Timing Review
FESA Requirements with respect to the current Timing implementation

FESA Team

FE section

Outline

- CERN Timing context
- What Timing services are required by Real-time Equipment software:
  - Runtime
  - Configuration
- Summary

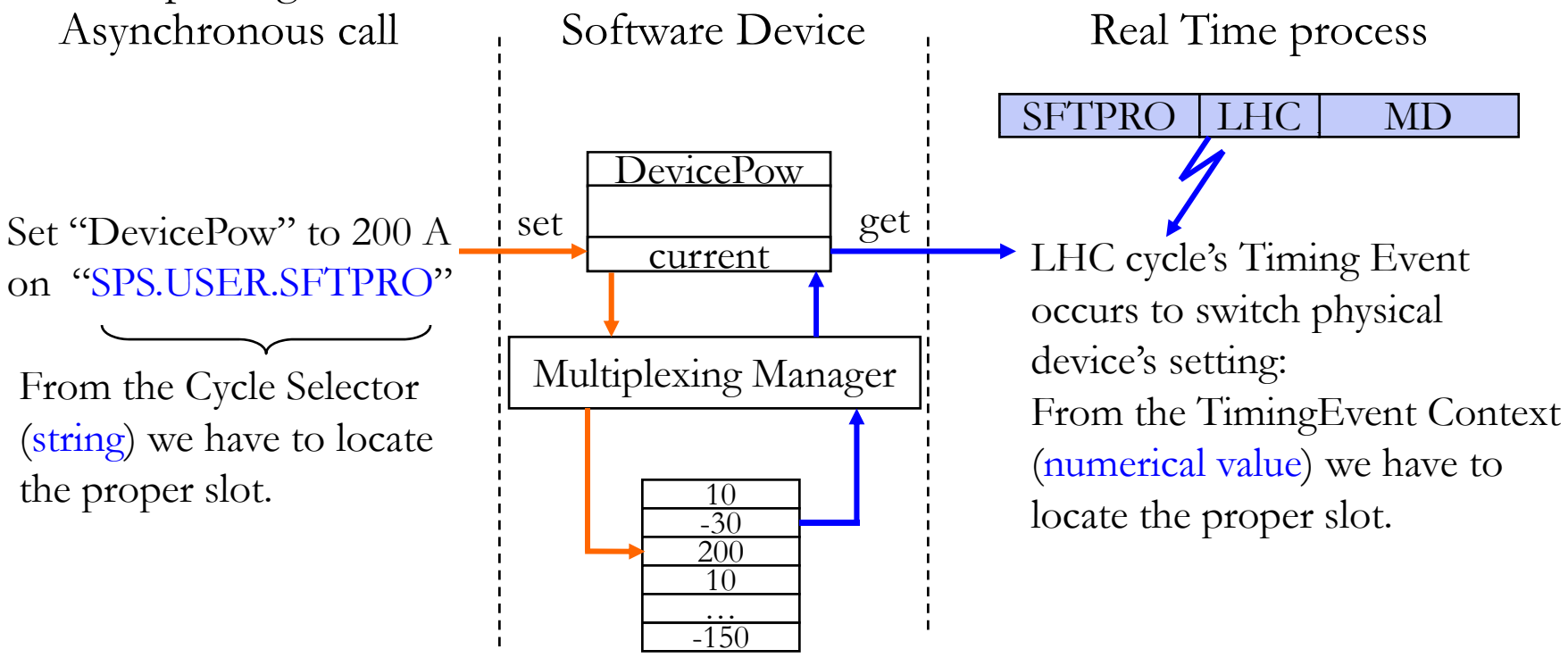
CERN Timing Context

- The Timing drives a Chain of accelerator linked together.
- The timing system acts as the pacemaker of the complex, by sending events to synchronize all the accelerators, by playing sequence of successive cycles indefinitely etc….

➔Cycling Accelerator defines Virtual Accelerator and as a result, Virtual Devices. Switching to the next cycle causes a switch of device's setting synchronized by timing events.

Runtime needs: Device's setting multiplexing on CycleType (USER) (1/3)

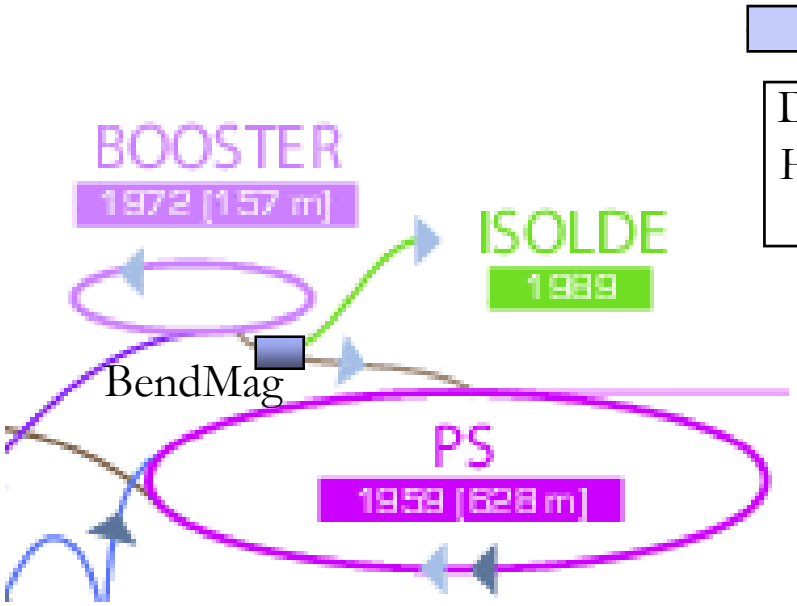
- Accelerator cycleType is identified by a Name (USER). Device setting multiplexing is done on USER.

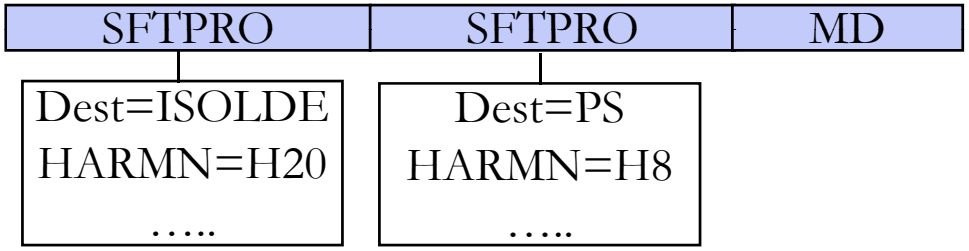


Timing key can be string or value, and therefore we need additional timing information to know the matching.

Runtime needs: Device's setting multiplexing on Telegram's Groups (2/3)

- Sometimes device's setting multiplexing is done on Telegram's data because setting can differ between occurrences of the same cycle. Telegram's data allows to refine a CycleType (USER)





Several Software devices linked to the same physical device in order to hold additional setting: BendMag_Isolde, BendMag_PS. So it is implemented in order to respect the rule "Multiplexing on USER" even if what is done is "Multiplexing on DEST".

Runtime needs: Equipment behavior multiplexing on Telegram's Groups (3/3)

- In some cases Telegram's Data are used not only to handle setting but:
  - To handle equipment behavior.
  - To apply different algorithm for processing data:

```
if (TelegramGroupMICS==“Staggered Ejection”)
    do this computation;
else
    do that one;
```

Run Time Timing Services (1/2)

- Telegram: read telegram's group values for several purpose: multiplexing setting, information, applying different algorithm, etc...
- Equipment Software receives Cycle information in different format:
  - High level java application send request (device/property call) using a string: "SPS.USER.SFTPRO1".
  - TimingEvent returns numerical value to identify a CycleType.

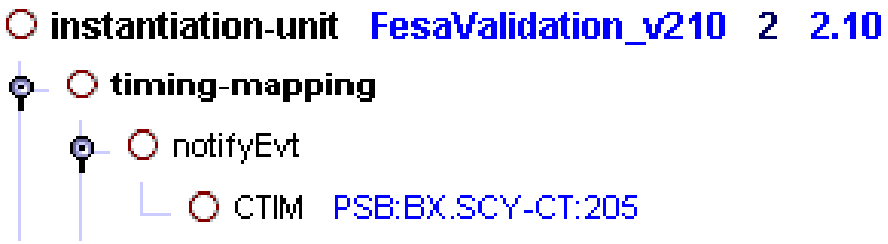
  ➔We need to read the Telegram's Group description (Line Names), providing the mapping string/value.
- Telegram's Group Line Names can change during accelerator run. FESA uses a timing service ("a light subscription") to be notified on change in order to reload the group description.

Run Time Timing Services (1/2)

- Connect to Timing Event
  - Using event Id : event Name is not used because names are less stable than Id. Quite often Event Name embeds the functionality of the timing which may change depending on the way the accelerator is operated BX.WCY350 (warning 350ms before the start cycle).
- TimingEvent wakeup context: all the required information should be available that is to say:
  - cycleTime: time of occurrence of the event in the cycle (ms).
  - timeStamp : time of occurrence of the event (ns: Unix time).
  - isForwarning : is a forwarning event or not.
  - isPayloadCycleId : is the payload can be used for multiplexing purpose.
  - payloadValue : payload value (CycleId, Intensity, TransactionId,…).

Equipment Software Configuration needs (1/2)

- Simulation: timing network is not always available (labs, outsourcing..). As a result, Accelerator cycling simulation is required (definition of Supercycle, Cycle and events). For the time being, is not a Timing service, and FESA has implemented this feature. Could be reconsidered?
- Equipment Software configuration consist, among other, to assign "concrete event" to logical "event name"



Equipment Software Configuration
Timing Services (2/2)

- The service is provided, but the current implementation makes the FESA configuration tools dependant of timing implementation details:
  - Timing Event classes:
    - LTIM, CTIM. LTIM_HARDWARE.
  - TgmNetwork.
  - TimingConfiguration XML file, containing all the central Timing events.
  - Other events (LTIM, etc…) are retrieved from the data-base
- We would like:
  - to see only TimingEvents.
  - to retrieve from the Data Base the complete list of Timing Events available for a particular Front-end.

Summary

- Timing Services covers well Real-time Equipment software needs, but, we can profit from this timing review to:
  - Revise some API.
  - Hide timing implementation details and export only real concepts.
  - Benefit from this unique opportunity to revise historical concepts in depth, having in mind:
    - CERN Users requirements.
    - Simplification.
    - Exportability of the control system.