VISA: Netstation's Virtual Internet SCSI Adapter

Rodney Van Meter

USC/Information Sciences Institute

rdv@isi.edu

http://www.isi.edu/netstation/

July 2, 1997

Talk Outline

- Netstation
- STORM & Derived Virtual Devices
- IP for NAPs
- VISA
- Conclusion

The Netstation Project

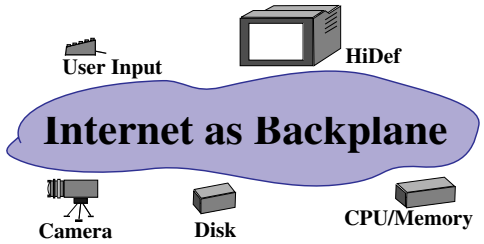
Gregory Finn (project leader), Rodney Van Meter, Steve Hotz,
(Bruce Parham and Reza Rejaie)

Objective:

Overcome fixed bus-induced limitations by utilizing improved scaling properties offered by gigabit networking.

Netstation

Netstation is a system composed of **network-attached peripherals (NAPs)** created by replacing the system bus in a workstation with a gigabit network.



Based on ATOMIC 640 Mbps switched network,
use Internet protocols for ubiquitous device access.

Why Netstation?

- Traditional buses don't scale in distance or bandwidth.
- Support efficient device-to-device transfer without consuming resources at main CPU.
  - ❏ e.g., incoming video data direct to display.
- Construct systems flexibly.

Netstation Problems

Closed, bus-centric architecture allows simplifying assumptions about resource identification, security and sharing.

- Set of resources not constrained by architecture.
- Control of devices not limited to bus master.
- Non-dedicated network.
- Security now paramount.

Accomplishments

- DTP: 30,000 RPCs/sec
- Network-Attached Peripheral (NAP) Security Model: Derived Virtual Devices (DVDs)
- Netstation Display
- X on Netstation Display
- Zero-Pass Checksumming
- Netstation Keyboard
- ZCAV Disk Work

Current Work

- DVD Implementation w/ Kerberos
- IP Disk
- VISA: Virtual Internet SCSI Adapter
- STORM: A DVD File System
- Third Party Transfer
- Netstation Camera
- More Network Protocols (mostly TCP)

Related Work

- MIT Viewstation
- Cambridge Desk Area Network
- SGI Origin 2000?
- CMU Network-Attached Secure Disk (NASD)
- LLNL's Network-Attached Peripheral (NAP) RAID
- National Storage Industry Consortium's NASD Committee
- Fibre Channel Disk Drives
- Palladio at HP Labs

Talk Outline

- Netstation
- **STORM & Derived Virtual Devices**
- IP for NAPs
- VISA
- Conclusion

Derived Virtual Devices

A **derived virtual device** (DVD) is an *execution context* at a network virtual device (NVD); i.e. a set of resources and procedures to access them.

DVD concept provides a mechanism to support
safe sharing of resources.

- Enforces resource bounds checking.
- Constrains operation functionality (e.g., read only).
- Checks authentication of **user**.

Who a request is from is **much** more important than *where*.

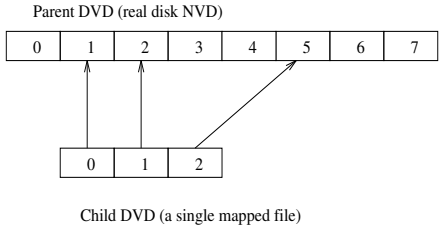
STORM: A DVD File System

STORM (STORage Manager) is the file system code, running at the Netstation CPU. Its purpose is to mediate **block-level** access to the Netstation disk drives by creating DVDs for clients.

- STORM accepts file open requests from clients,
  does protection checks and file mapping to blocks.
- STORM (STORage Manager) creates a DVD for client access to a file, mapping only the file blocks.
- Client then accesses device directly.
- STORM doesn't cache file pages for DVD-mapped files.

DVD File Mapping

A DVD can be defined to map the blocks of a file, which the client can then treat like a raw device.



This is how resource protection boundaries are enforced.

DVD Conclusions

Derived virtual devices are safe sharing. They offer safe, transferable, low-level access to Internet-attached devices. This supports a variety of device types and makes third-party device-to-device more efficient.

- DVDs enforce policy established by parent's owner.
- Direct access to device supported for efficiency.
- DVD creation is expensive, but subsequent access is both cheap and safe.
- Recursion simplifies support for third-party transfer.

Talk Outline

- Netstation
- STORM & Derived Virtual Devices
- **IP for NAPs**
- VISA
- Conclusion

Networking Problems for NAPs

HiPPI-6400❋gigabit Ethernet❋Myrinet❋FC-AL

as I/O Nets Get Larger and More Complex:

- Media Bridging
  (Routing, Addressing)
- Congestion
- Flow Control
- Demultiplexing @ Endpoints
  (Destination Addr Calculation, Control/Data Sifting, UL Protocols)
- Latency Variation
- Security
- Reliability
- Heterogeneity
  (Hosts, Traffic Types, Nets)

All Become Bigger Problems!
But...

1394❋HiPPI-800❋ATM❋SSA❋Fibre Channel

The Internet Community Has Solved Most of the Problems

- Strengths of IP: Issues of Scale and Heterogeneity
- Weakness: Performance

Advantages of IP

- Heterogeneous Interconnects
  - Intra-Machine Room
- Wide-Area Access
  - Enables Remote Mirroring and Backups
- Future Growth
  - Not Media-Specific
- Lower R&D Investment in Networking

Solving TCP/IP Performance Problems

Protocols

- Larger Packets (IPv6 MTU discovery)
- Zero-Pass Checksumming

Host Implementation

- Zero-Copy TCP (IPv6 Flow IDs)
- Early Demultiplexing

Device Controller

- Link & CPU Speeds Climbing Faster than Device Transfer Rate
- Implementation Can be Simple
- Scatter-Gather Real Memory Interface

Transport Layer Issues

- Want to Retain TCP's Reliability and Flow Control
- Need Application Framing

Application Layer Issues

- RPC Formatting
- App-Directed Out-of-Order Delivery

Conclusions

- IP Offers Significant Benefits with Little Cost
- Some Transport Issues are Still Open

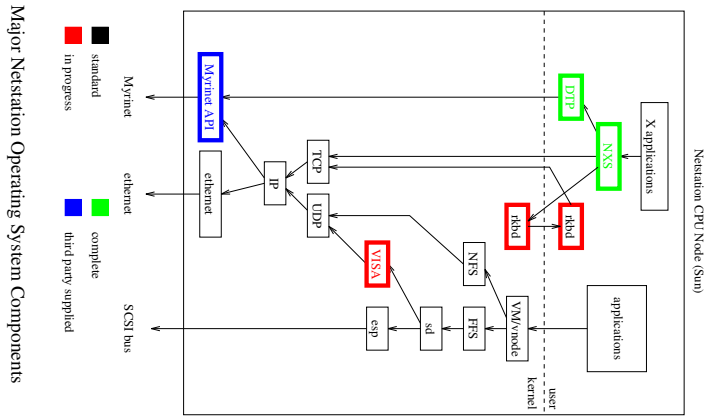
Talk Outline

- Netstation
- STORM & Derived Virtual Devices
- IP for NAPs
- **VISA**
- Conclusion

VISA: Virtual Internet SCSI Adapter

Goals:

- Demonstrate IP Acceptable for Peripherals
  - •Device Performance
  - •CPU Load
- Single Host to Device
- Platform for:
  - •Further NAP Protocol Research
  - •DVD/STORM
  - •Security
  - •Third Party Transfer
- Proof of Concept, not Production



Architecture

- Sends SCSI RPCs, Receives Data via Network
- Accesses IPdisk
- SunOS 4.1.3 scsi_transport Layer
- Simple Reliability over UDP
- Single-threaded Pseudo-process
- Prefetch/Consolidation Handled in FS Code Above
- Single Command per Target -- No Command Queueing
- Meshes with SCSI-3

IPdisk

- Emulates Disk NAP as User Process
- Three Types of Store:
  - •RAM (done)
  - •SCSI Disk (coming soon)
  - •SAM Solid State Disk (later)
- UDP (simple reliability) or TCP
- Third-party SCSI COPY w/ DVDs Planned

Transport Protocol

- Simplest Possible Reliability over UDP
- Fixed Size (Negotiated?):
  - •Packet Size (4KB)
  - •Window (32KB)
- Handles Errors, but not Efficiently
- Assumptions:
  - •Low Latency LAN
  - •In-Order Arrival
  - •Highly Reliable

Early Results

- 60 Mbps Read & Write Through File System
  (Sparc 20/71, Myrinet, 4KB pkts, 32KB Window)
- Currently Limited by:
  - •CPU at IPdisk
  - •Brain-dead Reliability & Limited Buffering
- Compares to:
  - •60 Mbps NFS
  - •90 Mbps TCP
  - •150 Mbps UDP Blast (8KB pkts)
- Requests up to 128KB Seen

Comparison to SCSI Bus

- Sun 4 ~1991: >75 Mbps SCSI Raw Device
- LOTS More Interrupts
- Lower Channel Efficiency
- SCSI Coprocessors Very Effective

Lessons Learned

- SunOS Layered/OO Modularity Made VISA Possible
- SCSI Configuration Happens EXTREMELY Early in Boot:
  - •No Timers, No Mbufs, No Networking
  - •Fake Device Config
  - •Kernel Rework Necessary to Correctly Identify Devices
- Packet Size Important (as Expected)
- CPU Load Significant Due to:
  - •Packet Overhead
  - •Extra Data Copies
  - •Underpowered/Underutilized Coprocessors

Future VISA Work

- Transport Protocol:
  - •TCP
  - •Better Custom
  - •Preferring Framing/ACK Patterns
  - •Acceptable Assumptions
- Performance Measurement:
  - •CPU Utilization
  - •Macro FS Effects (File Create Time, Seeks, etc.)
  - •Paging & Raw Disk Performance
  - •Comparison of Same Disk Locally & via IP
  - •Host Saturation Point
  - •SSD

Future VISA Work(2)

- Clean Up (Multi-Device Support, etc.)
- DVD Integration
- Test w/ Other Device Types (Tape Drive?)
- Fast Demultiplexing/Copy Reduction

Future Netstation Work

- STORM
- Camera
- 3rd-Party Transfer
- Kerberos Integration

Conclusions

- Netstation: Exploring Space of Network-Based Architecture
- Derived Virtual Devices (DVD)s Provide Safe Sharing of Network-Attached Peripherals

- Virtual Internet SCSI Adapter (VISA) Working, Results Pending
- Assertion that IP for NAPs is:
  - •Possible -- Done
  - •Appropriate -- not yet Proven

- **`http://www.isi.edu/netstation/`**

Supporting Slides

Netstation Components

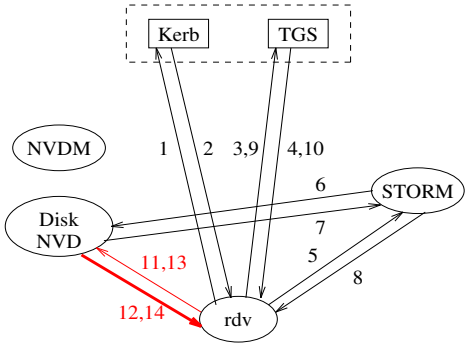
- NVDM: NVD Manager
  - ❑ Database provides configuration info.
- NVDs: Network Virtual Devices (physical resources)
  - ❑ NVD is named resource.
- DVDs: Derived Virtual Devices
- Access Procedures
- Network Protocols

History

- Conceived in 1991
- Grew out of ATOMIC (now Myrinet) 640 Mbps LAN work
- ...which grew out of the MOSAIC multiprocessor at Caltech
- Separately funded by ARPA since Oct. ‘94

DVD File Read

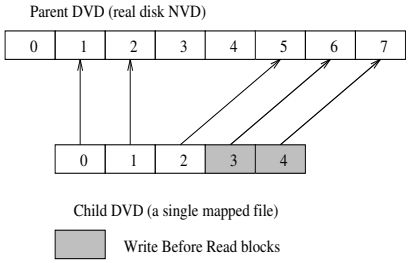
Once STORM establishes DVD for rdv, data requests go directly from rdv to the device (11-12, 13-14).



- **1-4 rdv gets ticket to access STORM**
- **5 requests DVD**
- **6 STORM makes DVD for rdv**
- **11 first data request**
- **13 second data request**

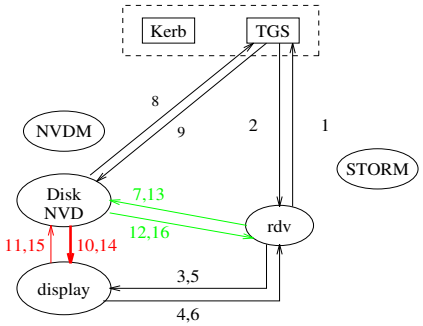
Write Before Read (WBR)

Example optimization: DVDs enable marking of blocks as **write before read**, eliminating erase while staying safe.



DVD Third-Party Transfer

Third-party transfer from disk to display.



- **5 rdv creates DVD at display for disk**
- **7 rdv tells disk to send data to display**
- **10 data transfer**
- **13 second request**
- **14 data transfer**

DVD 3rd party transfer uses minimal msgs on later iterations (four per request).

Implementation Status and Issues

- DVD definitions are downloaded Scheme (Lisp) code.
- Display working with non-recursive DVDs and custom X server.
- STORM file system and disk device in early prototyping.
- Authentication (Kerberos) not yet incorporated.

Block v. File Servers

Should a network-attached disk drive present a file system interface (e.g. NFS) or a block device interface (e.g. SCSI)?

We believe a low-level block-oriented interface is appropriate under many circumstances:

- Non-file system uses: striping, RAID, HSM cache, etc.
- NFS semantics “wrong” for other FS types: naming, directory semantics, commit semantics, etc. differ for log FSes & other different block allocation, non-Unix FS, etc.
- Low-level operations both necessary and promote efficiency: eject, format, erase, rotational positional allocation, partial block handling, etc.