Virtualization and Databases: State of the Art and Research Challenges

Ashraf Aboulnaga
University of Waterloo

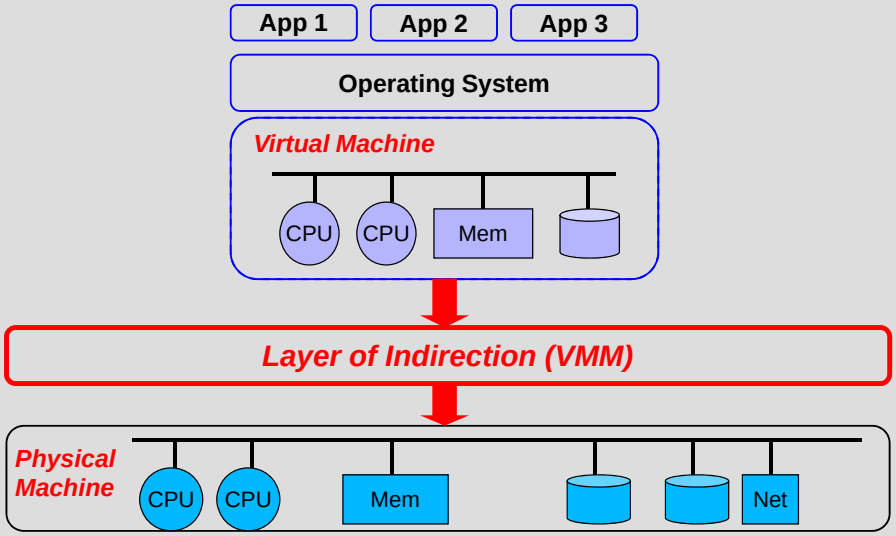
prepared jointly with
Cristiana Amza
University of Toronto
Kenneth Salem
University of Waterloo

What is Virtualization?

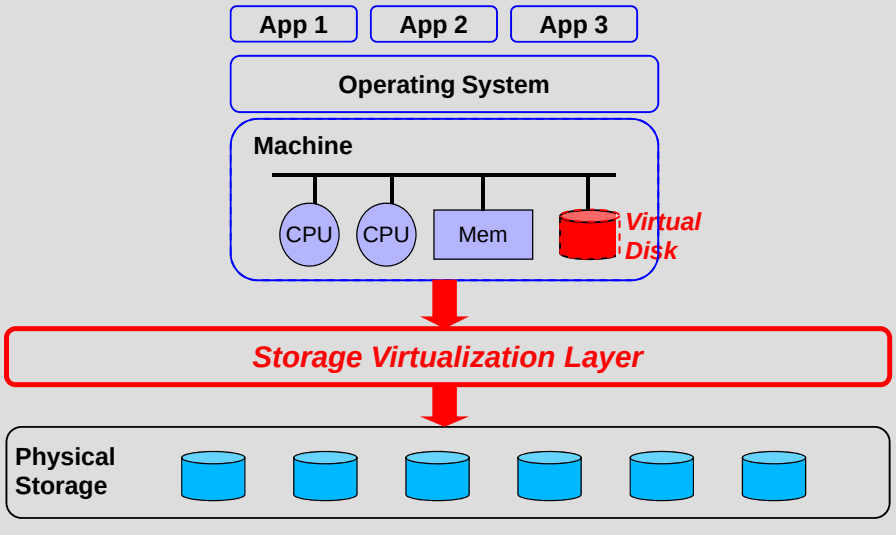
- Separating the abstract view of computing resources from the implementation of these resources
- ***A layer of indirection between abstract view and implementation of resources***
  - Hides implementation details
  - Controls mapping from abstract view to implementation

"any problem in computer science can be solved with another layer of indirection"
– David Wheeler

Example: Virtual Machines



Example: Virtual Storage



Why Virtualization?

- Virtualization adds ***flexibility*** and ***agility*** to the computing infrastructure
- Can be used to solve many problems related to provisioning, manageability, security, …
  - Pool and share computing resources
  - Simplify administration and management
  - Improve fault tolerance
- For organizations: ***Lower total cost of ownership*** for computing infrastructure
  - Fewer computing resources
  - More resilient and simpler to manage

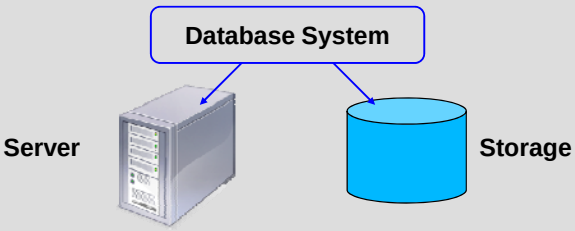
Why Should We Care?

- Computing infrastructure is becoming more and more virtualized
- Database systems are increasingly being run in virtualized environments
- Does this introduce new opportunities or challenges for database systems?

"virtualization will be a $20 billion market by 2010"
– IDC, January 2007

This Tutorial

- Virtualizing computing ***resources***



- The term virtualization is also used in other areas
  - Virtual teams
  - Virtual enterprises
  - Java virtual machines
  - Virtual reality
  - …

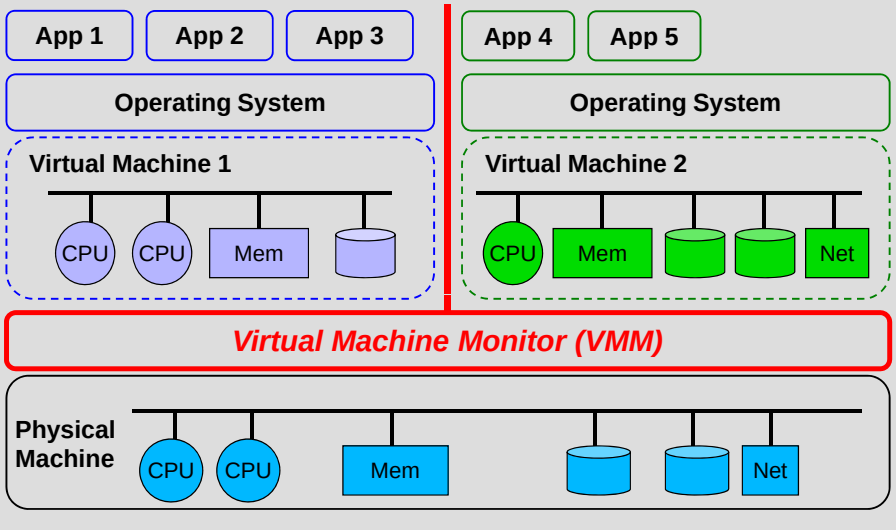
Outline

- Introduction
- Machine Virtualization
  - Overview of machine virtualization and its uses
  - Virtual machine technologies
  - Virtualization and databases
- Storage Virtualization
- Virtualization-like Capabilities in the DBMS
- Conclusion

Machine Virtualization

- A ***virtual machine*** "abstracts" the computing resources of a physical machine into virtual resources
- Introduces a level of ***indirection*** between virtual resources and physical resources
- End users only see the virtual resources
  - Can install their operating systems and run their applications on the virtual machines
- A ***Virtual Machine Monitor*** (or ***Hypervisor***) is a software layer that implements the mapping from virtual resources to physical resources

Machine Virtualization



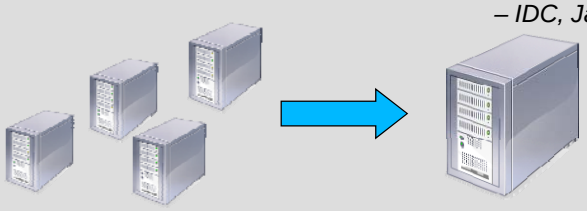
Virtual Machine Monitors

- ***Strong isolation*** between virtual machines
- ***Flexible mapping*** between virtual resources and physical resources
  - Can have more virtual resources than the corresponding physical resources
  - Can reallocate physical resources among VMs
- Pause, resume, checkpoint, and migrate virtual machines

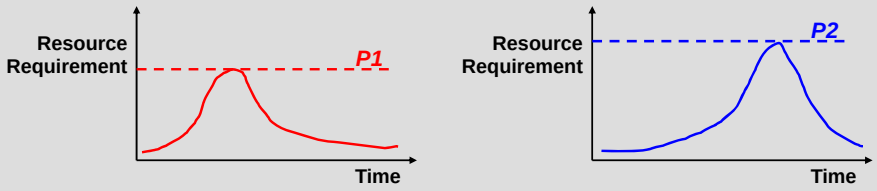
Why Use Virtual Machines?

- Server consolidation
  - Typical setup today: one machine per application (DBMS, web server, mail server, …)
  - Provisioned for peak load. Usually under-utilized
  - Instead, can run multiple applications on virtual machines that ***share the same physical machine***
  - Save hardware costs, administration costs, ***power***, etc.

"$140 billion worth of server assets go un-utilized every year"
– IDC, January 2007



Server Consolidation



P12 < P1 + P2

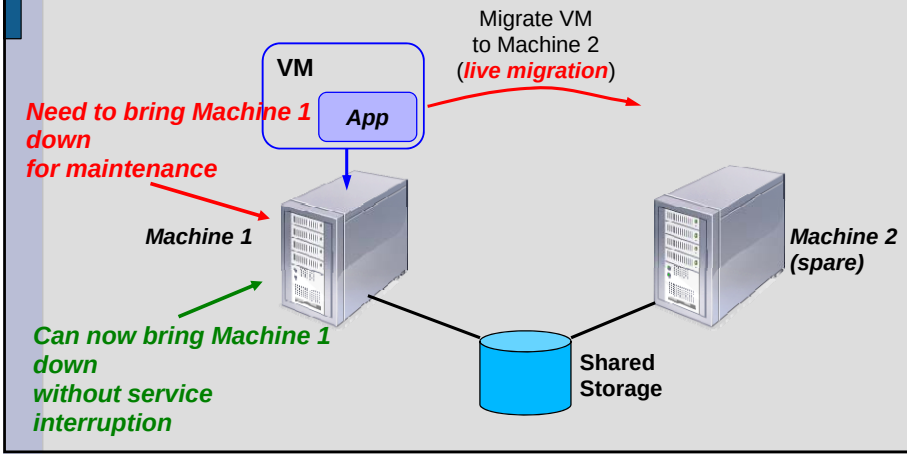
- Consolidate into a single machine with capacity P12
  - Easier to manage
  - Less total capacity and operating costs than original two
  - Better utilization than the original two

Why Use Virtual Machines?

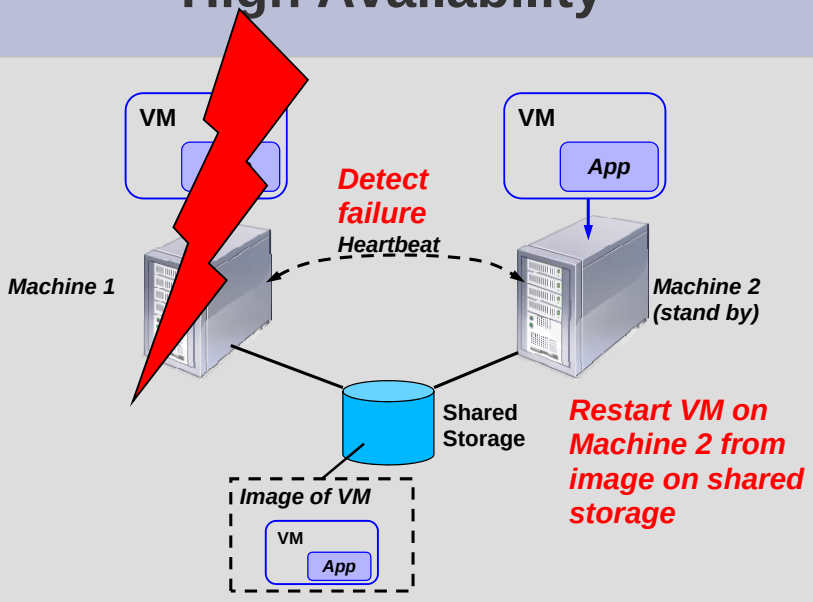
- Improved manageability
  - Dynamic provisioning of resources to VMs
  - Migration of VMs to avoid down time during upgrades
  - Migration of VMs for load balancing
- Isolation between VMs
  - Security
  - Privacy
  - Fault tolerance

Eliminating Scheduled Down Time

- Use live migration of virtual machines [clfr05]
- Could also be used for ***load balancing***

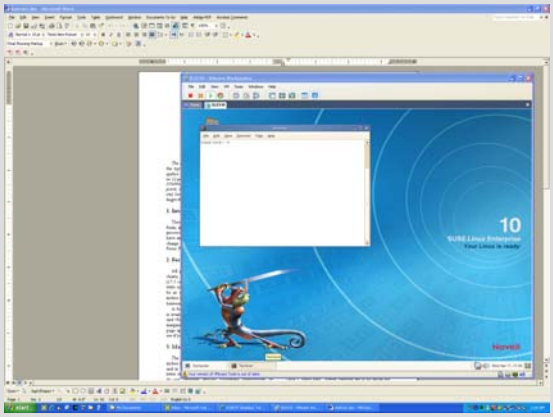


High Availability



Why Use Virtual Machines?

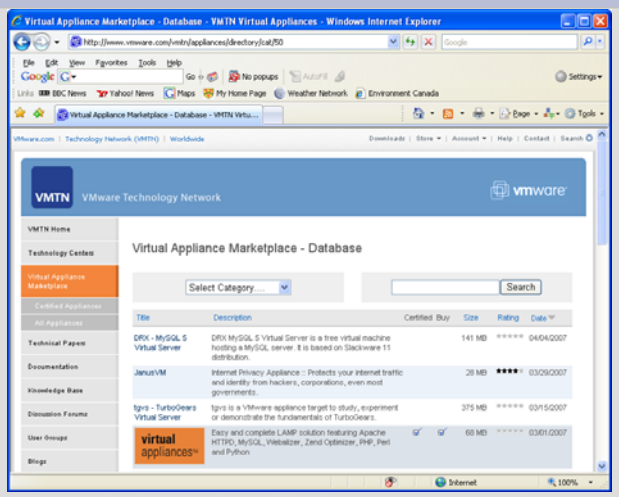
- Application compatibility
  - Different environments for different applications



Why Use Virtual Machines?

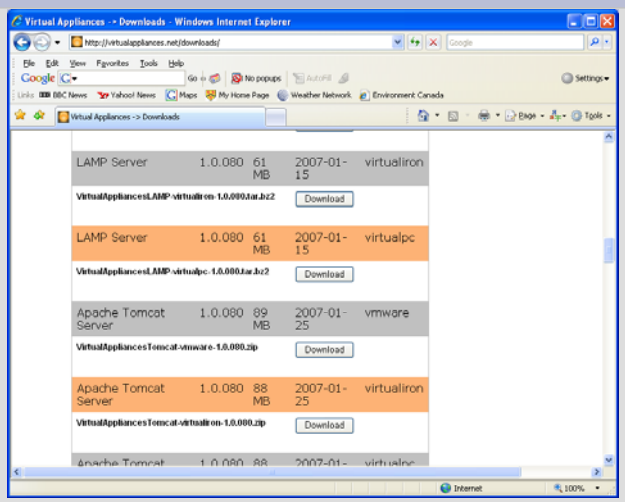
- Software development and testing
  - Multiple environments for development and testing
- Software deployment
  - Preconfigured ***virtual appliances***
  - Repositories of virtual appliances on the web

Virtual Appliances



http://www.vmware.com/vmtn/appliances

Virtual Appliances



http://virtualappliances.net/downloads/

Why *not* Use Virtualization?

- ***Performance penalty***
  - Indirection through VMM adds overhead
  - Can be reduced through better virtualization support
- ***Hiding details of physical resources***
  - Some applications (e.g., DBMS!) make decisions based on assumptions about the physical resources

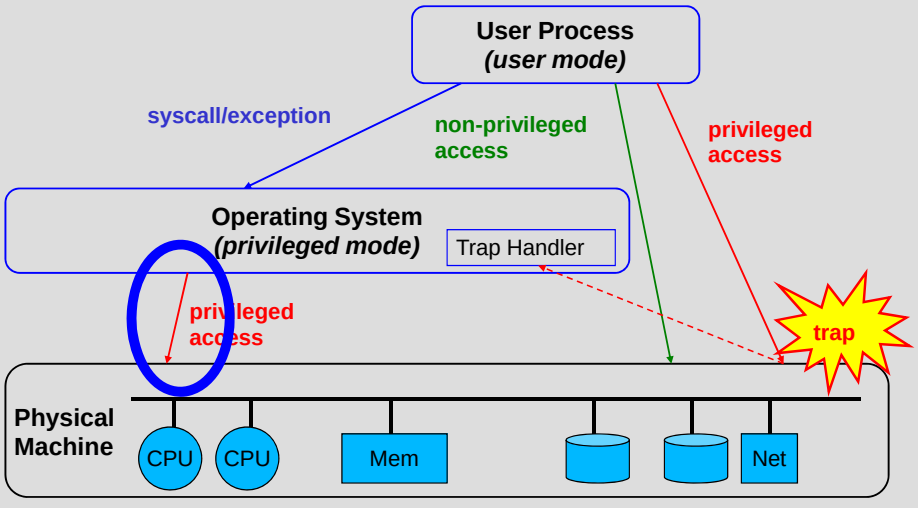
Outline

- Introduction
- Machine Virtualization
  - Overview of machine virtualization and its uses
  - Virtual machine technologies
  - Virtualization and databases
- Storage Virtualization
- Virtualization-like Capabilities in the DBMS
- Conclusion

History

- Virtualization has been around since the 1960's [gold74, fidi05]
- Prominent since IBM 370 mainframe series
  - Allowed expensive hardware to be shared by multiple applications running on different operating systems (i.e., server consolidation)
- Virtualization today
  - ***Larger scale***
  - ***Commodity hardware and operating systems***
  - ***Renewed interest in benefits of virtualization***

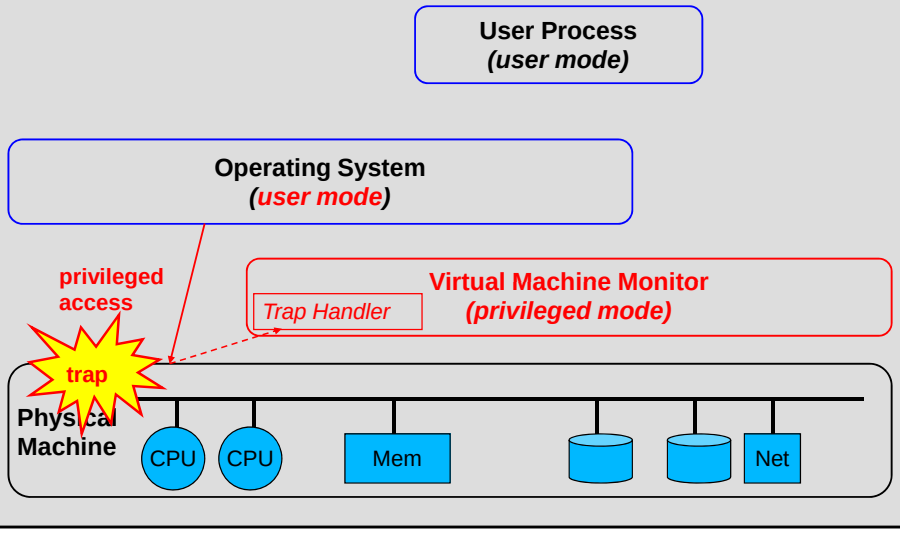
Operating Systems 101



Trap-and-Emulate Virtualization

- Run ***VMM in privileged mode***
- Run ***OS in user mode***
- Privileged operations by the OS will trap
- Trap handler in VMM emulates these operations as if they were run on the virtual machine
- Non-privileged operations can proceed as before with no intervention from the VMM [pogo74]

Trap-and-Emulate Virtualization



Example Technologies

- ***Not a comprehensive list!***
- VMware Workstation
- VMware ESX Server
- Xen
- XenServer
- Virtual Iron
- Microsoft Virtual Server

Related Terminology

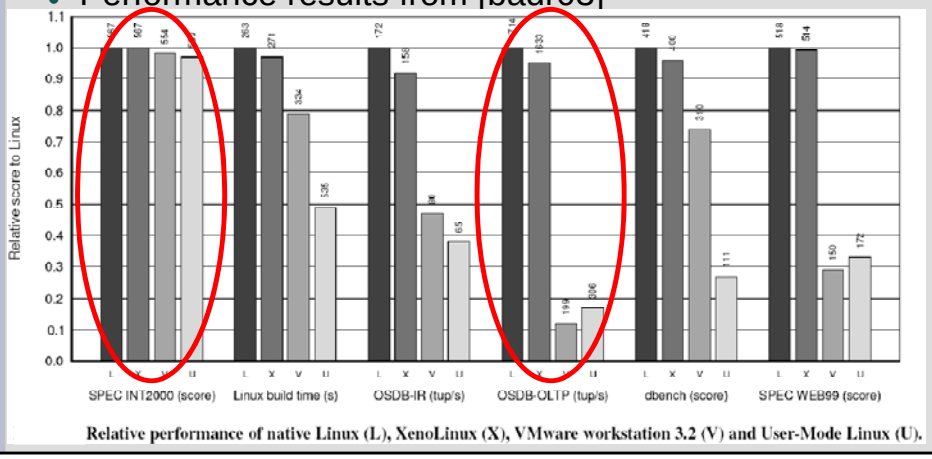
- ***Emulation***
  - Simulate complete hardware, allowing unmodified OS for ***different CPU*** to be run
  - Must simulate each instruction so ***slow***
  - Example: Bochs
- ***Operating system virtualization***
  - Isolated virtual servers within a server
  - Guest and host OS use ***the same kernel***
  - Example: Solaris Containers, Virtuozzo
- ***What we are discussing is often termed native virtualization or full virtualization*** [smna05]

Virtualization and Databases?

- Are database systems just another application running in the virtualized environment?
- ***No! Virtualization poses several interesting research questions for database systems***
  - Understanding the performance of database systems on virtual machines
  - Configuring and tuning virtual machines running database systems
  - Taking advantage of virtualization capabilities in the database system

Performance Overhead

- DBMS is a heavy user of the operating system
- Performance results from [badr03]



Relative performance of native Linux (L), XenoLinux (X), VMware workstation 3.2 (V) and User-Mode Linux (U).

Detailed Study of Overhead

- Study overhead of running ***TPC-H*** on ***PostgreSQL*** in a ***Xen*** virtualized environment [miya08]
- Run TPC-H queries on PostgreSQL running on
  (a) bare hardware
  (b) Xen virtual machine on indentical hardware
- Two questions:
  - How much performance degradation from (a) to (b)?
  - What are the causes of this overhead?

Results of Warm Experiments

	Base Runtime (secs)	Xen Runtime (secs)	Abs SlwDwn (secs)	Rel SlwDwn (%)
Q1	14.19	15.30	1.11	7.82
Q3	5.20	6.98	1.78	34.35
Q5	4.53	5.99	1.46	32.21
Q7	4.09	5.32	1.23	30.14
Q9	10.99	12.81	1.81	16.49
Q10	5.04	6.36	1.32	26.17
Q13	14.02	15.27	1.25	8.93
Q18	9.38	11.54	2.17	23.12
Q19	5.26	6.33	1.07	20.41
Q21	2.79	3.65	0.86	31.03

- Most significant slowdown in ***system time***
  - System calls considerably slower, but DBMS does not spend much time on system calls
  - ***Page fault handling*** is a major cause of slowdown

Results of Warm Experiments

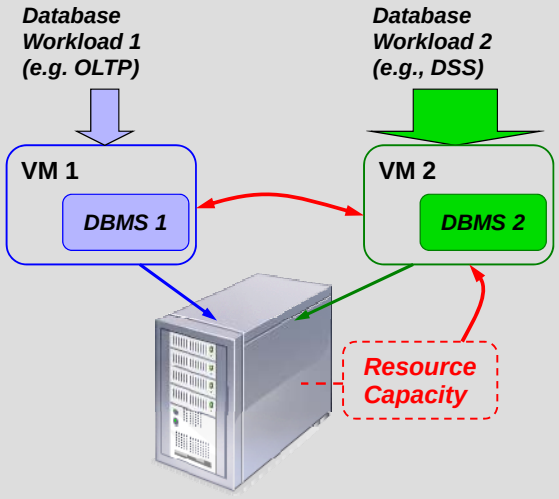
TPC-H Query	Base Runtime (secs)	Xen Runtime (secs)	Abs SlwDwn (secs)	Rel SlwDwn (%)
Q1	13.30	14.04	0.74	5.55
Q3	4.61	5.82	1.21	26.23
Q5	4.14	4.97	0.84	20.22
Q7	3.52	3.66	0.14	3.91
Q9	10.52	11.36	0.83	7.91
Q10	4.57	4.69	0.12	2.58
Q13	13.36	14.10	0.75	5.59
Q18	8.86	10.13	1.27	14.36
Q19	4.84	5.05	0.22	4.46
Q21	2.30	2.48	0.18	7.84

- Can reduce number of page faults by using one client process for all queries
- Slowdown = ***9.8%*** for warm, ***6.4%*** for cold

Discussion of Results

- Overhead of virtualization is not unacceptably high
- Can be made lower with better virtualization support
- Some new types of overheads are introduced and must be carefully managed
  - E.g., extra page fault handling overhead
- ***Not considered in this study***
  - Other workloads and database systems
  - Other virtualization environments
  - Multiple concurrently running database workloads in the same or different virtual machines

Resource Provisioning

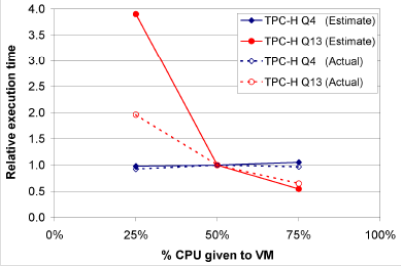


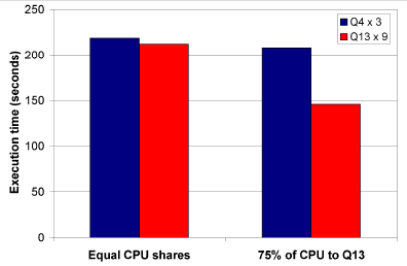
Resource Provisioning

- ***What level of resources to give to each DBMS?***
  - Configuring VM parameters
- ***How to tune the DBMS for a given level of resources?***
  - Configuring the DBMS parameters
- Need a ***model*** of how resource allocation affects database performance
- Need optimization or control algorithms to decide on the optimal resource allocation [pazh07]

Example Modeling Approach

- Use ***query optimizer*** as the cost model [soab07]
  - Calibrate it to reflect virtual machine resource levels
- Example (from [soab07]): Xen VMs running TPC-H queries on PostgreSQL and sharing the same physical machine
  - What is the best CPU allocation?



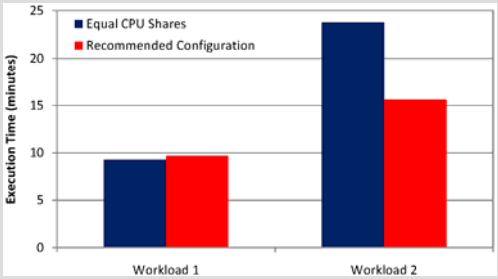


Virtual Machine Configuration

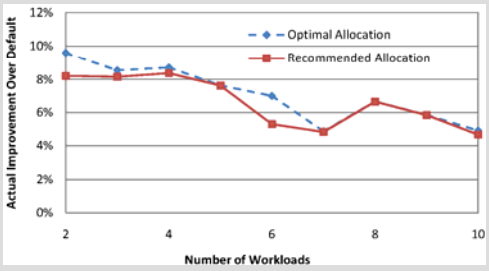
- Given a set of database workloads, each running in its own virtual machine, sharing one physical machine, ***how much resources to give to each virtual machine?***
  - Server consolidation scenario
- ***Virtualization design advisor*** [somi08]:
  - Use calibrated optimizer for cost modeling
  - Refine cost model based on on-line cost observations
  - Employ greedy search to find optimal allocation while satisfying quality of service constraints

Effect of Resource Allocation

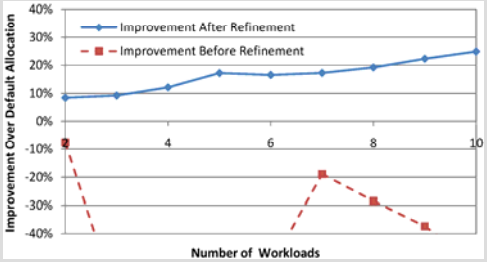
- Workloads on 10GB TPC-H database
  - PostgreSQL on Xen virtual machines
- Virtualization design advisor:
  - ***20%*** of CPU capacity to Workload 1, ***80%*** to Workload 2



Multiple Workloads



10 TPC-H Workloads

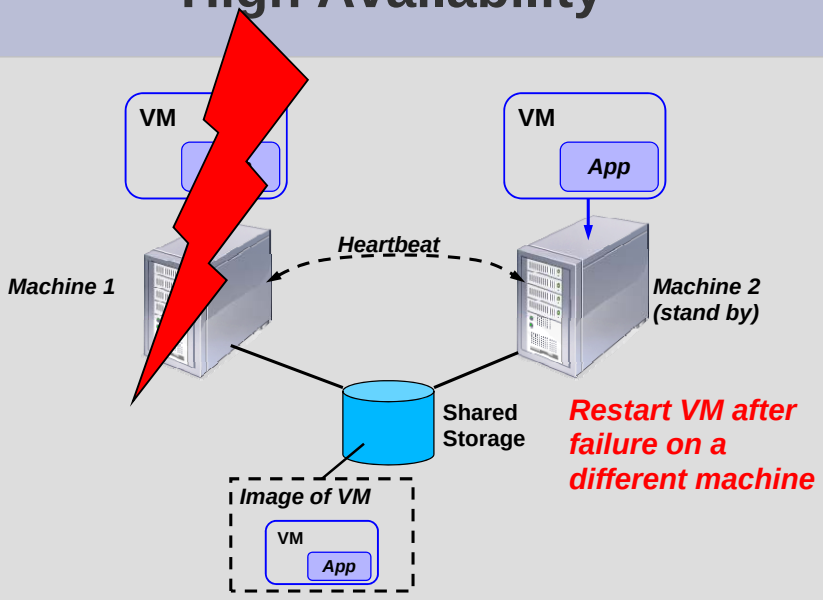


10 TPC-H + OLTP Workloads

Possible Future Directions

- Dynamically changing workloads
- Better modeling of resource costs
  - Concurrent queries
  - Non-database workloads
- Configuring the DBMS parameters
- ***Hints*** between DBMS and VMM
  - Performance objectives
  - Resource allocation constraints

High Availability



High Availability

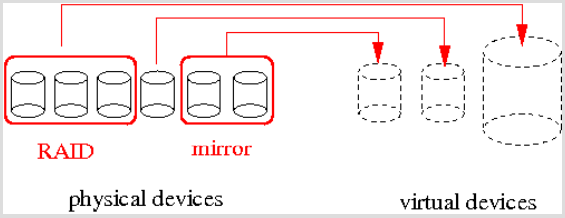
- Problem: Need to protect ***large amounts of state***
  - Database
  - DBMS processes
- Research questions: Can we…
  - rely on database recovery to reduce necessary work?
  - restart from the original VM image?
  - do better by having more recent images?
  - speed up database recovery after restart?
  - avoid losing connections and buffer pool?
  - leverage work on persistent DBMS sessions [lowe98, balo00, balo04]?
  - do this without shared storage?

Outline

- Introduction
- Machine Virtualization
- Storage Virtualization
  - Overview of storage virtualization
  - Why use virtual storage?
  - Implementations of storage virtalization
  - Challenges and opportunities for DBMSes
- Virtualization-like Capabilities in the DBMS
- Conclusion

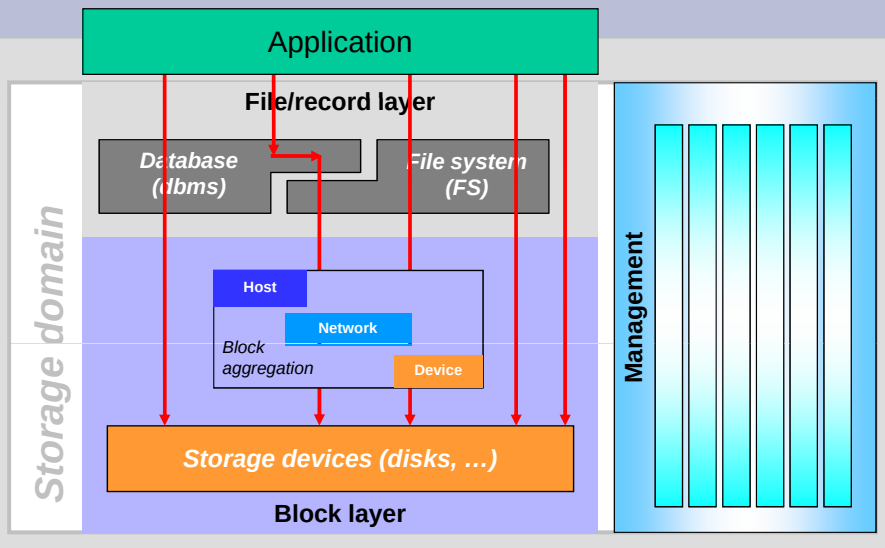
What is Storage Virtualization?

- storage virtualization is a layer of indirection that allows the definition of virtual storage devices



- virtualization isolates storage clients from the physical reality of the storage system

SNIA Shared Storage Model



Basic Capabilities of Virtual Storage

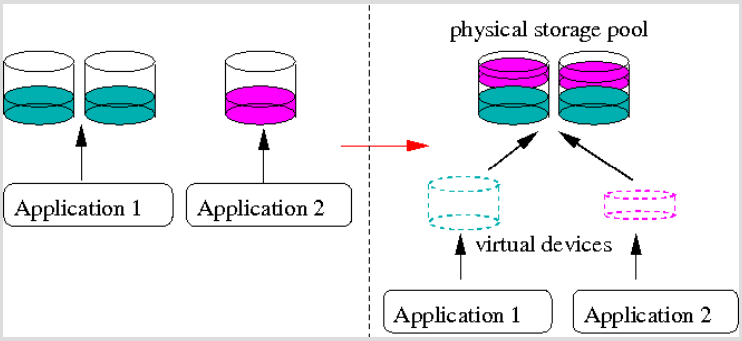
- create, destroy virtual devices using available *pool* of physical storage
- grow, shrink virtual devices
- control properties of virtual devices
  - size
  - performance
  - Reliability
- dynamic provisioning of physical storage

Additional Capabilities of Virtual Storage

- versioning, snapshots, point-in-time copies
- local and remote mirroring
- migration of virtual devices
  - supports provisioning, hierarchical storage management
- auto-administration
  - policy-based management
- storage QoS and performance isolation
  - active research area: [kaka05, utyi05, hape04,wech04, goja03, lume03, brbr99]

Why Virtualize Storage?

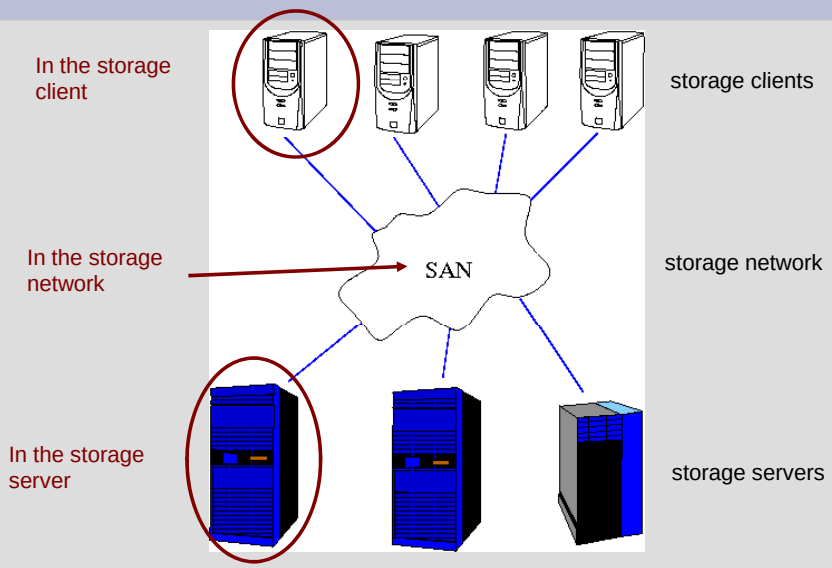
- improve storage utilization
- reduce storage costs



Why Virtualize Storage?

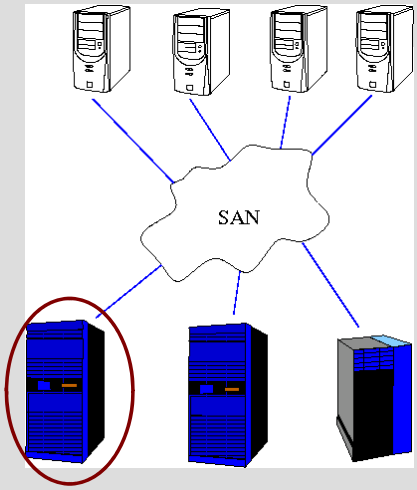
- minimize/avoid downtime
  - simplify maintenance tasks
  - transparent redundancy
- improve performance
  - distribute and balance storage loads
  - dynamic storage provisioning
  - control placement
- reduce cost of storage administration
  - single point of administrative control
  - simplified operations
  - automation, e.g., policy-based management

How to Virtualize Storage



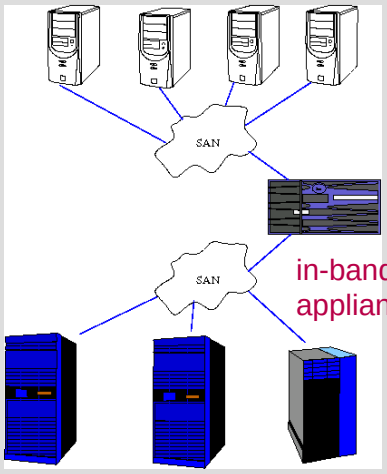
Virtualization in Storage Servers

- virtual devices limited to a single server
- storage server coordination, e.g., mirroring



Virtualization Appliances

- integrate heterogeneous servers
- centralized administration
- potential performance bottleneck

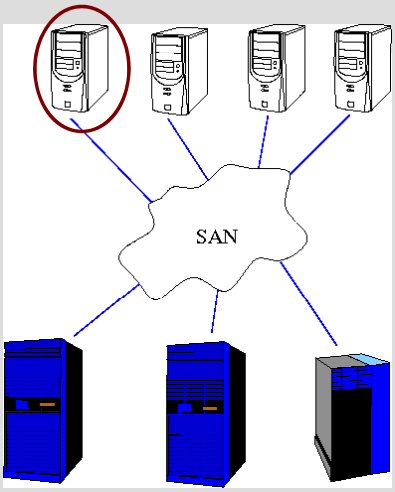


Example: HP StorageWorks SVS200

- in-band appliance
- pooling of heterogeneous storage servers
- transparent data migration
- local and remote mirroring
  - 1-safe and 2-safe mirroring
  - split mirrors

Virtualization in Storage Clients

- via *logical volume management* (LVM) in the storage client
- e.g., Linux LVM2
  - create, destroy, resize, snapshot, migrate logical volumes



The DBMS Perspective: So What?

- Virtual storage devices are dynamic
  - affects DBMS storage management, e.g., resizable tablespaces
- Virtual storage devices are opaque
  - affects DBMS configuration and tuning
- Virtual storage devices are based on shared physical resources
  - separate administration, distinct goals
- Virtual storage devices more capable than physical devices
  - CPU/memory, functionality

DBMS Configuration and Tuning

- characteristics of physical storage hidden from the DBMS (and DBA)
- how to:
  - layout DB objects?
  - set DBMS parameters?
    - page/extent sizes
    - prefetching
    - costs

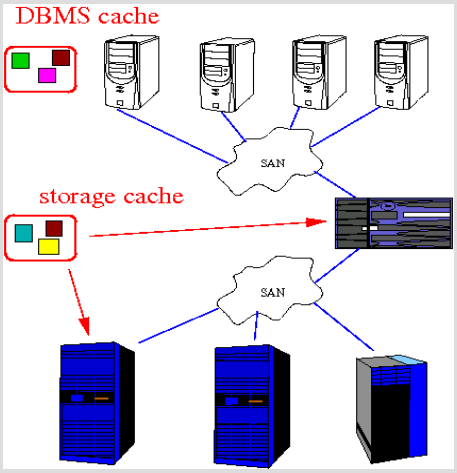
```
CREATE TABLESPACE ts
MANAGED BY DATABASE
USING (
  DEVICE '/dev/rlv1' 10000,
  DEVICE '/dev/rlv2' 10000)
OVERHEAD 6.0
TRANSFERRATE 0.05
PAGESIZE 8192
```

Exploiting Storage Capabilities

- storage snapshots and versioning
  - backup and recovery
  - concurrent applications
- storage replication and mirroring
  - dynamic DBMS provisioning
- dynamic storage resource allocation
  - accommodate workload fluctuations
- CPU and memory in the storage system
  - caching
  - offload DBMS functions

Multi-Tier Caching

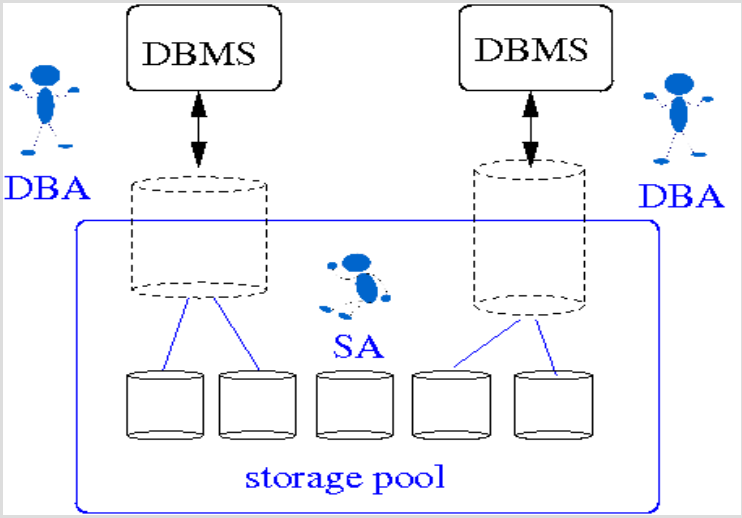
- exploit memory in the storage system
- DBMS can improve 2nd tier cache performance
- example: hinting [liab05]



DBMS and Storage Administration

- Textbook:
  - DBA understands and controls dedicated physical devices
- Reality:
  - storage is virtual
  - virtual storage is separately administered
  - DBA and storage administrator (SA) must coordinate

DBAs and SAs

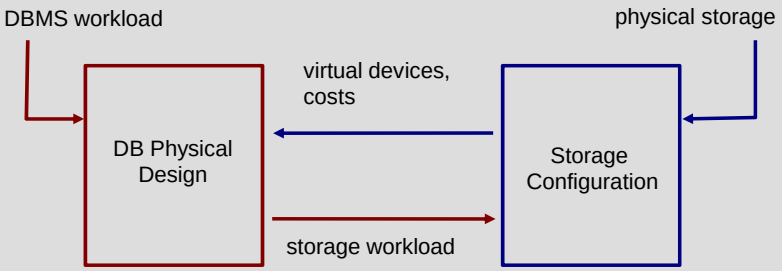


Tasks of DBAs and SAs

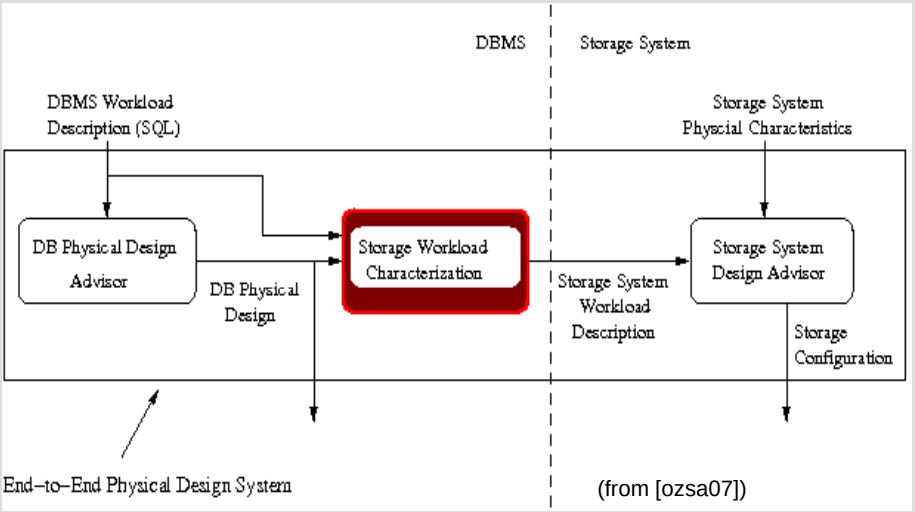
- database administration
  - DB physical design [brch05, coba05, vazu00, agch00]
  - layout [agch03]
- storage administration:
  - define and configure virtual devices
  - storage allocation, capacity planning
  - tools [anho02, waos02, anka01, albo01, dech03]

DB+Storage Co-Design

- DB Physical Design
  - index selection, layout
- Storage Configuration
  - define, layout, configure virtual devices



Storage Workload Estimation



(from [ozsa07])

Outline

- Introduction
- Machine Virtualization
- Storage Virtualization
- Virtualization-like Capabilities in the DBMS
  - Workload management
  - Provisioning through database replication
- Conclusion

Workload Management in DB2

- Control allocation of resources to database workloads ***from within database*** systems without using virtualization
- Different ***workloads*** get different levels of resources
- Available in DB2 v9.5

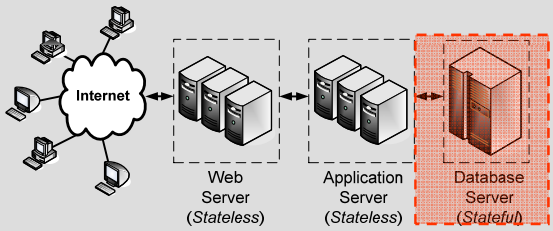
Workload Management in DB2

- Queries are classified into workloads based on the characteristics of their ***database connection***
  - Application name, user id, …
- Can set thresholds on the amount of work performed by different workloads
  - Elapsed time, idle time, rows returned, concurrent instances of workload, …
- Stop workload if a threshold is exceeded

Discussion

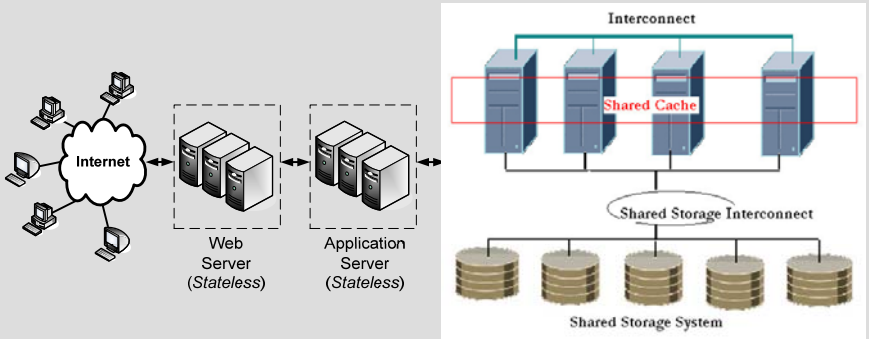
- Coarse grained approach to controlling resource consumption
- Resource consumption not measured directly
  - Difficult to measure CPU and I/O consumed by a query
- Workload manager can be integrated with operating system workload manager (AIX) to provide more fine grained control of CPU
  - Can be viewed as a form of ***hint***
- Workload manager provides monitoring data, but ***thresholds have to be set by DBA***

Database Replication



- Replication of front-end already possible
  - through dynamic server provisioning e.g., IBM's Tivoli, WebSphereXD, [Benn05], [Urga05], [Kar06]
- Database tier typically ***not replicated***

Replication with Oracle RAC

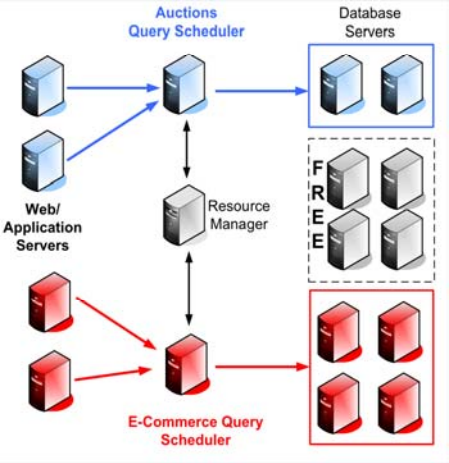


- Network attached shared storage solution, allows CPU provisioning in the DB back-end [Lah01]

Virtualized DB Tier Architecture

Technical challenges

- State management
- Replica allocation
  - Decide how many instances of each application should run
- Replica mapping
  - Decide on which node each instance should run

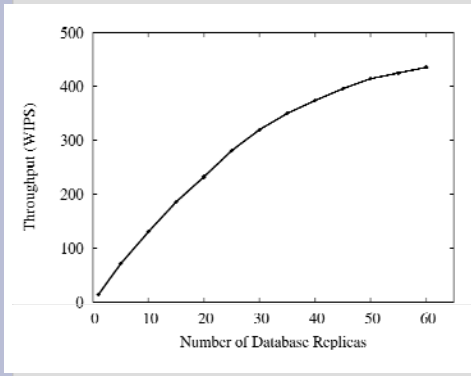


Resource manager controls allocations and mapping

State Management Approaches

- Use full database replication
- Asynchronous replication with consistency guarantees
  [Platt04], [Lin05], [Amza05], [Da06]
- Read-one write-all workload scheduling
  - within each application's allocation

Why Database Replication ?

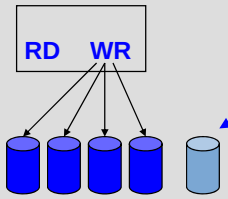


Scaling for E-Commerce (TPC-W) [Amza03]

Shown to scale well
- **[Platt04]** MW'04
- **[Lin05]** SIGMOD'05
- **[Amza05]** ICDE'05
- **[Da06]** VLDB'06

Unified approach to load peaks and fault handling

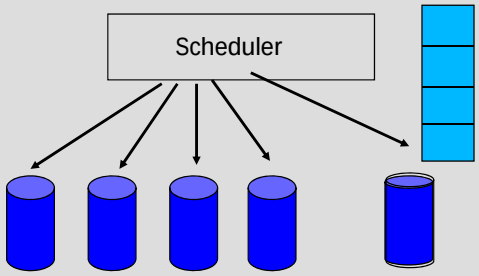
State Management Challenge



Adaptation steps:

1. Updating replica [Das05], [Kemme01], [So06]
2. Load balancing and buffer pool warm-up

Updating replica

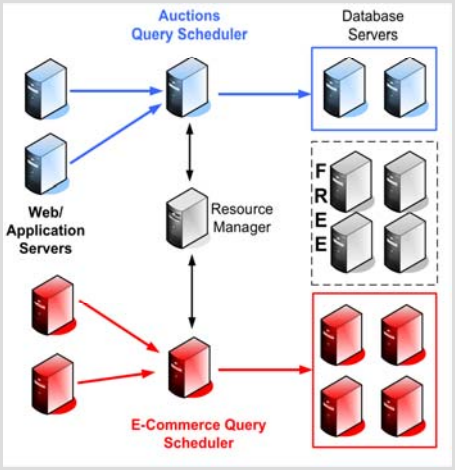


Update log replay on new replica
When do we start sending new updates to replica ?
[Kemme01], [Das05], [So06], [SA06]

Virtualized DB Tier Architecture

Technical challenges

- State management
- Replica allocation
  - Decide how many replicas of each application should run
- Replica mapping
  - Decide on which node each instance should run



Conclusion

- Virtualization
  - Powerful mechanisms for improving computing infrastructure
  - Adopted by a wide range of organizations
- Database systems are increasingly being run in virtualized environments
  - Significantly changes the operating environment
  - At the same time can be very useful
- ***Many opportunities for database researchers***

References

- **[adag06]** Keith Adams and Ole Agesen. A Comparison of Software and Hardware Techniques for x86 Virtualization. International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), October 2006.
- **[agch00]** Sanjay Agrawal, Surajit Chaudhuri, and Vivek R. Narasayya. Automated selection of materialized views and indexes in SQL databases. In Proc. International Conference on Very Large Data Bases, pages 496-505, 2000.
- **[agch03]** Sanjay Agrawal, Surajit Chaudhuri, Abhinandan Das, and Vivek Narasayya. Automating layout of relational databases. In International Conference on Data Engineering (ICDE'03), pages 607-618, 2003.
- **[albo01]** Guillermo A. Alvarez, Elizabeth Borowsky, Susie Go, Theodore H. Romer, Ralph Becker-Szendy, Richard Golding, Arif Merchant, Mirjana Spasojevic, Alistair Veitch, and John Wilkes. Minerva: An automated resource provisioning tool for large-scale storage systems. ACM Transactions on Computer Systems, 19(4):483-518, 2001.
- **[Amza03]** C. Amza, A. L. Cox and W. Zwaenepoel. Conict-Aware Scheduling for Dynamic Content Applications. In USENIX Symp. on Internet Tech. and Sys. 2003.

References

- **[Amza05]** C. Amza, A. Cox, W. Zwaenepoel. A Comparative Evaluation of Transparent Scaling Techniques for Dynamic Content Servers. In ICDE '05.
- **[anho02]** Eric Anderson, Michael Hobbs, Kimberly Keeton, Susan Spence, Mustafa Uysal, and Alistair Veitch. Hippodrome: running circles around storage administration. In Conference on File and Storage Technology (FAST'02), pages 175-188, January 2002.
- **[anka01]** E. Anderson, M. Kallahalla, S. Spence, R. Swaminathan, and Q. Wang. Ergastulum: quickly finding near-optimal storage system designs. Technical Report HPL-SSP-2001-5, HP Laboratories, July 2001.
- **[badr03]** Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and the Art of Virtualization. ACM Symposium on Operating Systems Principles (SOSP), pp. 164-177, October 2003.
- **[balo00]** Roger S. Barga, David B. Lomet, Thomas Baby, Sanjay Agrawal: Persistent Client-Server Database Sessions. International Conference on Extending Database Technology (EDBT), pp. 462-477, March 2000.
- **[balo04]** Roger S. Barga, David B. Lomet, German Shegalov, Gerhard Weikum: Recovery guarantees for Internet applications. ACM Transactions on Internet Technology 4(3): 289-328, August 2004.

References

- **[Benn05]** M. N. Bennani and D. A. Menasce, Resource allocation for autonomic data centers using analytic performance models, in ICAC 2005.
- **[Bowm07]** I. T. Bowman, P. Bumbulis, D. Farrar, A. K. Goel, B. Lucier, A. Nica, G.N. Paulley, J. Smirnios, M. Young-Lai. SQL Anywhere: A Holistic Approach to Database Self-management.SMDB 2007.
- **[brbr99]** John L. Bruno, Jose Carlos Brustoloni, Eran Gabber, Banu Ozden, and Abraham Silberschatz. Disk scheduling with quality of service guarantees. In IEEE International Conference on Multimedia Computing and Systems (ICMCS 1999), Vol. 2, pages 400-405, 1999.
- **[brch05]** Nicolas Bruno and Surajit Chaudhuri. Automatic physical database tuning: A relaxation-based approach. In Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data (SIGMOD'05), 2005.
- **[Brown93]** Brown, K.P., Carey, M.J., Livny, M. Managing memory to meet multiclass workload response time goals. In VLDB 1993.
- **[Brown96]** Brown, K.P., Carey, M.J., Livny, M. Goal-oriented buffer management revisited. In SIGMOD 1996.
- **[Chen06]** J. Chen, G. Soundararajan, C. Amza. Autonomic Provisioning of Backend Databases in Dynamic Content Web Servers. In ICAC 2006.

References

- **[Chen07]** J. Chen, G. Soundararajan, M. Mihailescu and C. Amza. Outlier Detection for Fine-grained Load Balancing in Database Clusters . In the 2nd International Workshop on Self-Managing Database Systems (SMDB) 2007.
- **[chzh05]** Z. Chen, Y. Zhang, Y. Zhou, H. Scott, and B. Schiefer. Empirical evaluation of multi-level buffer cache collaboration for storage systems. In Proceedings of the International Conference on Measurements and Modeling of Computer Systems (SIGMETRICS'05), pages 145-156, 2005.
- **[clfr05]** Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hansen, Eric Jul, Christian Limpach, Ian Pratt and Andrew Warfield. Live Migration of Virtual Machines. USENIX Symposium on Networked Systems Design and Implementation (NSDI), May 2005.
- **[coba05]** Mariano P. Consens, Denilson Barbosa, Adrian M. Teisanu, and Laurent Mignet. Goals and benchmarks for autonomic configuration recommenders. In Proc. ACM SIGMOD International Conference on Management of Data (SIGMOD'05), 2005.
- **[Da06]** K. Daudjee, K. Salem. Lazy database replication with snapshot isolation. In VLDB'06.

References

- **[Das05]** K.Dasgupta, S.Ghosal, R.Jain, U.Sharma, A.Verma, QoSMig: Adaptive Rate-Controlled Migration of Bulk Data in Storage Systems, In *ICDE'05.*
- **[dech03]** Murthy Devarakonda, David Chess, Ian Whalley, Alla Segal, Pawan Goyal, Aamer Sachedina, Keri Romanufa, Ed Lassettre, William Tetzlaff, and Bill Arnold. Policy-based autonomic storage allocation. In Proc. 14th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM), number 2867 in Lecture Notes in Computer Science, pages 143-154. Springer-Verlag, 2003.
- **[Diao02]** Y. Diao, J. L. Hellerstein, AND S. Parekh, Optimizing quality of service using fuzzy control. In *Proceedings of the 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management, 2002*.
- **[Elni07]** S. Elnikety, S. Dropsho and W. Zwaenepoel. Tashkent+: Memory-Aware Load Balancing and Update Filtering in Replicated Databases. In EuroSys 2007.
- **[engu95]** Dawson R. Engler, Sandeep K. Gupta, and M. Frans Kaashoek. AVM: Application-level Virtual Memory. Workshop on Hot Topics in Operating Systems (HotOS), May 1995.
- **[fidi05]** Renato Figueiredo, Peter A Dinda, Jose Fortes. Resource Virtualization Renaissance. IEEE Computer, 38(5):28-31, May 2005.

References

- **[Ghan07]** S. Ghanbari, G. Soundararajan, J. Chen, and C. Amza Adaptive Learning of Metric Correlations for Temperature-Aware Database Provisioning, In ICAC 2007.
- **[goja03]** Pawan Goyal, Divyesh Jadav, Dharmendra S. Modha, and Renu Tewari. CacheCOW: QoS for storage system caches. In Eleventh International Workshop on Quality of Service (IWQoS 03), 2003.
- **[gold74]** Robert P. Goldberg. Survey of Virtual Machine Research. IEEE Computer, pp. 34 45, June 1974
- **[hape04]** Lan Huang, Gang Peng, and Tzi cker Chiueh. Multi-dimensional storage virtualization. In Proc. Joint International Conference on Measurement and Modeling of Computer Systems, pages 14-24, 2004.
- **[kaka05]** Magnus Karlsson, Christos T. Karamanolis, and Xiaoyun Zhu. Triage: Performance differentiation for storage systems using adaptive control. ACM Transactions on Storage, 1(4):457-480, November 2005.
- **[Kar06]** A.Karve, T.Kimbrel, G.Pacifici, M.Spreitzer, M.Steinder, M.Sviridenko, A. Tantawi. Dynamic placement for clustered web applications. In *WWW 2006*.
- **[Kemme01]** B. Kemme, A. Bartoli, Ö. Babaoglu, Online Reconfiguration in Replicated Databases Based on Group Communication. In *DSN 2001.*

References

- **[kidu03]** Samuel T. King, George W. Dunlap, Peter M. Chen. Operating System Support for Virtual Machines. USENIX Annual Technical Conference, June 2003.
- **[Lah01]** T. Lahiri, V. Srihari, W. Chan, N. MacNaughton, and S. Chandrasekaran. Cache fusion: Extending shared-disk clusters with shared caches. In VLDB 2001.
- **[liab05]** X. Li, A. Aboulnaga, A. Sachedina, K. Salem, and S. Gao. Second-tier cache management using write hints. In USENIX Conference on File and Storage Technologies (FAST'05), pages 115-128, December 2005.
- **[liiy05]** Q. Lin, B. R. Iyer, D. Agrawal, and A. El Abbadi. SVL: Storage virtualization engine leveraging DBMS technology. In Proceedings of the 21st International Conference on Data Engineering (ICDE'05), pages 1048-1059, 2005.
- **[Lin05]** Y. Lin, B. Kemme, M. Patino-Martinez, and R. Jimenez-Peris. Middleware based Data Replication providing Snapshot Isolation, In *SIGMOD'05*.
- **[lowe98]** David B. Lomet and Gerhard Weikum. Efficient and Transparent Application Recovery in Client-Server Information Systems. SIGMOD Conference, pp. 460-471, June 1998.

References

- **[lume03]** Christopher Lumb, Arif Merchant, and Guillermo Alvarez. Facade: virtual storage devices with performance guarantees. In Proceedings of the 2nd USENIX Conference on File and Storage Technologies, pages 131-144, 2003.
- **[Mart06]** P. Martin, W. Powley, X. Xu, W. Tian. Automated Configuration of Multiple Buffer Pools. The Computer Journal, 2006.
- **[Mena01]** D. A. Menasce, D. Barbara, AND R. Dodge. Preserving QoS of e-commerce sites through self-tuning: A performance model approach. In Proceedings of the 3rd ACM Conference on Electronic Commerce,2001.
- **[miya08]** Umar Farooq Minhas, Jitendra Yadav, Ashraf Aboulnaga, and Kenneth Salem. Database Systems on Virtual Machines: How Much do You Lose? In Proc. International Workshop on Self-Managing Database Systems (SMDB '08), 2008.
- **[ozsa07]** O. Ozmen, K. Salem, M. Uysal and M. H. Sheikh Attar. Storage Workload Estimation for Database Management Systems. In Proc. ACM SIGMOD International Conference on Management of Data (SIGMOD'07), 2007.
- **[pazh07]** Pradeep Padala, Xiaoyun Zhu, Mustafa Uysal, Zhikui Wang, Sharad Singhal, Arif Merchant, and Kenneth Salem. Adaptive Control of Virtualized Resources in Utility Computing Environments. European Conference on Computer Systems, (EuroSys), March 2007.

References

- **[Peri02]** R. Jiménez-Peris, M. Patiño-Martínez, B. Kemme AND G. Alonso. Improving Scalability of Fault Tolerant Database Clusters. In *ICDCS'02*.
- **[Platt04]** C. Plattner and G. Alonso. Ganymed: Scalable replication for transactional web applications. *In Middleware 2004*.
- **[pogo74]** Gerald J. Popek and Robert P. Goldberg. Formal Requirements for Virtualizable Third Generation Architectures. Communications of the ACM, 17 (7): 412 –421, July 1974.
- **[Qin07]** Ye Qin, Kenneth Salem, and Anil Goel. Towards adaptive costing of database access methods. In SMDB 2007.
- **[roga05]** Mendel Rosenblum and Tal Garfinkel. Virtual Machine Monitors: Current Technology and Future Trends. IEEE Computer, 38(5):39-47, May 2005. NIX Security Symposium, August 2000.
- **[roir00]** John Scott Robin and Cynthia E. Irvine. Analysis of the Intel Pentium's Ability to Support a Secure Virtual Machine Monitor. USE
- **[smna05]** James E. Smith and Ravi Nair. The Architecture of Virtual Machines. IEEE Computer, 38(5):32-38, May 2005.
- **[So06]** G. Soundararajan, C. Amza, A. Goel. Database Replication Policies for Dynamic Content Applications. In *EuroSys 2006*.

References

- **[So06]** G. Soundararajan, C. Amza. Reactive provisioning of backend databases in shared dynamic content server clusters. In ACM Transactions on Autonomous and Adaptive Systems (TAAS), *p151 – p188, Vol. 1, Issue 2 (December 2006).*
- **[soab07]** Ahmed A. Soror, Ashraf Aboulnaga, and Kenneth Salem. Database Virtualization: A New Frontier for Database Tuning and Physical Design. International Workshop on Self-Managing Database Systems (SMDB 2007), April 2007.
- **[somi08]** Ahmed Soror, Umar Farooq Minhas, Ashraf Aboulnaga, Kenneth Salem, Peter Kokosielis, and Sunil Kamath. Automatic Virtual Machine Configuration for Database Workloads. In Proc. ACM SIGMOD International Conference on Management of Data (SIGMOD '08), 2008.
- **[suve01]** Jeremy Sugerman, Ganesh Venkitachalam, and Beng-Hong Lim. Virtualizing I/O Devices on VMware Workstation's Hosted Virtual Machine Monitor. USENIX Annual Technical Conference, June 2001.
- **[Tes05]** G. Tesauro, R. Das, W. E. Walsh, and J. O. Kephart. Utility-function-driven resource allocation. in ICAC 2005.
- **[Tes06]** G.Tesauro, R.Das, N.Jong, M. Bennani. A hybrid reinforcement learning approach to autonomic resource allocation. In *ICAC 2006*.

References

- **[Tot06]** A. Totok and V. Karamcheti. Improving performance of internet services through reward- driven request prioritization. In *IWQoS 2006*.
- **[Urga05]** B. Urgaonkar, G. Pacifici, P. Shenoy, M. Spreitzer and A. Tantawi. An analytical model for multi-tier internet services and its applications. In *SIGMETRICS 2005.*
- **[utyi05]** Sandeep Uttamchandani, Li Yin, Guillermo A. Alvarez, John Palmer, and Gul Agha. CHAMELEON: A self-evolving, fully-adaptive resource arbitrator for storage systems. In Proc. USENIX 2005 Annual Technical Conference, pages 75-88, 2005.
- **[vazu00]** Gary Valentin, Michael Zuliani, Daniel C. Zilio, Guy M. Lohman, and Alan Skelley. DB2 Advisor: An optimizer smart enough to recommend its own indexes. In 16th International Conference on Data Engineering, pages 101-110, 2000.
- **[vome04]** K. Voruganti, J. Menon, and S. Gopisetty. Land below a DBMS. SIGMOD Record, 33(1):64-70, March 2004.
- **[wal02]** Carl A. Waldspurger. Memory Resource Management in VMware ESX Server. Symposium on Operating Systems Design and Implementation (OSDI), December 2002.

References

- **[Wal04]** W.E. Walsh, G. Tesauro, J. O. Kephart, and R. Das. Utility functions in autonomic systems. In ICAC 2004.
- **[waos02]** Julie Ward, Michael O'Sullivan, Troy Shahoumian, and John Wilkes. Appia: automatic storage area network design. In Conference on File and Storage Technology (FAST'02), pages 203-217, January 2002.
- **[wech04]** Wei Jin, Jeffrey S. Chase, and Jasleen Kaur. Interposed proportional sharing for a storage service utility. In Proc. International Conference on Measurements and Modeling of Computer Systems (SIGMETRICS'04), pages 37-48, June 2004.
- **[whco05]** Andrew Whitaker, Richard S. Cox, Marianne Shaw, and Steven D. Gribble. Rethinking the Design of Virtual Machine Monitors. IEEE Computer, 38(5):57-62, May 2005.
- **[whsh02]** Andrew Whitaker, Marianne Shaw, and Steven D. Gribble. Denali: A Scalable Isolation Kernel. ACM SIGOPS European Workshop, September 2002.
- **[Wood06]** M. Woodside, T. Zheng, M. Litoiu. Service System Resource Management Based on a Tracked Layered Performance Model. In *ICAC 2006*.