Event Builder and Level 3 Trigger at CDF experiment

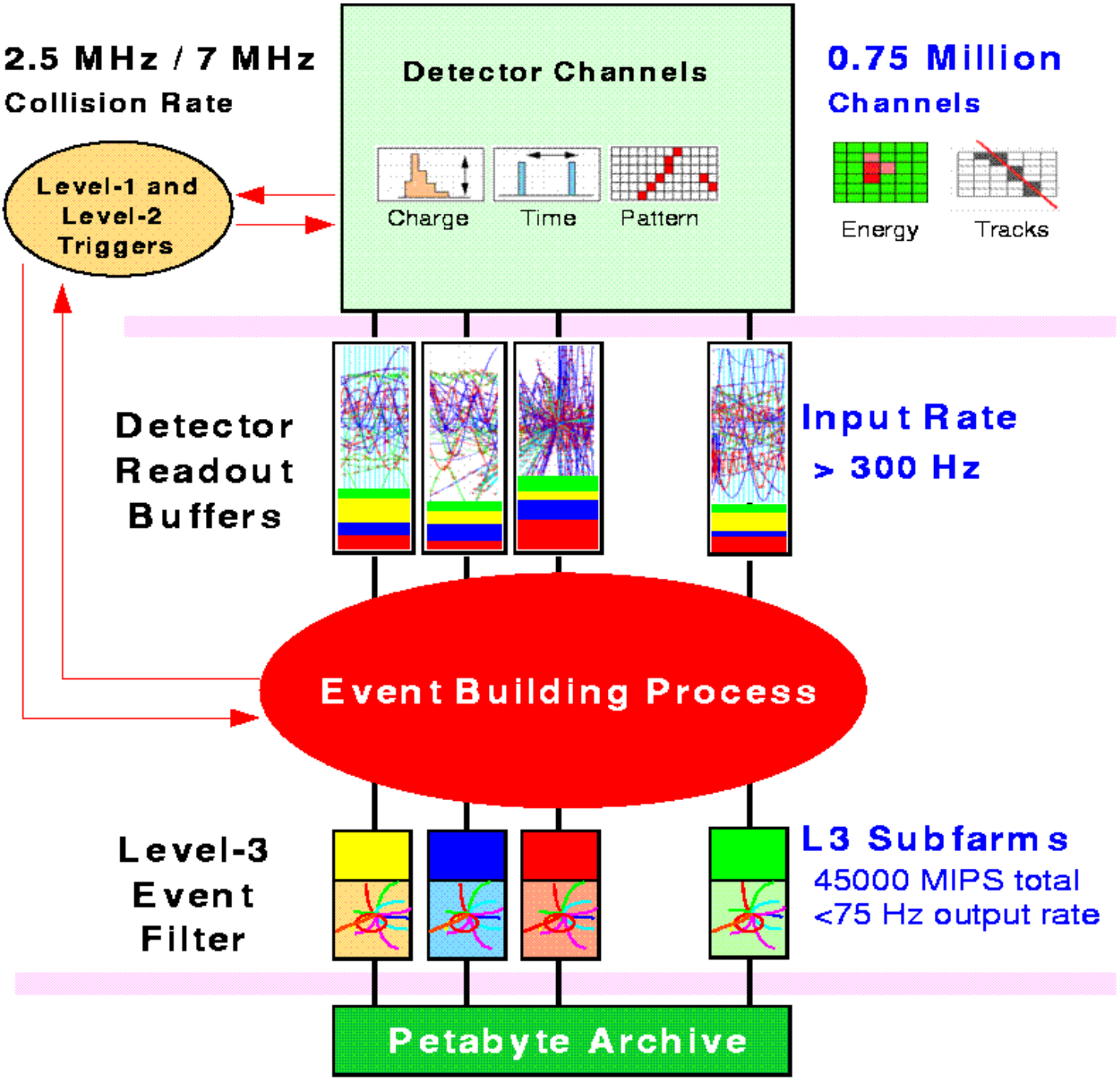
I. Kravchenko

Massachusetts Institute of Technology

CHEP 2000, Padova, Feb. 7 – Feb. 11, 2000

- CDF DAQ Overview
- Event Builder and Level 3 Architecture
- DAQ Integration Tests
- EVB/L3 Control and Monitoring
- Summary

CDF DAQ Overview



Overview: Event Builder and Level 3

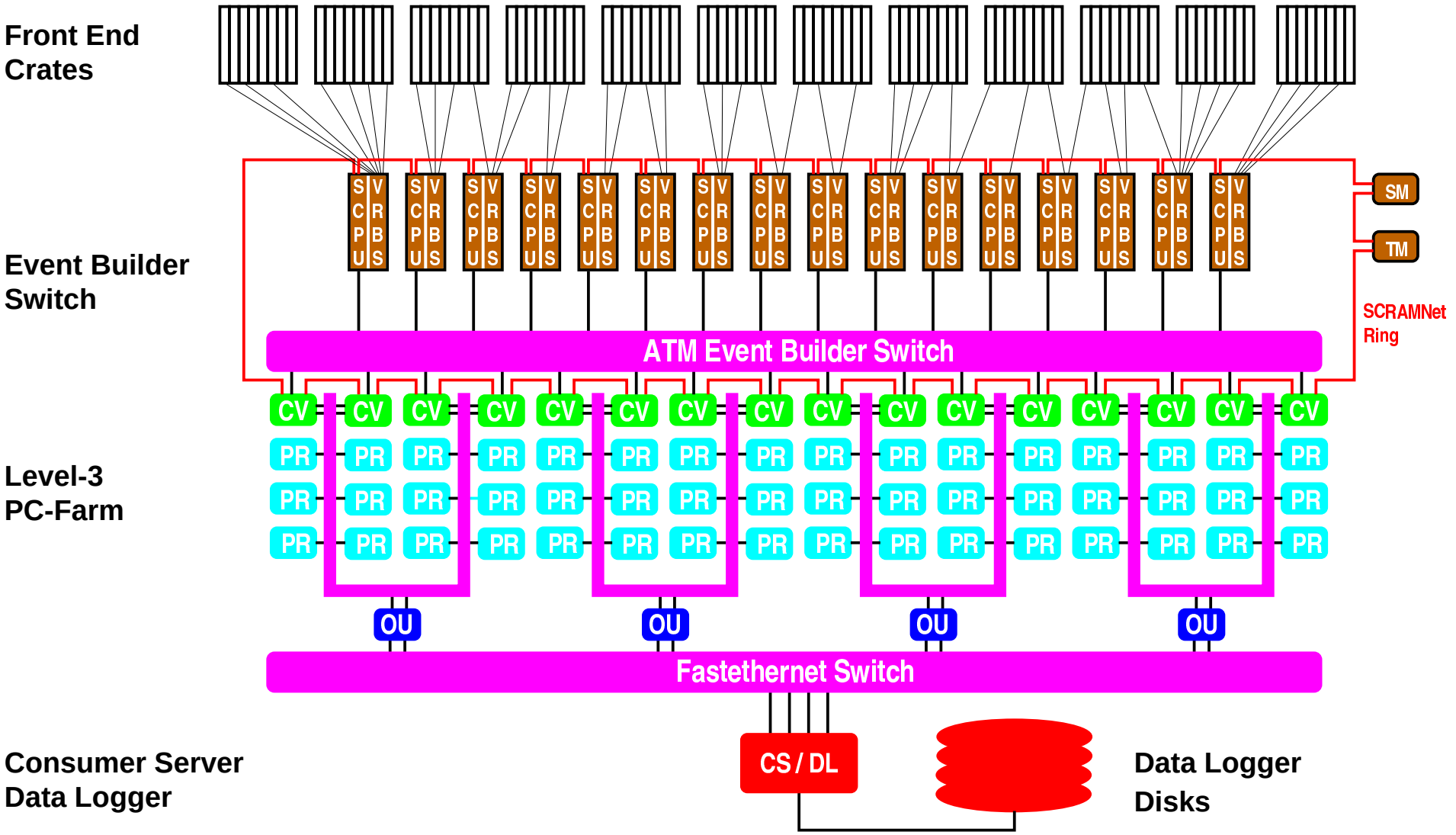
Event Builder

- purpose: assembles event fragments at one location
- design: network topology is a star, based on ATM switch
- specifications: event rate ≈300Hz (up to 1kHz), event size 200kB (up to 250kB)

Level 3

- purpose: run filtering algorithm on reconstructed events
- design: PC farm
- specifications: reduce event rate from 300 Hz(up to 1kHz) to 30Hz (up to 75Hz)

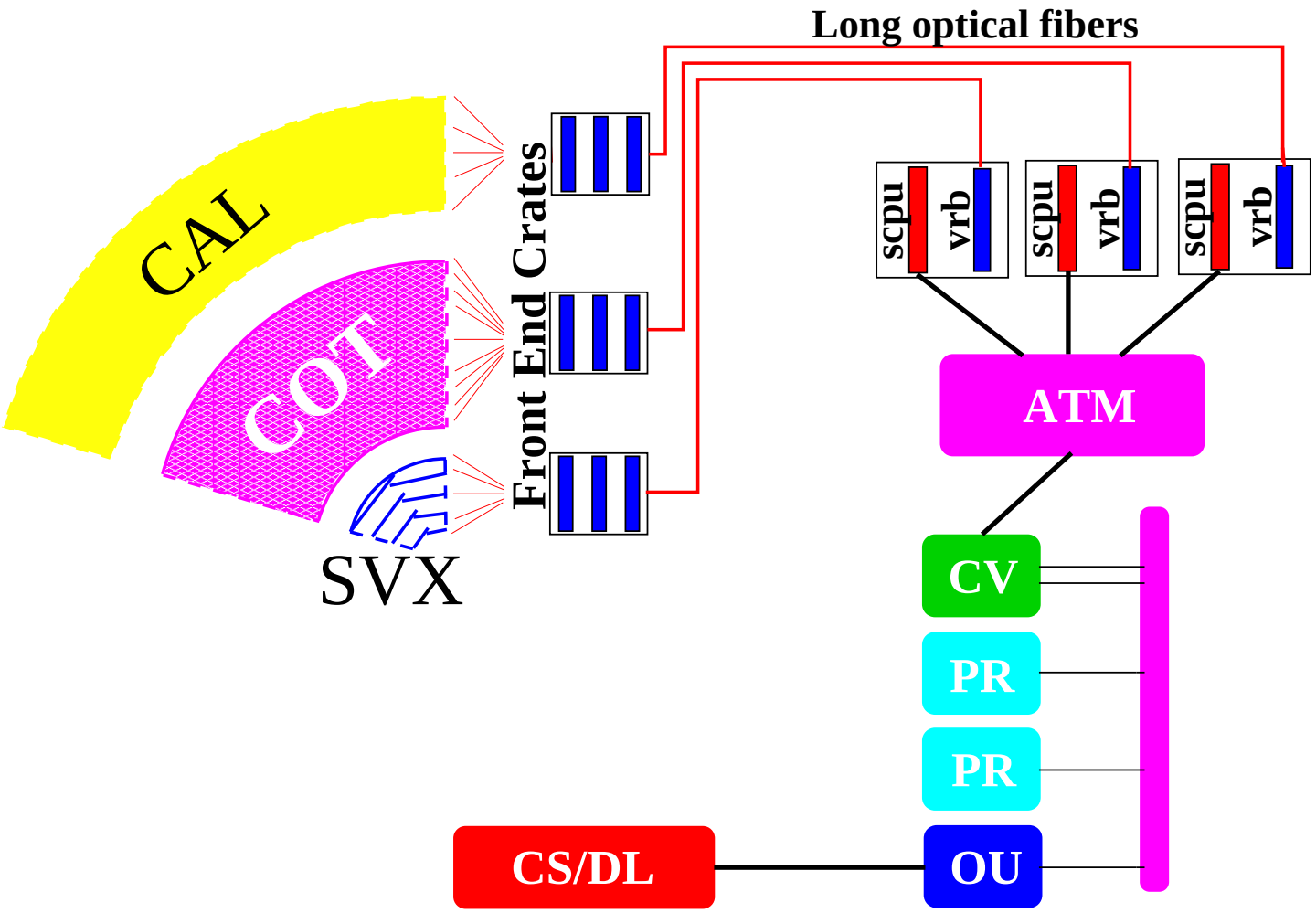
Event Builder and Level 3 Architecture



Integrated DAQ tests

Three-subdetector readout test:

- trigger on cosmics
- read out subset of Front End crates of Drift Chamber, SVX and Calorimeter

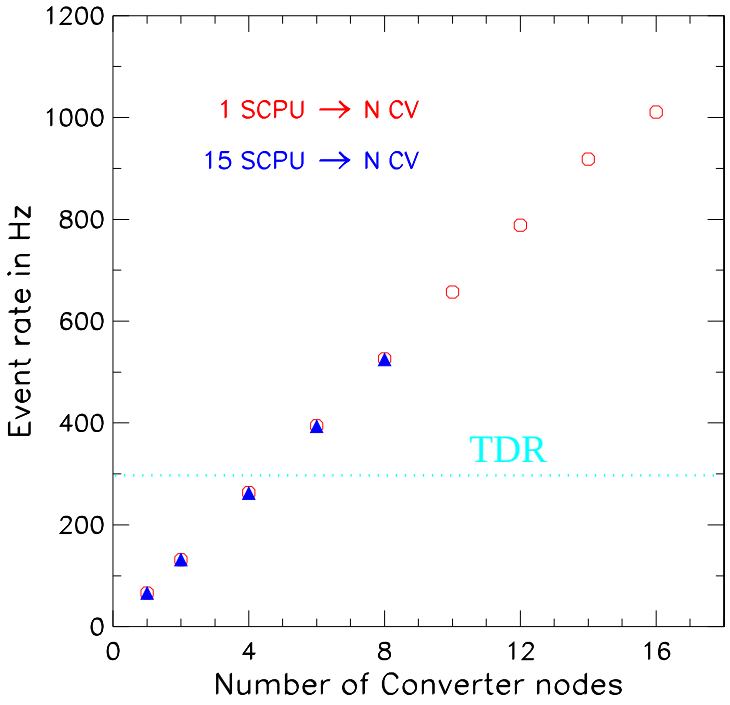


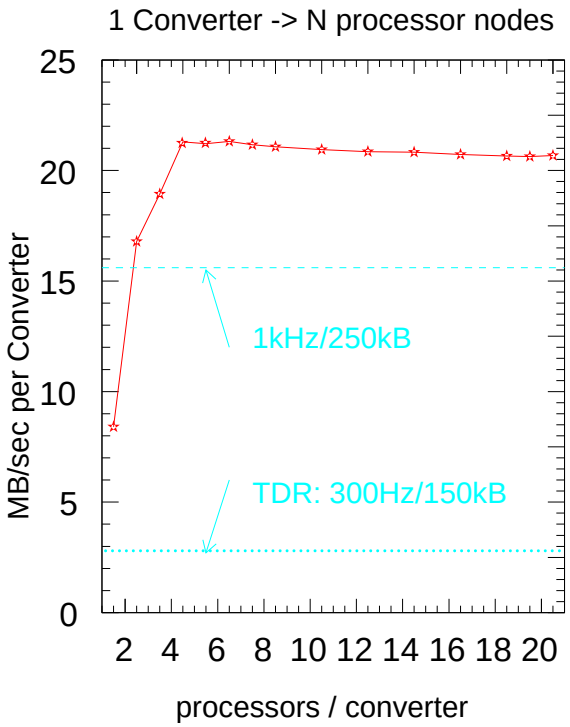
The test exercises the full chain of event flow and demonstrates the full connectivity.

Integrated DAQ tests, cont'd

Event Builder and Level 3 throughput test:

- simulate events on Scanner CPUs
- pass events to Level 3 farm
- realistic event size (16kB/SCPU, 250kB max)
- high event rates (up to 500Hz)





EVB and Level 3 exceed 500Hz (250kB events) in stable operation.

→ comfortably exceed TDR specifications

Now aiming for 1kHz.

Overview: Control and Monitoring System

Functions:

- start and coordinate processes and tasks on the nodes of EVB and L3
- interact with top-level CDF Run Control
- facilitate expert operations
- provide status and statistics information when needed

Difficulties:

- deal with up to 300 computers
- accommodate 2 operating systems: Linux (PC) and VxWorks (VME CPUs)

Software:

- Languages: C, C++ and Java
- ROOT for Level 3 framework
- Talk to ORACLE database using JDBC

Control and Monitoring: Communications

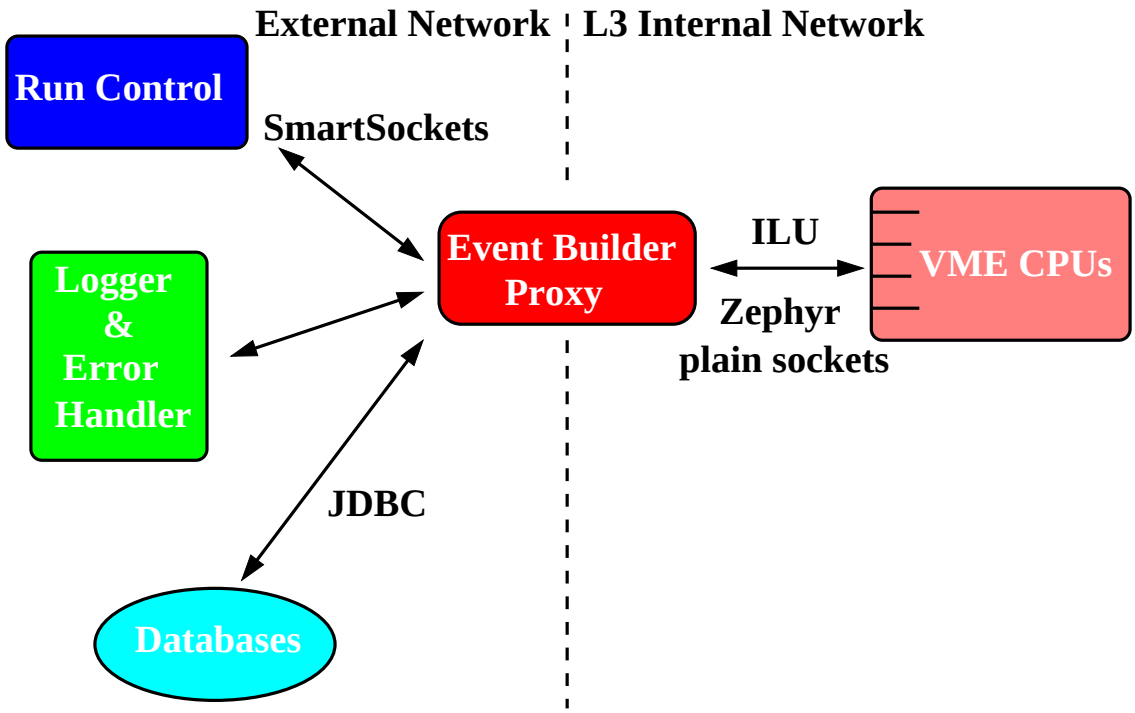
CORBA-based

- ILU (by Xerox, free) for VxWorks
  - supports C binding
  - fits into <1MB of memory
  - ported to VxWorks by our group
- ORBacus (by OOC, inc, free) for Linux
  - supports C++ and Java
  - easy to work with
- communication ILU-ORBacus works well

Publish-subscribe messaging systems

- SmartSockets (by Talarian, commercial) used extensively at CDF
- Zephyr (by MIT) used in EVB/L3

Event Builder Control and Monitoring

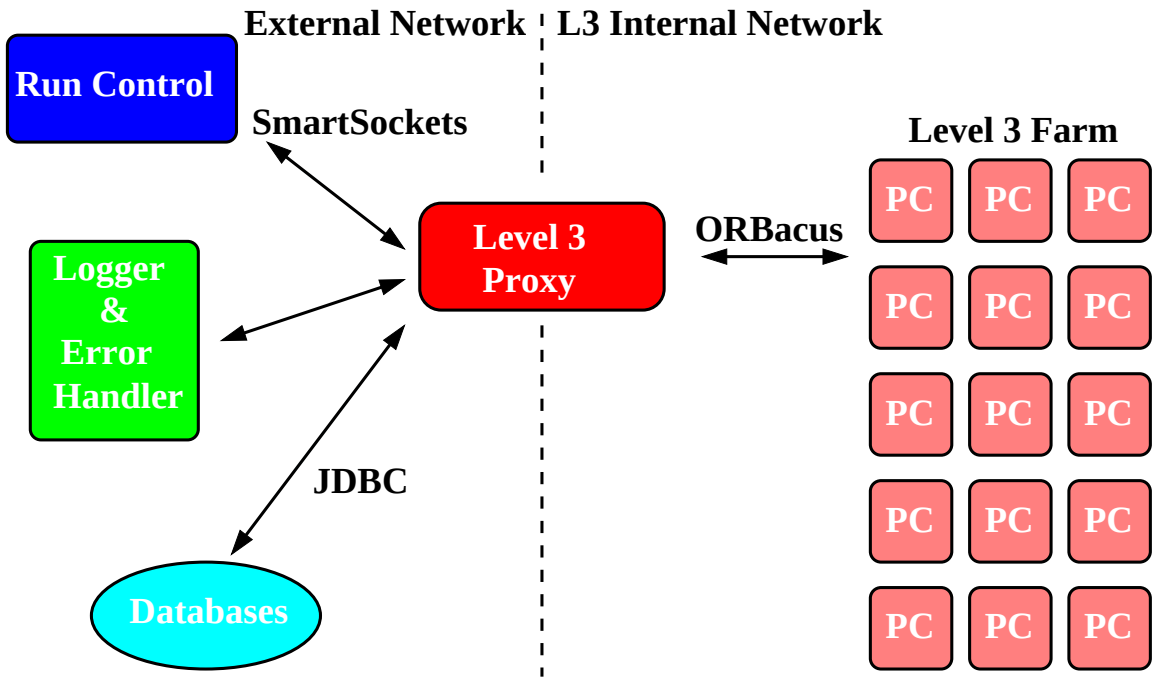


VME CPUs are controlled by Event Builder Proxy

The Proxy:

- works in accordance with top-level CDF Run Control (via SmartSockets)
- performs system state transitions (e.g., setup, configure, activate)
- allows expert operations
- collects monitoring information (ILU, Zephyr, plain sockets)

Level 3 Control and Monitoring



- Central control executable runs on Level 3 Gateway computer
- Control framework is based on ROOT
- Currently using scripts, Level 3 Proxy and communication with top-level CDF Run Control is under development
- CORBA IIOP communication with the Level 3 computers
- Relay mechanism is employed for interaction with individual nodes

L3 Control: ROOT script example

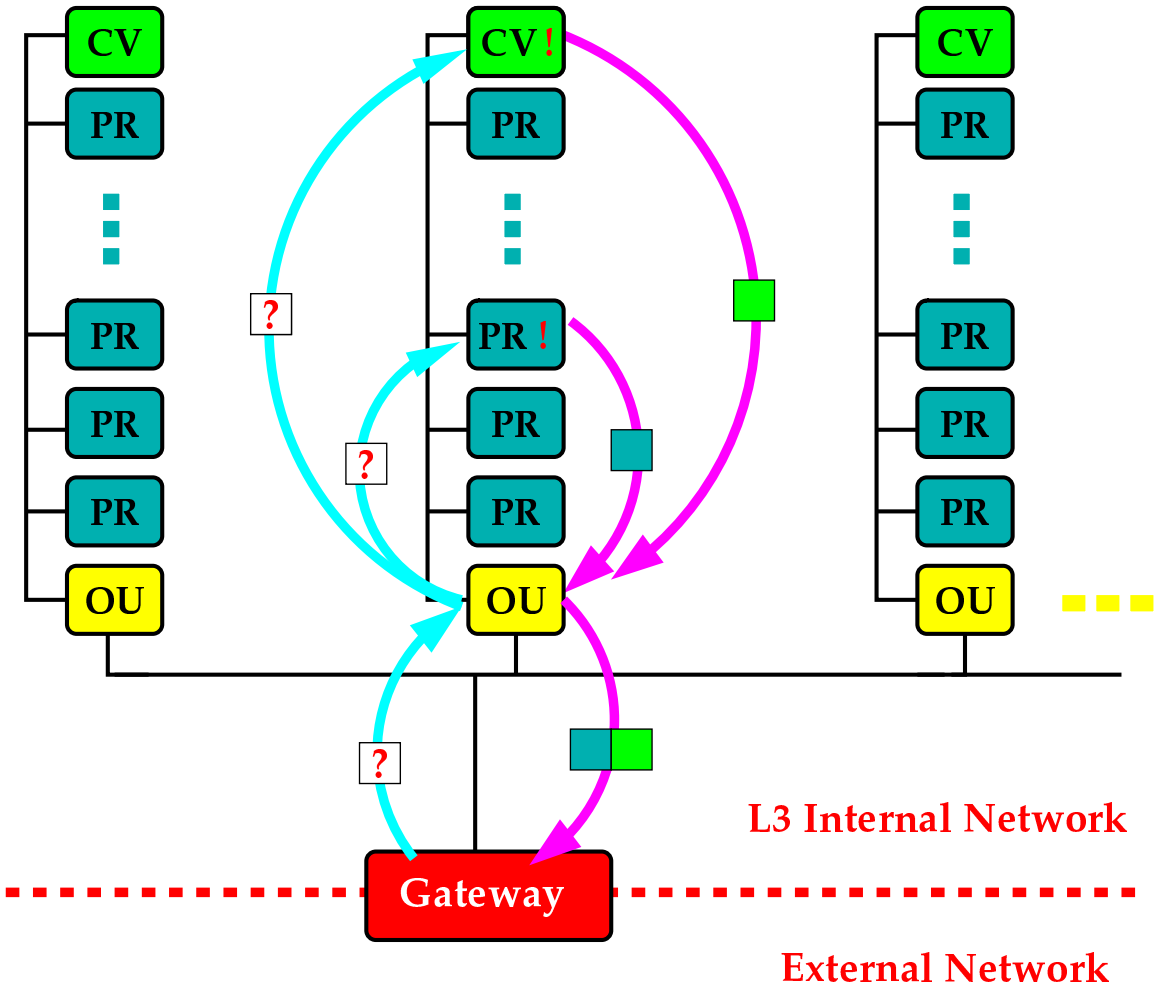
In this example, we configure and run Level 3 system consisting of 1 converter, 2 processor and 1 output node.

```
{
  // Node objects
  L3Converter cnv(‘‘b0l3c01’’);
  L3Processor p05(‘‘b0l3005’’);
  L3Processor p06(‘‘b0l3006’’);
  L3Output    u01(‘‘b0l3u01’’);

  // Processor node sets
  L3ProcessorSet set1;
  set1.AddNode(&p05);
  set1.AddNode(&p06);

  // Subfarm definition
  L3Subfarm sub1(&cnv)
  sub1.AddSet(&set1)

  // Add output nodes to subfarms
  sub1.SetOutputHost(u01.GetName());
```

L3 Control: ROOT script example cont'd

```
  // Define Partition
  L3Partition part;
  part.Add(&sub1);

  //State Transitions for output nodes
  u01.Initialize();
  u01.Start();
  u01.Config();
  u01.Activate();

  //State Transitions for partitions
  part.Initialize();
  part.Start();
  part.Config();
  part.Activate();

  gSystem->Sleep(60000); // one minute

  part.End();
}
```

Level 3 Relay Mechanism



- start and stop processes
- request and collect information
- distribute text, script and binary files
- implements multicasting and routing for arbitrary network architecture

Level 3 Monitoring System

Objectives:

- provides sufficient information on all components of L3
- has low impact on event flow

Monitored information:

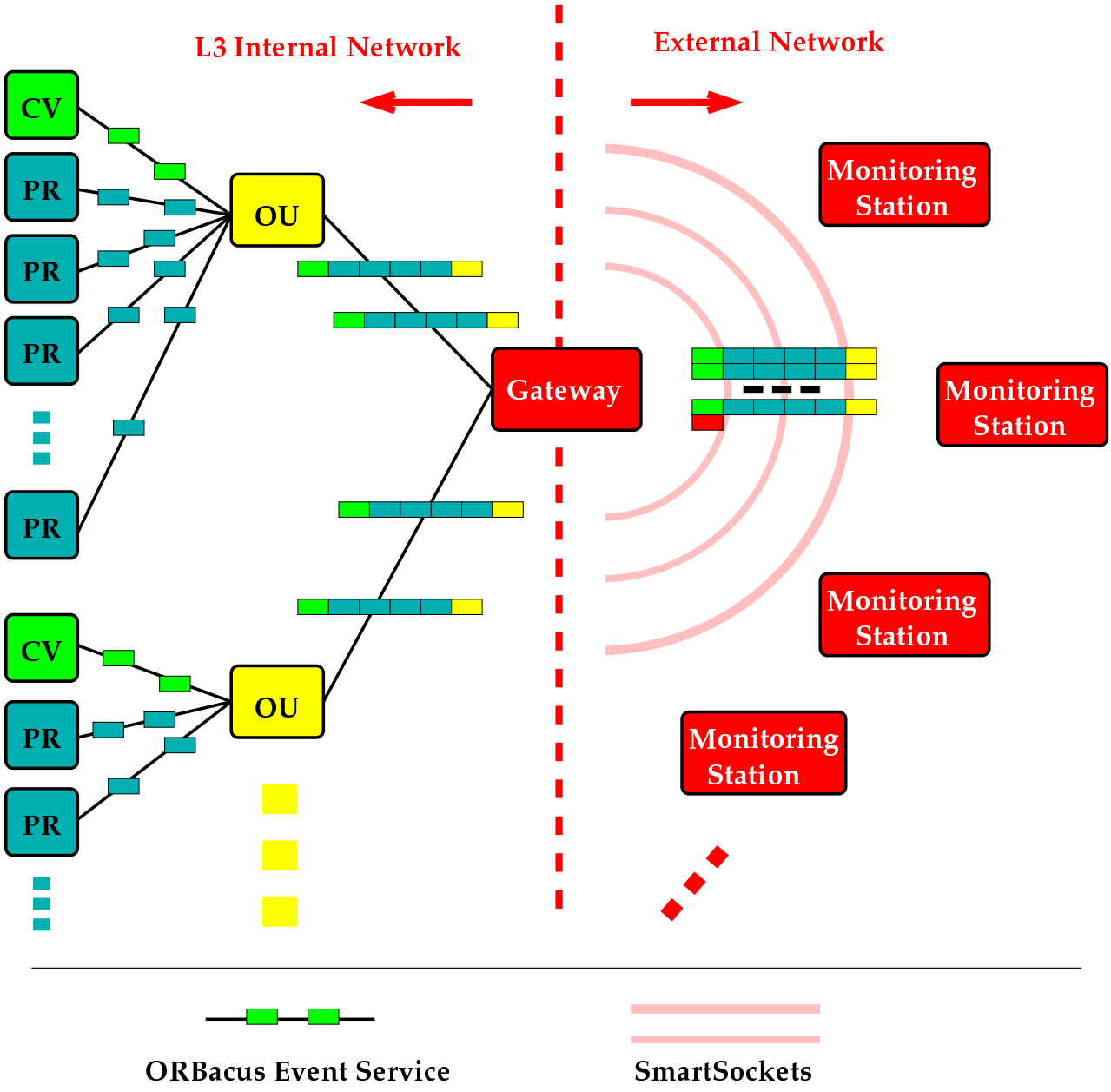
- Level 3 filter statistics (event count, Event Buffer status, etc)
- Operating System information (process count, load average, etc)
- Hardware (CPU temperature, voltages, etc)

Subsystems: Routine Monitoring and Expert Monitoring

Implementation:

- Singleton requests using Relay Mechanism for Expert Monitoring
- Periodic publishing using CORBA IIOP and SmartSockets for Routine Monitoring

Level 3 Routine Monitoring



Status and statistics information is periodically pushed to monitoring clients

Summary

- System architecture finalized, all major hardware components are installed and tested
- Connectivity tests with other DAQ subsystems are successful
- Basic functionality of the Control and Monitoring for Level 3 and Event Builder is implemented, based on:
  - ORBs
  - publish-subscribe systems
  - ROOT + Relay mechanism
  - interface to ORACLE database
- Control and Monitoring tested on medium-size systems (30-40 nodes)
- EVB and Level 3 exceed TDR, further improvements in performance are envisioned.

Glossary

- EVB -Event Builder
- VRB - VME Readout Boards
- SCPU - Scanner CPUs in VME crates
- SM - Scanner Manager, VME CPU coordinating SCPUs
- L3 - Level 3
- CV - converter node (PC), receives events from EVB
- PR - processor node (PC), runs filtering software
- OU - output node (PC), sends data to CS/DL
- CS/DL - Consumer Server Data Logger
- TDR - Technical Design Report