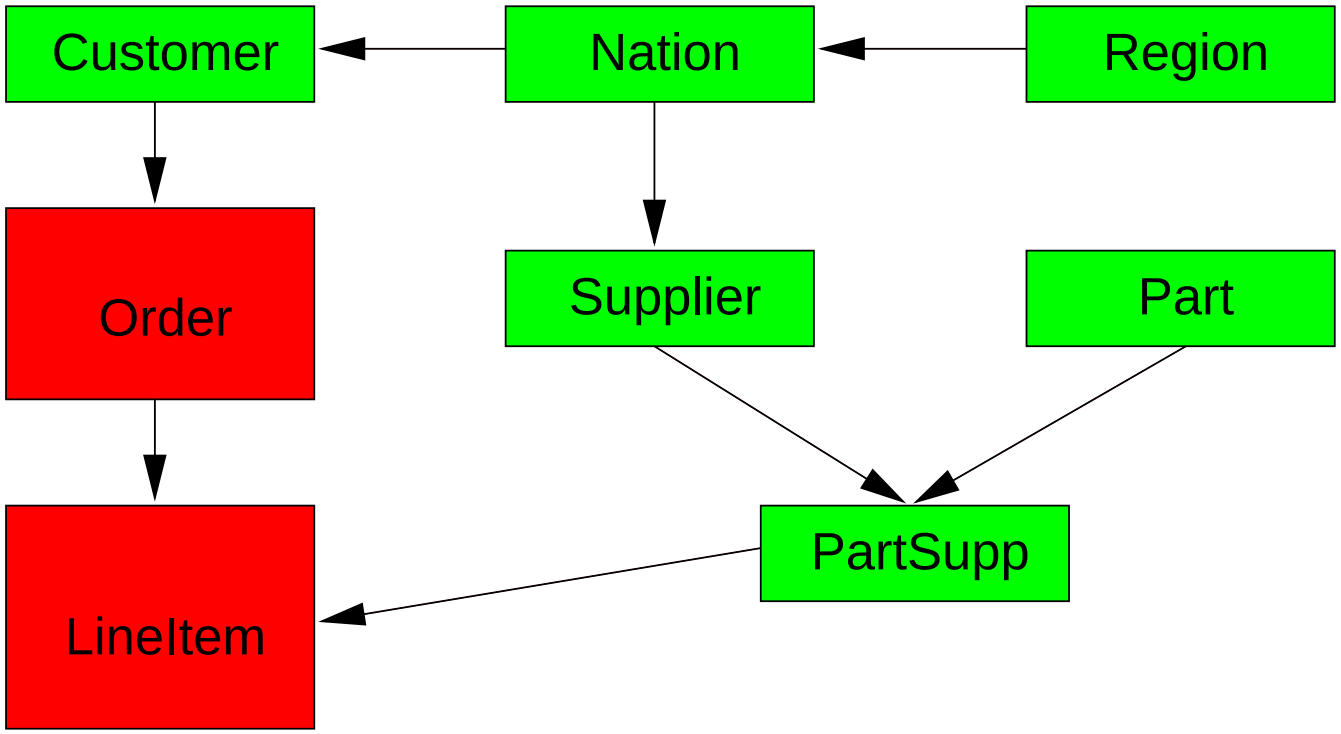
OS Support for a Commodity Database on PC Clusters — Distributed Devices vs. Distributed File Systems

Felix Rauch (National ICT Australia)
Thomas M. Stricker (Google Inc., USA)
Laboratory for Computer Systems,
ETH Zurich, Switzerland

Australian Government
Department of Communications, Information Technology and the Arts
Australian Research Council

NICTA Members
THE AUSTRALIAN NATIONAL UNIVERSITY
UNSW
Department of State and Regional Development
First for Business
BusinessACT
ACT GOVERNMENT
Victoria
The Place To Be
THE UNIVERSITY OF MELBOURNE
The University of Sydney
Queensland Government
Griffith UNIVERSITY
Queensland University of Technology
THE UNIVERSITY OF QUEENSLAND
AUSTRALIA
NICTA Partners

Commodity Solutions for OLAP Workloads



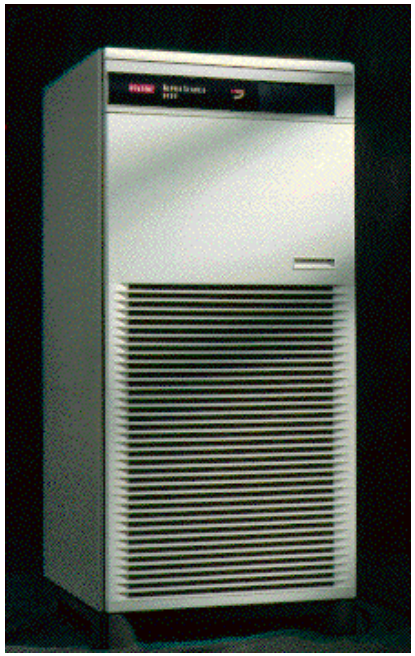
TPC-D
data model

Database size:
10-100 GByte

What kind of system architectures are suitable for this type of workload?

Platforms

Traditionally:



Symmetric Multi-processor (SMP)

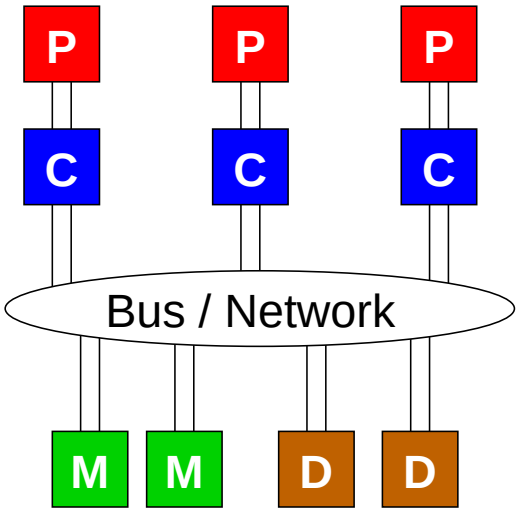
E.g. DEC 8400

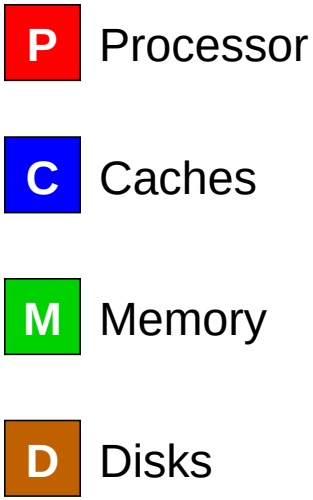
More recently:

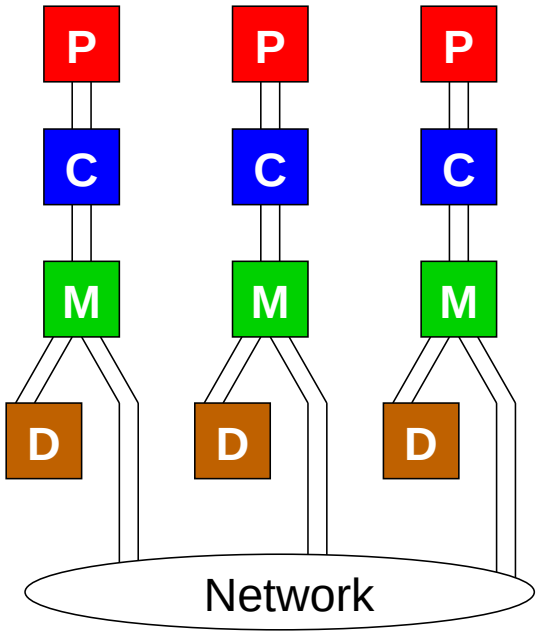
Cluster of commodity PCs

E.g. Patagonia multi-use cluster at ETH Zurich

Killer SMPs vs. Clusters of PCs







Killer SMP

- Killer performance!
- Killing price...

Cluster of commodity PCs

- Killer price!
- Killing performance?

Overview

- Introduction
- Motivation
- Distributed storage architectures
- Evaluation
- Analysis of results
- Alternative: Middleware
- Conclusion

Research Goal

Turn PC clusters into "killer SMPs" for OLAP.

Combine excess storage and high-speed network already available on cluster nodes.

Provide transparent distributed storage architecture as database's storage backend for OLAP applications.

System architect's point of view.

Focus on performance and understanding.

Storage Architectures for Clusters of PCs

Traditional:

- Big server with RAID
- Storage-area networks (SAN)
- Network-attached storage (NAS)

$\rightarrow$ Additional hardware and costs

Our proposed alternative:

Use available commodity hardware and distribute data in software layers.

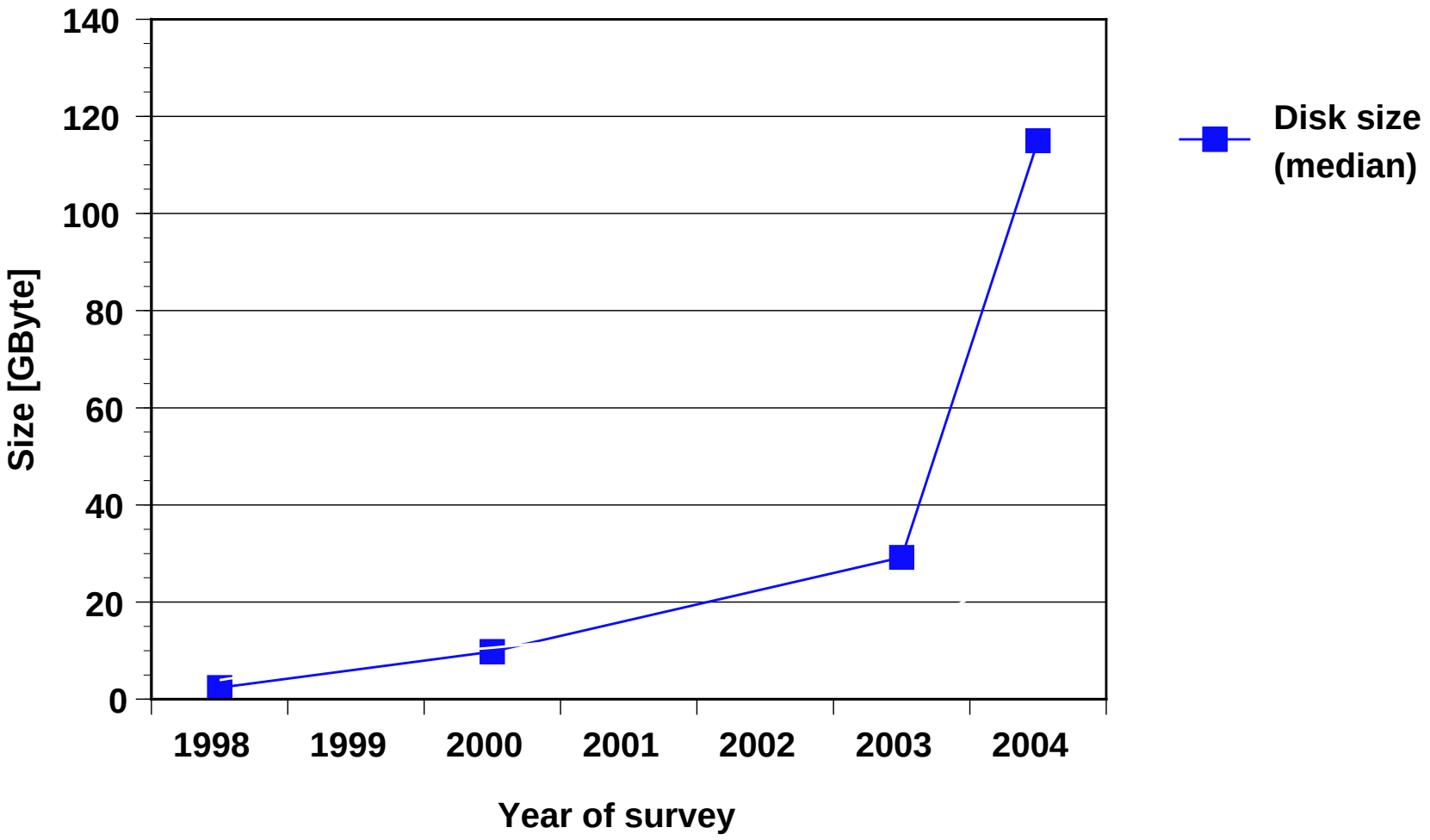
Why Should Such an Architecture Work?

Commodity hardware and software (OS) allows high cost effectiveness.

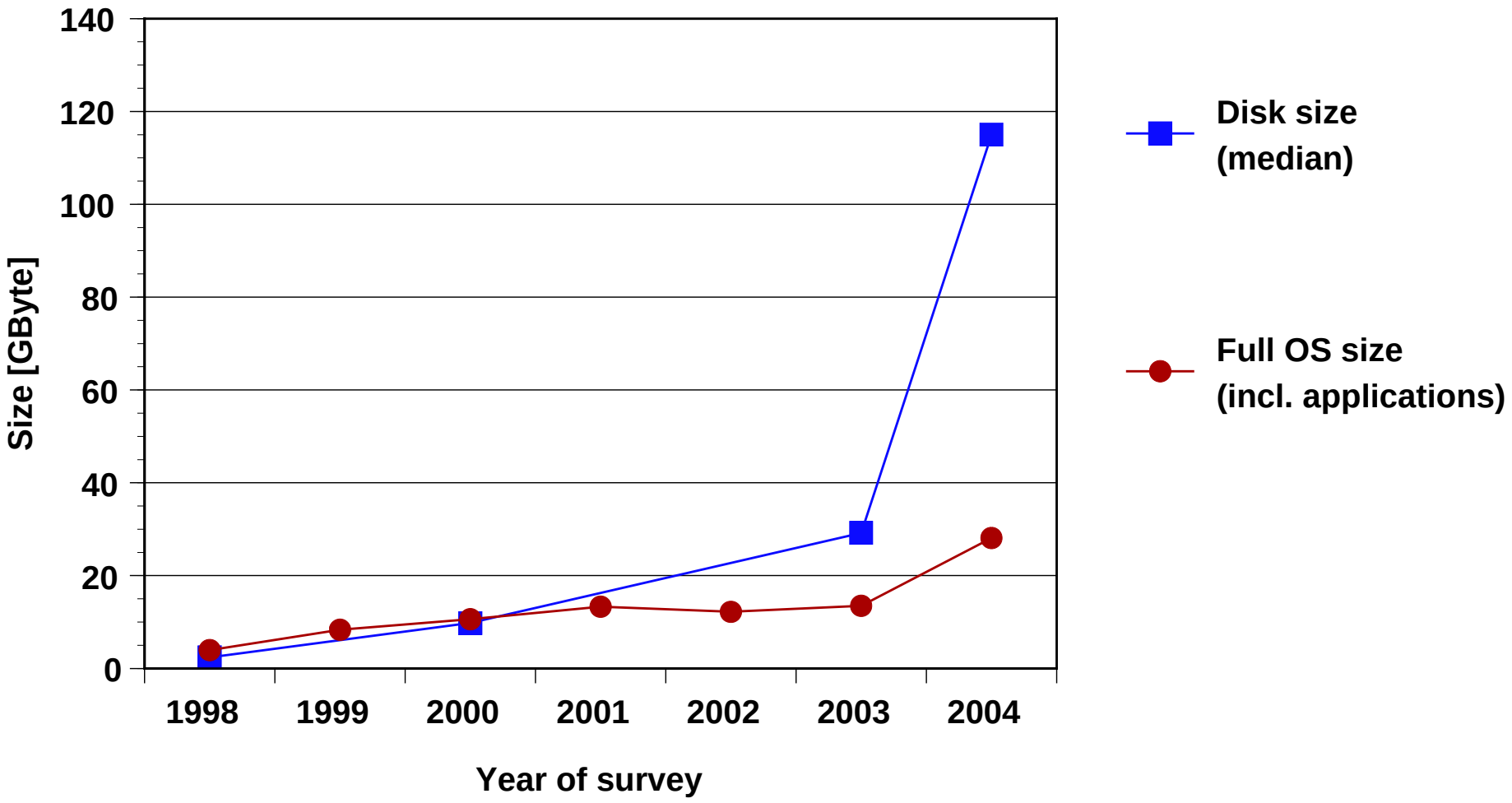
Trends:

- Disks becoming larger and cheaper
- Built-in high-speed network

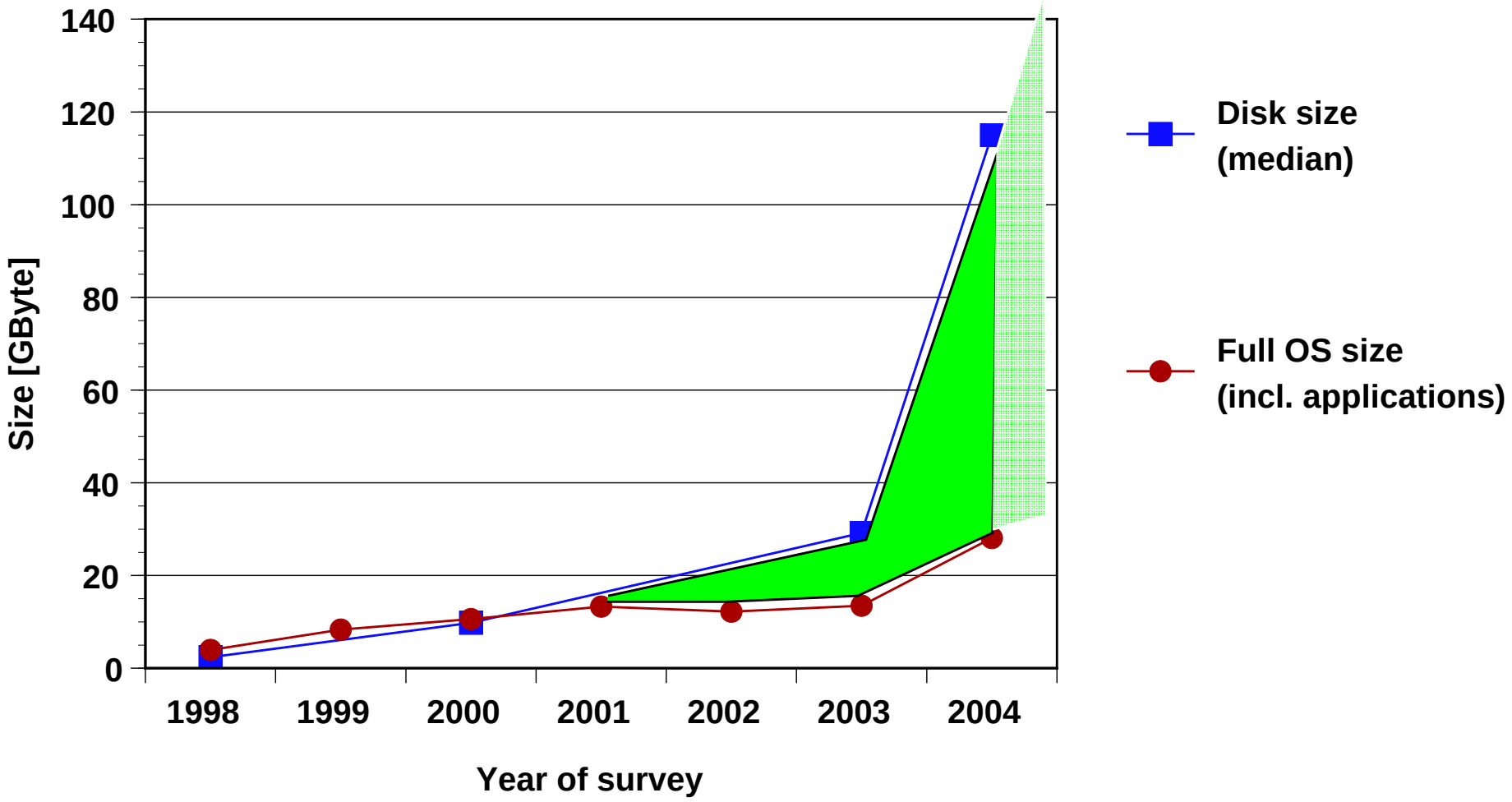
Large Hard-Disk Drives



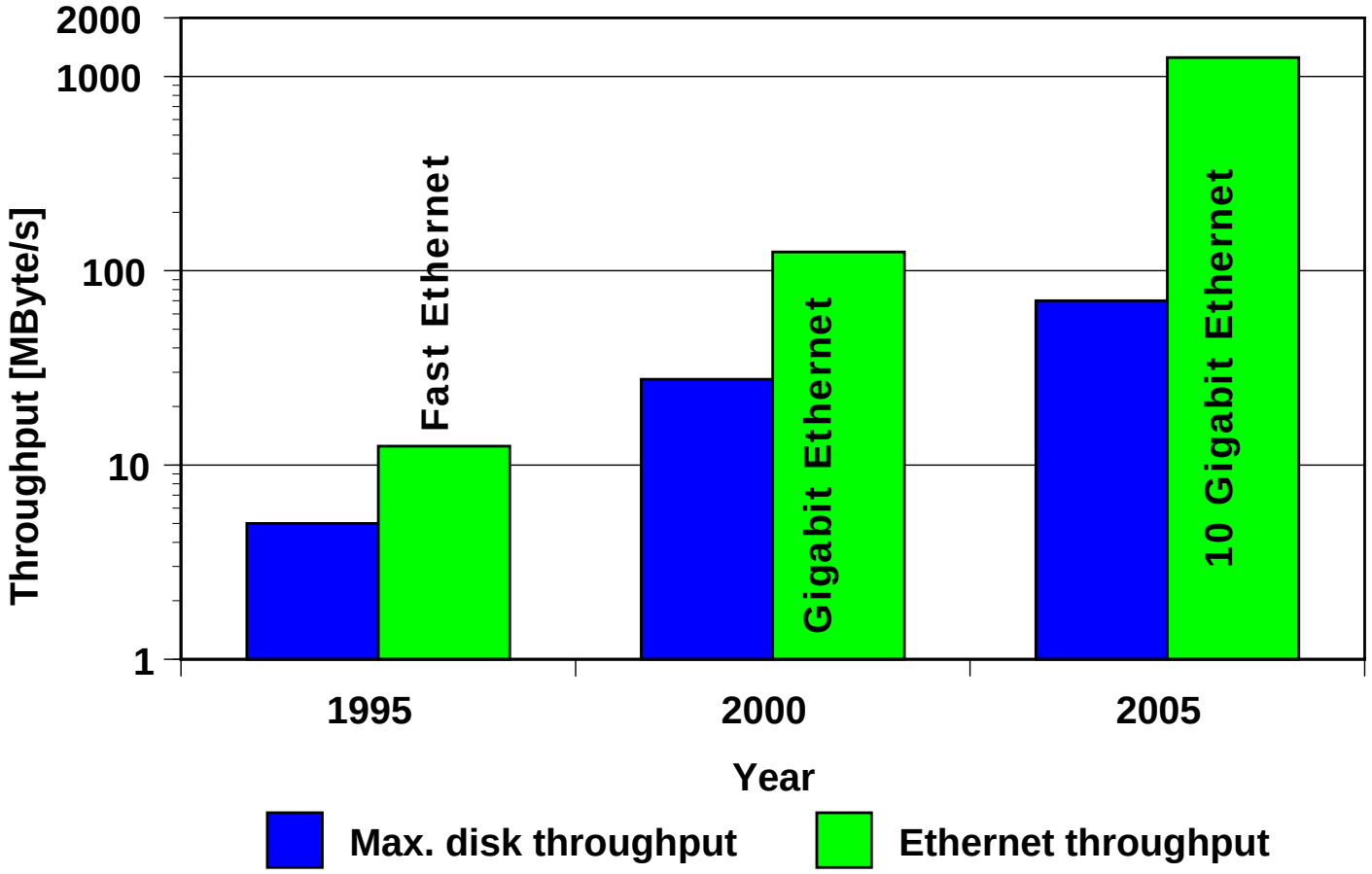
Large Hard-Disk Drives



Large Hard-Disk Drives



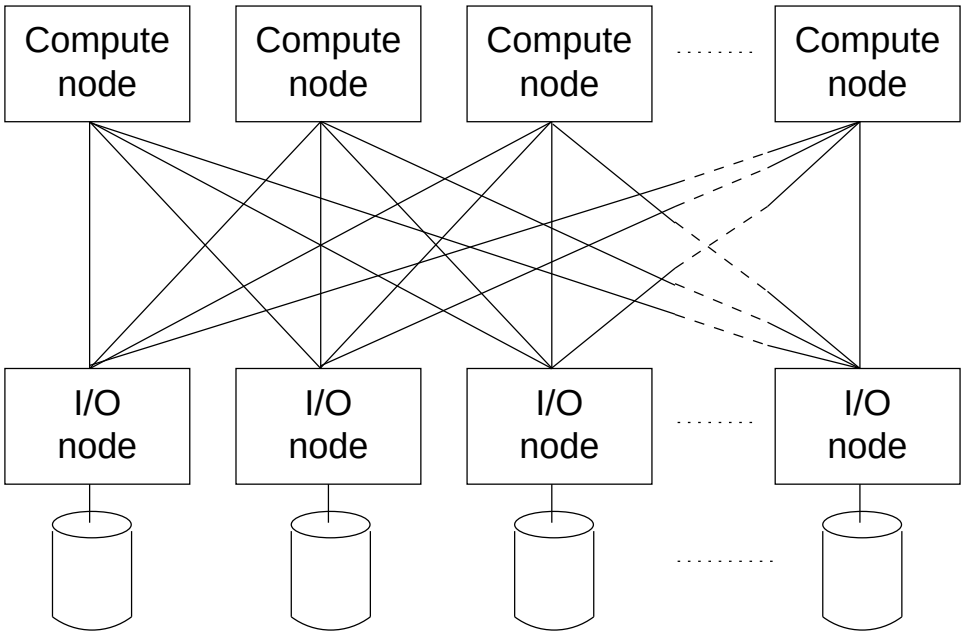
High-Speed Network



→ Enough bandwith to support distributed storage.

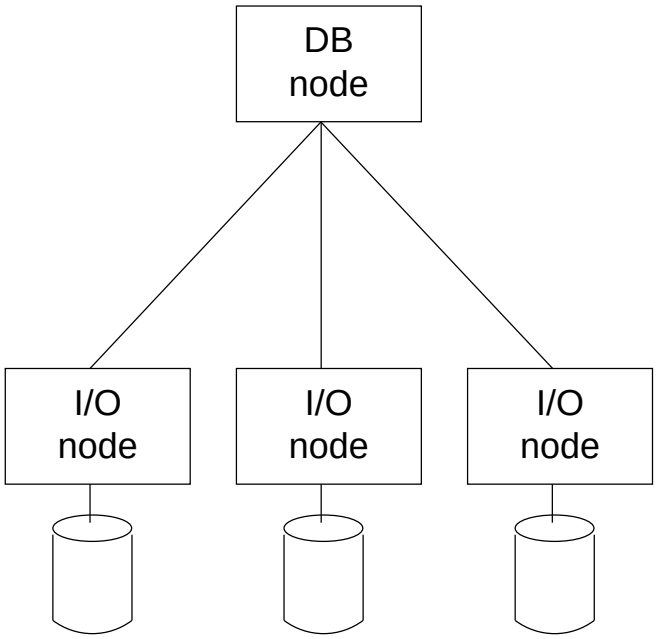
Our Scenario

Parallel file systems for high-performance computing



Scalable (Lustre, PVFS)

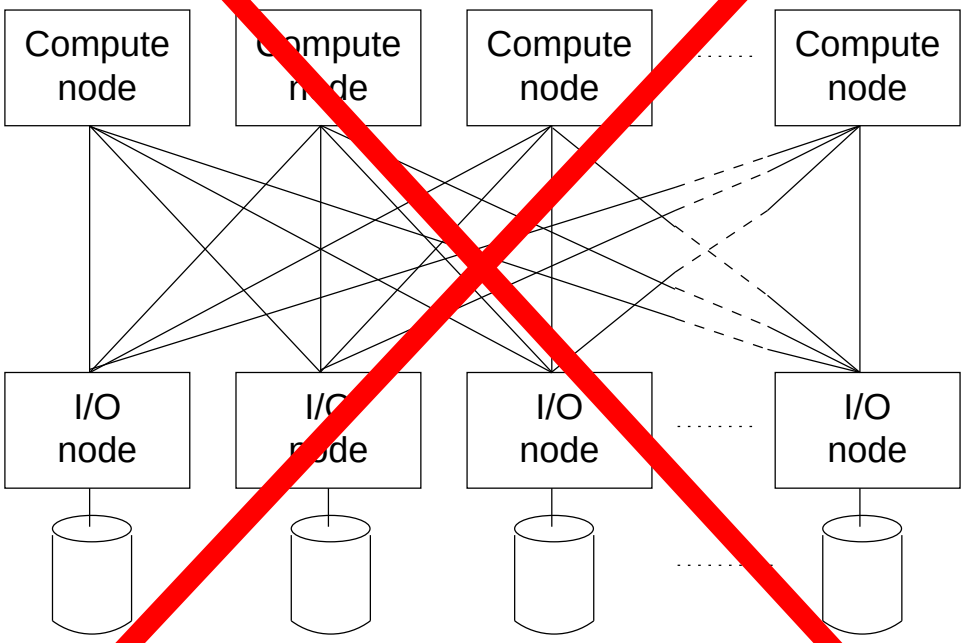
Distributed File System (network RAID0)



Boost DB performance

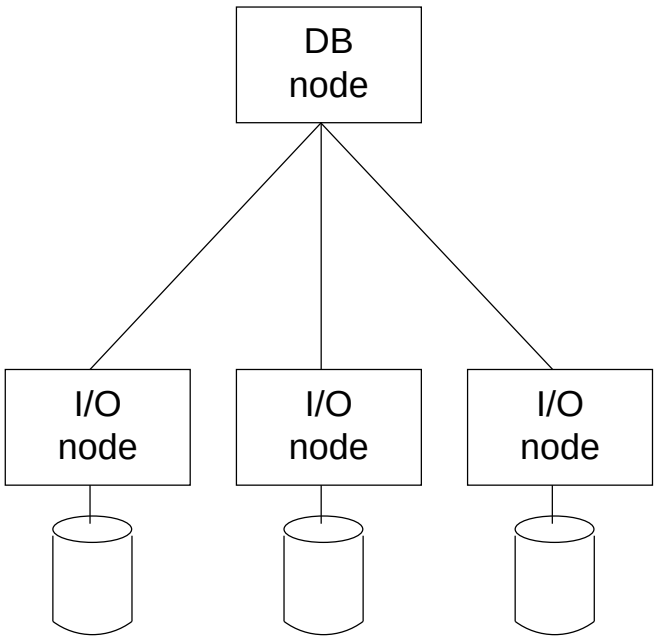
Our Scenario

Parallel file systems for high-performance computing



Scalable (Lustre, PVFS)

Distributed File System (network RAID0)



Boost DB performance

Alternative Systems

- Petal [Lee & Thekkath, 1996]:
  Distributed virtual disks with special emphasis on dynamic reconfiguration and load balancing.
- Frangipani [Thekkath, Mann & Lee, 1997]:
  Distributed file system that builds on Petal.
- Lustre [Cluster File Systems, Inc.]:
  Object oriented file system for large clusters.

Investigated Architectures

Fast Network Block Device (**FNBD**)

- Maps hard-disk device over network
- No intelligence, but highly optimised

Parallel Virtual File System (**PVFS**)

- Integrates nodes' disks into parallel FS
- Fully-featured file system

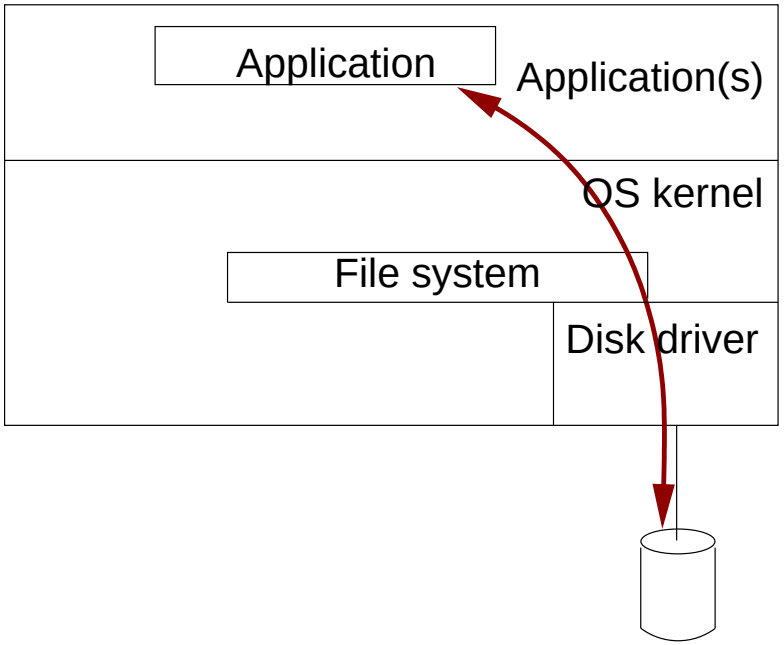
Fast Network Block Device (FNBD)

- Loosely based on Linux network block dev.
- Implemented as kernel modules
- Maps remote disk blocks over Gigabit Ethernet (from 3 servers)
- Uses hardware features of commodity network interface to implement zero copy
- Multiple instances into RAID0-like array of networked disks

Parallel Virtual File System (PVFS)

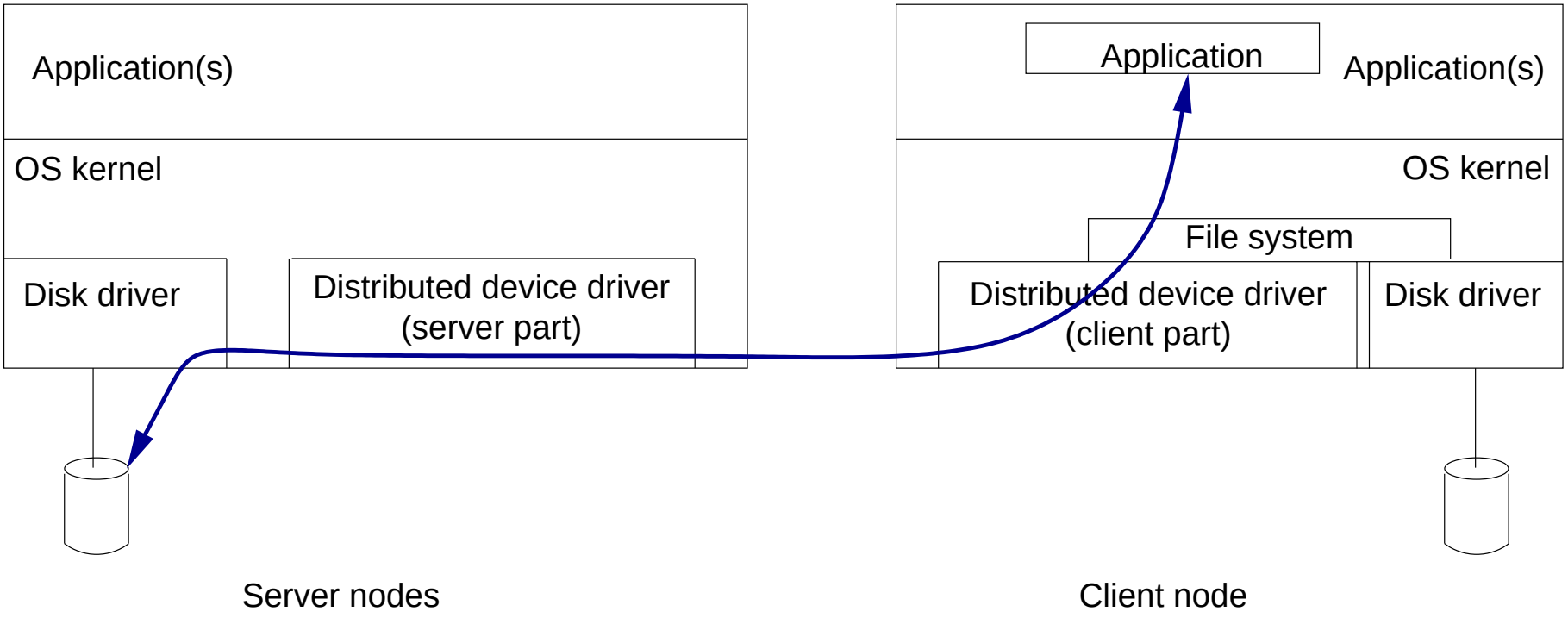
- Widely used for PC clusters
- Implemented as dynamically linked library
- Fully featured distributed file system
- Can be accessed by any participating node
- Combines special directories on server nodes into large file system
- 6 servers due to space limitations

Architecture of Reference Case

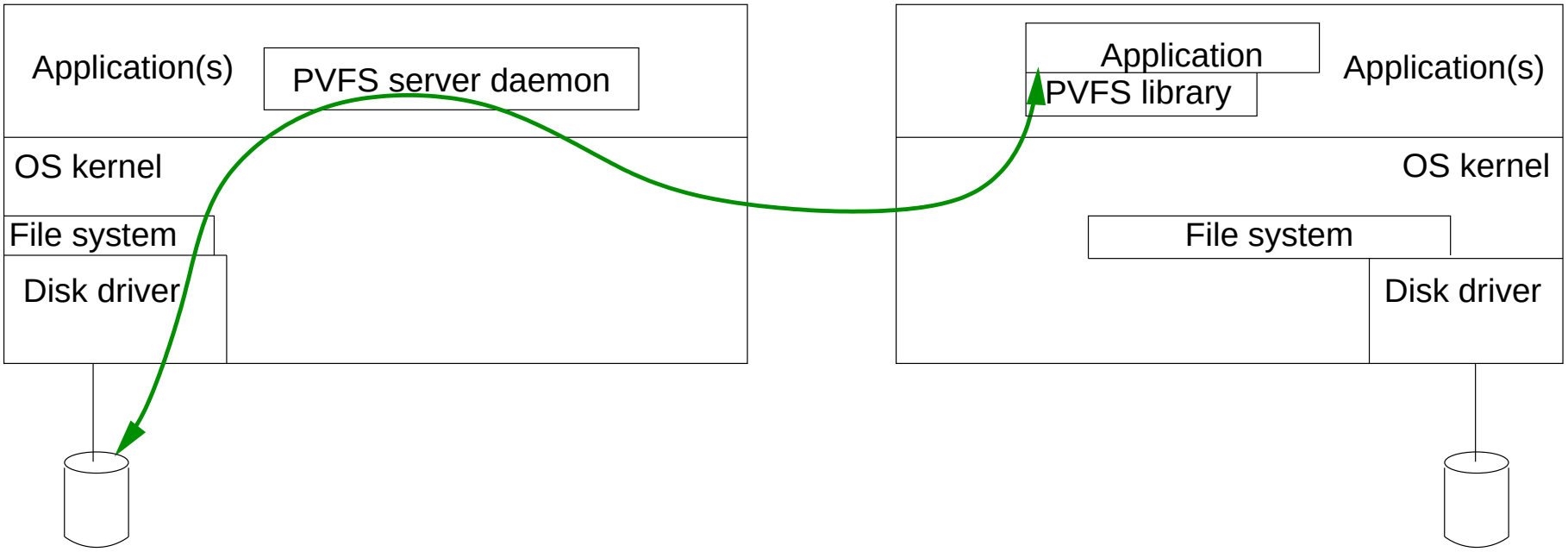


Single node

Architecture of FNBD



Architecture of PVFS



A Stream-Based Analytic Model

Presented at EuroPar 2000 conference.

Considers flow of data stream and limits of building blocks.

$\rightarrow$ Set of (in)equations.
Solve to find maximal throughput of stream.

Simple, works well for large data streams.

Modelling Workload

Need to know performance characteristics of all involved building blocks.

- Easy for small and simple parts (HW, OS functionality): Measurements or data sheets.
- Very difficult for complex, closed software (RDBMS): Black-box.

$\rightarrow$ Calibration model with know queries.

Calibration of Database Performance

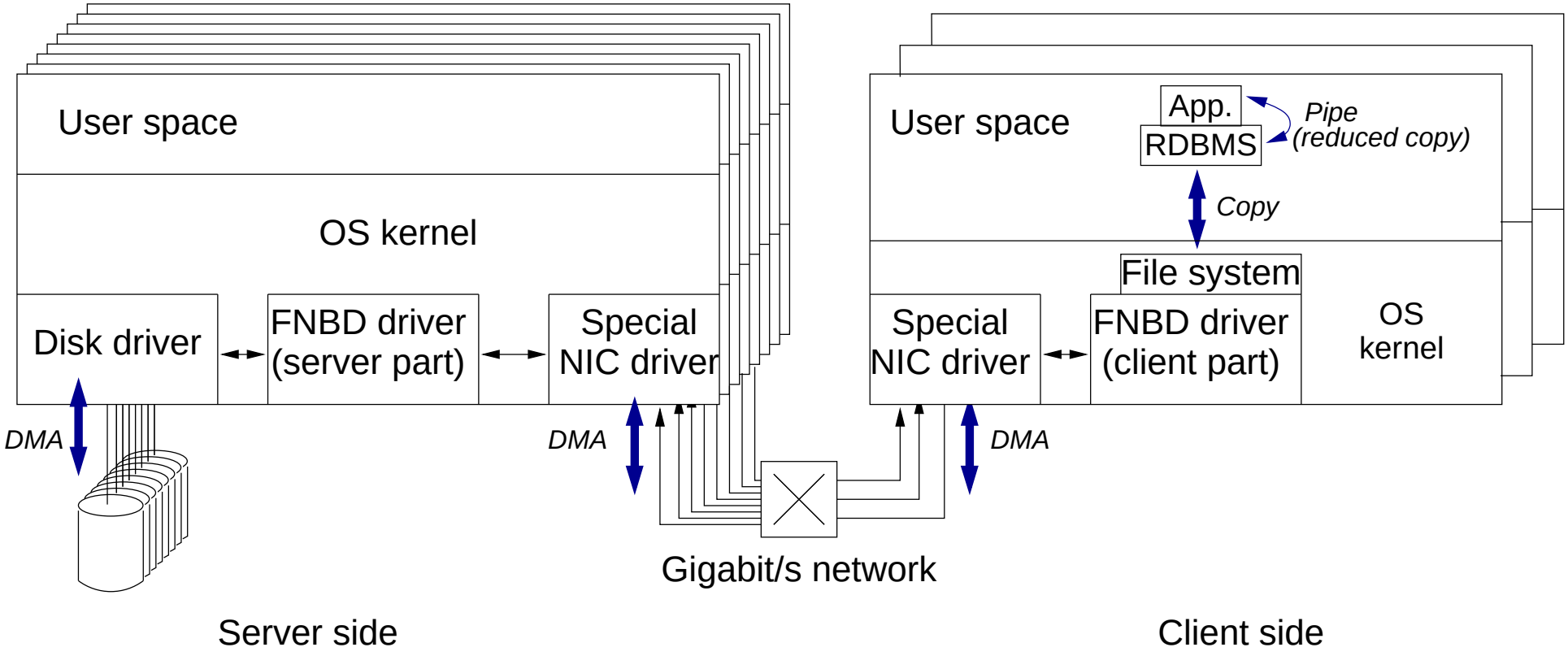
Two cases:

- "Simple" case: Full table scan (find max.)
- "Complex" case: Scan including CPU (sort)

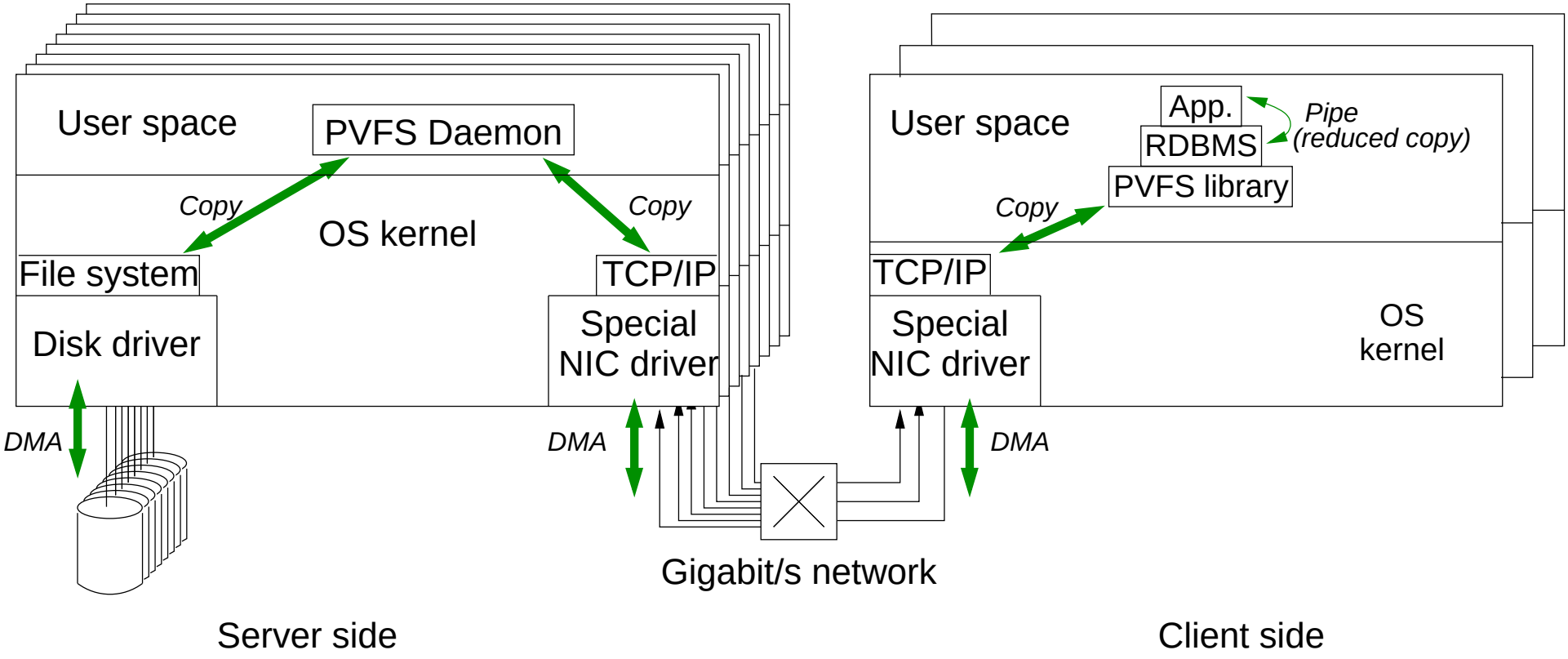
Experimental calibration with data in RAM:

- 140 MByte/s throughput for simple case
- 7.75 MByte/s throughput for complex case

Modelling OLAP on FNBD



Modelling OLAP on PVFS



Evaluation Criteria

Small microbenchmark "speed":

- Throughput for large contiguous I/O operations with varying user-level block sizes.

Application benchmark TPC-D:

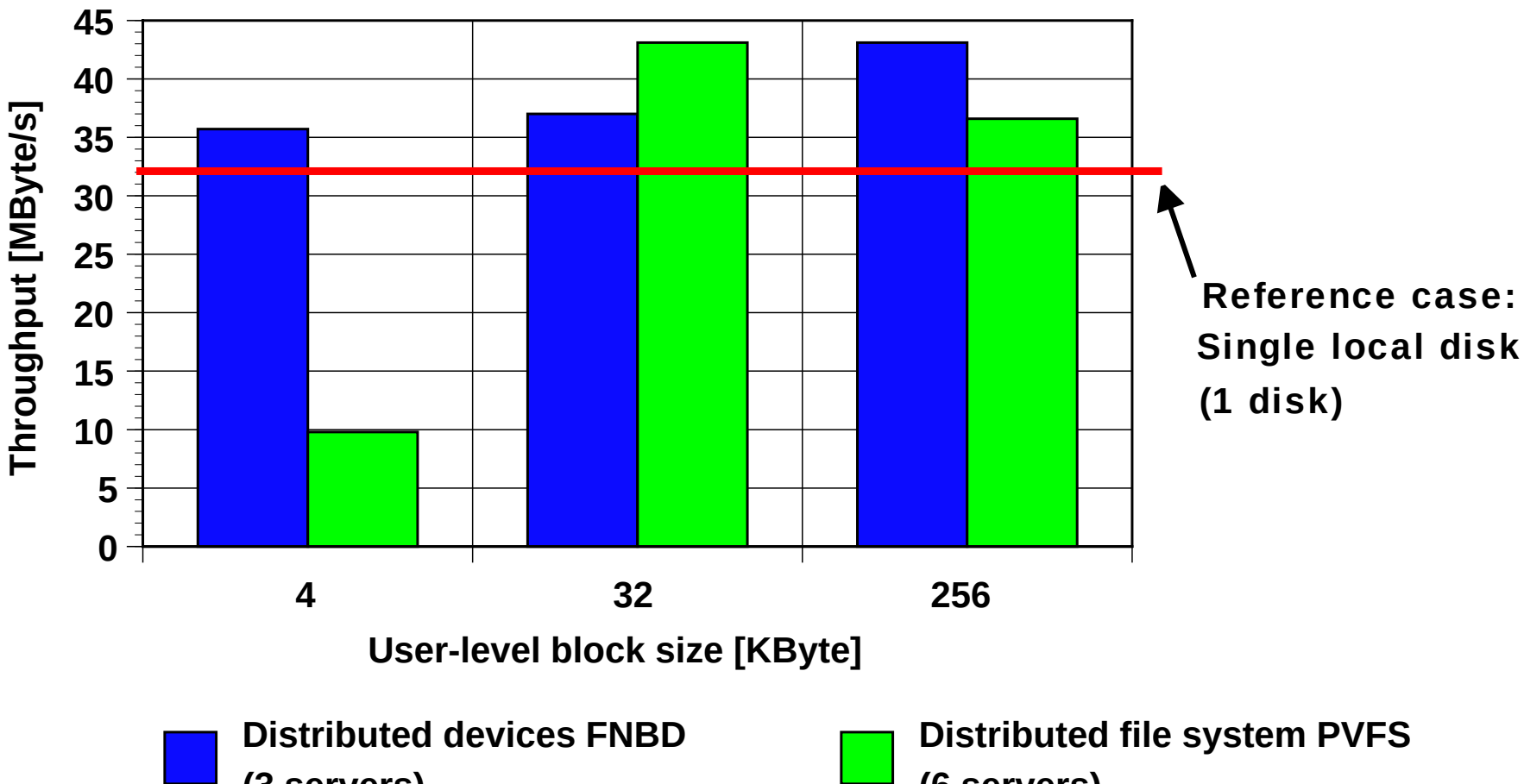
- Broad range of decision support applications, long-running, complex ad-hoc queries.
- New TPC-H and TPC-R include updates.

Experimental Testbed

Multi-use cluster with 16 nodes, each with:

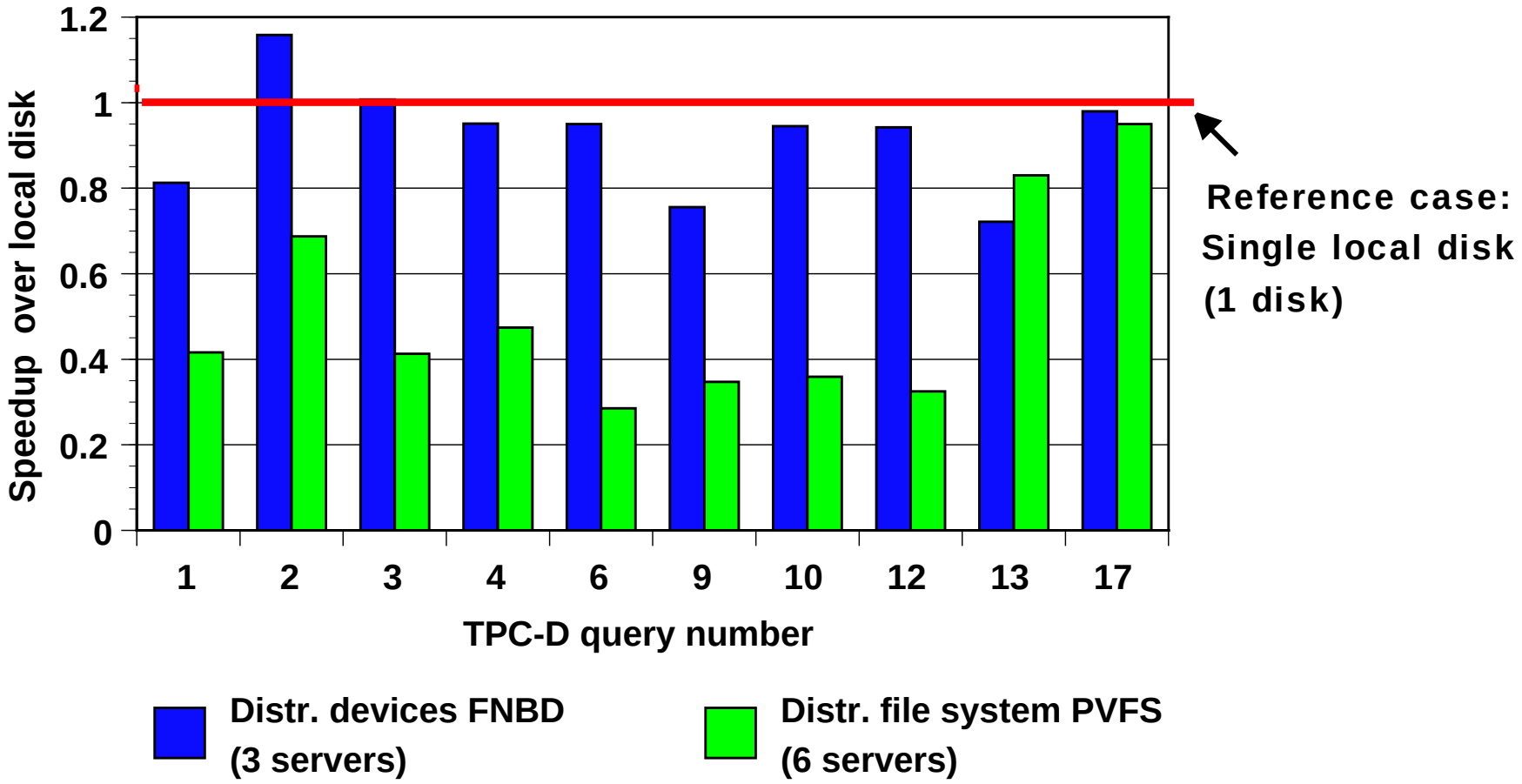
- Two 1-GHz PentiumIII CPUs
- 512 MByte ECC SDRAM
- 2 x 9 GByte disk space
- 2 Gigabit Ethernet adapters
- Linux kernel 2.4.3

Microbenchmarks



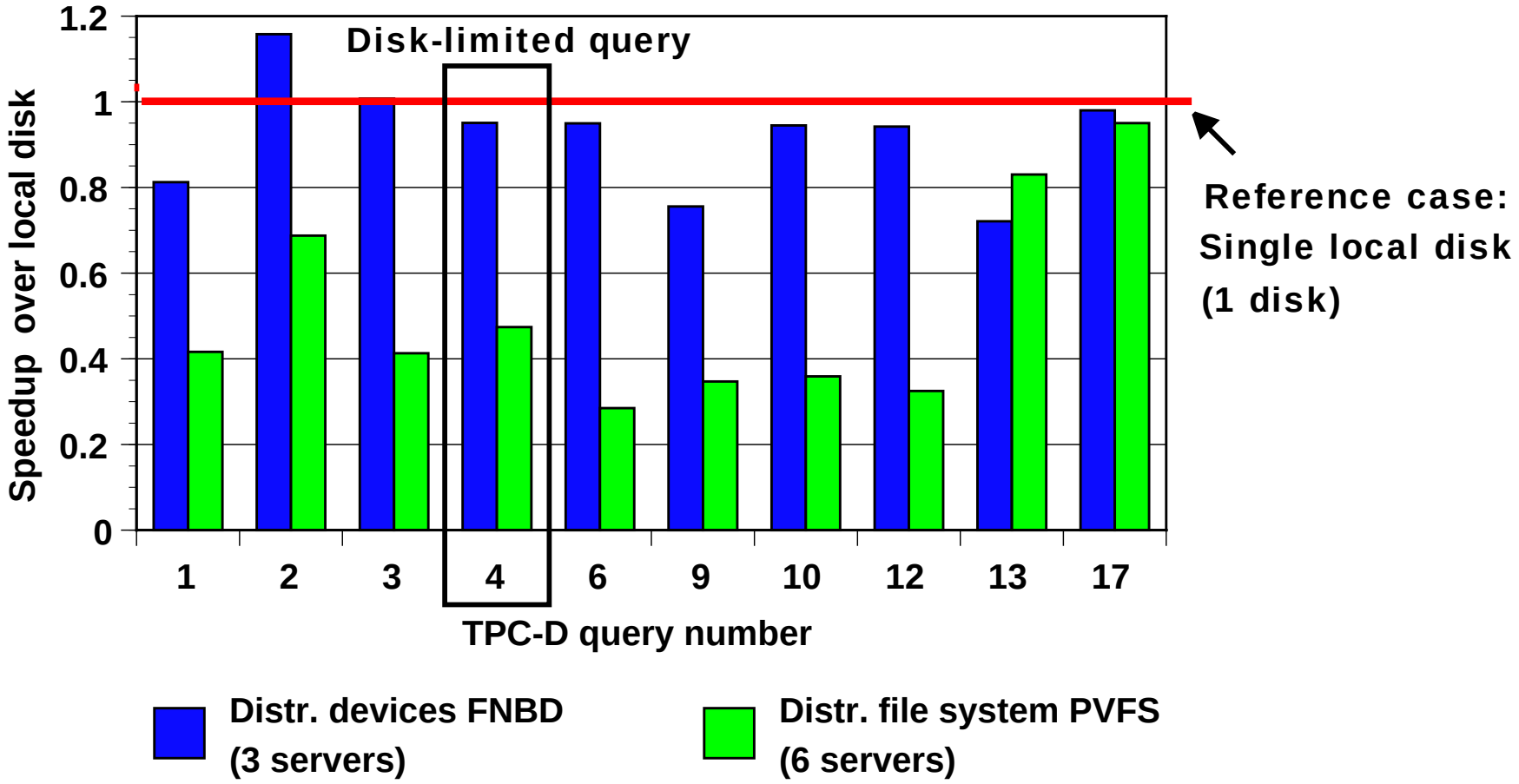
Experimental Evaluation with OLAP

TPC-D decision support benchmark on ORACLE

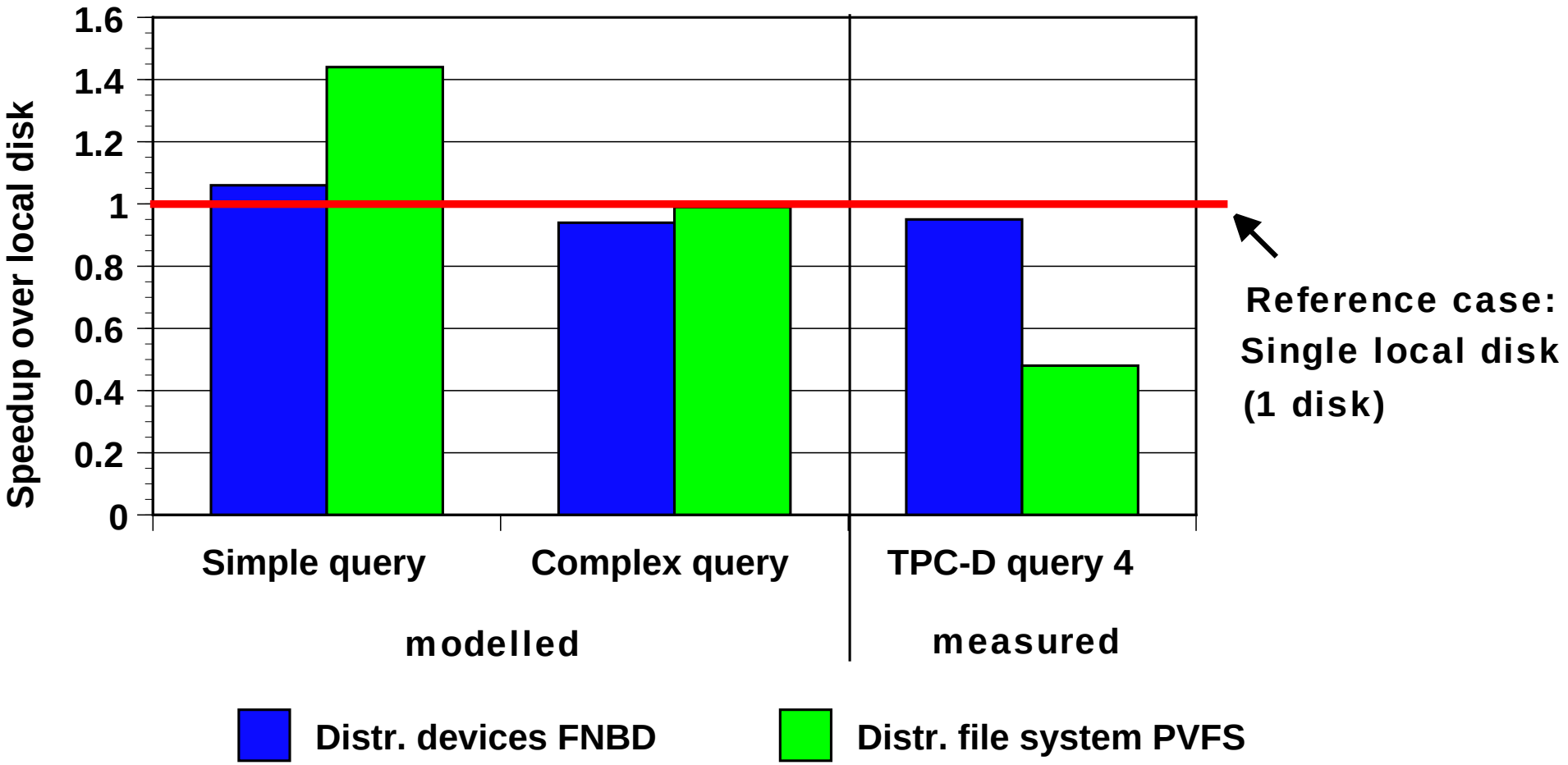


Experimental Evaluation with OLAP

TPC-D decision support benchmark on ORACLE



Quantitative Performance: Model vs. Measurements



Analysis of Results

Performance lower than expected.

Aggregation of distributed disks did not increase application performance.

Fully-featured distributed file system failed to deliver decent performance.

Stream-based analytic model too simple for complex workload.

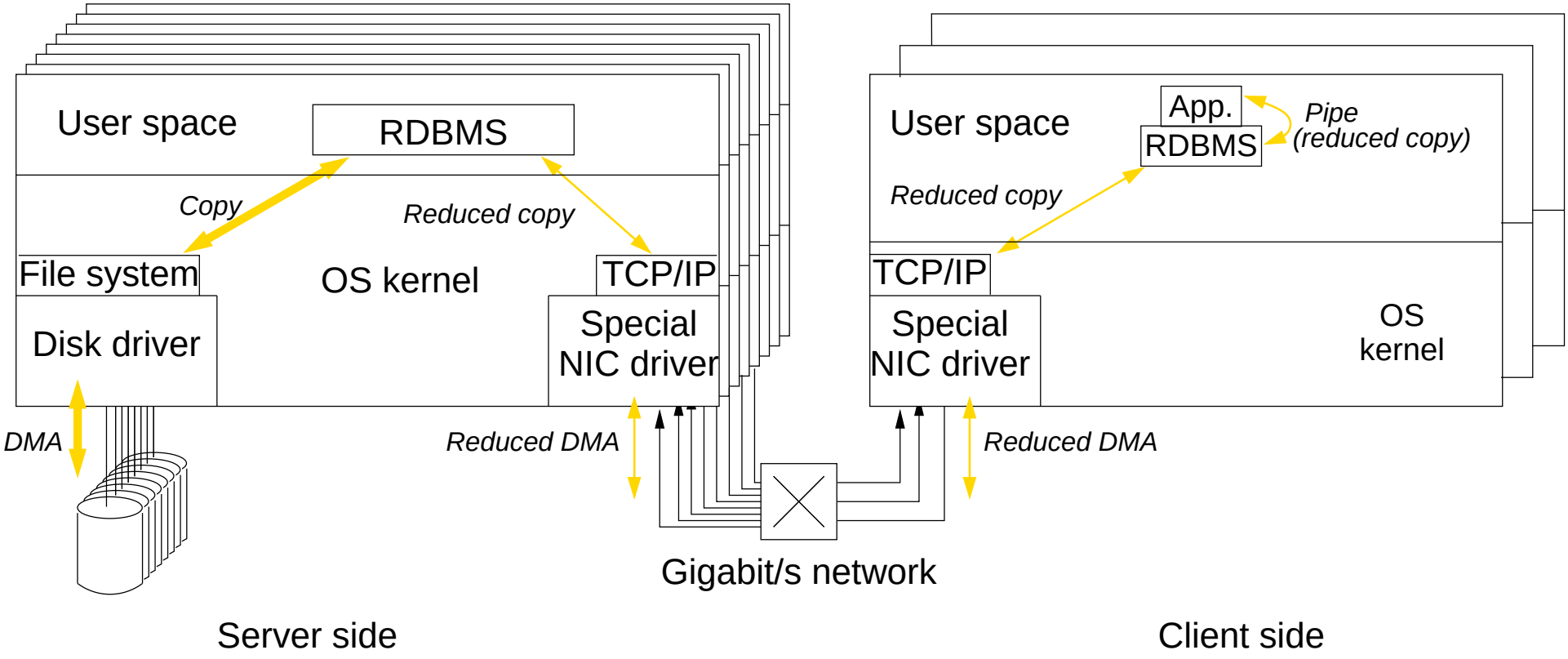
Alternative: Performance with TP-Lite Middleware

Data distribution in middleware layer:

TP-Lite by [Böhm et al, 2000]

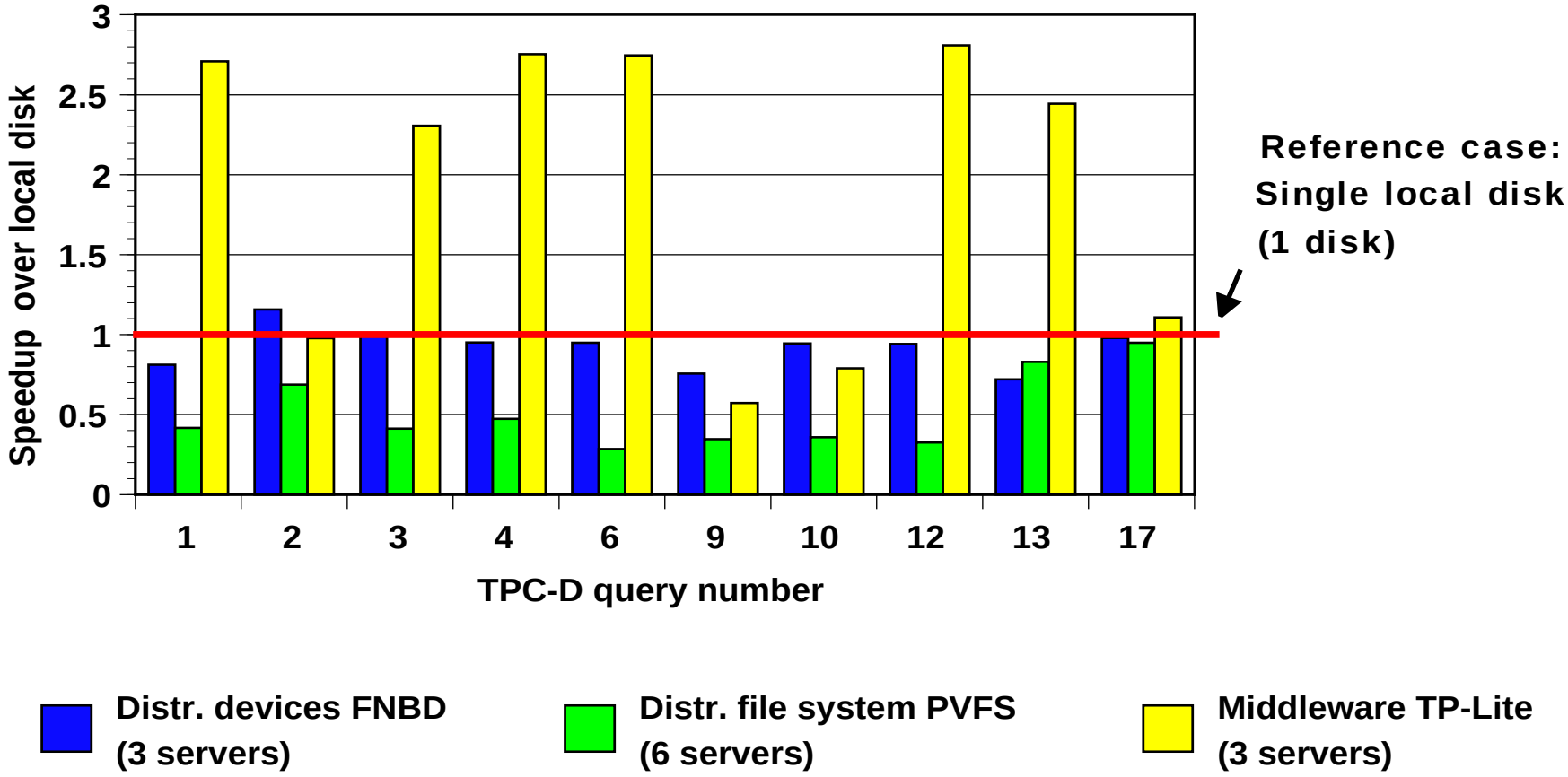
- Distributes queries to multiple database servers in parallel
- Needs multiple servers (costs)
- Small changes to application (not always possible)

Modelling OLAP with TP-Lite



Performance of TP-Lite

TPC-D decision support benchmark on ORACLE



Conclusions

We tried to turn clusters of PCs into "killer SMPs" as an architecture to boost OLAP performance.
Our cost-effective approach uses excess storage on clusters nodes and a transparent parallelisation.

Simple network RAID can not boost performance.

Fully-featured scalable parallel file system failed.

To model the workload is almost impossible.

We system architects can not help database community with system tricks (sorry).

Questions?

National ICT Australia (NICTA)

Embedded, Real-Time, and Operating Systems Research Program (ERTOS)
`http://www.ertos.nicta.com.au/`

CoPs Project (Clusters of PCs)
1996-2004 @ ETH Zurich
`http://www.cs.inf.ethz.ch/CoPs/`