ATCA at UIUC

M. Haney, M. Kasten
High Energy Physics
Z. Kalbarczyk, T. Pham, T. Nguyen
Coordinated Science Laboratory

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

UIUC/SLAC Collaboration

- SLAC hardware, and funding
  - UIUC Physics engineer, UIUC Coordinated Sciences grad students
- Goals of the Collaboration:
  - Advance the state-of-the-art of Standard Instrumentation for particle accelerator controls, beam instrumentation and physics experiments
  - Evaluation and adaptation of commercial standards for particle physics use
  - High Availability engineering of instruments and control systems
  - Adaptation of application-specific prototype designs to new and/or more general platforms
  - Development and evaluation of new controls and diagnostics systems for future accelerators and experiments
  - Development and promotion of standards among particle physics research communities
  - Other activities deemed to be mutually beneficial.
- Part of the High Availability Electronics Program for the ILC

Hardware Environment

- Hardware from SLAC
  - Shelf manager
  - 2 Intel Blades
    - Dual Xeon processors
    - Three watchdog timers
    - Redundant/embedded BIOS
    - Hotswappable
  - Switch: ZNYX ZX5000
    - Layer 2 switching and Layer 3 routing
    - 16 ports 10/100/1000 Mbps Ethernet
  - Host PC: server

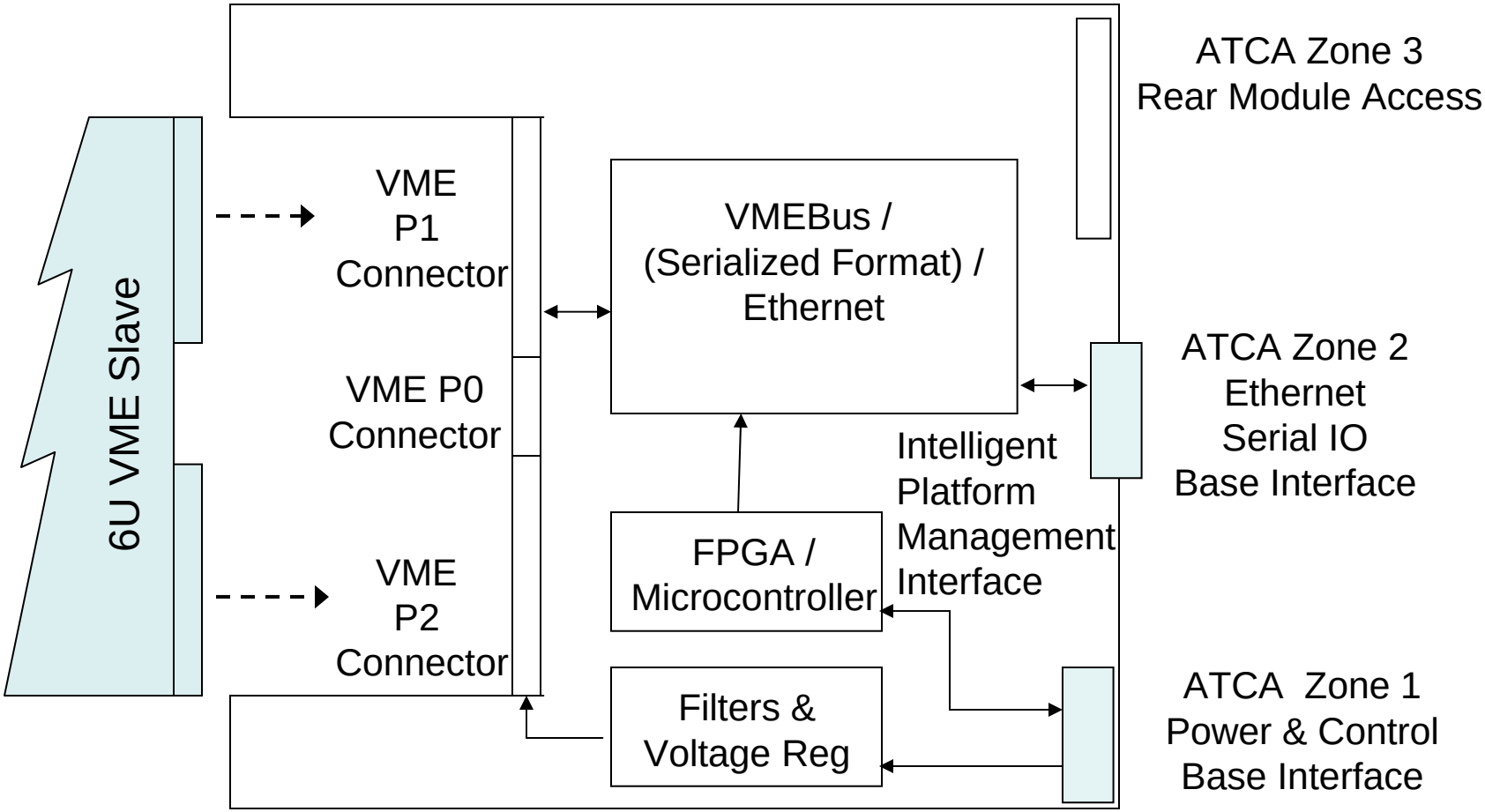
UIUC Physics - Past

- Installed FC5 on blades
  - via CD drive in USB/ATA carrier
- Combined "heartbeat" with Apache
  - automated failover of simple web service
    - Much simpler than EPICS ☺
- Examined
  - RMCP remote management control protocol
    - ipmitools – allows access to Shelf Manager
  - SNMP simple network management protocol
    - Preferred over command line (serial port or telnet), web, or RCMP for controlling the Shelf Manager
- Detailed notes available: http://web.hep.uiuc.edu/Engin/ILC/atca_report/ILC_ATCA_journey%20II.doc

UIUC Physics - Current efforts

- Development of a VME-ATCA adaptor
  - generic ATCA support for 6U VME
    - single (slave) board
  - Key issues
    - VME (master) serialized abstraction
    - Ethernet connection to Base Interface
    - Flexible P2 (user I/O) mapping to Zone 3
    - IPMC microcontroller(s)
    - -48V to VME DC-DC power

VME-ATCA Adaptor Board

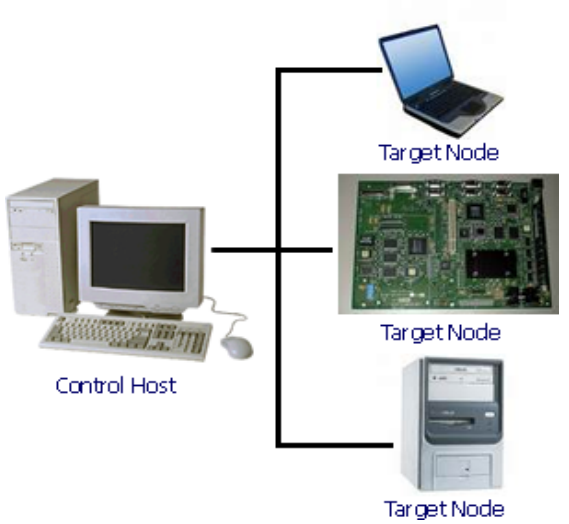


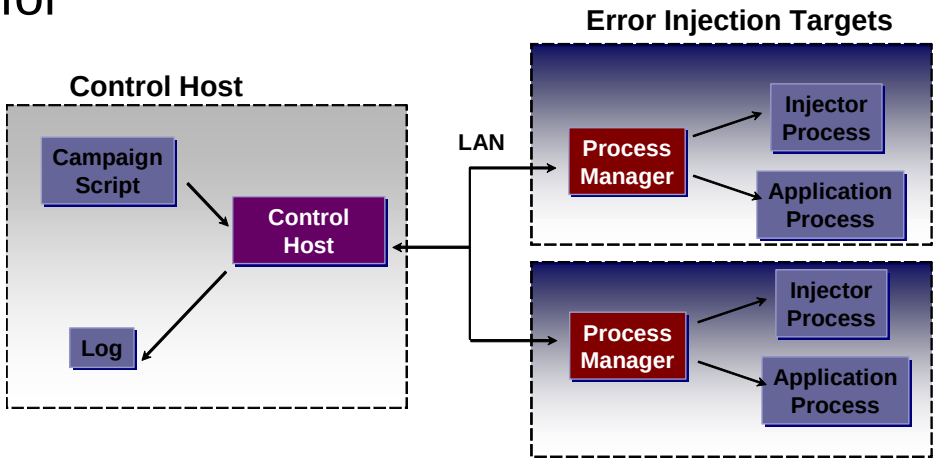
UIUC Coordinated Science Laboratory

- Fault Injection Based Characterization of Shelf Manger
  - Objectives & Approach
    - Characterize failure behavior of Shelf Manager on ATCA platform using automated fault/error injection
    - Faults/errors injected (using NFTAPE) to stress
      - Shelf Manager software
      - Underlying operating system (Linux)
    - Collect and analyze results to
      - characterize system response to failures,
      - identify dependability bottlenecks,
      - propose reliability enhancements

NFTAPE: Networked Fault Tolerance and Performance Evaluator

- Framework for conducting automated fault/error injection based dependability characterization
- Enables user to:
  - specify a fault/error injection plan
  - carry on injection experiments
  - collect the experimental results for analysis
- Enables assessment of dependability metrics including reliability, and coverage



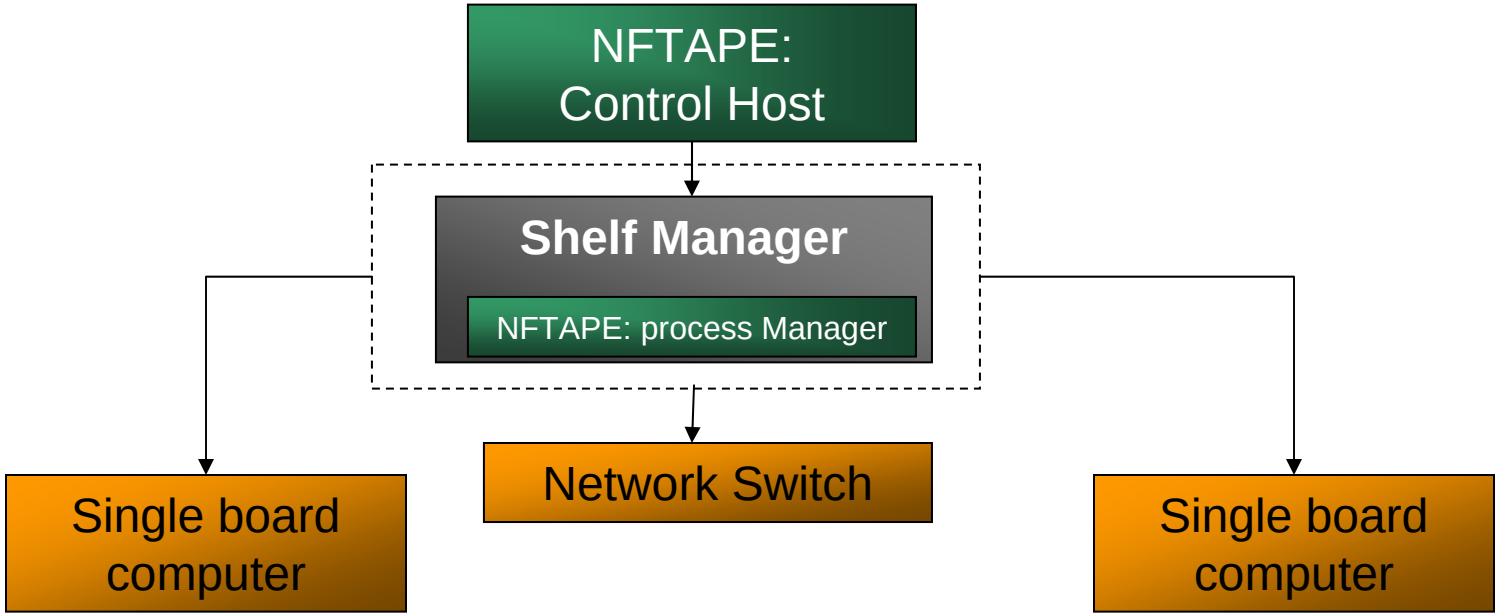


NFTAPE: Control Host & Process Manager

- Control Host
  - Common mechanism to setup and control fault/error injection experiments
  - Processes a Campaign Script, a file that specifies a *state machine* or control flow followed by the control host during the fault injection campaign
- Process Manager
  - Daemon to manage (execution and termination) processes on target nodes
    - processes include: injectors, workloads, applications, monitors
    - all processes are treated the same – as an *abstract process object* – rather than a process of some specific type
  - Facilitates communication between control host and target nodes

Study of Shelf Manager

- Fault/error injection to Shelf Manager – single and multiple bit errors inserted into:
  - User and kernel memory space
  - Text & data segments



Results: Error Activation and Severity

- Around 100K faults/errors injected to user and kernel memory space
- Error activation rate is low (<10%) for random injections in both user space and kernel space
  - The error activation rate increases to over 55% for breakpoint-based injections when targeting most frequently used Linux kernel functions
- About 5% of activated errors in the kernel cause system hang
  - an external intervention (e.g., a watchdog) is required to restore the system operation
- Rather unexpectedly, occasionally, the system (operating system) hangs due to an error in application data
  - This should be prevented

Results: Error Sensitivity

- Error sensitivity (defined as a conditional probability that an error in a given function leads to the system hang, crash, or silent data corruption) of most frequently used functions
  - shelf manager < 25%
  - kernel > 25%
- Silent data corruption
  - Why this is important?
    - Shelf Manager takes actions based on the data obtained from computing nodes
    - Corrupted data can make the shelf manager to take an incorrect decision
  - No error propagation (due to instruction errors) from shelf manager to computing nodes
  - No silent data corruption observed
  - Reasons
    - Inability to detect this type of errors
    - Need to instrument Shelf Manager to enable verification of run time data

Conclusions

- Automated fault/error injection enables failure characterization of computing platforms
  - Error severity and sensitivity
  - Error propagation
  - Availability
- Evaluation of Shelf Manager platform
  - about 5% of activated errors in the kernel cause system hang
  - unexpectedly, the system may hang due to an error in application data
  - direct injections to frequently used application and kernel functions show dramatic increase in the number of hangs.
- Use primary-backup configuration to cope with hangs
  - preliminary fault injection experiments indicate that the primary-backup configuration is still susceptible to hangs
  - comprehensive study required to provide insight into causes of hangs

UIUC CSL - Future Work

- Evaluate chances of errors to propagate from the shelf manager to computing nodes
- Explore development of:
  - software middleware to provide low-cost fault tolerance to applications executing on ATCA platform
    - application/system fail-over
  - OS-level support for providing error detection and recovery
    - application-transparent checkpoint