulty (or wrongly connected) thermocouples. Figure 19 shows how power supervision can prevent overheating a vacuum component.

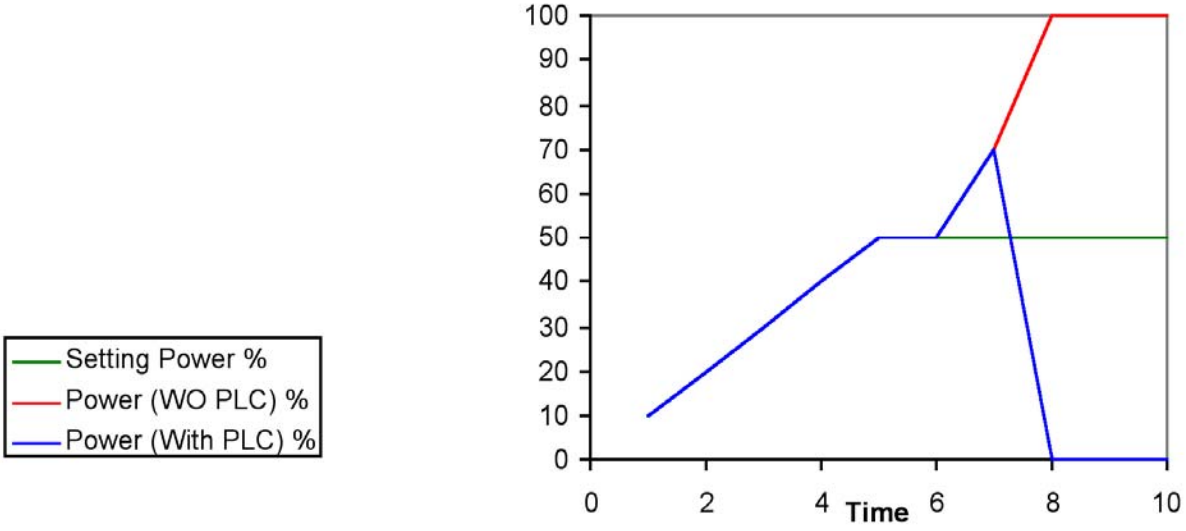


Fig. 19: Using power measurement for protection against overheating

Table 2 shows a set of possible scenarios for dealing with various faults, showing that the available processing power allows one to protect against or to recover from a number of failures using information from other equipment.

Using a PLC-based controller is also economically more attractive. A single PLC can control as many as 24 channels (the limitation is more the number of electrical connections to make, than the processing power) with a single interface to the control network.

Table 2: Failure recovery scenarios for a bakeout process

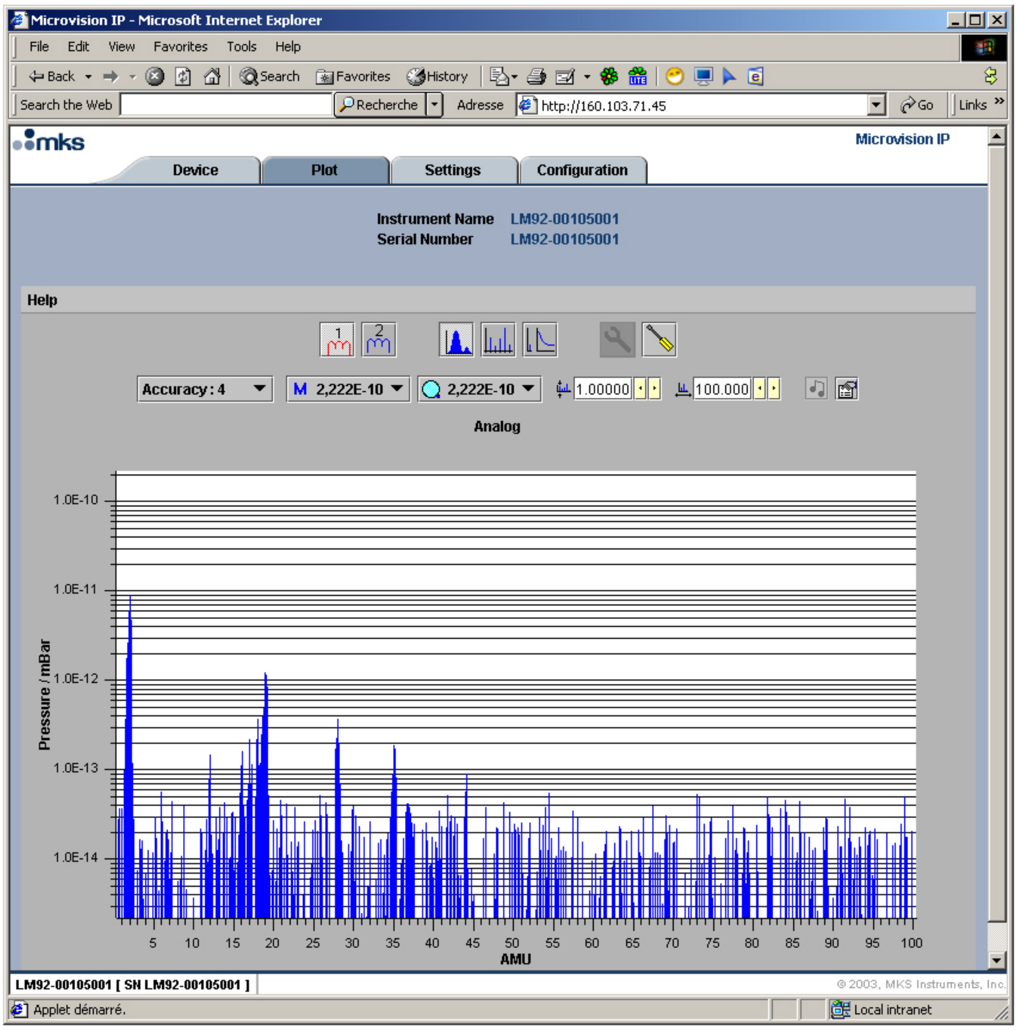
Events/ Actions	Non-NEG	NEG	Standard regulators	PLC BO rack
Short power cut (<5mn)	Y (1)	Y / N	Stop the bakeout	Either stop or continue the bakeout (setting) & generate an alarm
Long power cut (>5mn)	Y (2)	Y (2)	Stop the bakeout	Smoothly top the bakeout (with a decrease temp. slope) & generate an alarm
Too high temp. measurement	Y (1)	Y (1)	Nothing	Stop the channel, continue or smoothly stoop the other channels (with a decrease temp. slope) & generate an alarm
Too low temp. measurement	Y (1)	Y (1)	Nothing (100% of the power will be delivered)	Stop the channel, continue or smoothly stoop the other channels (with a decrease temp. slope) & generate an alarm
High pressure	Y (3)	Y (2)	Stop the bakeout (need hardware interlock)	Smoothly stop the bakeout (with a decrease temp. slope) & generate an alarm
Leak (RGA pick 40 leak detection)	Y (3)	Y (1)	Stop the bakeout (need hardware interlock)	Smoothly stop the bakeout (with a decrease temp. slope) & generate an alarm
Pumping group stopped or VVR closed	Y (3)	Y (3)	Nothing	Smoothly stop the bakeout (with a decrease temp. slope) & generate an alarm

10 Web-server-based applications

An alternative to central user interfaces (at the presentation layer) is to use embedded Web servers. This technique has been tested at ESRF for residual gas analysers and PLCs. Embedded Web servers are essentially interface modules that can access the equipment on one side and accept html commands from Ethernet on the other side. Web pages generated by the embedded server can be called from any browser.

Commercial Web servers come with graphical tools which are fine to develop simple control panels and display simple graphics, but their limits are quickly reached when approaching any scientific display. Also, the Web server is not designed to store any important amount of data, meaning that it does not replace the traditional control path. The main reason to use them is to have access to local processes even when the main control system is not available (typically during some shutdown periods). An alternative reason is to allow for quick prototyping of new user interfaces.

Figure 20 shows a Web page produced to control a residual gas analyser. The Web server module is part of the control unit, itself located in the synchrotron ring and not accessible during operation. Therefore, the Web interface is exclusively used during shutdown periods, for diagnostic while the residual gas analyser is connected to the main control system during operation.


Microvision IP - Microsoft Internet Explorer

Adresse http://160.103.71.45

Instrument Name LM92-00105001
Serial Number LM92-00105001

Accuracy : 4
M 2,222E-10
2,222E-10
1.00000
100.000

LM92-00105001 [SN LM92-00105001]

Applet démarré.

Fig. 20: Web page to control a residual gas analyser

A second example is shown in Fig. 21 with the use of an embedded Web server in a PLC dedicated to the control of valves, shutters, absorbers, and vacuum interlocks in beam lines. Using a Web server on beam lines appears really useful since the set-up changes very frequently, making changes and tests much quicker than from the central control system. In this case, the synoptic is displayed, a point and click approach allows one to open the specific control panel of each piece of equipment.

Other applications have been successfully tested with this approach at ESRF, like upgrades of bakeout controllers or even the controls for the recently installed NEG coating facility.

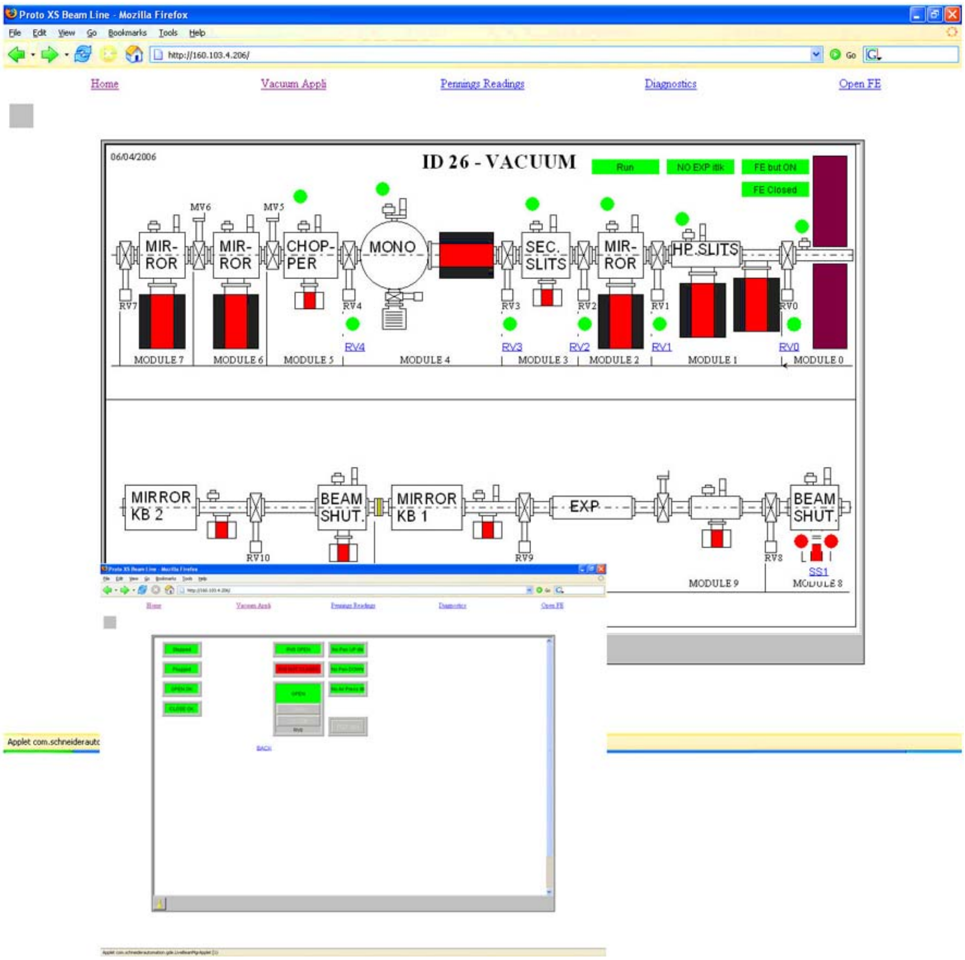


Fig. 21: Web page to control a beam line in ESRF

11 Interlocks

11.1 Protecting vacuum sectors

An accelerator vacuum system is normally divided into sectors to allow independent venting and pumping cycles. The choice of the length and location of the sectors can be driven by different reasons. For a room temperature accelerator, critical equipment like RF cavities or injection and ejection devices, are normally isolated by sector valves. For an accelerator including superconducting elements (like the magnets in the LHC), a good location to place sector valves is at the cold-to-warm transition. Large experimental facilities like in HERA and in the LHC are usually also isolated by valves. In all cases, the aim is to minimize the intervention time and to protect sensitive parts of the accelerator from major vacuum problems. The control of the sector valves needs to be interlocked with the vacuum conditions. The evaluation of the vacuum conditions is most often done using ion-pumps and cold-cathode gauges, as these devices are quite robust. A good practice is to use several instances in the same vacuum sector with a voting rule (e.g., 2 out of 3 alarms) to trigger the closure of the valve, but to require all devices to give a good indication to allow opening the valve. In most cases, the interlock chain for the sector valves is hardwired.

For high intensity machines (like the future LHC), closing a valve on a circulating beam would not only destroy the valve, but very likely severely damage adjacent equipment. An additional interlock is therefore required to trigger the beam abort in case of a vacuum problem, with a proper feedback to the valve controller to allow the valve to close only when the beam is ejected.

11.2 Protecting individual components

Many vacuum devices need to be protected against a too high pressure. As long as the equipment is on, monitoring a relevant parameter of the equipment (in most cases a current proportional to pressure) and switching the equipment off when a limit is reached is enough (self-protection). For more complex instruments, like hot-cathode ionization gauges, more than one parameter has to be monitored, for instance the value of the emission current. The situation is different when a piece of equipment is off. If some are quite robust (e.g., sputter ion pumps or cold cathode gauges) and in general will only suffer a degradation of performance if switched on at a too high pressure, some others can be destroyed (e.g., hot-cathode gauges or titanium sublimation pumps). One therefore has to establish a chain of interlocks based on various measurement devices, where at least one must be able to give a reliable indication at high or atmospheric pressure (e.g., a Pirani gauge).

11.3 Interlocks to other systems

In addition to the protection of the vacuum system proper, interlocks from the vacuum system to protect other systems are often required. As an example, RF cavities or electrostatic septa need a vacuum interlock, as they could suffer from sparking at high pressures. Although still often hardwired, the interlocks to protect the equipment are now also implemented as software signals. Figure 22 shows an example of the various interlocks which have been identified to run the LHC.

Interlocks should also be considered at a more abstract level and modelled, but little effort has been put into this so far in the field of vacuum. Basically, an interlock is just another process with a number of input signals (conditions) that trigger or filter actuations (commands to the equipment). With more and more distributed process control, it will become inevitable that the interlocks are separated from the equipment they protect. For example, during the bakeout and NEG activation of a vacuum sector in the LHC, the bakeout regulation has its own process controller, as do the pumping stations. The permanently installed pumps and gauges also have their controller. Every controller integrates its own protection mechanisms and can have a few external inputs or outputs (e.g., close the roughing valve if there is a fault in the pumping station). But when it comes to protecting the system as a whole, there should be a global monitoring process, which so far relies mostly on the operator.

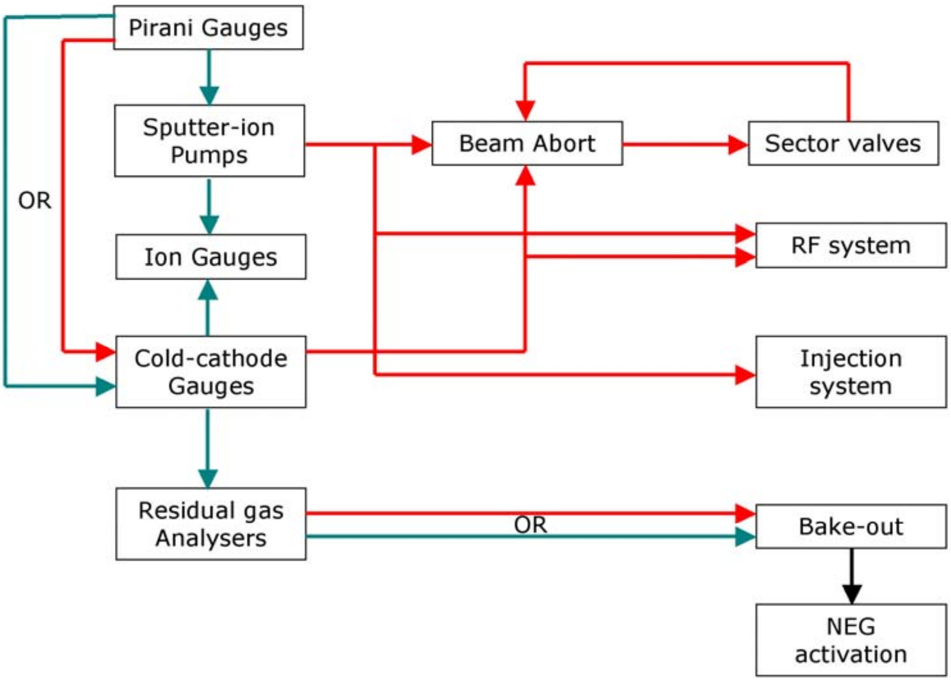


Fig. 22: Schematic of the interlocks of the LHC vacuum system

References

[1] R. Gavaggio, M. Steffensen and P. M. Strubin, Using models to allow for object oriented programming of the vacuum control systems, EPAC94, London, UK.

[2] P. Gayet and C. H. Sicard, Deployment and integration of industrial controls: the case of LHC cryogenics controls, Proceedings of ICALEPCS2003, Gyeongju, Korea.

[3] K. Rehlich, Status of TTF VUV-FEL control system, PCaPAC 2005, Hayama, Japan.

[4] R. Gavaggio and I. Laugier (CERN, Geneva, Switzerland), L. Kopylov, S. Merker and M. Mikheev (IHEP, Protvino, Russia), Development of the vacuum control system for the LHC.

[5] P.M. Strubin and N. Trofimov, Control and operational models for vacuum equipment, PAC97, Vancouver, Canada, Eds. M. Coryn *et al*. (IEEE, Piscataway, NJ, 1998), pp. 2440–42.

[6] http://itcobe.web.cern.ch/itcobe/Services/Pvss/

[7] U. Hahn, M. Hesse, T. Kracht, E. W. Weiner, K. Porges and U. Miße, The residual gas analyzer system for the HASYLAB beamline vacuum system, Nucl. Instrum. Methods Phys. Res. A **467/468** (2001), 805–808.