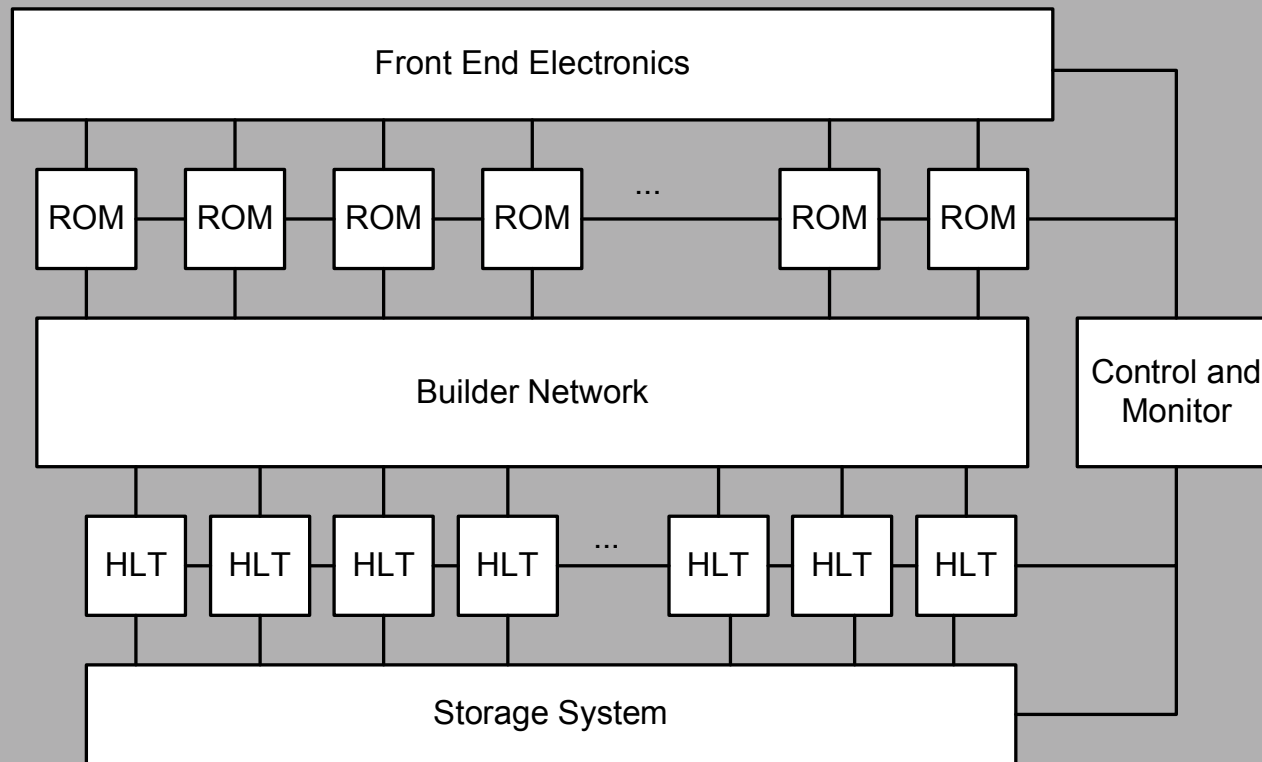


SuperB Online Overview & Progress

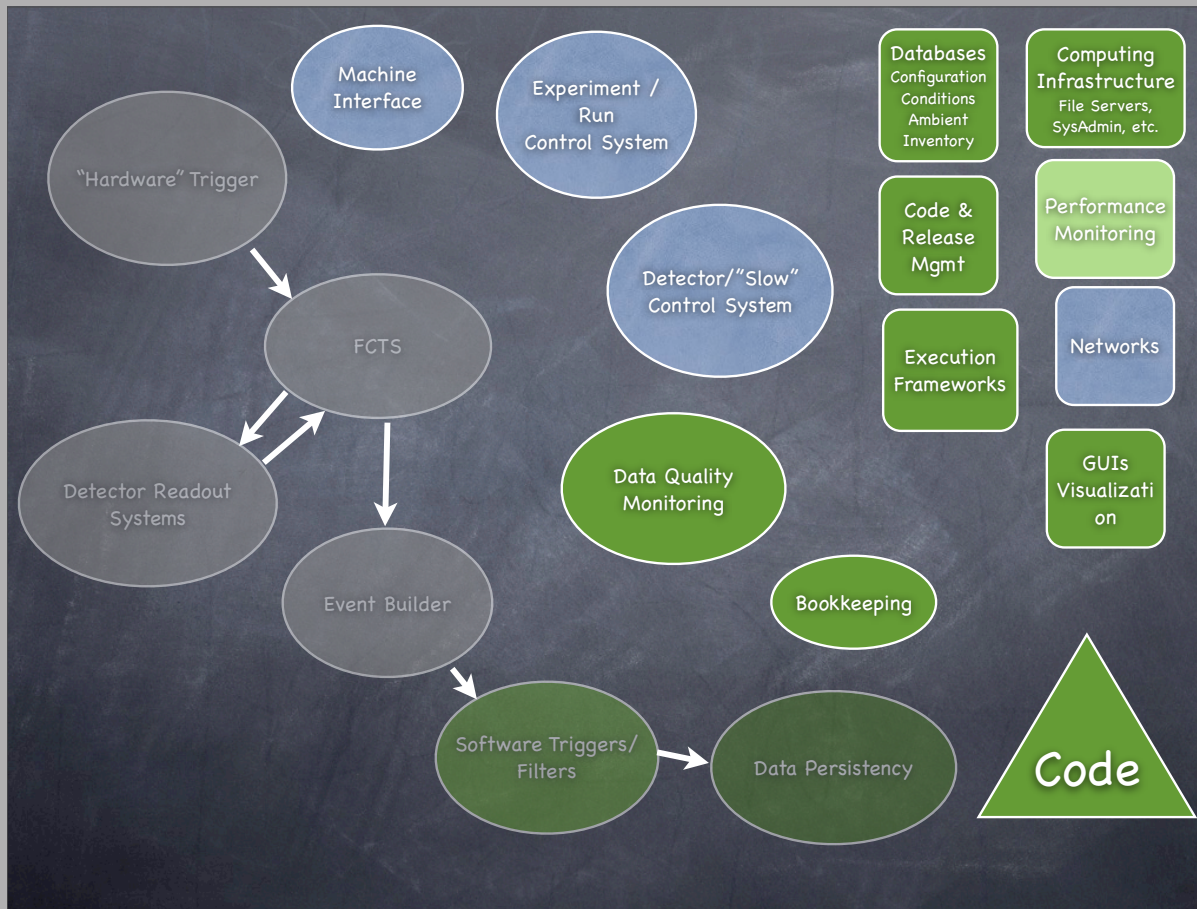
Steffen Luitz, Dominique Breton, Umberto Marconi
Annecy Meeting, 16.3.2010

- ROMs
- Network Event Builder
- HLT Farm and Logging
- Run Control
- Detector Controls / Slow Controls
- Other Systems

Components



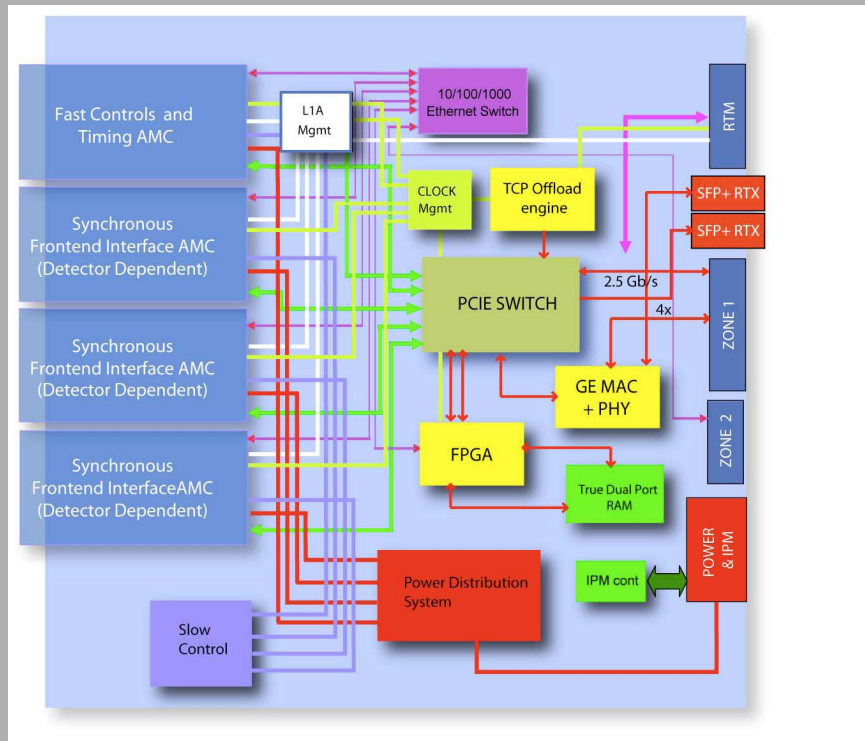
System Overview



Online System Components

- **Baseline assumptions:**
 - 150kHz L1-accept rate, 75kByte Event size
 - HLT (BaBar L3-equivalent) accepts 25nb
 - 25kHz logging rate at 1×10^{36}
 - ca. 12 Gbyte/s input rate
 - Assume x2 "safety" (can't run at 100%) → 24GByte/s
 - ca. 2 Gbyte/s output/logging rate
 - Extrapolated from BaBar
 - Currently best estimate
 - Event size may increase (e.g. SVT Layer 0)
 - Need to design size capability (and/or safety factor)
 - After-FEX event size estimate needed soon from Sub-Detectors
 - L1 accept rate may increase (design for lumi upgrades)
 - Not an issue for Online now (if designed to be scalable)

Rates and Sizes



ROMs

- Process and forward FCTS information to FEE, implement FEE-specific requirements
- Receive data from the sub-detectors over optical links
- Reconstitute linked/pointer events
- Process data (FEX, data reduction)
- Send event fragments into HTL farm (network)
- Would like to use off-the shelf hardware as much as possible (i.e. off-the shelf computers with OL PCIe cards?) → R&D
- Will need to determine processing requirements from sub-detectors.

- Combines event fragments from ROMs into complete events in the HLT farm
 - In principle a solved problem
 - We would like the fragment routing determined by FCTS
 - FCTM decides to which HLT node all fragments of a given events are sent (enforces global synchronization)
 - Choice of network technology
 - Combination of 10Gbit/s and 1Gbit/s Ethernet prime candidate
 - UDP vs. TCP ... a long contentious issue?
 - Pros and cons to both
 - Can we use DCB/DCE end-to-end flow control in switches?
 - Design choices for protocol and network / node congestion control
 - Can we re-use some other experiment's event builder?
 - Interaction with protocol choices

Network Event Builder

- Standard off-the shelf rack-mount servers
- Network event builder receivers
 - Receive event fragments from ROMs, build complete events
- HLT trigger (L3)
 - 10ms/event (baseline assumption, almost 10x BaBar) → 1500 cores needed (~150 servers)
- Data logging & buffering
 - Local disk (few TB/node)?
 - Storage servers over back-end network?
 - Probably 2 day's worth of local storage (2TByte/node)? Depends on SLD/SLA for data archive facility.
 - No file aggregation into "runs" → bookkeeping
 - Back-end network to archive facility

HLT Farm & Logging

- System-wide collection of information
 - Histograms, scalars, etc.
 - L1, HLT
 - small farm that reconstructs sub-samples of events and performs specialized tasks like beam-spot monitoring
 - Automatic monitoring + operator GUIs
- Distributed histogram collection problem
- No specific thoughts have gone into this yet, but will most likely NOT re-use BaBar infrastructure (e.g. DHP)

Data Quality Monitoring

- Run Control
 - Coherent management of the ETD and Online systems
 - User interface, managing system-wide configuration, reporting, error handling, start and stop data taking
- Detector Control / Slow Control
 - Monitor and control detector and detector environment
- No specific thoughts have gone into designing these systems, but we assume that we can use/re-use LHC experiment and commercial technology

Run Control / Detector Controls

- Electronic Logbook
 - Web based – integrated with bookkeeping
- Databases
 - Configuration, Conditions, Ambient
- Configuration Management
 - Authoritative source of configuration
 - Log trail of configuration
- Software Release Management
- ETBD (eventually to be designed)

**Auxiliary & Support Systems
(some experiment-wide)**

- For this meeting focus on ETD issues
- For now most Online design issues are on “computing timescales”
- Online next steps:
 - Define roadmap and timetable for Online
 - FEX / Data reduction in ROMs
 - Work with sub-detectors to
 - Identify processing requirements for FEX/data reduction
 - Determine output data size (needed for network design and initial farm scaling) – some flexibility there but would be good to settle event size for all downstream system design and sizing
 - Map processing requirements on processing units
 - CPU (preferred), FPGA, GPU???
 - Online Software & Infrastructure
 - Look at what others are doing. Pros and cons of certain approaches.
 - Investigate potential use of existing tools (such as CMS xDAQ)
 - More research on Online/Offline code sharing reqs
 - Code, build infrastructure, frameworks databases, etc.
 - ROM R&D (what’s the best way to build a ROM?)

Online Next Steps