



Logistical Networking

Micah Beck, *Research Assoc. Professor*
Director, Logistical Computing &
Internetworking (LoCI) Lab

Computer Science Department
University of Tennessee

SInRG Workshop

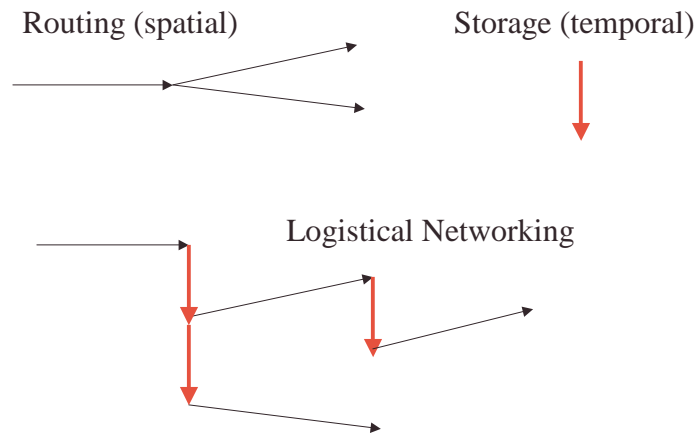
Jan 23, 2002



Logistical Computing & Internetworking

- Principle Investigators
 - Micah Beck
 - James S. Plank
 - Jack Dongarra
 - Rich Wolksi (UCSB)
- Research Staff
 - Alex Bassi
 - Henri Casanova (UCSD)
 - Jeremy Millar
 - Terry Moore
 - Graziano Obertelli (UCSB)
 - Stephen Soltesz
 - Martin Swany (UCSB)
- Graduate Students
 - Scott Atchley
 - Erica Fuentes
 - Ding Jin
 - Wo Ling
 - Sharmila Kancherla
 - Xiang Li
 - Sathish S. Vadhiyar
 - Yong Zheng
 - Luke Zhou
- Funding: DoE, NSF

Dimensions in Communication



Why “Logistical” Networking

- Analogy to military or industrial logistics
- Distributing goods requires not only highways but also warehouses (depots)
- Data transmission is the highway
- Storage servers are the depots
- “Scalably sharable” network storage
- Enabling new users and applications



Internet Backplane Protocol (IBP)

- Servers that make allocation of primitive “byte arrays” available to clients
- Byte arrays are not blocks (more abstract)
 - Network capabilities (primitive security)
 - Variable extents
- Byte arrays are not files (weaker semantics)
 - Size & duration are limited
 - “Volatile” allocations
 - Best effort reliability and availability
 - No directory structure, accounting
 - No caching, replication
- **An end-to-end approach to network storage**



IBP Software Structure

- IBP Depots (servers) are daemons that serve local storage to IBP clients.
- IBP clients only link to a protocol library.
- Clients talk to depots using TCP/IP.
 - Alternatives are being investigated
- Design is for high-performance/scalability.



Allocation Attributes

- Permanent vs. Time-Limited
- Volatile vs. Stable
- Read/Write semantics:
 - Byte Array
 - Pipe
 - Circular Queue
 - Completely Destructive Write



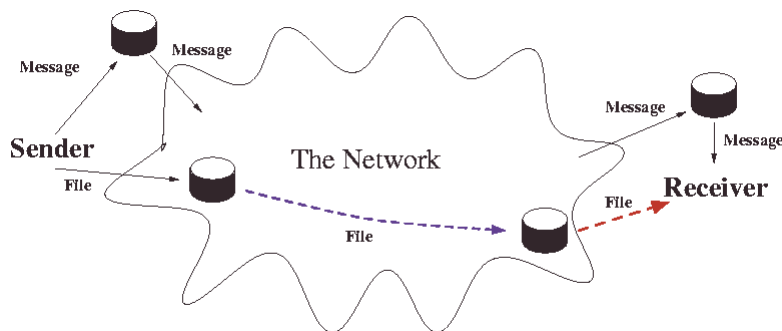
Illustration: IBP Mail

- The Problem: How to mail huge files?
- Possible solutions:
 - Tape and Federal Express
 - SMTP and MIME
 - SMTP and HTTP/FTP
- There are problems with every solution.

The IBP Solution

- Store the file at an IBP depot near the sender.
- Send capability with the mail message.
- Copy the file from the IBP depot near the sender to an IBP depot near the receiver.
- The receiver gets the file from the IBP depot.

IBP Mail





Data Movement & Multimedia

- Moving data is complementary to storing it
- IBP API must support advanced depot-to-depot data movement
- **mcopy**: point-to-multipoint copy
- Overlay networking
- data mover plug-ins
 - optimized TCP
 - parallel TCP (GridFTP)
 - UDP-based protocols
 - IP multicast
 - etc.
- Can we stream MM using only IBP?



Other Application Areas

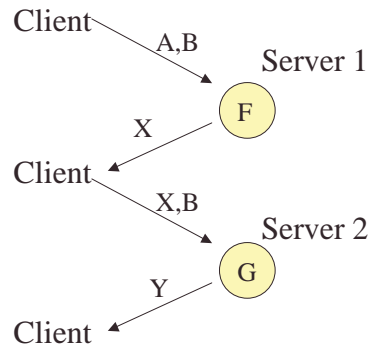
- Source routing
- Bandwidth adaption
- Reducing (BW×delay)
- Reliable multicast
- Content Distribution
- Remote access to structured data
- Managing computation state (NetSolve caching)
- Temporary storage
- Very large data sets
- Collaborative computing & visualization

State Management in NetSolve

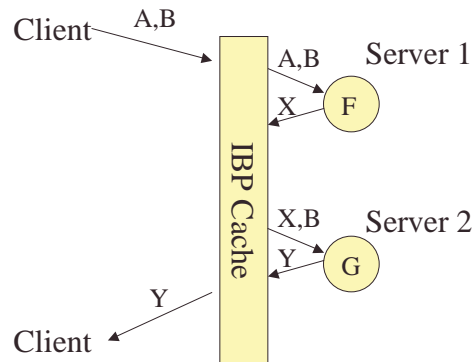
- The Problem: NetSolve calls are functional
- Excessive data transfers

For example:

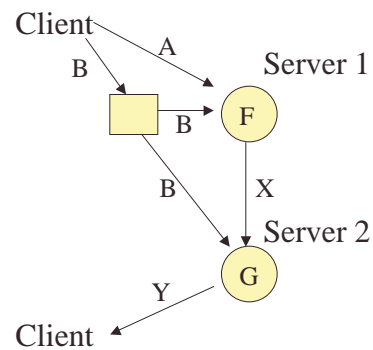
```
X = F(A, B);
Y = G(X, B);
```



Caching



Dependence Flow





An Experiment Using NetSolve

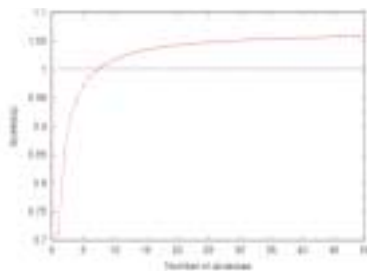
- NetSolve Client at UC San Diego
- Computational Servers at UT Knoxville
- MA28 solver library used to solve systems of equations from the Harwell-Boeing collection of the Matrix Market repository
- Uncached to client-directed caching



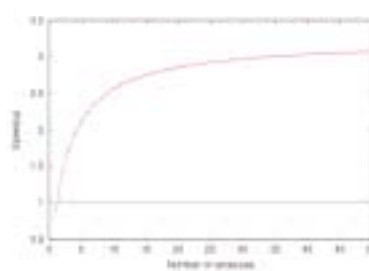
Preliminary Results

- Unenhanced NetSolve vs.
NetSolve w/IBP caching

16.1 KB



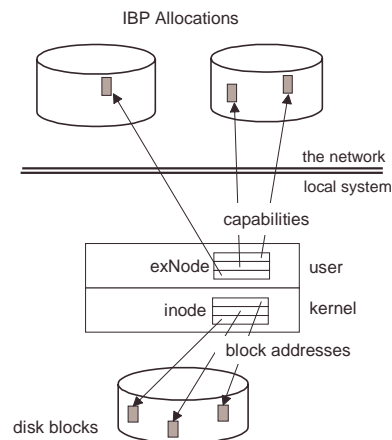
2.68 MB



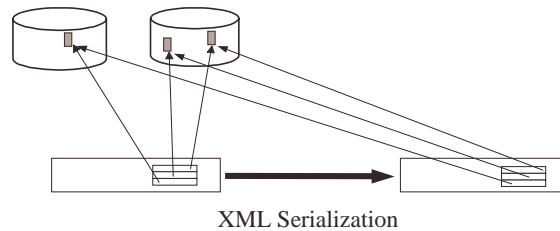
Building on IBP

- Many applications assume file semantics
 - Unbounded size & duration
 - High reliability & availability
 - Caching & replication
- In a layered architecture, these are implemented through aggregation and additional intelligence at the next level

ExNode vs inode



ExNode Mobility



The exNode serialization is a portable soft link

Building a Logistical Operating Environment

- Resource Discovery
 - Logistical Backbone registry (LBone: Plank)
 - Can also use DNS, could use MDS
- Programming Tools
 - exNode library, primitive commands
 - policy modules (replication, fragmentation)
 - cool applications (SC demo: IBPster MP3)



IBPster Demo

Presenter: Stephen Soltesz
Research Associate
Computer Science Dept.