XFEL DAQ and Control for Photon Beam Systems

XFEL Timing System

and Data Acquisition Concept

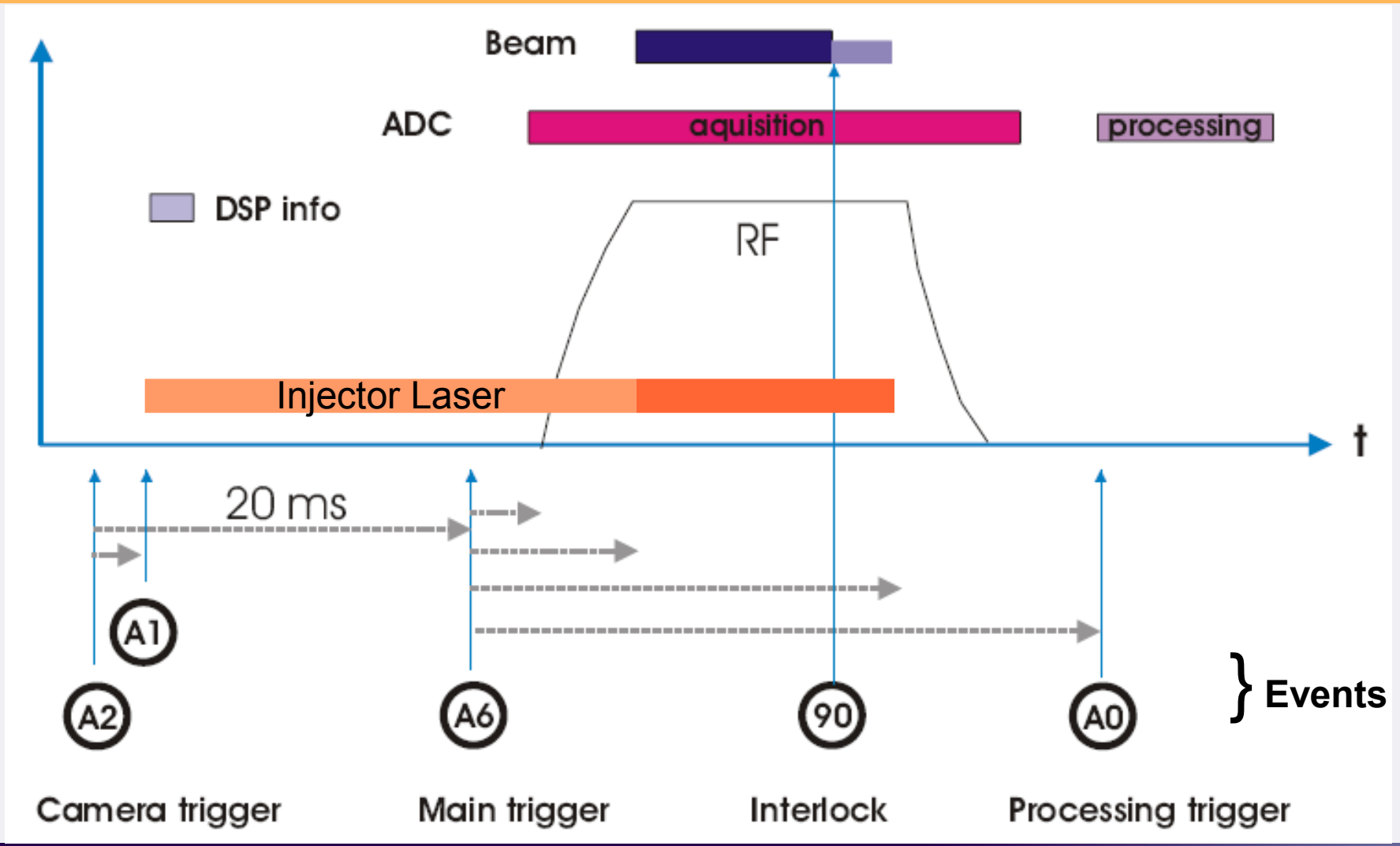
Kay Rehlich

DESY, MCS4, WP-28

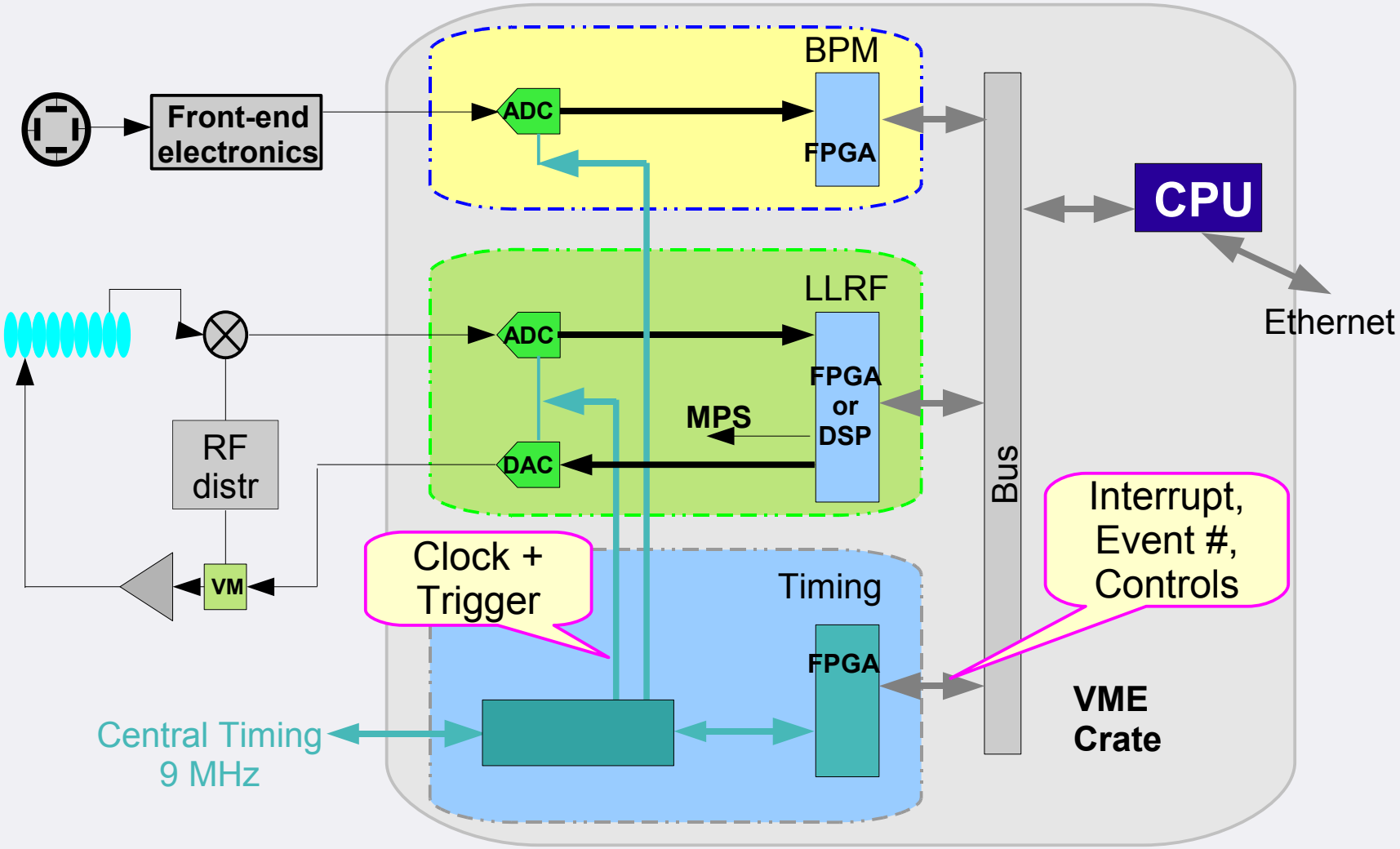
Our Vision

- **Bunch synchronous data acquisition**
  - **ps / fs stable clocks and triggers (hardware)**
- **Full software integration**
  - **Correlation of the data: accelerator and experiments**
  - **A single program can access all data:**
    - **On-line, off-line and Linac, Experiments**

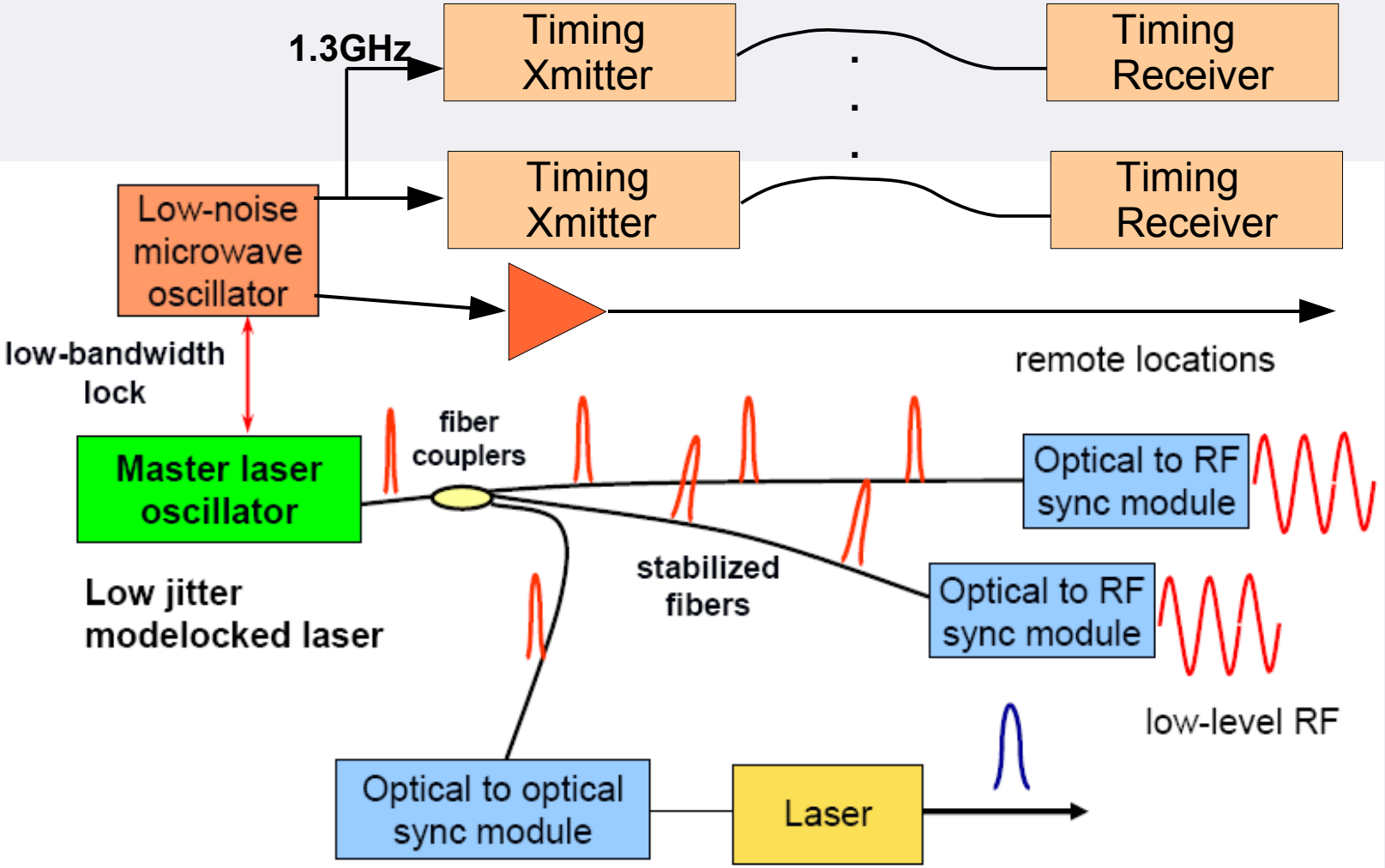
Timing Sequence FLASH



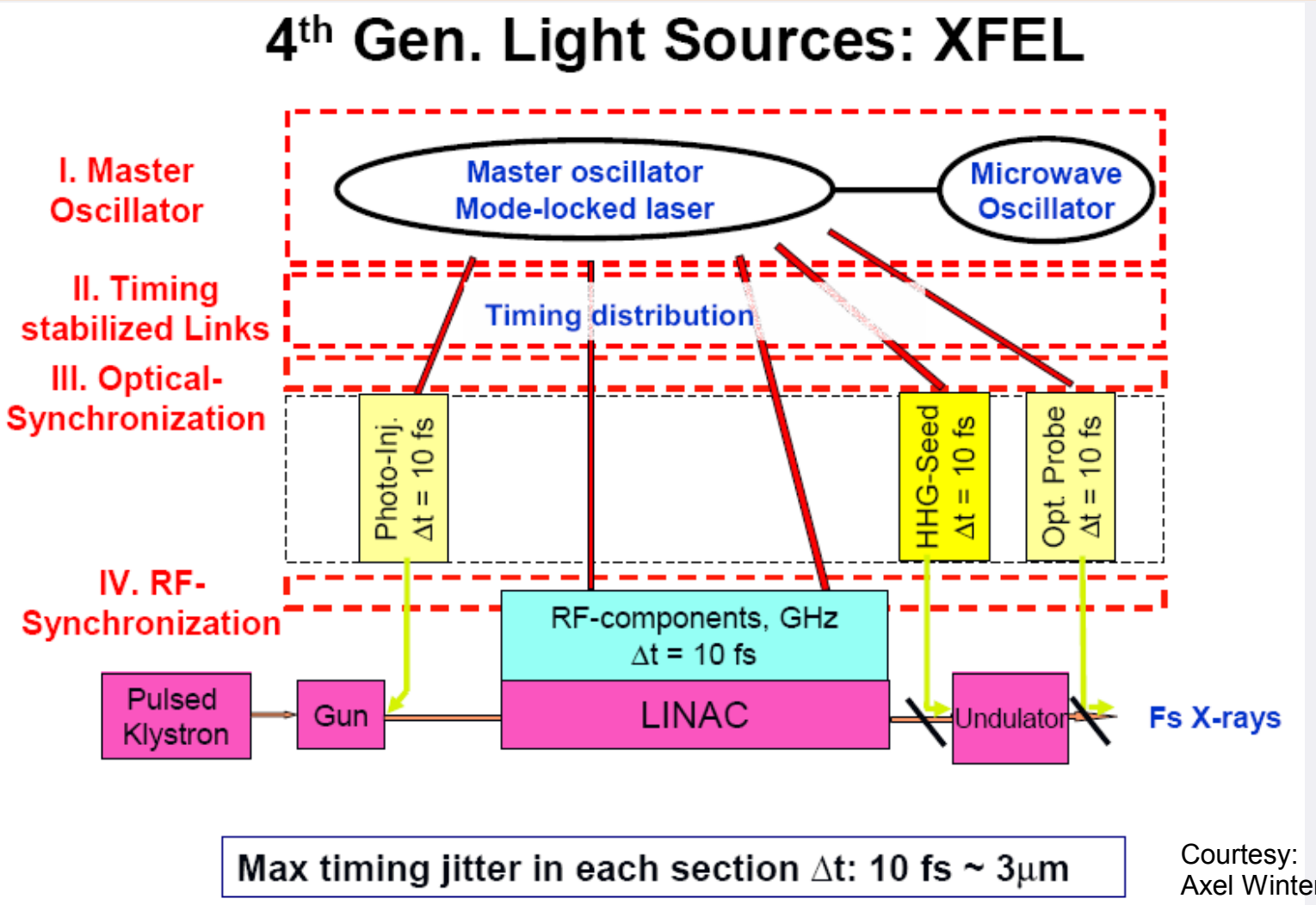
Timing @ FLASH



Timing and Synchronization Systems @ XFEL

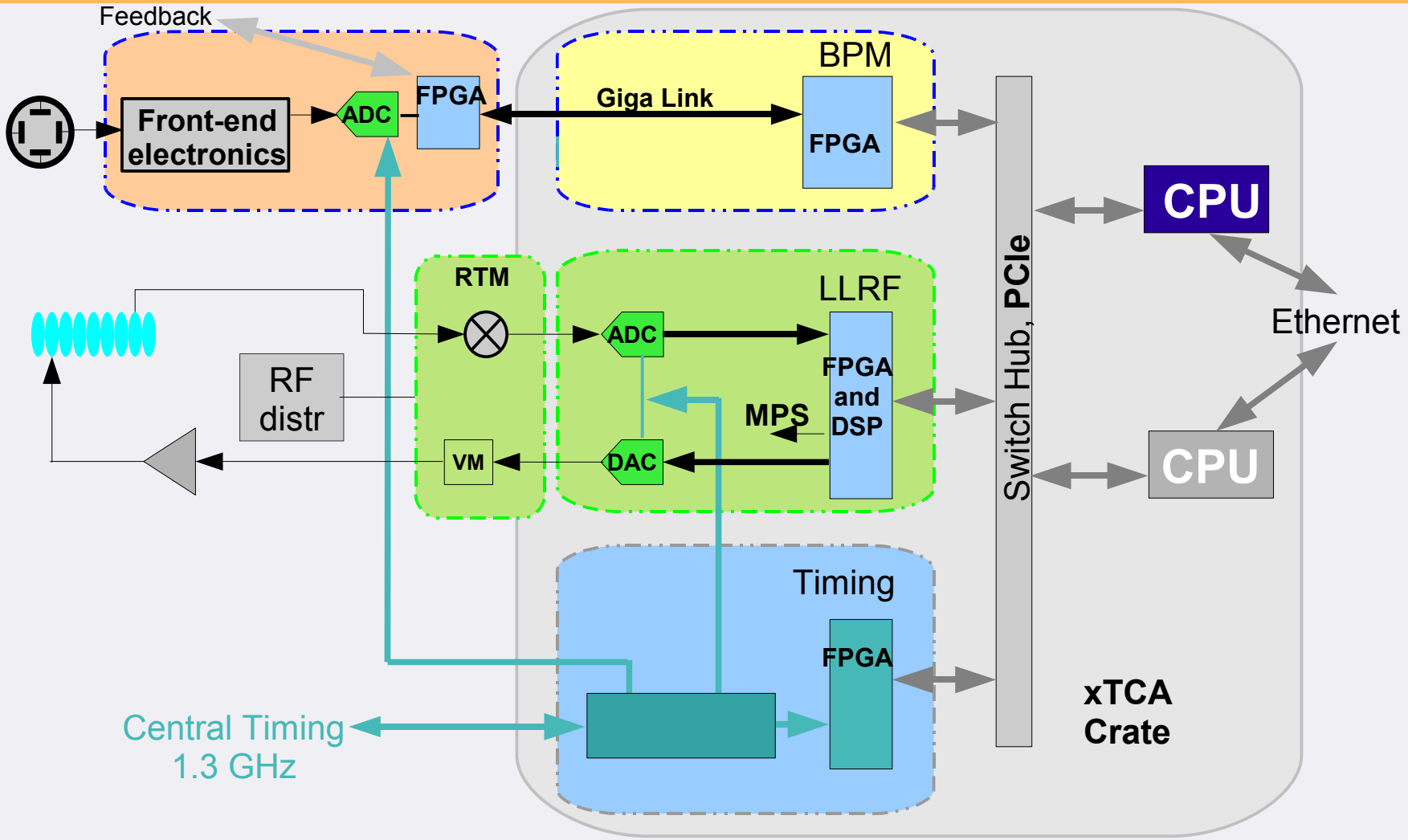


fs synchronization XFEL



Courtesy:
Axel Winter and MIT-Bates

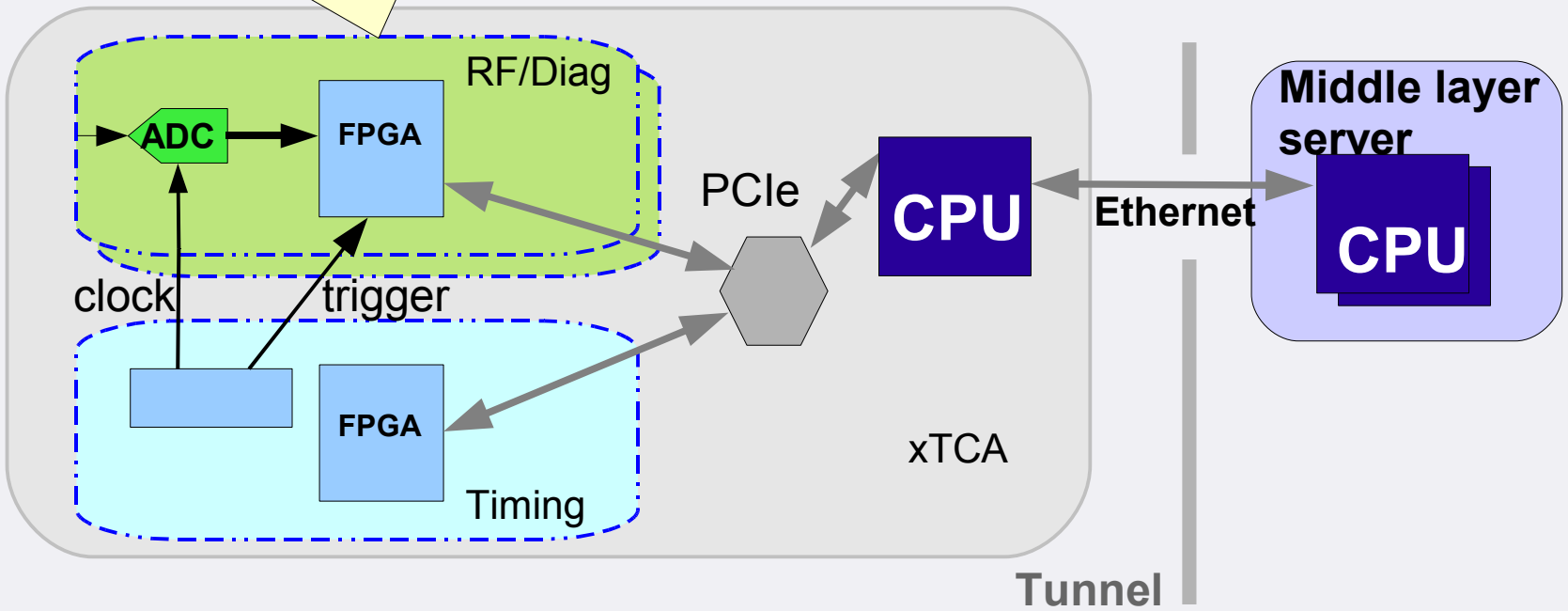
The Front-end: XFEL



Data Acquisition (Hardware)

Hardware:
receives clocks or patterns, triggers

to synchronize the bunches (ps stability)



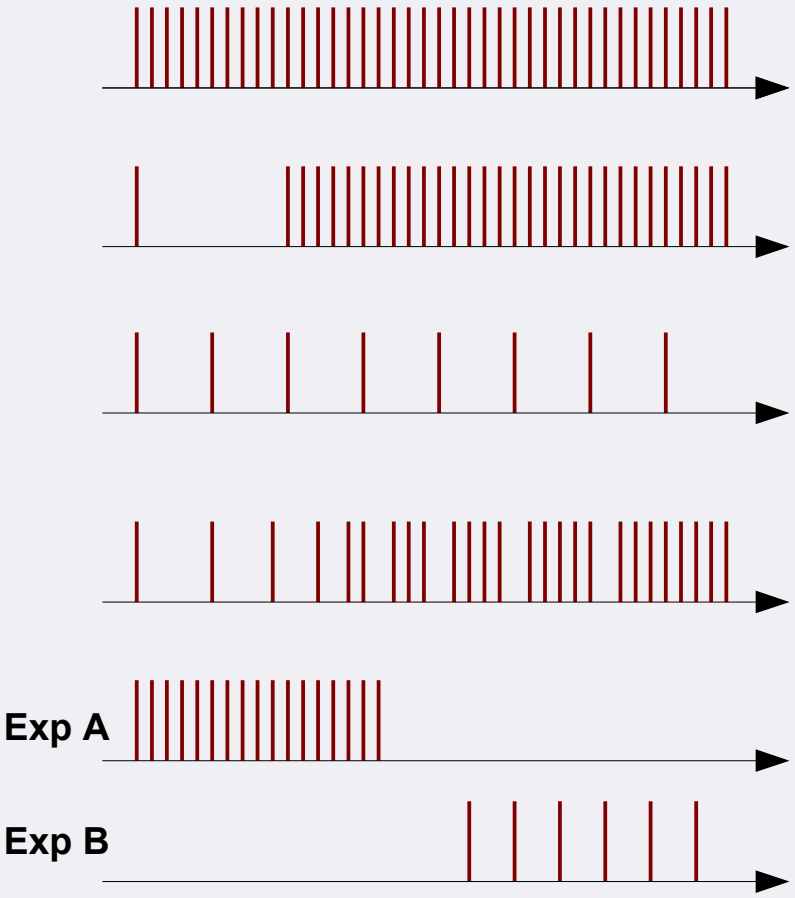
Data Acquisition

CPU:
receives event numbers, interrupts, modes

to synchronize the macro pulses for DAQ

RF/Diag
ADC
FPGA
clock
trigger
FPGA
Timing
xTCA
CPU
Ethernet
Middle layer server
CPU
Tunnel

Possible Bunch Patterns



Max: 5 Mhz, 3000 bunches

Pre bunch

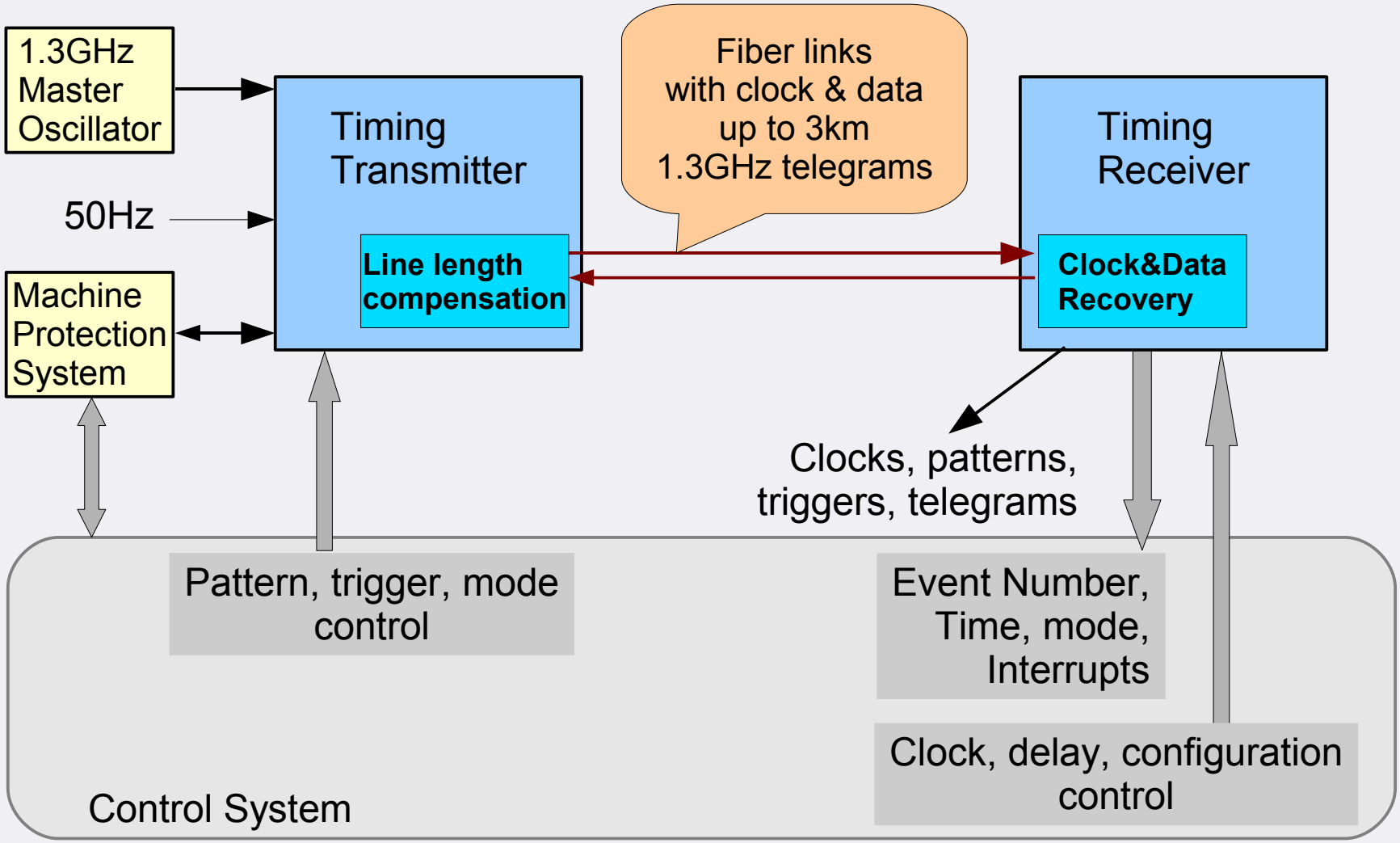
1 MHz or lower frequencies

Arbitrary patterns

} Different patterns @ different beamlines in one macro pulse

or varying patterns from shot to shot

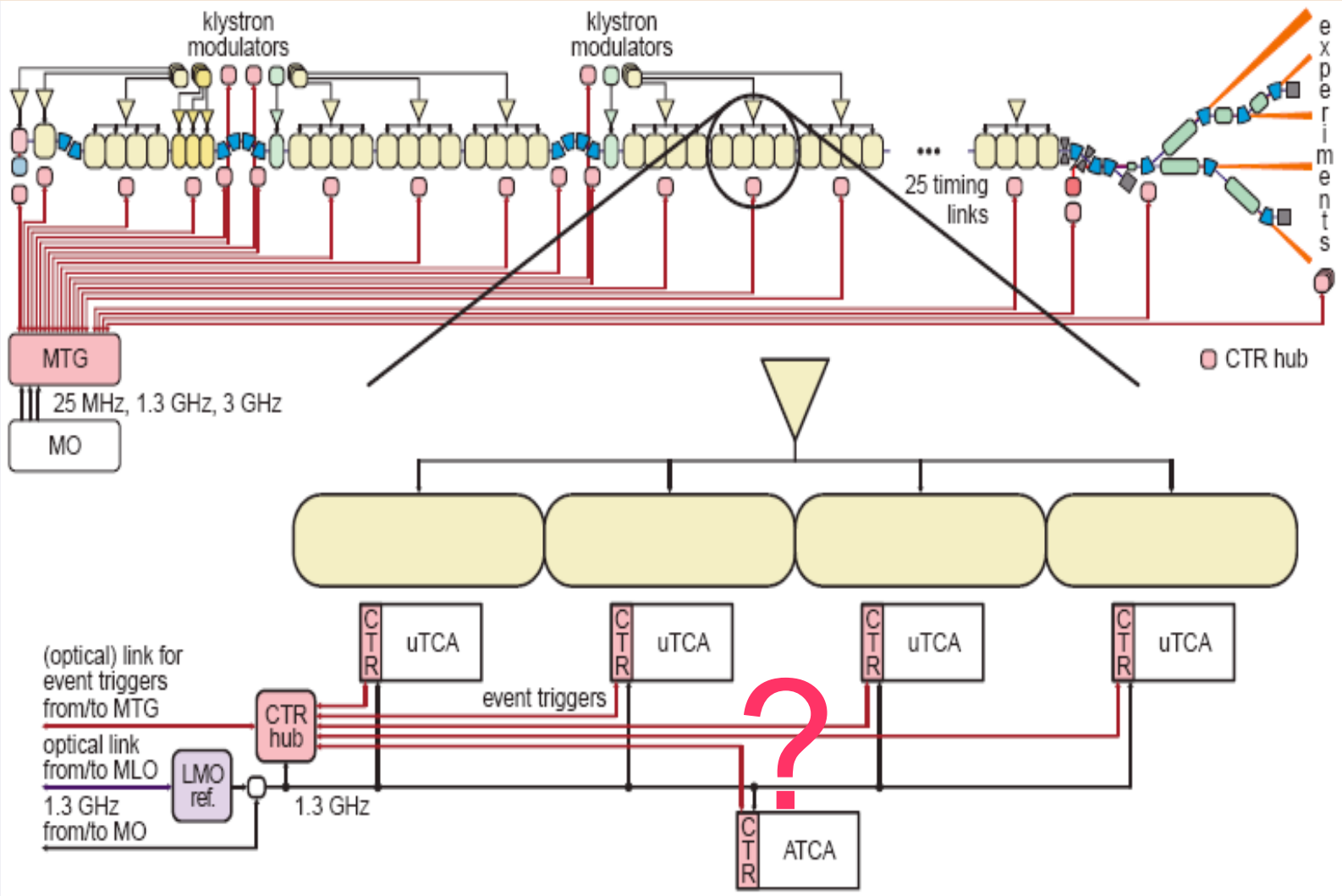
Timing System Blocks



Timing System Requirements

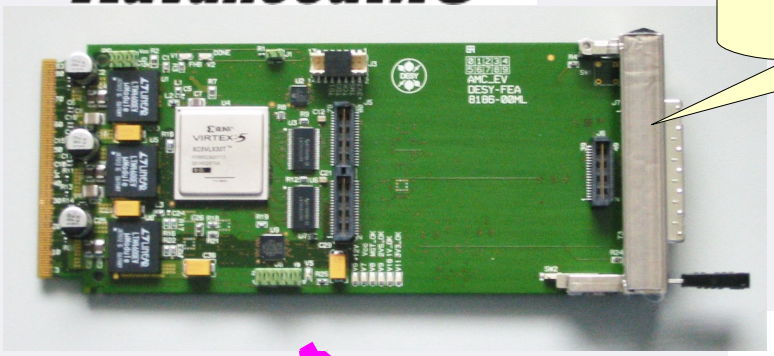
- **1.3GHz telegrams**
  - With clock recovery, few ps jitter
  - Events and data for triggers, event number, modes, bunch pattern, bunch charge?, ...
  - Sender compensates cable length, drifts and measures time delay from sender to receiver
- **Timing receiver outputs (hardware)**
  - Raw telegrams
  - Clock (and gated clocks) on front and backplane
  - triggers
  - Level (LVDS, LVPECL,..) to be defined
  - Connectors to be defined (e.g. Infiniband)
- **Goal: first prototype ready end 2008**

Timing Distribution



Hardware: xTCA

AdvancedMC™

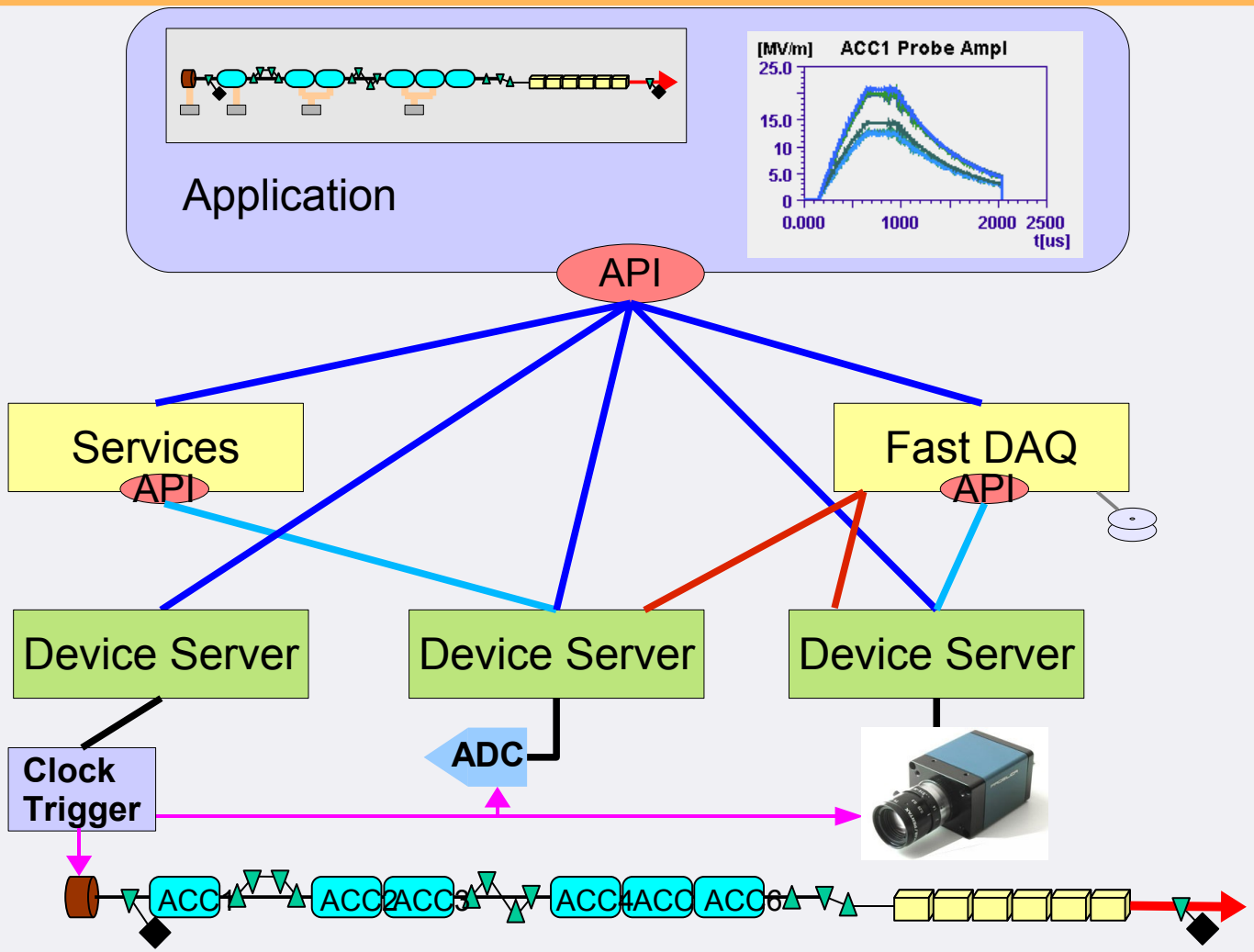


Timing receiver board size

AdvancedTCA®

µTCA™

Software Integration: DOOCS



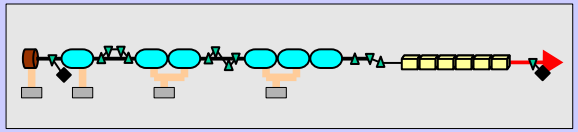
User Interface

Middle Layer

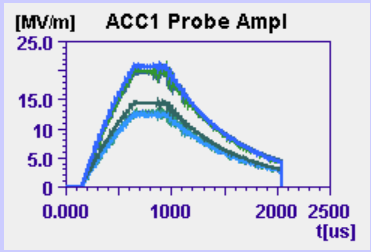
Front-end Tier

Accelerator, Experiments

Software Integration (2): DOOCS



Application



API

DOOCS Application Program Interface

- One interface to access all data
  - On-line
  - Off-line
  - Accelerator
  - Experiments

Fast DAQ

API

Device Server

ACC6

User Interface

Middle Layer

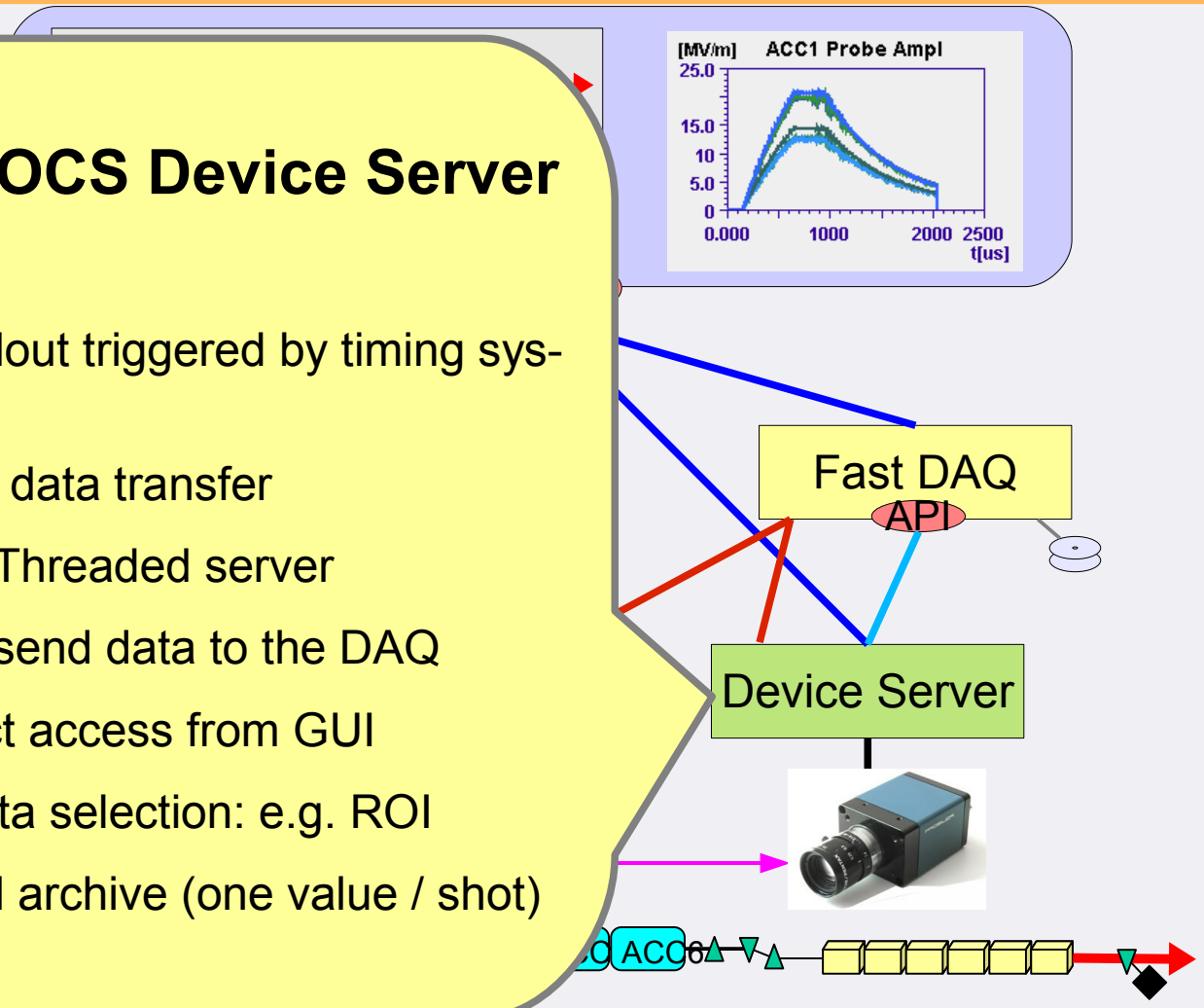
Front-end Tier

Accelerator, Experiments

Software Integration (3): DOOCS

DOOCS Device Server

- Readout triggered by timing system
- DMA data transfer
- MultiThreaded server
- Can send data to the DAQ
- Direct access from GUI
  - Data selection: e.g. ROI
- Local archive (one value / shot)



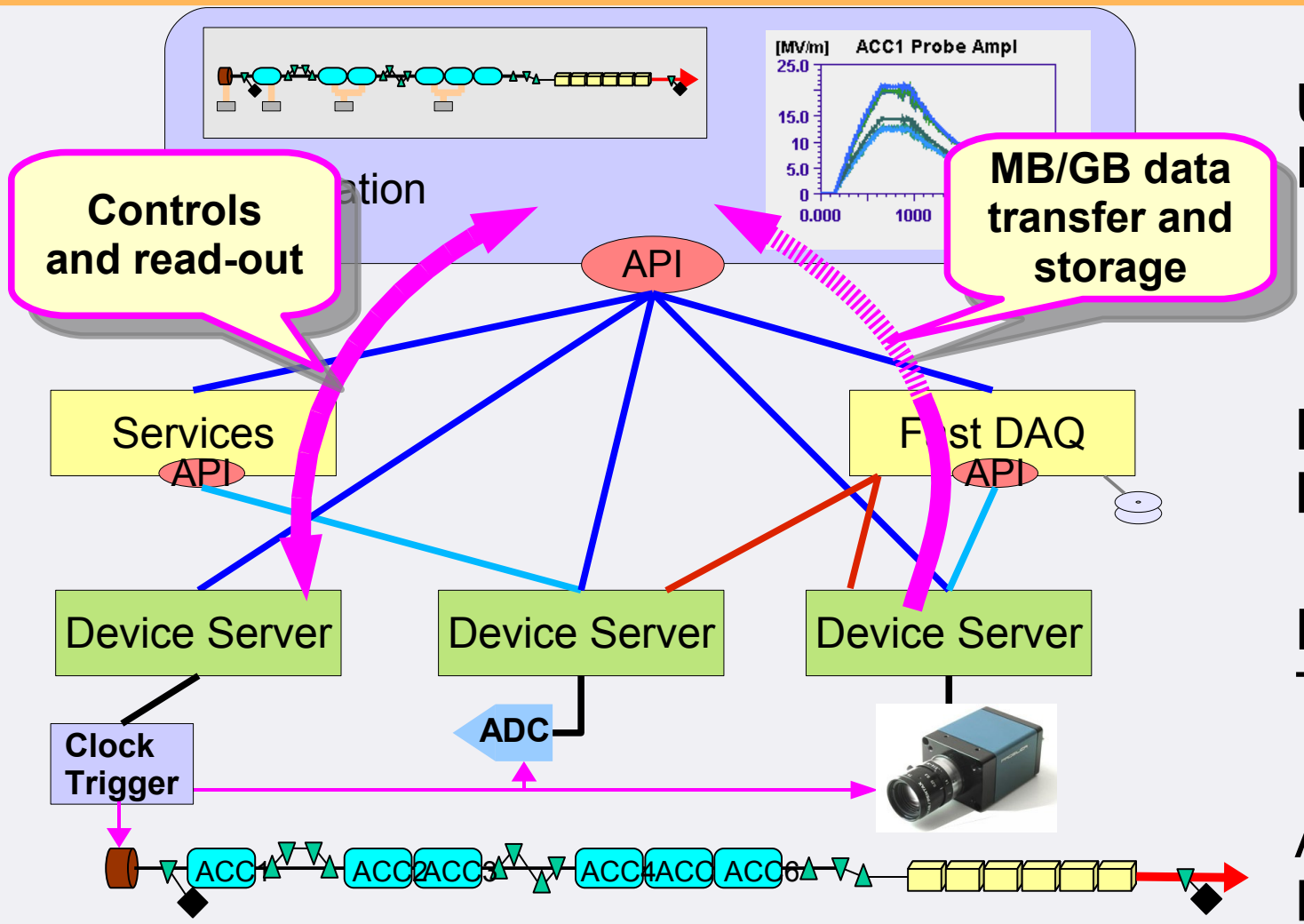
User Interface

Middle Layer

Front-end Tier

Accelerator, Experiments

DOOCS: Integrated environment



User Interface

Middle Layer

Front-end Tier

Accelerator, Experiments

Conclusions

- Hardware timing system
  - ns stable system: a lot of experience gained at FLASH
    - Provides clocks, triggers and event numbers
  - ps and fs stable systems for XFEL are under design
- Software integration
  - DOOCS provides synchronized data acquisition for the accelerator and experiments at FLASH
  - Future developments of DOOCS will provide more featured tools for a full access to online and offline data