Database involvement in Timing

Ronny Billen

Accelerators and Beams Department
Controls Group
Data Management Section

Timing Review – 29 February 2008

Some timing stuff already covered by DM

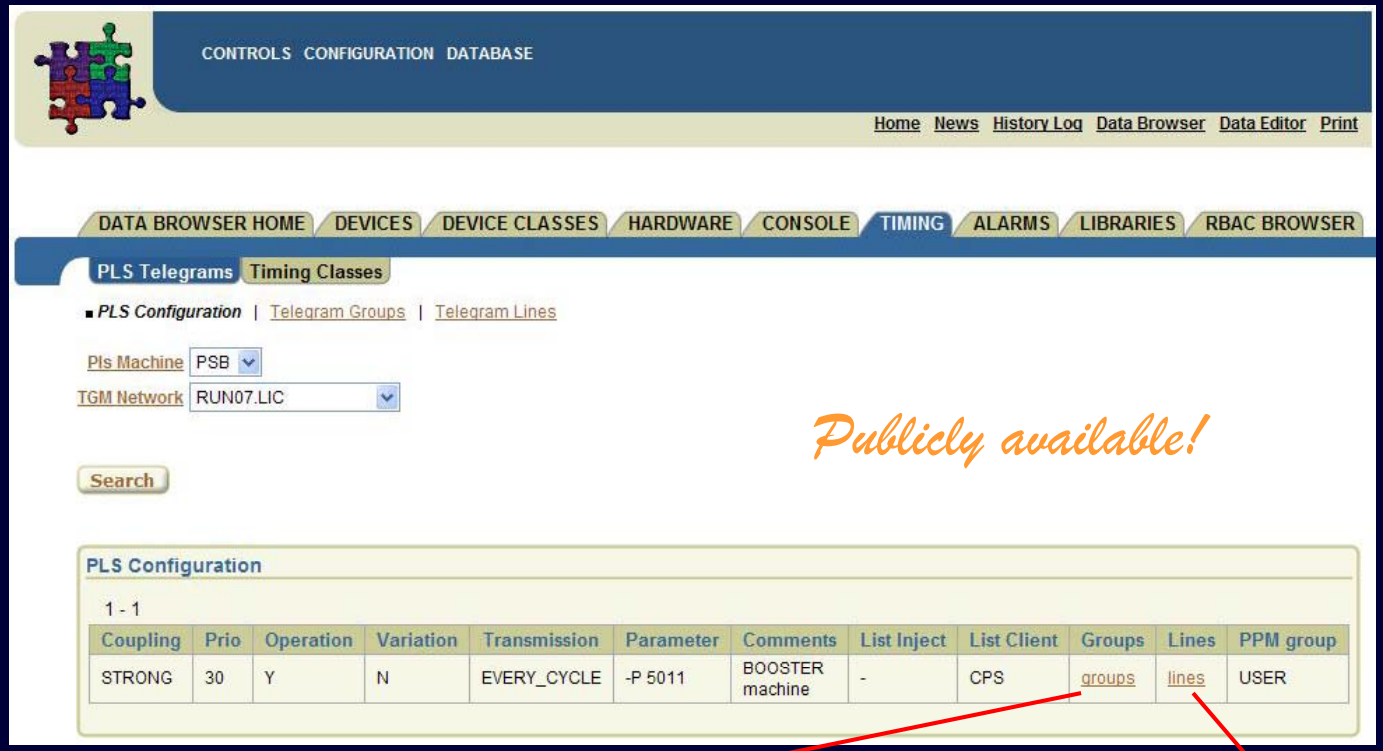
- Controls Configuration data for PLS[?] i.e. the original implemented concepts, currently in the database
- PLS Configurations for {PLS-machine, TGM[?]-network}
  - PLS-machine: CPS, PSB, SPS, LHC,...
  - TGM-networks: LIC[?], CTF, LHC
  - Configuration data: coupling, transmission, injector/client
- PLS Groups for each PLS Configuration
  - PLS Group: PARTY, CYCLE, USER, DEST,...
  - Group meta-data: type, min/max values, size, treatment
- PLS Lines for each PLS Group
  - PLS Line: e.g. NOBEAM, PROTON, PB58 for PARTY
- This is *meta-data*, definitions, for timing telegrams
- PLS data model re-factored in 2007 with AB-CO-HT

Extra information what is being played, historically distributed by the timing system

PLS – Program Line Sequencer
TGM – anagram of MTG
LIC - LHC Injector Chain

R. Billen

PLS Configuration Browser



PLS Configuration

1 - 1

Coupling	Prio	Operation	Variation	Transmission	Parameter	Comments	List Inject	List Client	Groups	Lines	PPM group
STRONG	30	Y	N	EVERY_CYCLE	-P 5011	BOOSTER machine	-	CPS	groups	lines	USER

Telegram Groups

Ppm Group USER

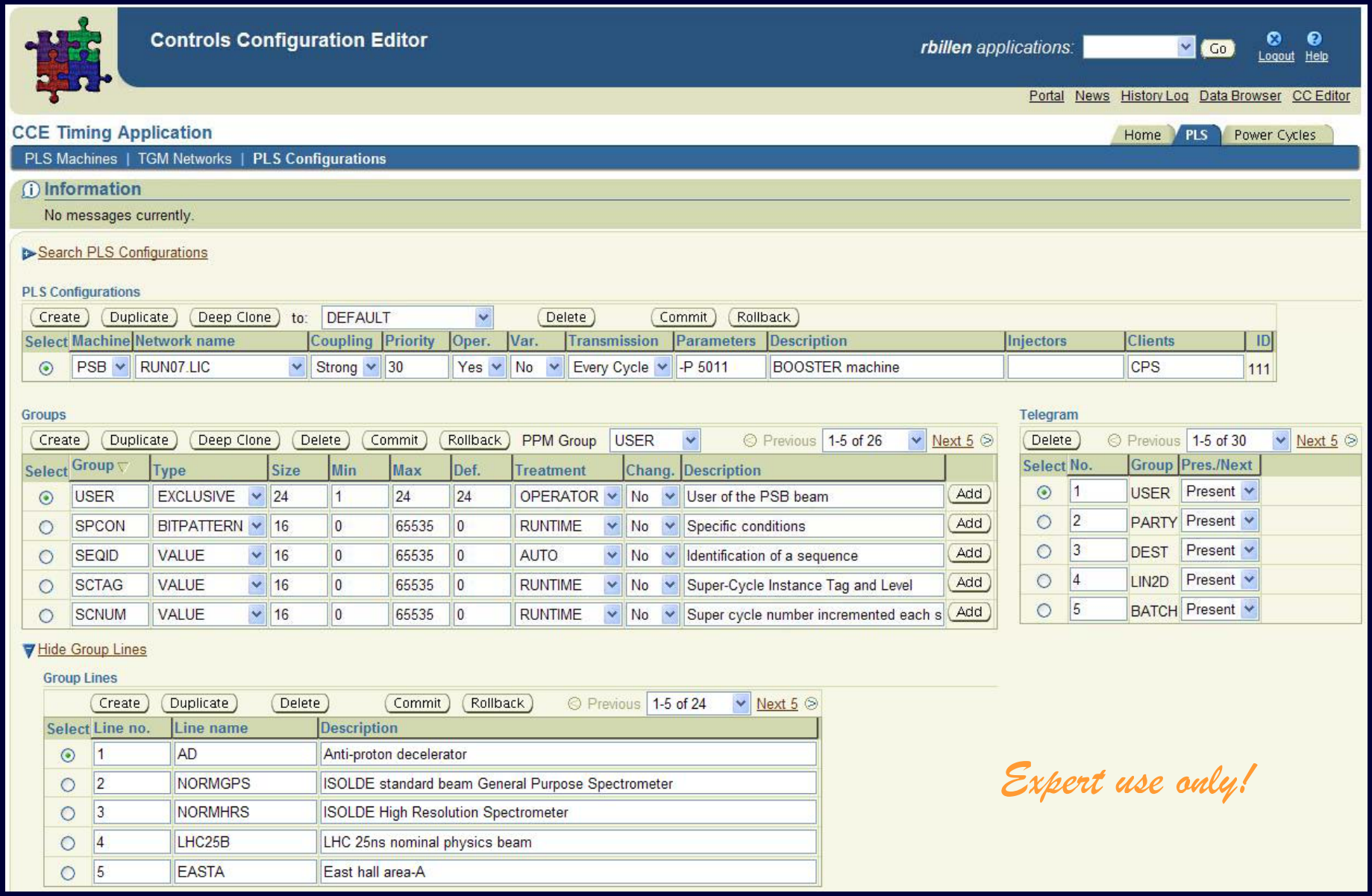
1 - 30

Seq	Group	P/N	Type	Size	Min	Max	Default	Treatment	Description
1	USER	P	EXCLUSIVE	24	1	24	24	OPERATOR	User of the PSB beam
2	PARTY	P	EXCLUSIVE	8	1	4	1	OPERATOR	Partical type
3	DEST	P	EXCLUSIVE	8	1	4	2	AUTO	PSB beam destination
4	LIN2D	P	EXCLUSIVE	8	1	2	1	OPTION	Linac II destination
5	BATCH	P	VALUE	16	0	16	0	AUTO	SPS batch for LHC filling
6	MISC	P	BITPATTERN	16	0	1	0	OPERATOR	Misceleaneous lines

Group lines

Seq	Group	P/N	Type	Size	Min	Max	Default	Treatment	Lineno	Name	Description
1	USER	P	EXCLUSIVE	24	1	24	24	OPERATOR	1	AD	Anti-proton decelerator
									2	NORMGPS	ISOLDE standard beam General Purpose Spectrometer
									3	NORMHRS	ISOLDE High Resolution Spectrometer
									4	LHC25B	LHC 25ns nominal physics beam

PLS Configuration Editor

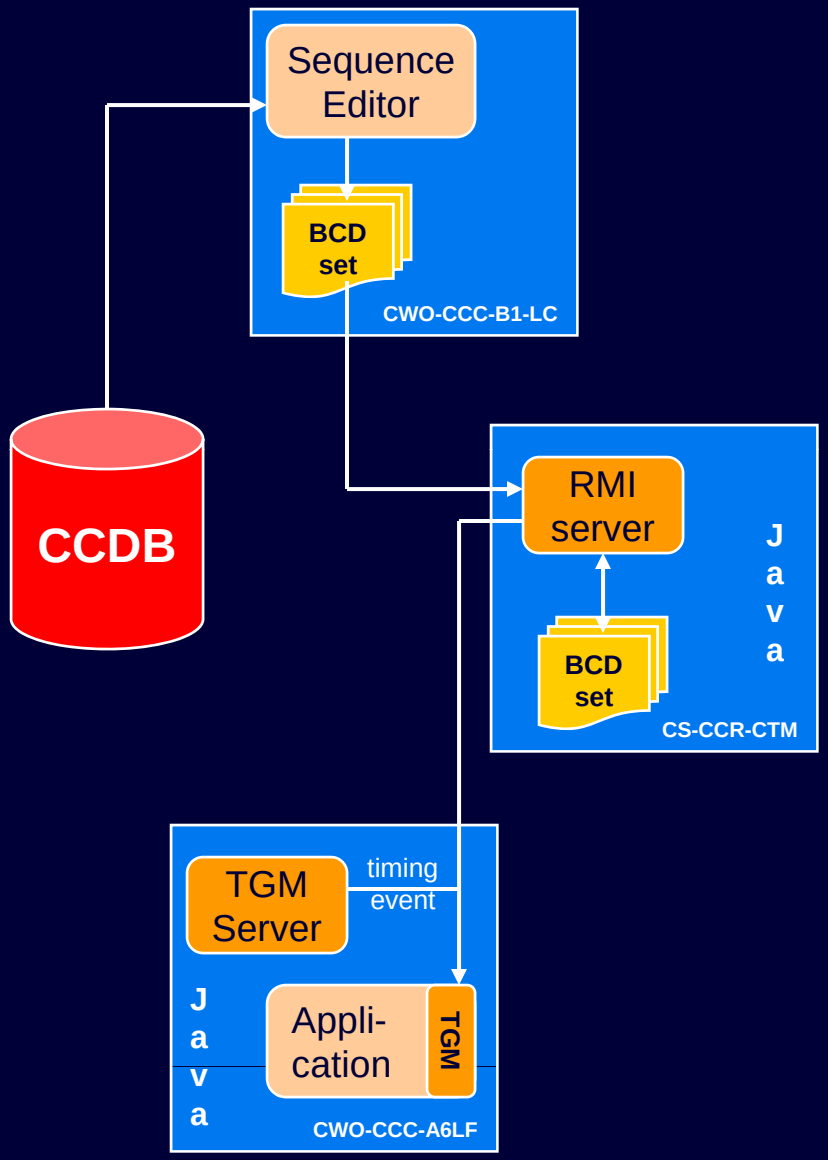


Expert use only!

R. Billen

The current real-time telegram information

- Preparation of BCD with Sequence Editor, by CCC operator, based on the existing meta-data
- Resulting telegrams are stored in serialized Java objects over RMI on the CBCM server CS-CCR-CTM
- Java applications running on workstations can get the actual (and historic) BCD data via the included TGM library
- Current BCD data is re-actualized by the Java TGM-server, triggered by timing events, on the concerned workstation
- ☑ This will continue to exist during 2008

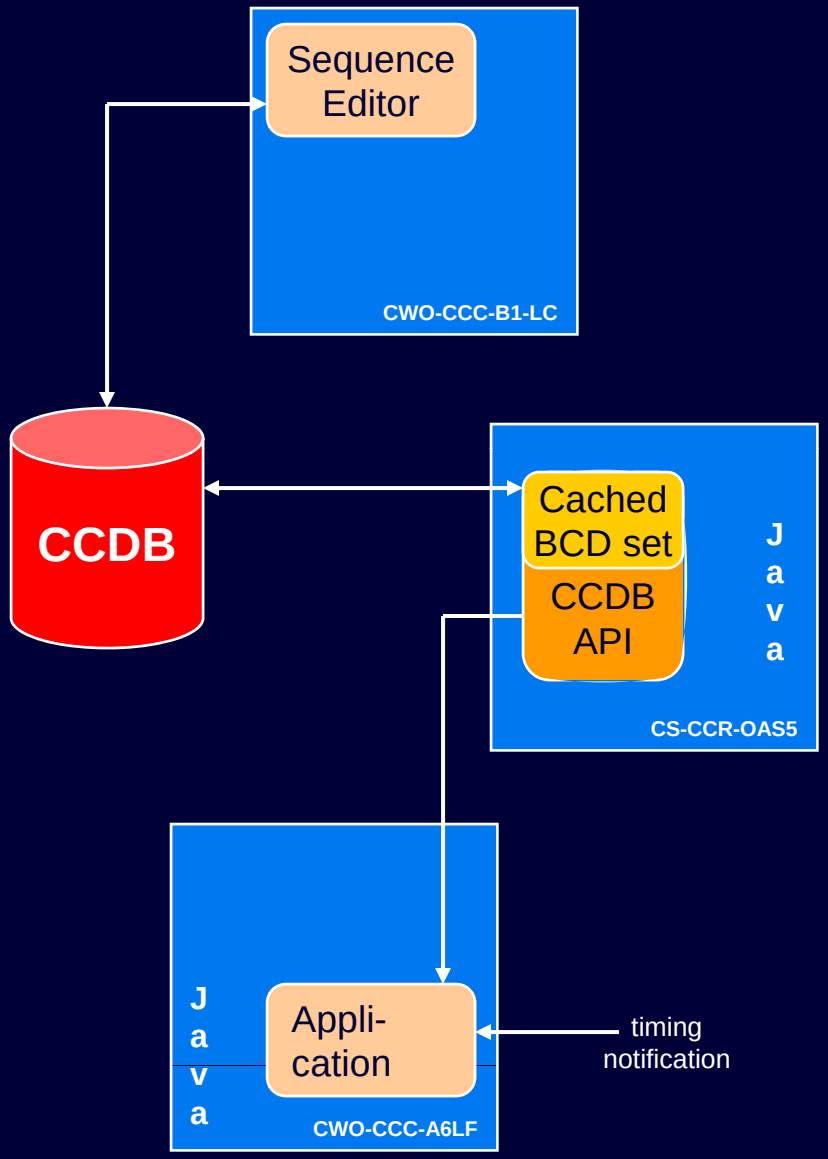


BCD – Beam Coordination Diagram
RMI – Remote Method Invocation
CBCM – Central Beam and Cycle manager

R. Billen

Proposed evolution of the real-time information

- The data in the BCD set are the *currently playing* objects of those already in the CCDB
- Extend the database model to hold this data for *semi-on-line* usage
- Three-tier architecture where cached data on an Application Server is available on-line
- Application gets notified on BCD set change, fetches the new data *only* when necessary
- Data communication via API
- This approach is quite *different* than the standard publish-and-subscribe in our Java environment
- Availability requirements are pushed to the database and application server



R. Billen

Requirements to be satisfied

- Sequence editing is not *actively deployed* until explicitly *confirmed* in the database and available for *propagation*
- Every time a BCD is deployed, it is tagged as a new *version*
- Possibility to retrieve preceding versions of the BCD by the applications
- The API to the CCDB, provided by AB/CO/DM will be the *only interface* for most applications
  - ⇨ Applications will not depend on the TGM package any more
  - ⇨ AB/CO/HT expert applications may continue using it
- This API covers a sub-domain of the future 3-tier deployed Java Directory Services
- Availability, performance, scalability, reliability to be ensured
  - ⇨ Depending on correct API usage by the client applications
  - ⇨ Not to forget scheduled interventions e.g. security patches

R. Billen

Work to do

- AB/CO/DM
  - Capture detailed requirements and functionality from timing experts
  - Extend current CCDB PLS schema
  - Develop Java API with appropriate methods
  - Deploy API (on dedicated application server)
  - Adapt PLS data browser, possibly data editor
- AB/CO/HT
  - Provide domain knowledge
  - Test, feedback, iterate
- Workload - timeline
  - Always more work than originally estimated
  - Milestones to be proposed by the client
  - Implementation of new ideas will inevitably require more study