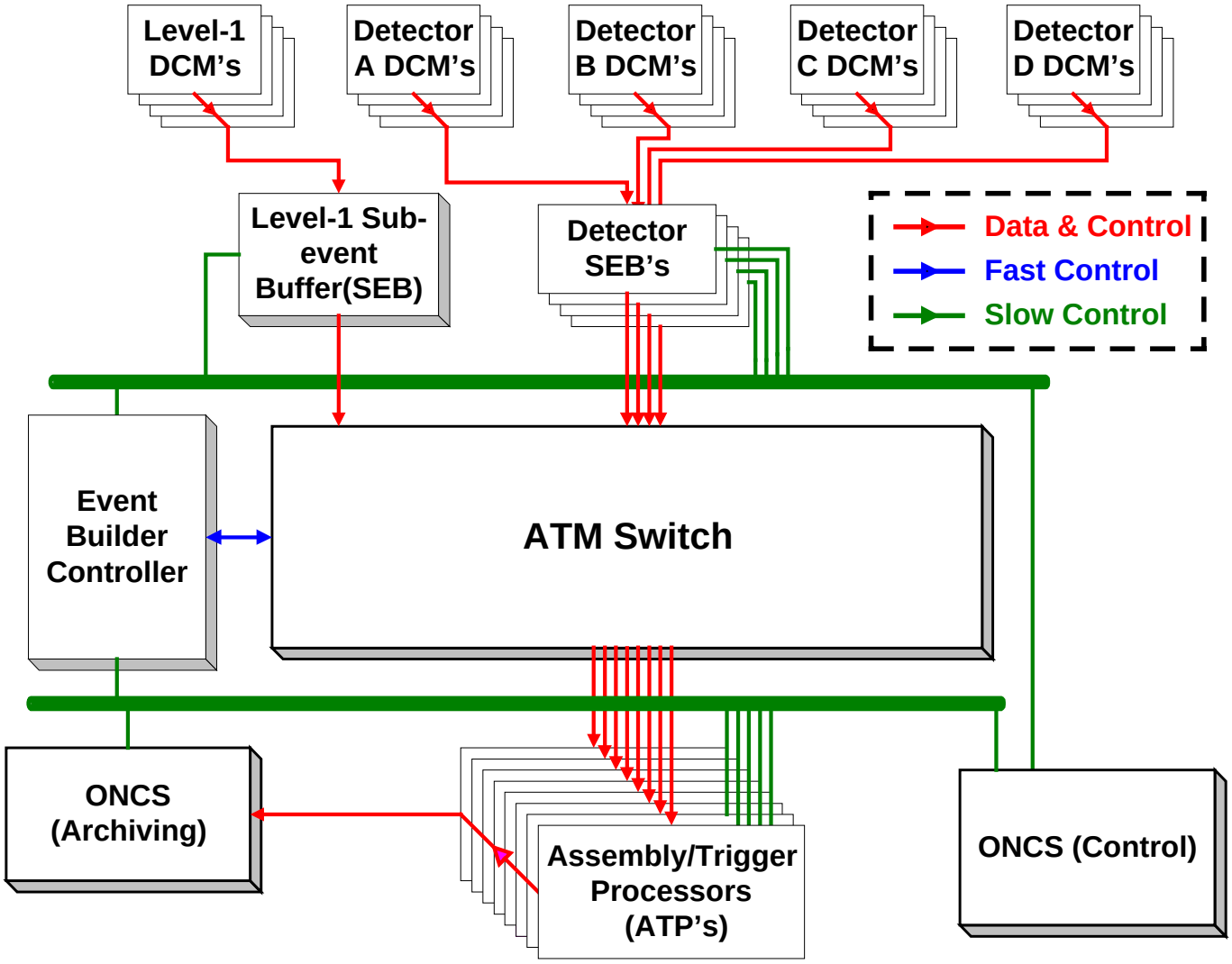
The PHENIX Event Builder
A Progress Report

Brian Cole
Columbia University, Nevis Labs

PHENIX Mini-Collaboration Meeting
Tokyo, Japan
December 7, 1997

1. Overview
2. Event builder design
3. Status
4. Problems/issues

Event Builder Block Diagram



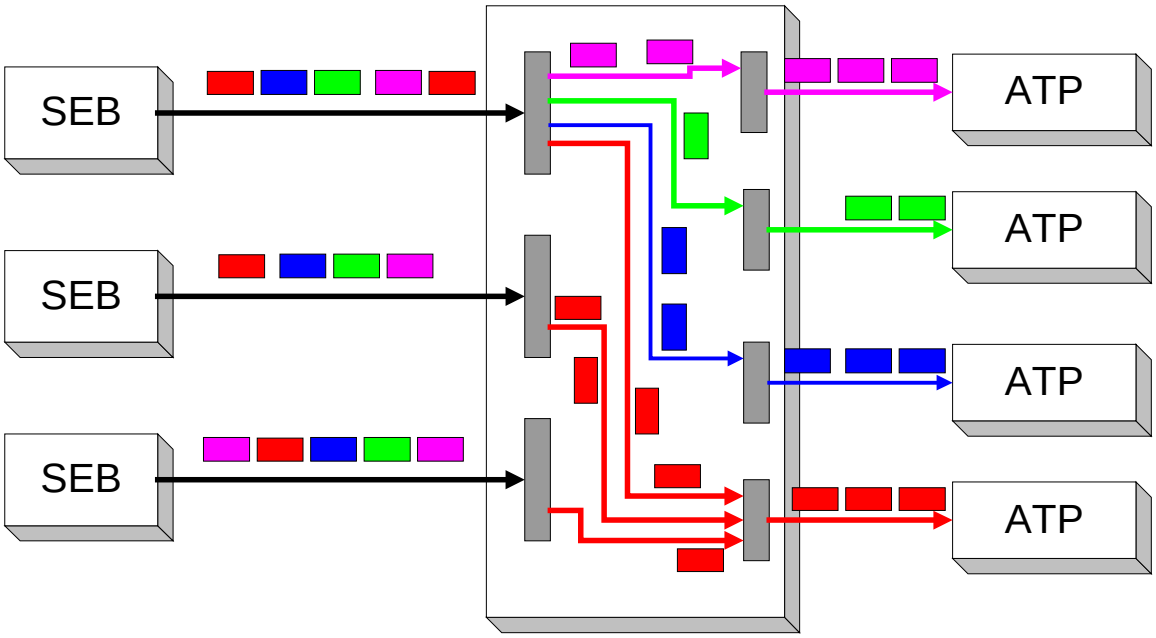
ATM Switch-based Event Builder

- **SEB's:** **Receive, buffer data from DCM's, send data to ATP's.**
- **Controller:** **Allocates destination ATP's based on "availability".**
- **ATP's:** **Pull data, assemble events, run trigger algorithms, send accepted events to ONCS for archiving.**

Event Builder Implementation

Data Flow

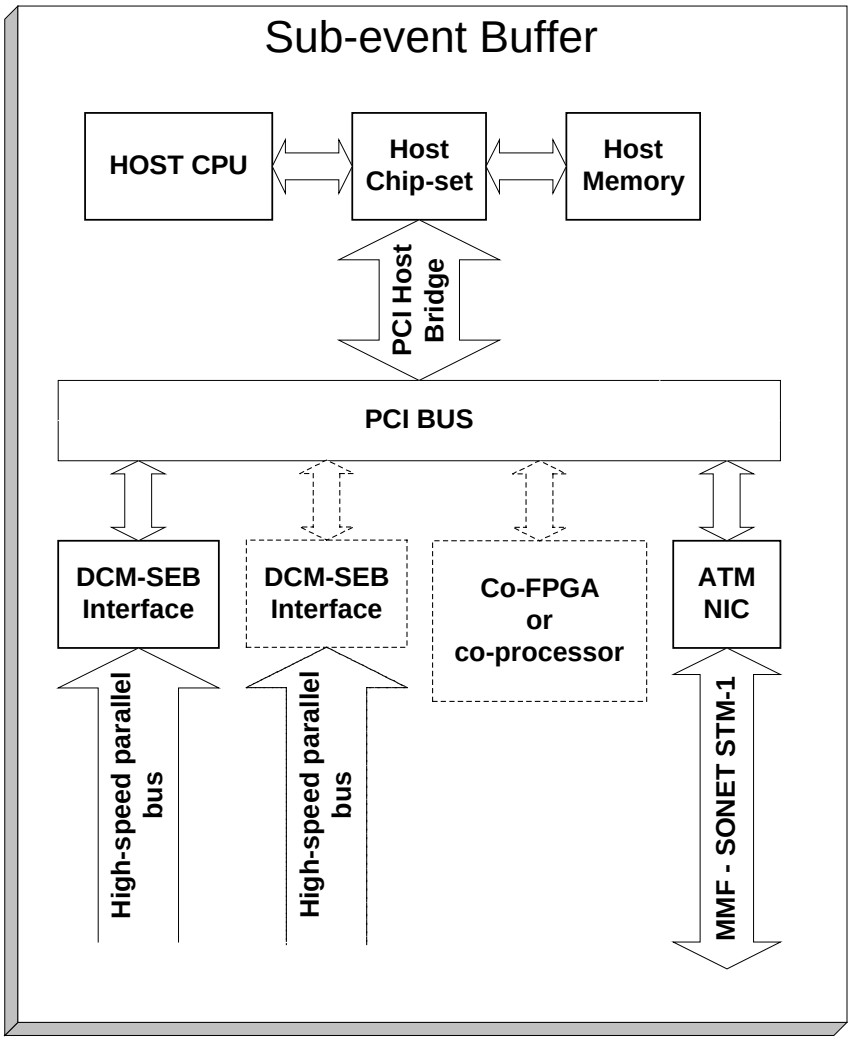
- Data will be "pulled" by ATP's
  - Simplifies flow control, congestion avoidance, event assembly logic, implementation of partitioning
  - Allows partial event building before trigger
- New event signalled by presence of Level-1 data.
- Use ATM AAL5 PVC's for messages, data.
- ACKnowledge, re-send control messages
  - Do not re-send lost data !? (comments ?)
- Use virtual-circuit based source flow control.



Event Builder Hardware

SEB/ATP/Conroller

- "PC" with PCI bus
- ATM network interface card
- (SEB only) DCM-SEB interface card
  - DEC Pammette co-FPGA + custom interface card

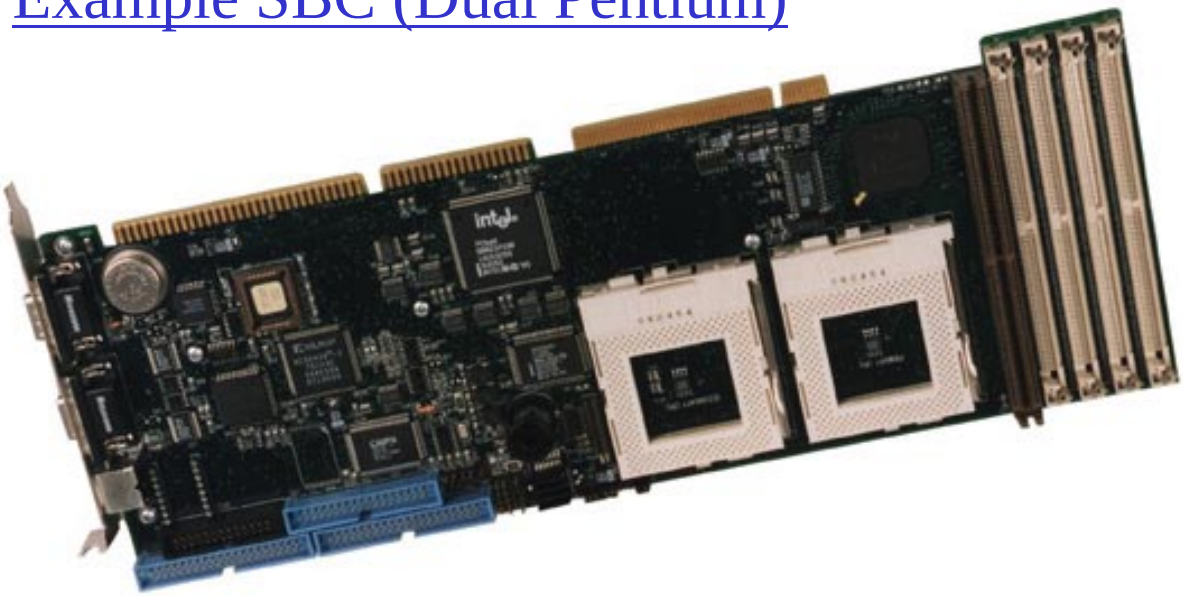


Event Builder Hardware (2)

"PC" Implementation

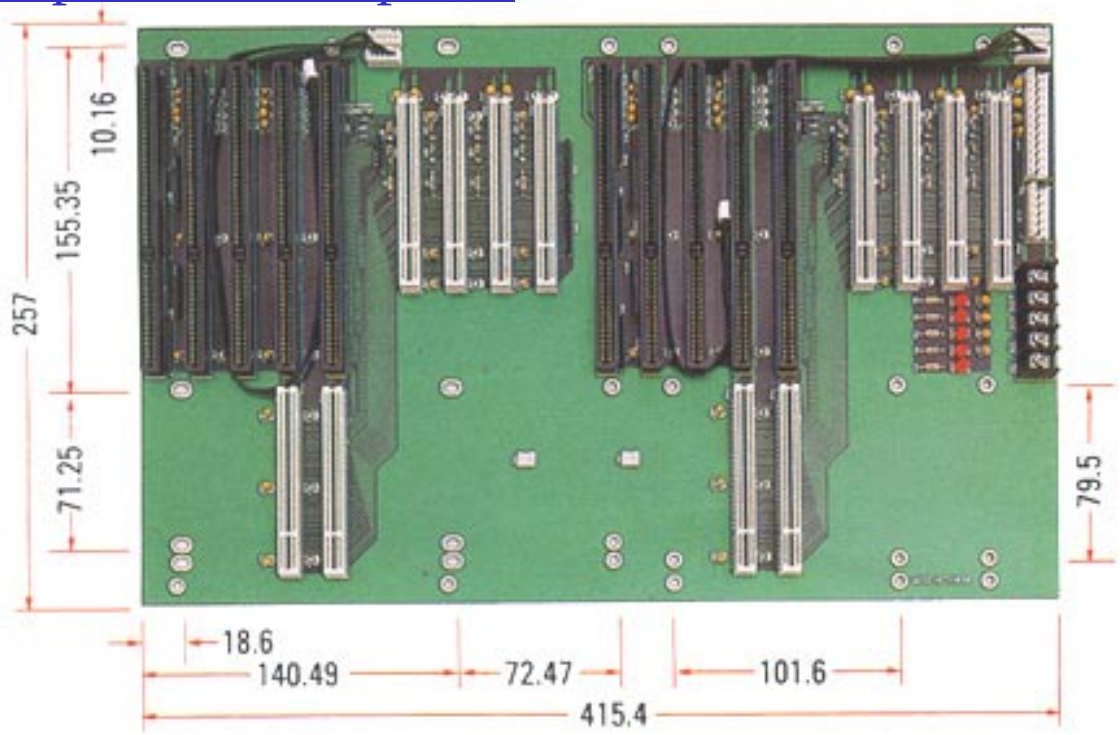
- Currently looking at so-called "Industrial PC's"
  - Mounted in well-designed rack-mount boxes
  - Allow local and external monitoring of PS, fan
  - Provides more robust card support, cooling, ...
- Passive backplane + Single Board Computer
  - allows trivial replacement of processor on failure
  - easy to "upgrade" processor/add additional processors
  - simpler to diagnose problems
  - Allows multiple "systems" per box (passive backplane)

Example SBC (Dual Pentium)



Event Builder Hardware (3)

Example passive backplane



ATM Hardware

- FORE ASX-1000 switch
  - Maximum 64 ports of 136 Mbit/s (data rate).
  - Maximum aggregate throughput of 32*136 Mb/s
  - = 544 Mbyte/s
- *undetermined* 155 Mbit/s network interface card
  - currently testing three such cards
- ATM “network” analyzer
  - not yet chosen

Event Builder Software

Logical "Components"

- Common and identical for SEB, ATP (& Controller)
  - ATM transport
  - Control message handler
  - Monitor/Logger
  - Memory manager
  - Frame handler
  - Database interface
- Common but different versions for SEB, ATP, controller
  - CORBA Control interface
  - Processing loop
  - Thread/process control
- Specific to SEB
  - Pamette interface
  - (Level-1) event initiator
- ATP
  - ONCS data handler interface
  - Level-2 algorithm manager
- Controller
  - Switch configuration interface

Event Builder Software

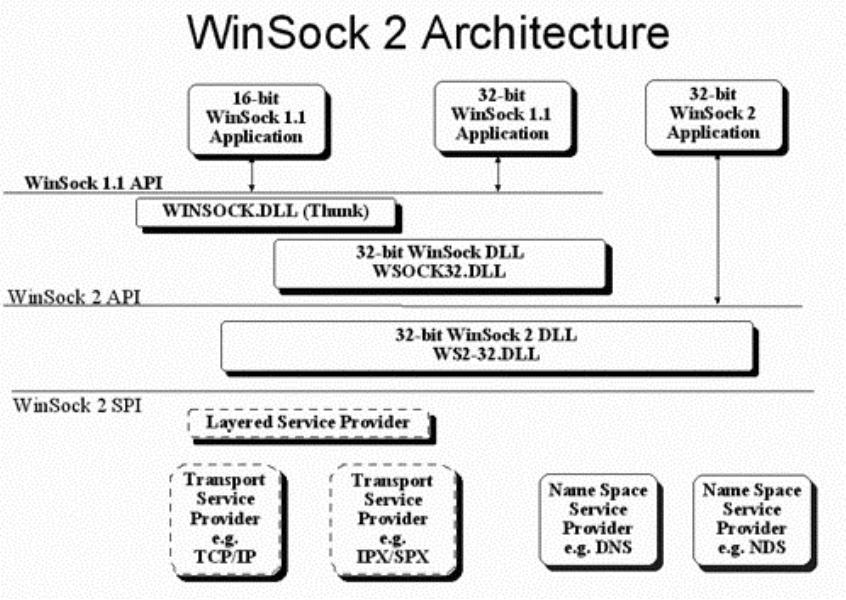
Current Development

- Working on the ATM transport (below).
- Frame handling part of my work on data format.
- Ed and I are “working” on control interface
- **This leaves much to be done !!**

Enter Atlas Demonstrator-C group

- Have a “functional” ATM event builder system
- Has been used to test performance of ATM EvB
  - See Masa’s talk immediately following
- “Missing” pieces
  - Data manager
  - Control interface
  - Monitoring/logging
- Goal
  - Start with Demonstrator-C system
  - Add missing pieces
  - “Replace” the transport.
  - Add DCM interface
  - When/where required adapt remainder to PHENIX’s needs.

ATM on Windows-NT

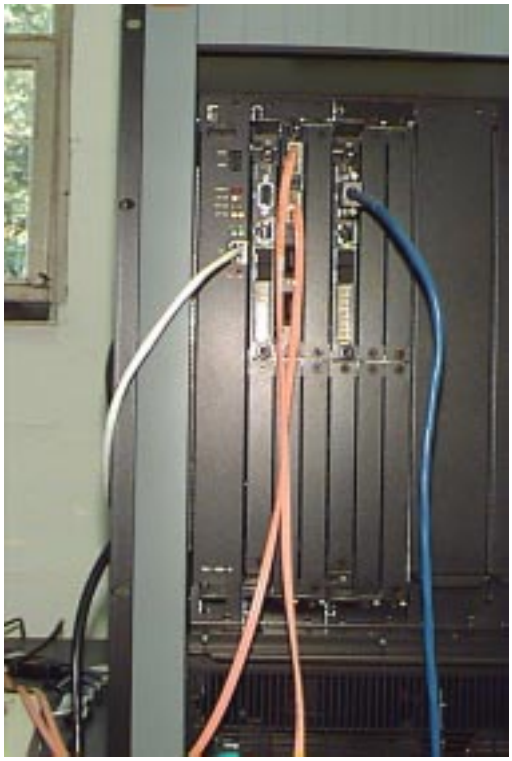


Winsock2

- Socket-based API extended from BSD sockets.
- "Transport independent" API for network I/O
- High-level interface to
  - ATM quality-of-service control
  - ATM "raw" AAL1/ 5 transports
  - ATM signalling
- Explicit support for
  - scatter-gather transfers
  - asynchronous I/O (using NT I/O services)
  - "pre-declaration" of receive buffers for "zero-copy" RX's
- Uniform interface to NIC's from different vendors
  - Allows same **executable** to run on different NIC's

ATM "Prototype" System

Hardware



- FORE ASX-1000 Switch
  - 2.5 Gbit/s switching capacity
    - ⇒ Upgradeable to 10 Gbit/s
  - Currently instrumented with 4 ports
- Network interface cards (155 Mbit/s)
  - FORE
  - Efficient
  - IDT
- PC's
  - Micron 200 MHz Pentium
  - Running Windows-NT 4.0

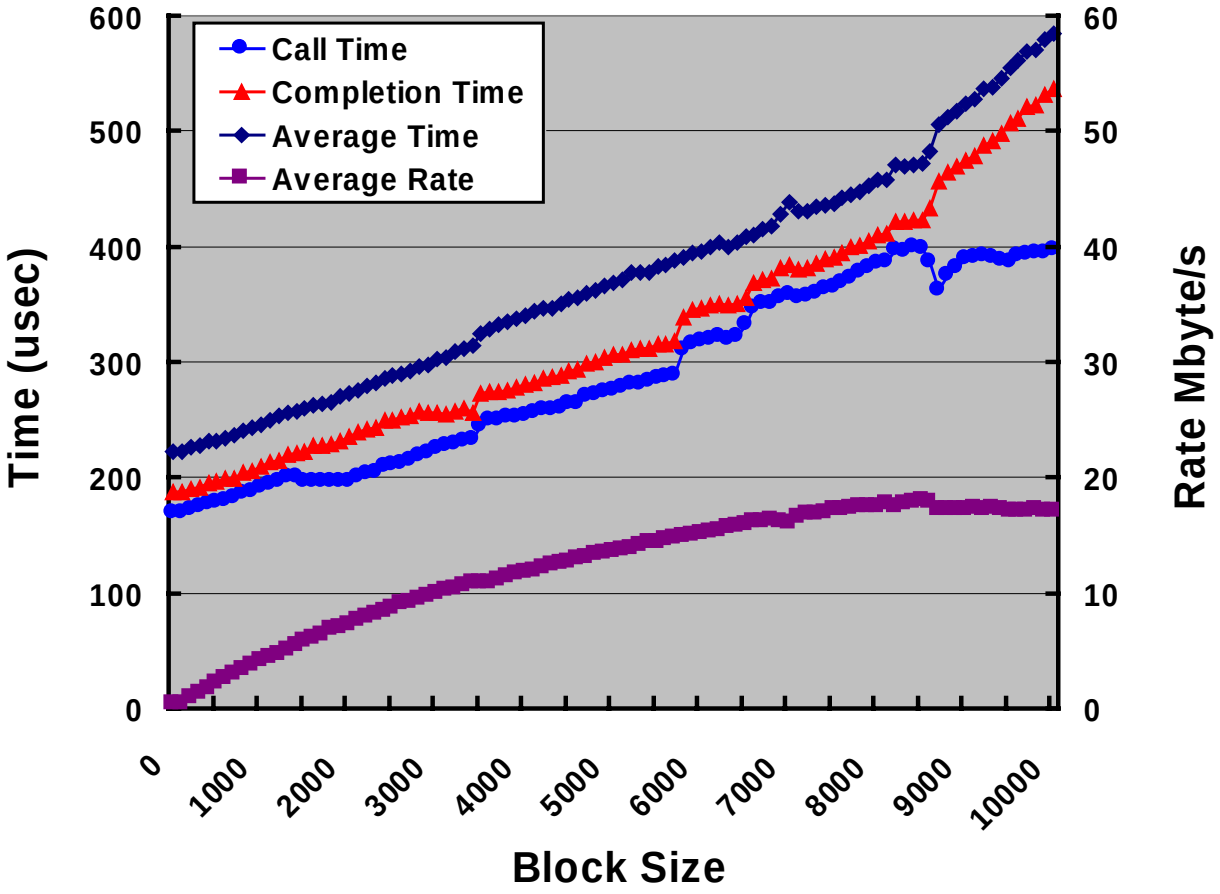
Technical Milestones

- Switch operational, configured with PVC's - July 97
- Single (Fore) NIC loopback tests - July 97
- Communication between PC's - August 97
- Performance tests (FORE NIC) commence - August 97
- Efficient NIC installed, tested - September 97
- NT head-beating - Sepember thru December 97

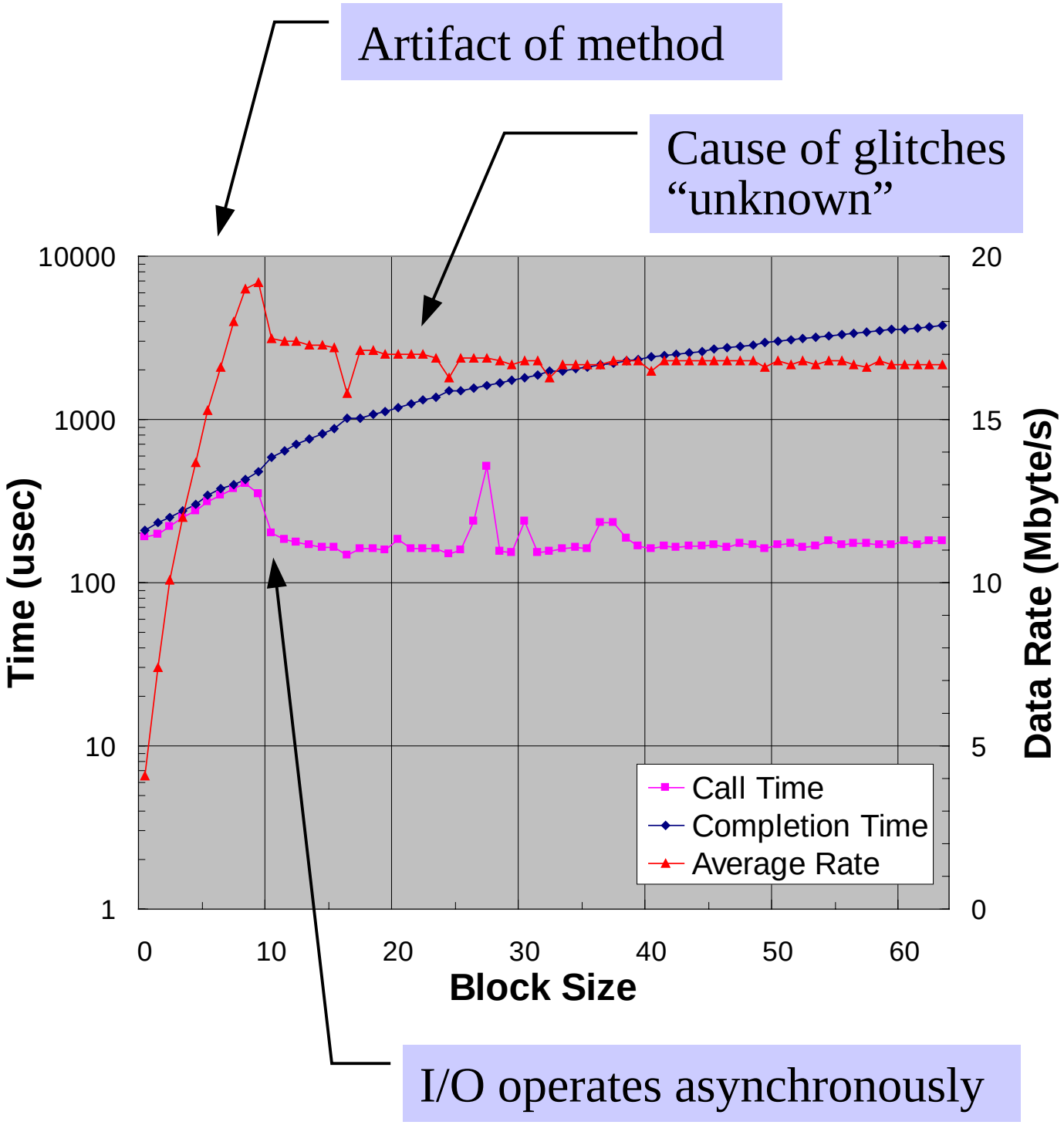
Winsock Performance

Method

- Measure send/receive rates on two different NIC's
- Use Saclay timer routines
  - cross-checked with vendor NIC monitoring programs
- Use asynchronous I/O and measure vs block size:
  - I/O call time
  - time to completion routine
  - average time to execute extended loop



Winsock Performance (2)



Winsock Performance (3)

Good News

- Software can saturate and sustain OC-3 line speed.
- Basically works “out of the box”
- No fussing with PCI issues (yet)
- Results are consistent for 2 NICs+Winsock SPI’s

Bad News

- overhead of 170 μs to every transfer.
  - Max TRX rate of 5 KHz
- Asynchronous I/O shows “odd” NT behavior.
- Results are consistent for 2 NICs+Winsock SPI’s

Future

- Existing SPI’s based on Intel example Microsoft will provide “native ATM” Winsock SPI in NT5.0.
- Is this good news or bad news ?

Conclusion

- Performance sufficient for ATP’s
- Performance sufficient for SEB’s in first few years.
- Forge ahead using Winsock for ATM transport.
- Let commercial market push the technology (??).
- Plan for possible need to switch ATM API in future.
- Plan for possible need to switch OS in the future.

Problems/Issues

Manpower, Manpower, Manpower,

- Serious !! Manpower shortage @ Nevis
  - Most ATM work done by myself with some student help.
  - Finally resolved with hiring of two post-docs @ Nevis
    - ⇒ Will start in Jan, Feb 98
- Rd-31/Atlas Level-2 Demonstrator-C
  - Have continued to meet (June, September @ RT97)
  - Still no formal arrangement with Saclay
  - "New" addition to collaboration Osaka University
    - ⇒ More from Masaharu following my talk
- Georgia State
  - We're badly in need of Xiaochun's effort/experience
  - Need to help get his program going!!

Windows-NT

- Starting to get concerned about NT I/O performance
  - e.g. overhead discussed above wrt ATM
- May need to consider switching to (e.g.) Lynx-OS
  - In the future after we have a "functional" system
- Issue may be be resolved in NT5.0 ??
- Faster/dual processors will also help.

Participants

Columbia/Nevis

- 1 Faculty, 2 post-docs (New - starting Jan 98), 1 student
- Responsibilities
  - Overall management
  - Design
  - ATM communication
  - Hardware/Software testing/evaluation
  - SEB, ATP, Controller software development

BNL

- 1 Staff + (?), 1 student
- Responsibilities
  - DCM - SEB interface
  - SEB software development

Georgia State

- 1 Faculty + 1/2 staff
  - Trigger support software
  - ATP software development

Osaka/Nagasaki

- 1+ ? Staff, 1 student - starting late spring 97
- Responsibilities
  - unknown

Saclay

- No “direct” participation but large indirect contributions.