User interfaces for the SPIRAL2 Machine Protection System

L. Philippe, G. Normand, P.Gillette and the Ganil control group (Ganil / Caen, France)

Overview of the Machine Protection System

Goals

- Controlling the operating range
- Limiting heath increase resulting from the beam power inside the machine devices
- Minimizing radiations (function not adressed in this poster)

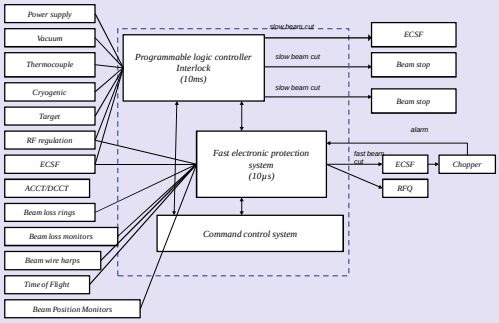
Architecture

Many types of device act upstream the system : failure detections, vacuum losses, beam diagnostics …

MPS system reacts on the beam by decresing the beam duty cycle (ECSF) and by inserting an ad hoc beam stop.

MPS system is composed of

- **PLC interlock** : triggers slow beam cut and controls operating range
- **Fast electronic protection system** triggers fast beam cut and warn PLC interlock
- **Command Control system** : provides HMI, operating alarms, archiving, threshold management

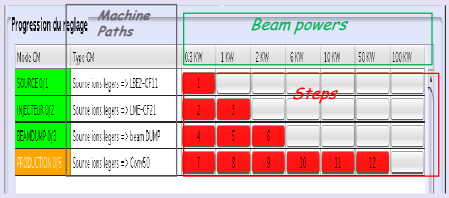


Beam Power Raise

Guides operators for raising progressively the beam power.

Steps are given by a 3D matrix, the 3 axes are :

- **Machine Path** : From the source to an ad hoc beam stop, corresponding to the beam progress level
- **Beam Power Mode** : 300W, 1kW, 2kW, 6kW, …, 200kW
- **Beam Type** : Deuteron, proton, H2+, H3 1+, heavy ion, …
- **PLC Interlock** is in charge of checking the machine status and conditions



Machine Protection System HMI

"MPS" HMI lets operators and accelerator engineers monitor MPS states, alarms and tune some beam losses thresholds.

Thresholds

For each diagnostic used in the MPS context a threshold management subsystem will be implemented either inside the device's electronic or in a separate electronic card.

When the threshold is exceeded an alarm is issued to the fast electronic protection system (Figure 2). The latter should decide to cut off beam .

The functions about threshold management are listed under mentioned :

- Compute threshold offline
- Apply threshold
- Modify some threshold inline
- Verify threshold

Réglages seuil SPM des diagnostics

Diagnostic	Diag 1	Diag 2	Diag 3	…
Seuil BD	5 mA	50 cp/s	30 cp/s	
Seuil effectif	5 mA	60 cp/s	25 cp/s	
Vigilance				

Code vigilance : Rouge si seuil demandé > seuil BD, Vert si seuil demandé < seuil BD

Réglages des plages de surveillances Courant/tension/phase

Courants	LME Q24	LME Q25	Tensions	LME REG1	Phase	LME REG1
Plage BD	0.5%	1%	Plage BD	1%	Plage BD	1%
Plage effective	0.8%	1%	Plage effective	1%	Plage effective	0.5%
Vigilance			Vigilance		Vigilance	

Code vigilance : Rouge si plage demandée > plage BD, Vert si plage demandée < plage BD

Mise en surveillance partielle: LBE/LME

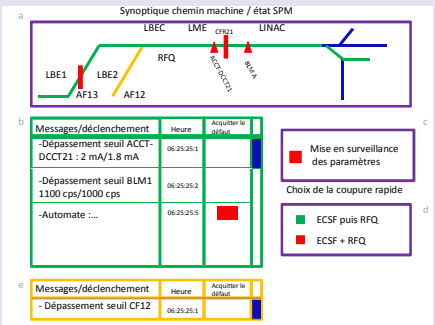
Rythme de hachage lors du déclenchement SPM

Rythme limite	Rythme BD	Rythme effectif
1/100	1/10000	1/5000

Code couleur : Rouge si le seuil effectif < rythme BD, Vert si rythme effectif > rythme BD

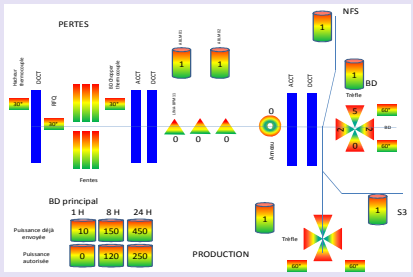
MPS State

- A synoptic that displays the main components (beam stops, diagnostics) related to the MPS context
- A table that summarizes MPS alarms.

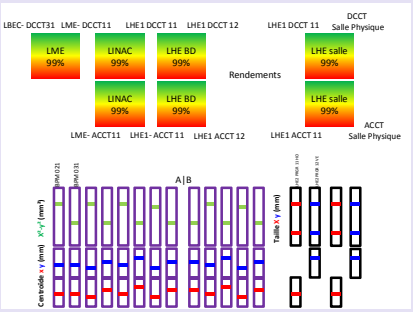


Beam Power Losses

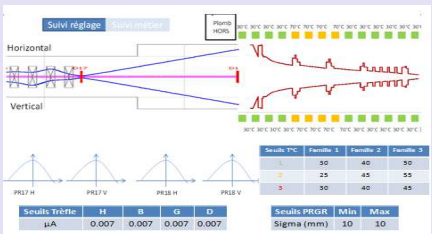
A synotic that displays beam losses



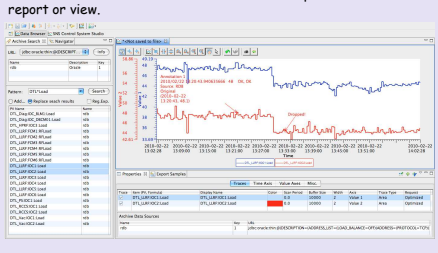
A visualization of beam alignment and beam efficiency



A special interest is accorded to the main beam dump



A stream archiving and retrieval data module. Assume by the global CSS archiver and databrowser mechanism with potential in-house report or view.

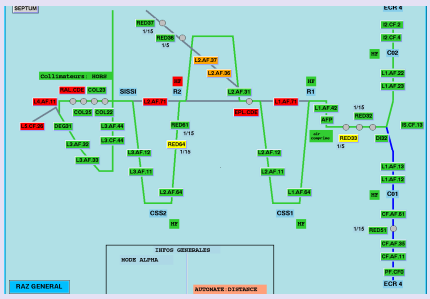


Interlock

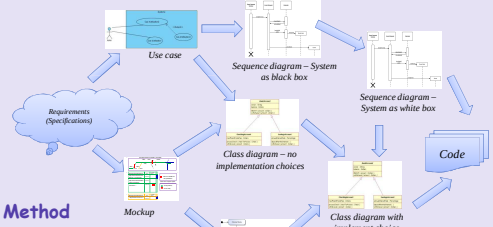
The "Interlock" HMI is a synoptic displaying the state of all beam stops and act on them. Four states are defined for a beam stop :

- IN
- OUT
- Indeterminate
- Radiological protection lock : inhibit the command by the control system

The figure bellow shows an example based on the GANIL control command's HMI interlock. For Spiral2, CSS Boy is envisioned for this application and is still to be developed.



Method and tools



Method

- Use UML Diagrams (principally 7 of them)
- Help to formalize requirements
- Help to formalize documentation

Steps

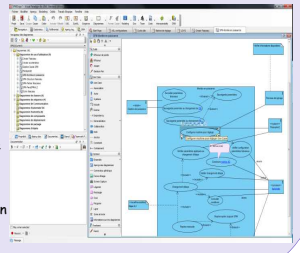
- Identify use case
- Make a mockup
- Specify navigation
- Specify interaction between actors and system (view as a black box)
- Specify interaction between component of the system (white box)
- Determined object and interaction between object
- Without implementation choice
- Add implementation choice
- Coding

UML

- Standardized Modeling language in the field of software engineering
- 14 types of diagram
- 7 for modeling structure
- 7 for modeling behavior

VP UML

- Case tools support UML, SYSML, ER diagram (database modeling), ….
- Roundtrip with JAVA
- Roundtrip with database
- Powerful navigation between diagrams
- Generate documentation



Conclusion and next steps

- Many people involved
- Current step : Consolidating requirements
- Next step : Determine with system should implements which function and catch links and sequences between subsystems
- Still a lot of work to do before implementing

See also

POSTER OF C. BERTHE ET AL, "MACHINE PROTECTION SYSTEM FOR THE SPIRAL2 FACILITY", THIS CONFERENCE.