

LHC Beam Operation Workshop – Evian 7-9 December 2010

Software and control issues

Delphine Jacquet

On behalf of the OP/LHC team

Thanks to OP/LHC and OP/SPS members, Greg Kruk, Vito Baggiolini, Ralph Steinhagen, Etienne Carlier, Mike Lamont for their feedback

a year of debugging...

- During 2010 run, the software in all domains has evolved and improved considerably:
 - **Reactivity**: issues and problems found and fixed as soon as possible
 - **Flexibility**: new requirements implemented to follow the fast evolution of the LHC.
- ⇒ We end up with an impressive amount of well working applications in the control room to run the machine.
- ⇒ We also end up with a list of things we would like to see improved.

Outline

- Equipment control
- Injection
- Sequencer, state machine
- LSA
- Alarms, diamon
- Other things to be improved
- Conclusion

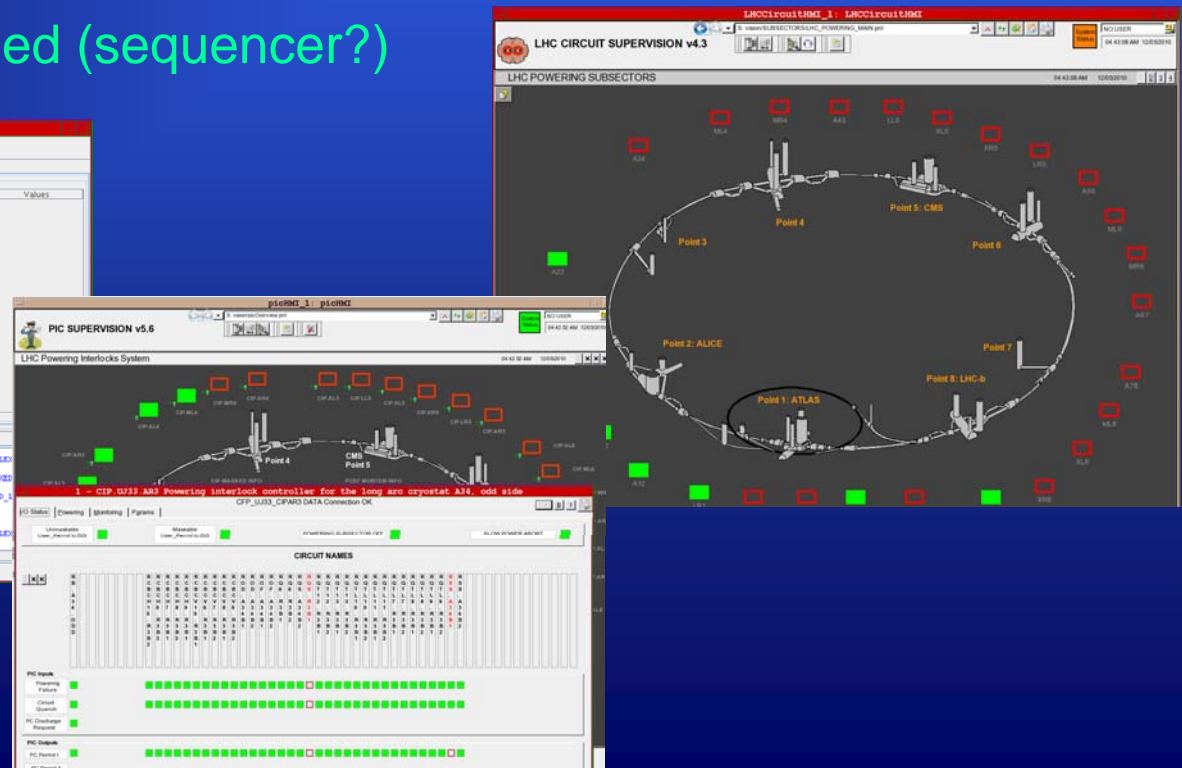
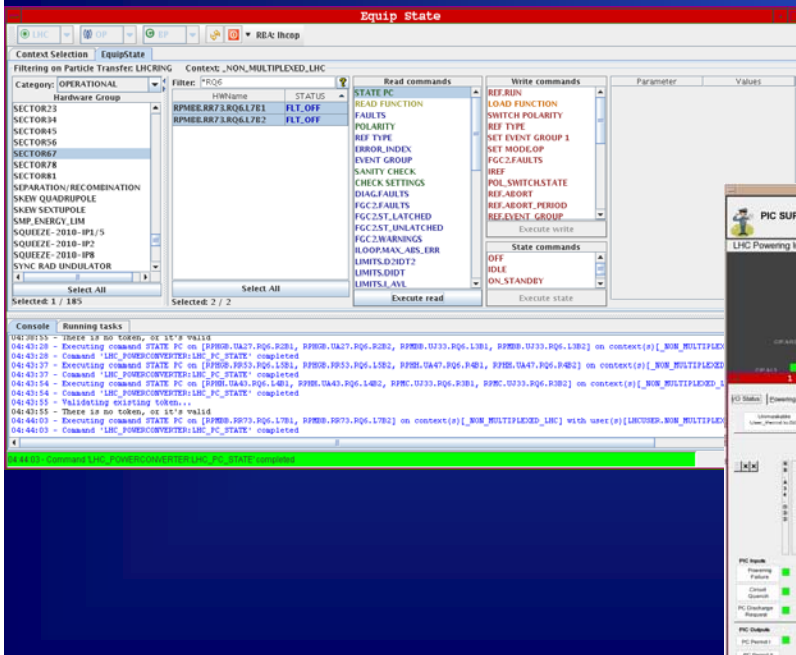
Equipment control

- TCDQ software problems: we need to regain trust in it
 - TCDQ statuses not handled properly by the FESA class: stayed armed, reporting idle, and then moved unexpectedly.
 - ⇒ Etienne's team is working on a new software version of the PLC
 - ⇒ Separated sequences will be created for the TCDQ
- RF
 - ⇒ RF interlocks: we need a detailed status of the RF interlocks
 - ⇒ RF signals for diagnostics: most of them not easily accessible in CCC but only in SR4

Equipment control

- Power converter, PIC, QPS :
 - Not efficient to restart a few power converters that tripped (we only have global restart sequences by sector):
 - 3 applications : Equip state, PIC and Circuit synoptic
 - Jump from one to the other to reset the circuit (entering twice our logging and password in the process.)

⇒ Need to be simplified (sequencer?)



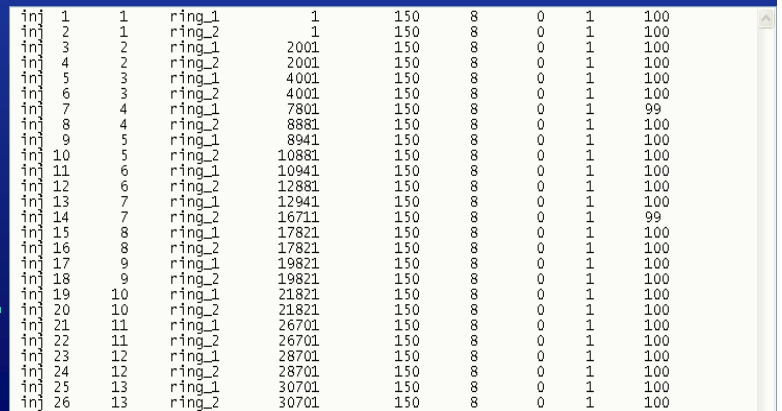
Injection

1-Scheme editor, bunch pattern

During the 2010 run: almost 100 filling schemes

⇒ Existing software to create them need to be improved

- bunch patterns: lack of flexibility
 - created in LSA DB with a SQL script (run locally by Delphine)
 - ⇒ Has to be done from an application that everyone can use
 - We need an extra attribute for the batch spacing (e.g. 150ns schemes)
 - ⇒ Update LSA database and software
- Injection schemes creation: could be more efficient
 - Requests can be created one by one with the application
 - Takes time, easy to make mistakes
 - A PERL script enters automatically all the requests from a file.
 - has to be run locally by Delphine.
 - ⇒ In the application should be able to create an entire scheme from the file.



inj	1	1	ring_1	1	150	8	0	1	100
inj	2	1	ring_2	1	150	8	0	1	100
inj	3	2	ring_1	2001	150	8	0	1	100
inj	4	2	ring_2	2001	150	8	0	1	100
inj	5	3	ring_1	4001	150	8	0	1	100
inj	6	3	ring_2	4001	150	8	0	1	100
inj	7	4	ring_1	7801	150	8	0	1	99
inj	8	4	ring_2	8881	150	8	0	1	100
inj	9	5	ring_1	8941	150	8	0	1	100
inj	10	5	ring_2	10881	150	8	0	1	100
inj	11	6	ring_1	10941	150	8	0	1	100
inj	12	6	ring_2	12881	150	8	0	1	100
inj	13	7	ring_1	12941	150	8	0	1	100
inj	14	7	ring_2	16711	150	8	0	1	99
inj	15	8	ring_1	17821	150	8	0	1	100
inj	16	8	ring_2	17821	150	8	0	1	100
inj	17	9	ring_1	19821	150	8	0	1	100
inj	18	9	ring_2	19821	150	8	0	1	100
inj	19	10	ring_1	21821	150	8	0	1	100
inj	20	10	ring_2	21821	150	8	0	1	100
inj	21	11	ring_1	26701	150	8	0	1	100
inj	22	11	ring_2	26701	150	8	0	1	100
inj	23	12	ring_1	28701	150	8	0	1	100
inj	24	12	ring_2	28701	150	8	0	1	100
inj	25	13	ring_1	30701	150	8	0	1	100
inj	26	13	ring_2	30701	150	8	0	1	100

Injection











2- Injection sequencer- IQC

- to inject B2, the IQC analysis of beam 1 has to be finished.
 - ⇒ Modify injection sequencer to inject B1 and B2 in parallel (a lot of time to gain)
 - Circulating bunch configuration: we must get it right
 - Distributed to the experiments and equipments via DIP and FESA
 - Used to prevent over-injection
 - updated by the injection sequencer according to the IQC response
 - Decision of IQC relies on 2 BCTs in injection lines and the kicker pulse
 - In case of measurement problem, IQC answer not correct
 - Database has to be updated by hand to sort out the situation
- ⇒ Check BQM meas versus database before injection to avoid over-injection
- ⇒ Transfer line BCT not reliable enough for IQC, cross check with ring BCT or BQM
- ⇒ In LHC BQM: new functionality to set the circulating bunch config with the measured bunches to avoid hacking the database

Injection

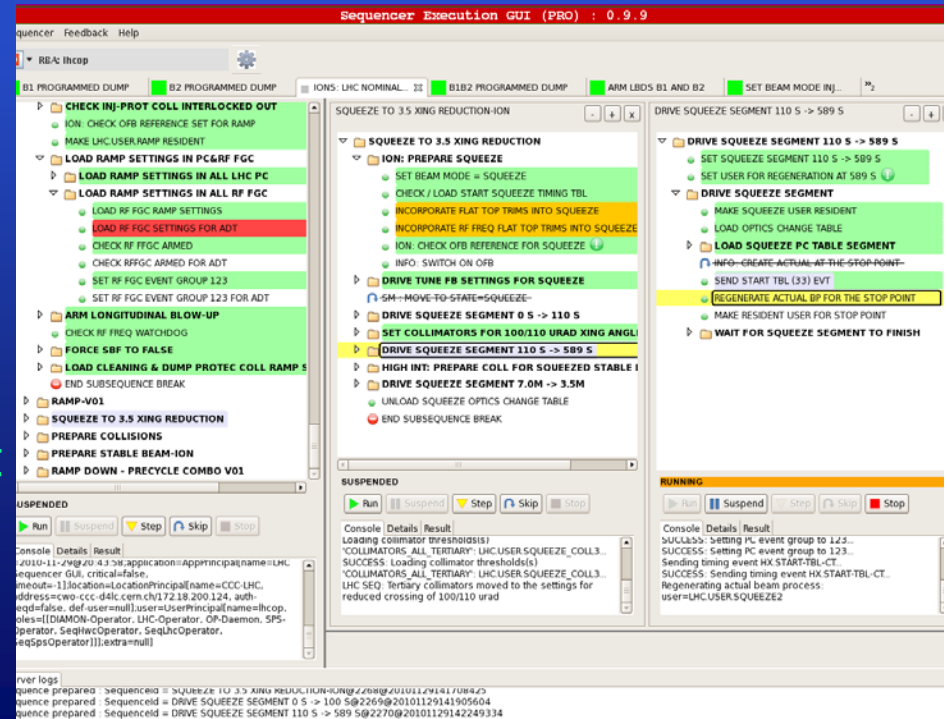
2- Injection sequencer- IQC

- **IQC latches too frequently:** BLM thresholds too tight, data missing etc...
 - Most of the time the recommended action was to unlatch and continue
 - ⇒ **IQC threshold needs to be revisited: a latch should be meaningful**
 - ⇒ **Missing BPM or BLM data problem has improved, but still there.**
- During months, OP was asked for each latch to put a screenshot of the IQC in the logbook (time consuming)
 - ⇒ **IQC playback could be used, or automatic images entries in the logbook should be implemented.**

29	03:07	 20100822030751.png
		latch of IQC on the 8th injection on beam 1
30	03:09	 20100822030957.png
		latch of IQC on the 9th injection on beam 1
31	03:11	 20100822031123.png  20100822031134.png
32	03:11	unequal bunch intensity in the same batch for beam 1, fi
		latch of IQC on the 10th injection on beam 1
33	03:12	 20100822031257.png
		latch of IQC on the 11th injection on beam 1
34	03:14	 20100822031437.png
		latch of IQC on the 11th injection on beam 2 (1 bunch)
35	03:15	 20100822031522.png
		latch of IQC on the 12th injection on beam 1
36	03:15	 20100822031552.png
		latch of IQC on the 12th injection on beam 2
37	03:16	 20100822031642.png
		latch of IQC on the 13th injection on beam 1
38	03:16	 20100822031718.png

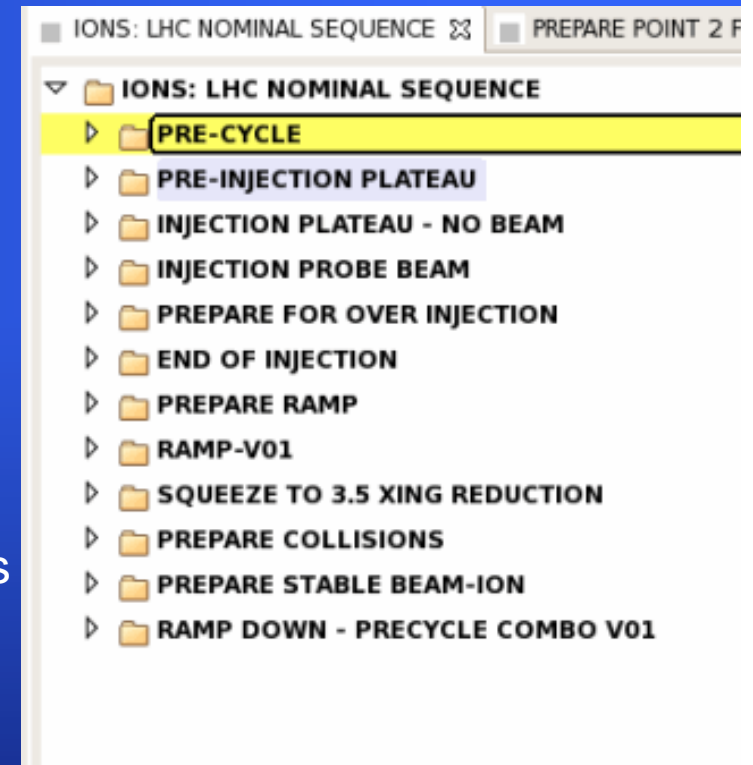
LHC sequencer GUI

- Difficult start with the first GUI version
 - Some bugs (i.e. went directly from ramp to rampdown with beam)
 - Not flexible enough (no parallelism), not intuitive, windows not resizable...
- In September, a new GUI developed by the CO/AP section:
 - parallelism possible
 - clear display of the task result
 - drag and drop subsequences, very easy to use.
 - Still to do
 - ⇒ Improve the GUI for the check list panel
 - ⇒ Functionality to interactively set a parameter (i.e. sequence to restart a power converter).



Nominal sequence

- Has evolved a lot during the run
 - a lot of tasks have been added to replace manual actions.
- Actual subsequences structure is obsolete. A subsequence by equipment would be more efficient (parallelism to be exploited)
- Lots of pop-up windows and unused tasks
- on-error tasks to be replaced by ensure tasks

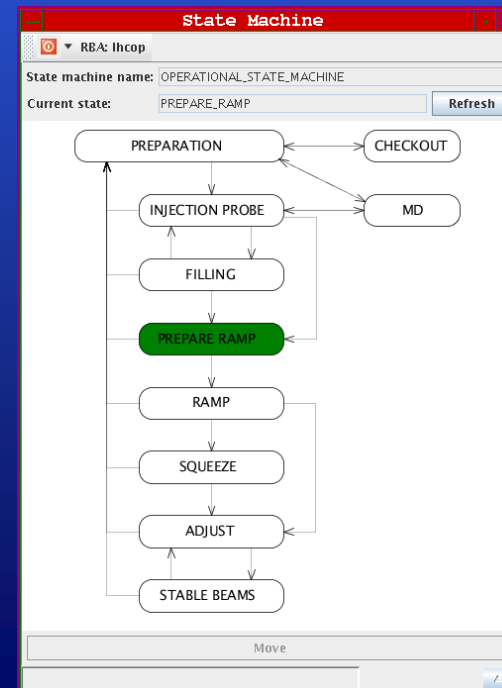
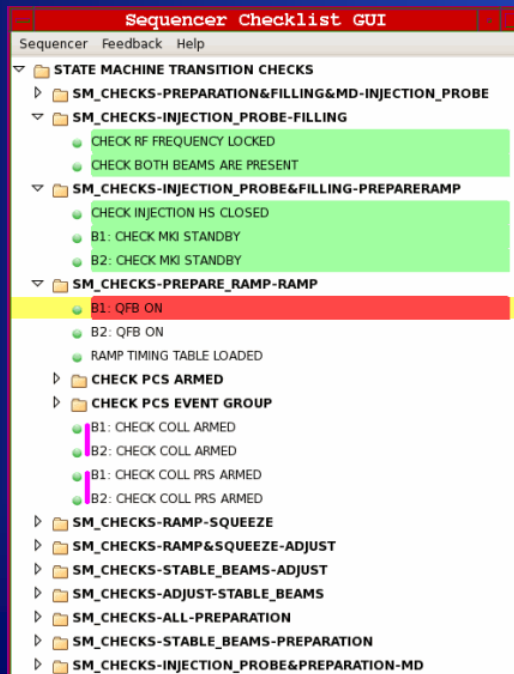


- ⇒ The nominal sequence has to be cleaned and re-organized
We should create a working group of EIC's and operators to end-up with consensus on a safe and efficient sequence.
- ⇒ Need to define how to manage the updates of the nominal sequence during operation. (communicate and keep track of the changes)

State machine

CONCEPT

- Functional states of the accelerator have been defined.
- For each state transition, a list of conditions has to be checked
- State changes driven by sequencer:
 - A task in the nominal sequence asks state machine to change state
 - State machine executes a little sequence with conditions to be checked
 - State change is done if all OK.



State Machine

STATUS

- End of the debug phase
- The check of transition is now operational
- Still some tasks to be added to all transition check lists

⇒ Present: use to safely go from one state to the other by ensuring that the needed equipments are in the right state with the right settings

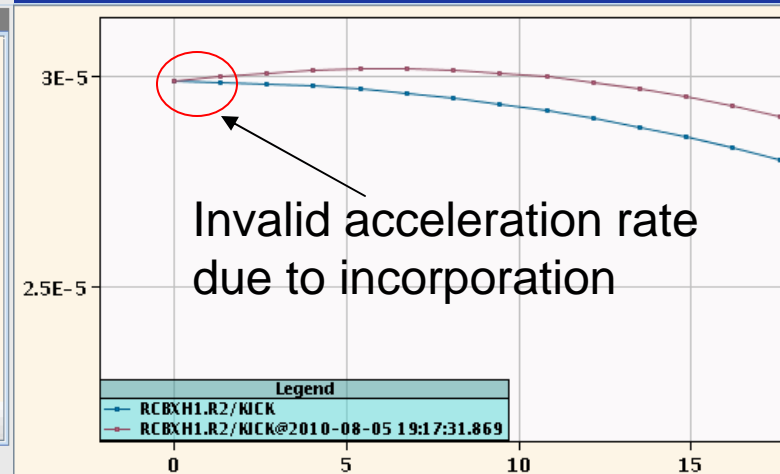
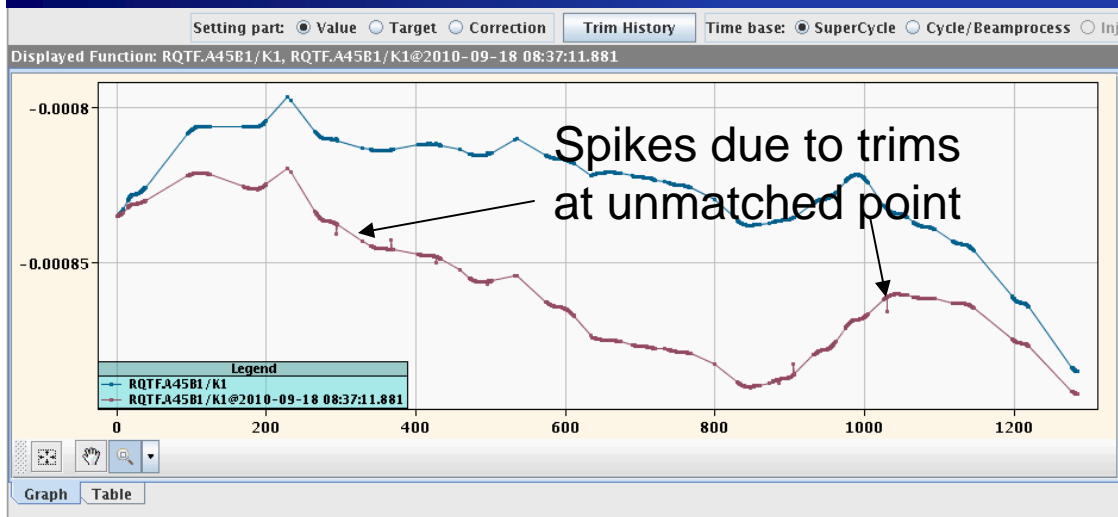
⇒ Future: behavior of certain control software (e.g. LSA, Sequencer) should be constrained by operational state (e.g. state should influence the LSA settings or sequencer tasks that can be used)

INFO: state machine review the 13th of December by Matteo

LSA-settings management

1-recurrent problems with some hw functions

- Constraints on hw functions
 - Multiple source settings and makerules
 - Fast optics changes in squeeze
 - Has to be smooth and continuous along the hypercycle (magnets constraints)
- Complex functions generated that have 2 types of problems
 - Function doesn't pass the FGC internal check (invalid times or di/dt out of limits)
 - Function loaded but provokes a QPS trip (acceleration rate too high)



LSA- settings management

1-recurent problems with some hw functions

- What solution do we have?

⇒ add filters that remove the points that are too closed one to each other (done)

⇒ Add a check of the di/dt in the makerules and incorporation rules: problem detected immediatly, when the function is trimmed or generated.

⇒ Work to be done at the makerules level on the smoothing of the function to avoid spikes (new Ksmooth parameters)

⇒ Estimate the acceleration rate of a generated function and prevent from loading (work started but very tricky)

LSA-settings management

2-other improvements needed

- **Incorporation**

- Not flexible enough : need rules defined for given point, defining a unique rule per parameter/BP type is not enough
- More sophisticated incorporation rules are needed (e.g. include snapback, dynamic B3 corrections at injection...)
- For non-expert, difficult to understand the mechanism: mistakes
- ⇒ **LSA team will review, complete and simplify the incorporation mechanism**

- **Traceability, settings rollback, settings checks**

- ⇒ **Logged history** of all the driven parameters and resident beam processes
- ⇒ **A proper tool to rollback parameters trims** (especially for orbit trims: should be able to rollback all trims from a same correction)
- ⇒ **A tool to easily compare settings**
 - LSA settings between 2 beam processes
 - Settings check against a reference BP (e.g. after an MD, detect wrong incorporation...)

LSA-settings management

2-other improvements needed

- To be considered: A unique ramp-squeeze-collide beam process
 - Possibility to stop at each phase or run through
- Knobs
 - application not user friendly, to be improved
 - Should be possible to delete old knobs and their settings
- LSA database: too easily accessible, should be well protected.
- Proper rounding and limits for all the parameters
- Hypercycle change (e.g. going from protons to ions)
 - ⇒ need a clear procedure and a sequence
- MCS problem : regeneration needs expert signature, complicated work around in place
 - ⇒ A solution has been found and tested, will be deployed for next start-up
 - ⇒ Still impossible to load a segment of function for critical settings, Greg is trying to find a solution with Verena.

ALARMS

- **Work to be done on configuration:** alarm screen permanently with red lines, we don't pay attention anymore
 - ⇒ To be reviewed by equipment experts and OP : what alarms do we really need, is the level correctly chosen...
- Would help a lot if alarms were mode dependent (e.g. what is critical at injection can be ignored during ramp)

Alarm screen when
Everything works perfectly

The screenshot shows the 'Laser Console' interface with two windows. The top window, 'Active List', displays a table of active alarms. The bottom window, 'Last-arrived List', displays a table of recent alarms, with several rows highlighted in red to indicate critical or high-priority issues.

Priority	#	Date	Time	System Name	Identifier	Problem Description
[+]		16/09	20:04:22			
[+]	(41)	29/11	14:23:35			
-	(63)	29/11	14:22:38			
[+]		09/02	10:08:26	ACCE_ZORA_LHC		
[+]	(21)	25/11	09:27:40	FGC power converters		
[+]	(16)	29/11	14:22:38	LHCCollimator		
[+]	(3)	2009	-	LHC_VACOK		
[+]	(16)	12/01	09:11:22	LHC_VACOK_WRN		
[+]		02/11	13:17:22	LSA_BI_CONCENTRATION		
[+]	(2)	29/11	14:06:20	QPS-CIRCUIT		
[+]	(3)	24/11	14:58:48	SURVEILLANCE		
-	(39)	29/11	13:53:13			
[+]	(26)	11/11	13:50:02	FGC power converters		
[+]	(9)	29/11	13:53:13	LHC SIS		
[+]	(4)	29/11	13:44:33	LHCCollimator		

#	Y	Y Date	Y Time	System Name	Identifier	Problem Description
19	10	10:27:41	10:27:41	FGC power converters	S78 - PC permit	[MULT 7] >3 PC Permit not present
19	10	10:27:41	10:27:41	FGC power converters	S45 - PC permit	[MULT 7] >3 PC Permit not present
N		13:43:59	13:43:59	LHC SIS	INJ1	SW PERMIT false
N		13:44:15	13:44:15	LHCCollimator	TCLIB.6R2.B1	At least one axis is out of dump constant limit
N		13:44:15	13:44:15	LHCCollimator	TCLIA.4R2	At least one axis is out of dump constant limit
N		13:44:19	13:44:19	LHC SIS	INJ2	SW PERMIT false
N		13:44:33	13:44:33	LHCCollimator	TCLIA.4L8	At least one axis is out of dump constant limit
N		13:44:33	13:44:33	LHCCollimator	TCLIB.6L8.B2	At least one axis is out of dump constant limit
N		13:53:13	13:53:13	LHC SIS	INJ	SW PERMIT false

Diamon

- When a GUI has a problem, it is often hard to find
 - What is the associated front-end
 - From what software layer the problem comes from.
 - If we can restart the server without dumping the beam
 - ⇒ We need a clear information of the hierarchy between application, middletears, proxy, front-ends...
- In addition, not easy to detect a server that has a problem. (some servers are always red, some others are always green)
 - ⇒ Need to work on the configuration

Diamon when everything works well

Diamon when some servers are down

The image displays two side-by-side screenshots of the Diamon console interface, showing a grid of server status indicators. The left screenshot, titled 'Diamon console 1.9.17 - LHCOF - http://abwww/~pca/diamon_config/bi/config.xml', shows a healthy state where all server components are green. The right screenshot, titled 'Diamon console 1.9.17 - LHCOF - http://abwww/~pca/diamon_config/bi/config.xml', shows a state where several server components are red, indicating they are down or in error. The red components include 'cfv-sx4-bq', 'cfv-ua23-bra', 'cfv-ua27-bra', 'cfv-ua47-bctfra', 'cfv-ua83-bra', 'cfv-ua87-bra', 'cfv-us45-bsra', 'cfv-us45-bsrtb1', 'cfv-us45-bsrtb2', 'cfv-usa15-bra', 'cfv-usc55-bra', 'cfv-sr8-blmc', 'cfv-sr8-blmr', 'cfv-sr8-blmi', 'Host cs-crr-oft', 'Host cs-crr-ofsu', 'Host cs-crr-cmw3', 'Host cfr-us45-bsrt', 'cfv-ua47-bq', and 'cfv-ua47-bqpll'. The interface includes a search bar, a tree view on the left, and a detailed view at the bottom for the selected component.

Other things to be improved

- **Front-ends** : still too many crashes, sometime with a big impact on the LHC efficiency.
- **Orbit and tune Feedbacks** :
 - Most of this year problem are solved
 - Still to do
 - ⇒ Dynamic reference change for orbit and tune feedback : implemented, to be tested and used next start-up
 - ⇒ Better estimate the measurement's quality before the feedback decides to use it.
 - ⇒ Sort out the compatibility with transverse dampers
- **Sequencer editor**:
 - the actual GUI is not user friendly, reflects underlying database tables too much , some bugs with sequence clone.
 - We need
 - ⇒ Tasks copy, cut and paste
 - ⇒ Possibility to create an independent subsequence
 - ⇒ Clear sequences and tasks catalogue
 - ⇒ To keep track of the sequences change (who did change what, when and why) and rollback possibility.

Other things to be improved

- **Fixed display:** we have a lot of fixed displays permanently sitting on the screens : space problem.
 - ⇒ Would be useful to define sets of fixed displays per machine mode that will appear and disappear dynamically with the change of mode.
- **Injection interlocks:** many interlock systems involved: LHC SIS, LHC BIC, SPS SIS, SPS BQM, SPS BIC
 - ⇒ A display with a summary of all the involved system would be helpful
- **Proxy:** improved, but still some problems (i.e. data missing for IQC or XPOC).
- **Software releases**
 - ⇒ ensure they are backward compatible and well tested
 - ⇒ avoid Friday evening releases.
 - ⇒ Communication of the changes very important
- **Documentation:** inexistent (or well hidden) for most of the applications

CONCLUSION

- A long list of requested improvement for different systems has been presented
- **Some of them are really important to**
 - reduce the turn around (e.g. injection)
 - minimize the down time (e.g. lsa settings, TCDQ)
 - improve the efficiency (e.g. sequencer)
 - minimize the risk of error and mistakes (e.g. state machine, settings checks)
- **Others are less important but would help OP to**
 - Diagnose problems before calling experts (e.g. RF interlocks, diamon)
 - Detect a problem as soon as it appears (e.g. alarms)
 - Improve the ergonomics (fixed displays)

CONCLUSION

- Lots of work for developers
- So better be organized!!

