SCHOONER

SCALE SMART

Flash Databases: High Performance and High Availability

Flash Memory Summit Software Tutorial

August 11,2011

Dr. John R. Busch
Founder and CTO
Schooner Information Technology
John.Busch@SchoonerInfoTech.com

Data

- Most important and valuable component of modern applications and websites

- Driving revolutionary changes in computing and the internet

    - New opportunities for generating revenue

    - More efficient use of current business processes and infrastructure

- Data access downtime or poor performance has a major cost to a business' bottom line

The Mission-Critical Imperative

“Let me tell you the difference between Facebook and everyone else, we don't crash EVER! If our service is down for even a day, our entire reputation is irreversibly destroyed!

Facebook and Google invest hundreds of millions of dollars every year on custom software and hardware infrastructure to optimize availability, performance, administration, and cost

Mission Critical Imperative

- Maintaining data availability and response time is critical for key classes of businesses
  - Web 2.0
  - eCommerce
  - High-volume websites
  - Telecommunications

- IT departments and application developers seek architectures and deployments providing
  - high service availability
  - resilient performance scalability

- Meet rising service demand while controlling capital and operating expenses

Mission-Critical Database Requirements

Mission-Critical Database Goals and Metrics

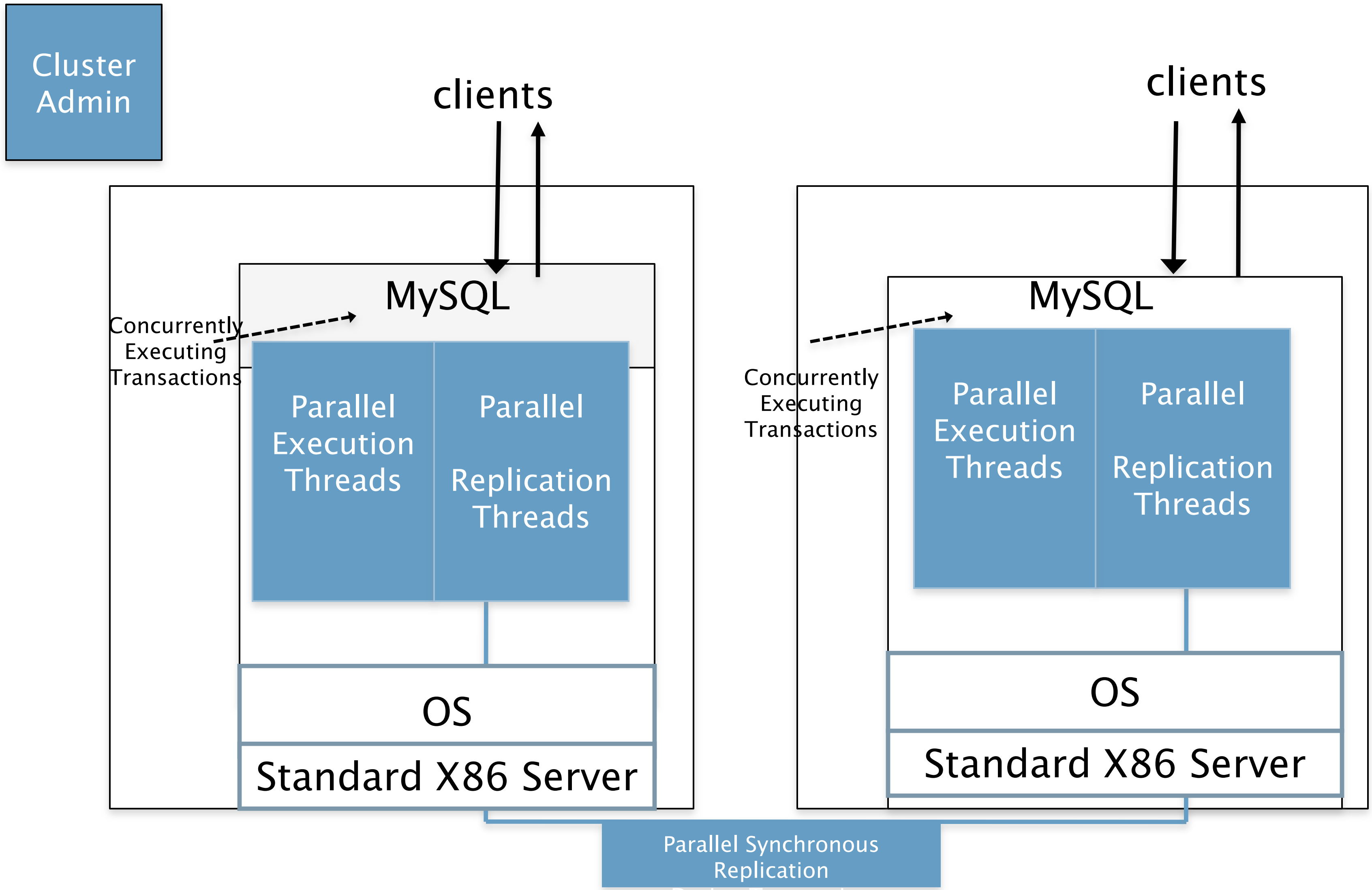
Goals	Metrics
High Availability	Service unavailability (minutes/year) from failures, disaster recovery, or during planned administration
High Data Integrity	Probability of data loss or corruption; data consistency levels
High Performance and Scalability	Transaction throughput, response time; performance scalability; performance stability
Simple and powerful administration	Ease of cluster administration; fail-over automation; monitoring and optimization tools
Cost effective	Total cost of ownership (TCO); return on investment (ROI)
Standards and Compatibility	Level of standards compliance and certification

SCHOONER
SCALE SMART

Opportunity for Large Impact on All Mission Critical Dimensions

- Flash
- Multi-core
- Optimized Database Architecture
- Cloud

Tightly-Coupled Flash Optimized Database with Synchronous Replication



Key Resource Management Algorithm Design Requirements

- Processor
  - Multi-core scalability
    - fine-grained locking, concurrent data structures
- Storage
  - Log files on HDD with persistent DRAM controller
    - Fast, saves flash for high access data
  - Concurrent DRAM buffer-pool management algorithms
    - Multi-threaded background write of dirty blocks so clean on misses
  - Batched commits
  - Highly-parallel multi-threaded flash-memory access
    - Utilizes ~150k IOPS for balancing a 2 socket Westmere Server with 64GB DRAM
    - Flash Cache give ~80% throughput if database working set fits in flash: must size
- Network
  - Memory to memory multi-threaded parallel synchronous

SCHOONER SCALE SMART

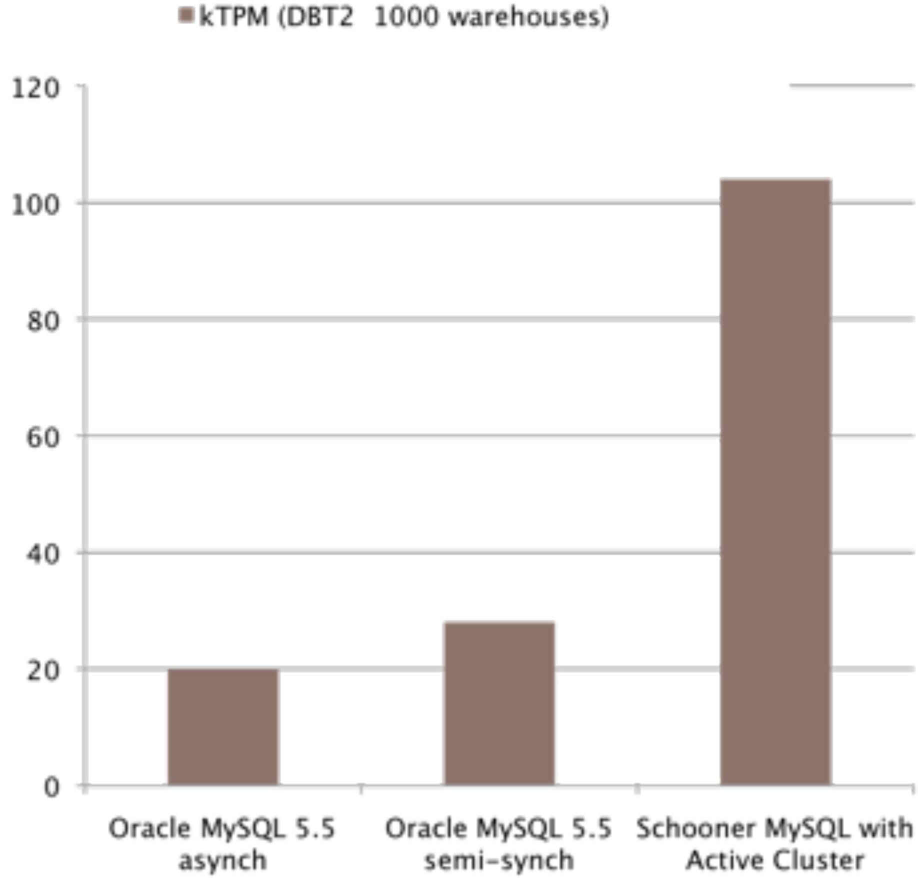
Tightly Coupled Database Design Enables Effective Vertical Scaling with Commodity Flash Memory and Horizontal Scaling

DBT2 open-source OLTP version of TPC-C
- 1000 warehouses, 32 connections
- 0 think-time
- Result metric: TPM (new order)

Measurement Configuration
- 2 node Master-Slave configuration
- 2 socket Westmere
- 72GB DRAM

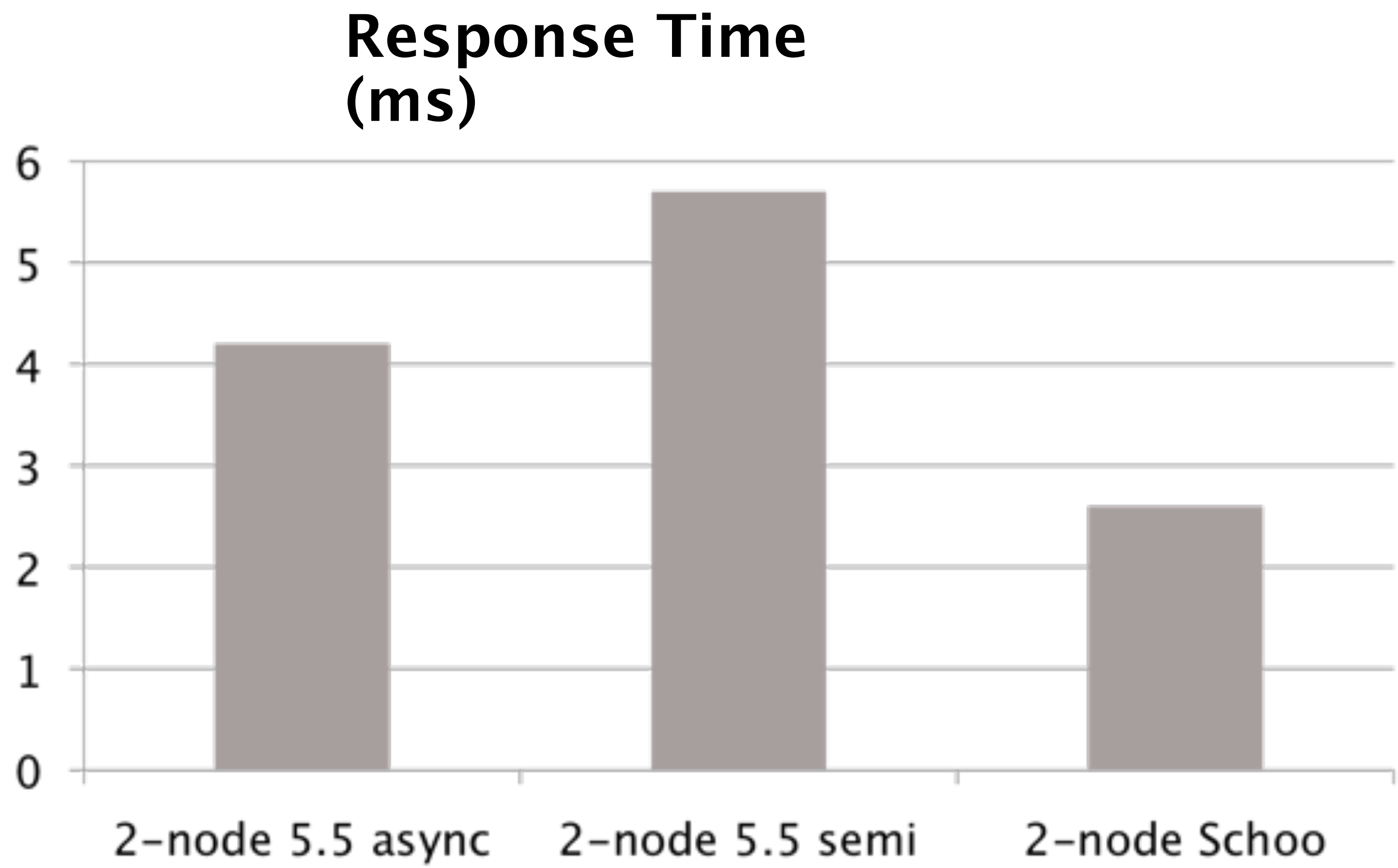
Transaction Throughput with Hard Disc Drives

Transaction Throughput with Flash



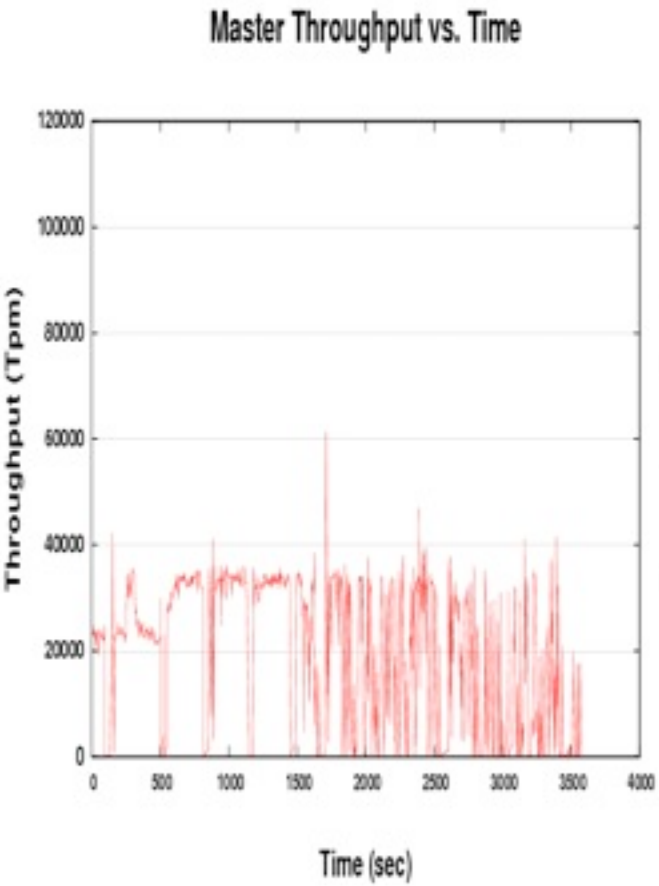
SCHOONER
SCALE SMART

Lower Response Times

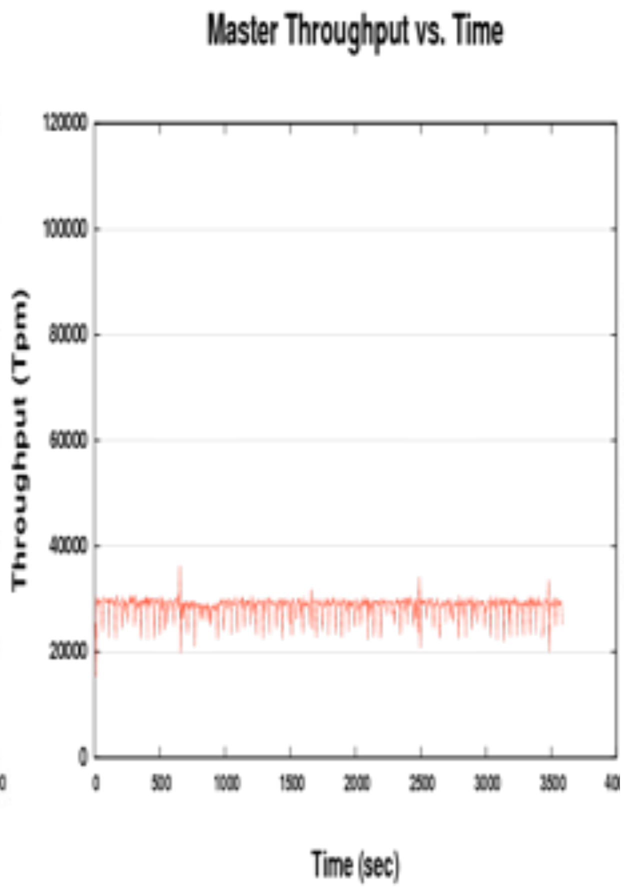


Higher Performance Stability

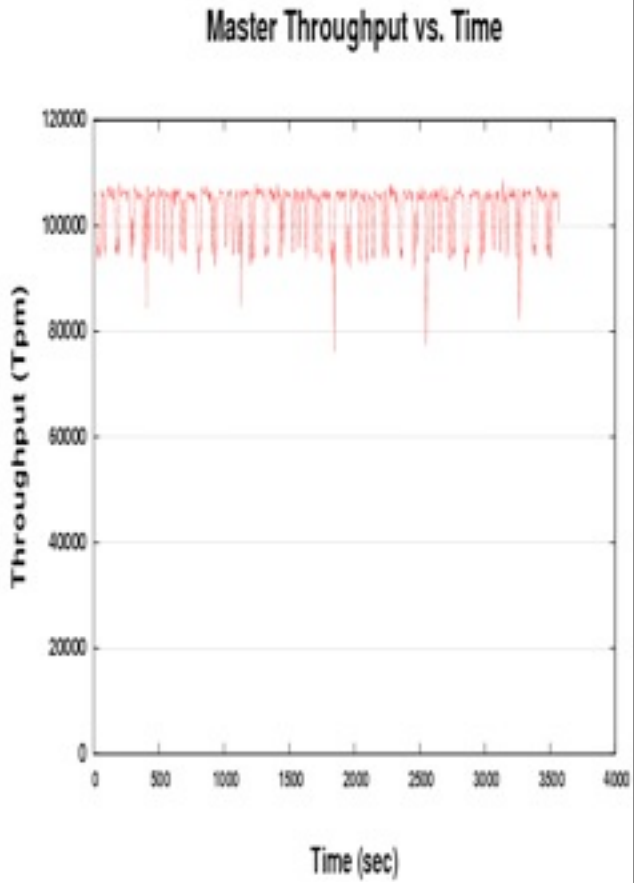
5.5 Async



5.5 Semi-sync

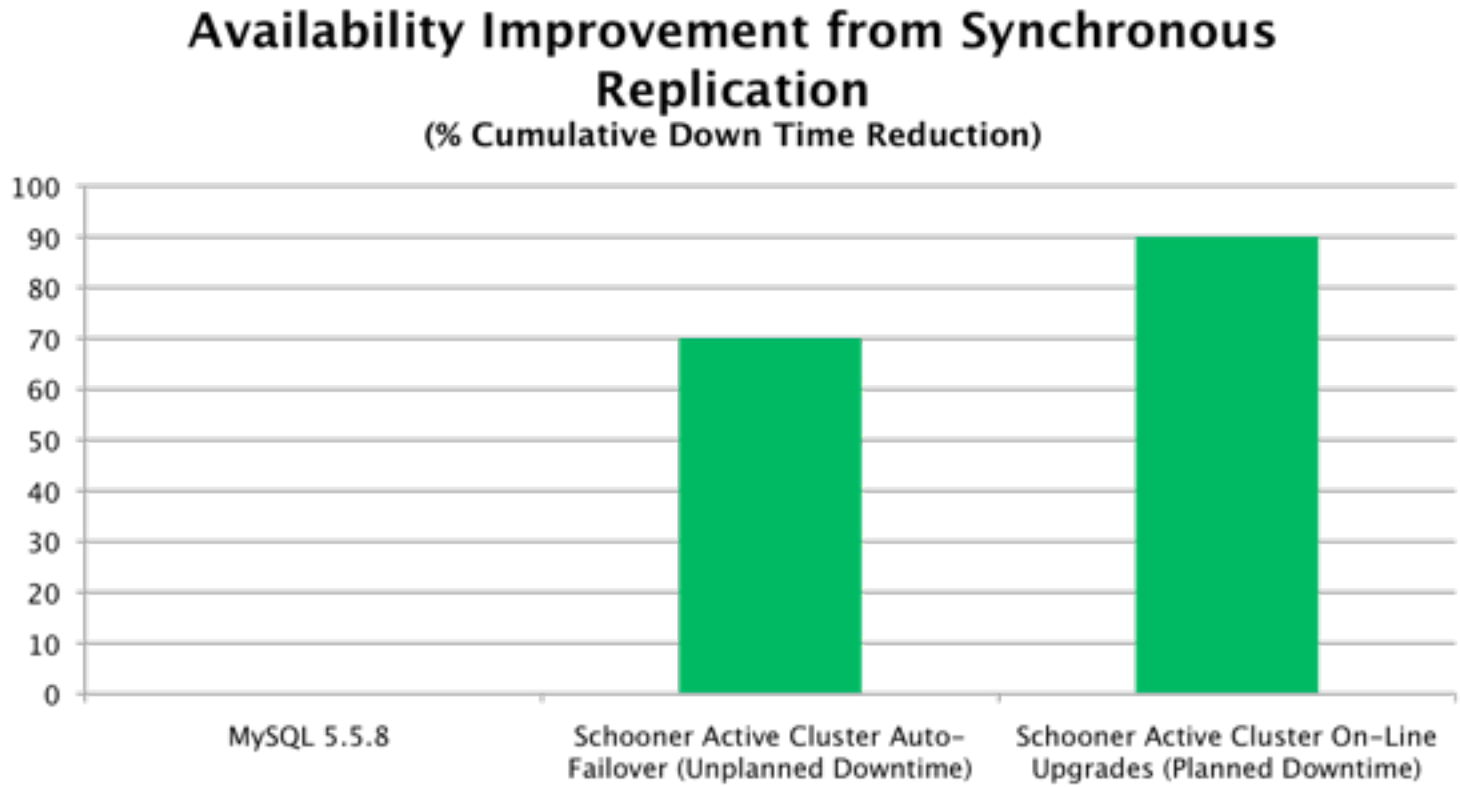


SAC



SCHOONER
SCALE SMART

Increased Service Availability and Data Integrity

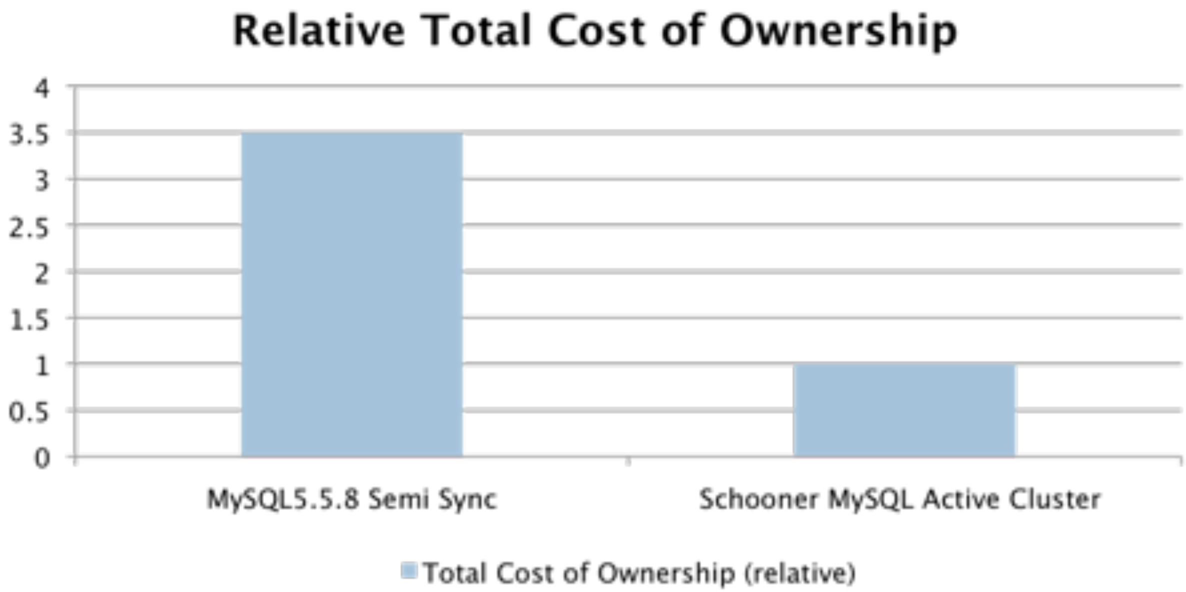


SCHOONER
SCALE SMART

Lower Total Cost of Ownership

Lower Cost

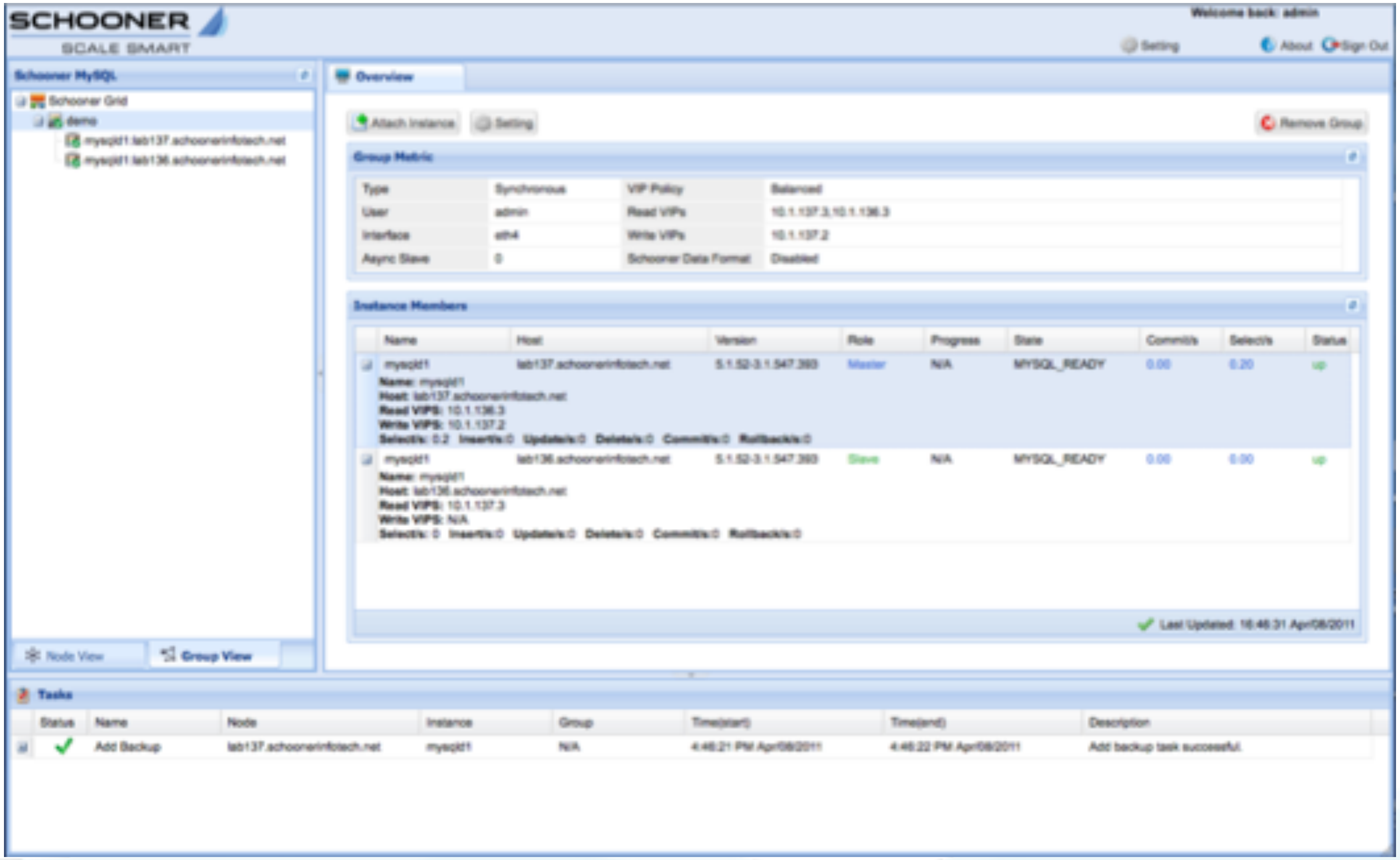
- Reduced capital and operating costs through reduction in servers, power, space, admin
- Savings from increased service availability and associated revenue and customer retention



- TCO and ROI models are customer and workload specific
- Function (throughput/server; server, rack, and network costs, software license and support costs, admin costs; space and power costs; cost of downtime)

Simplified Administration

- **Fail-over can be completely automatic and instant**
  - requiring no administrator intervention or service interruption
- **Cluster Administrator GUI and CLI can provide a single point for cluster-wide management**
  - single click slave creation and database migration; monitoring; trouble-shooting; tuning



Mission-Critical Database Best Practices

Goal	Best Practice
High Availability	Replication (synch local and asynch parallel WAN); automation of failure detection and recovery
High Data Integrity	Synchronous replication to eliminate data loss and fully consistent data; combined with parallel asynchronous replication for WAN disaster recovery
Excellent Performance and Scalability	Effective vertical and horizontal scaling for exploiting flash and multi-core
Simple and powerful administration	Centralized management; automation; visibility (statistics); alerts
Cost effective	Leverage commodity hardware and software; achieve high hardware utilization
Standards and Compatibility	100% standards compliance and certification

SCHOONER
SCALE SMART

Cloud Requirements and Challenges for Scaled Enterprise Services

- Cloud providers must deliver:
  - guaranteed service availability, performance, and elastic scale
  - multi-tenant management and security
  - and a net TCO savings vs. dedicated data centers

- Barriers in deploying enterprise class services into the cloud at scale
  - For many classes of applications and services:
    - the realized performance and availability characteristics of cloud deployments are disappointing at scale
    - the large quantity of cloud instances needed to support scaling a deployment drive the cost of cloud deployment to unacceptable levels
  - Opportunity for flash, but innovation is required

Current Cloud Virtualization : Successes and Limitations

- Cloud server-virtualization
  - Provisioning application instances in virtual machines on servers
    - combine existing applications with multi-core systems to increase utilization
    - elasticity of service capacity through dynamic provisioning of more or fewer application instances based on the current workload demand.
- Successes
  - applications that scale horizontally and can run under a VM hypervisor within a server's DRAM (eg web application tier)
  - works well for low volume apps and services (start-ups, new games, ...)
- Problems : scaled production databases
  - virtualization kills performance if they do not fit in DRAM

Cloud Virtualization Impact on Production Databases

- Databases in production cloud environments:
  - provide additional data partitioning (very small data bases)
  - provide additional caching layers to minimize I/O (breaks ACID)
  - provision many more database instances than in a non-virtualized environment

- Net Impact
  - drives up application and management complexity
  - increases cost
  - reduces service availability and data integrity

- Less than 10 percent of production data-tier server workloads are virtualized today.

Fusing Cloud + Flash + Optimized Databases

- Short term
  - virtualized machine instances for the web and application tiers
  - non-virtualized, vertically scaling data-tier solutions
    - Exploit balanced commodity, flash-based, multi-core system configurations
  - custom management APIs and tools to link together in a hybrid cloud

Fusing Cloud + Flash + Optimized Databases

- Longer Term : Innovation Required
  - Need improved virtualization technologies
    - Flash optimized virtualization cutting flash access overhead
    - unified virtual administration model
      - applicable to all tiers in the data center including flash-optimized data tier
      - dynamic provisioning, management, monitoring, and accounting
  - Large potential Quality of Service and TCO Benefits
    - increased performance, scalability, and service availability
    - reduced capital and operating expenses

Thank You!

Schooners, first built in the 1700s, applied an innovative design to the standard cargo sailing ship, enabling stupendous levels of speed and range. They enabled a set of visionary companies to enter new markets on a global basis. Where can a Schooner take your company?

The Schooner MySQL® with Active Cluster™ Advantage

Schooner MySQL with Active Cluster

High Availability

• No service interruption for planned or unplanned database downtime
•Instant automatic fail-over
• On-line upgrade and migration
• 90% less downtime vs. MySQL 5.5
•Full WAN support with master auto-failover

Data Integrity

• No lost data
• Cluster-wide data consistency

Great Performance and Scalability

• 4-20x more throughput/server vs. MySQL 5.5
•High performance synchronous and asynchronous replication

Visibility and Control

• Easy cluster administration
• No error-prone manual processes
•Monitoring and Optimization

Compelling Economics

• Cut server capex (consolidation)
• Cut opex (power, pipe, DBA time)
•Increase revenue (eliminate service interruptions)
• TCO 70% cheaper than MySQL 5.5

Out-of-the-box Product

• Full MySQL + InnoDB: not a toolkit
• Free your staff to build your business,
not a custom database

100% MySQL Enterprise Compatible and Certified

Broad Industry Deployments

• eCommerce, telco, financial,...

www.SchoonerInfoTech.com

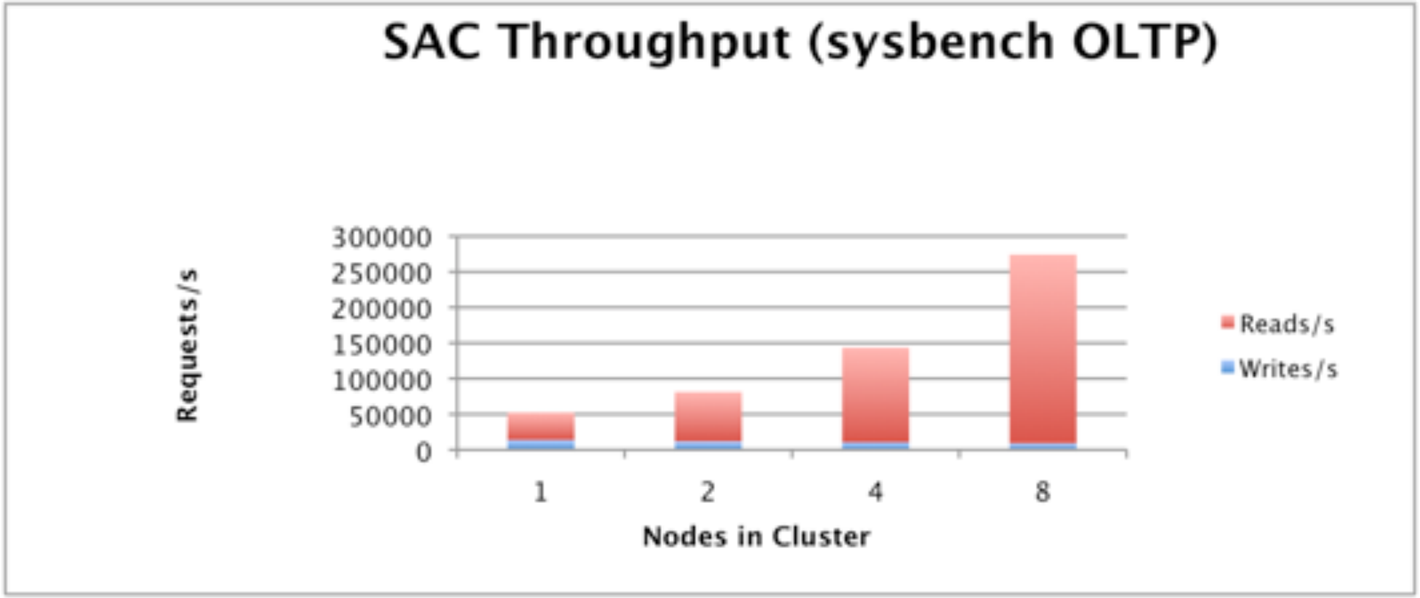
How Best to Provide WAN Replication and Disaster

- WAN/geographically dispersed data centers
  - Typically requires Asynchronous replication
    - Can’t add additional ~200ms with high potential variance to query response time for synchronous replication

- Goal: WAN asynch slave should automatically fail-over when synchronous master fail-over occurs
  - Requires WAN asynch replication to be loosely integrated with synch replication group

- Goal : Limit remote slave lag and recovery to ~ WAN latency
    - Maximize WAN data consistency
    - Minimize disaster recovery time
  - Requires high performance asynchronous replication
    - Need multi-threaded asynch for parallelizing updates

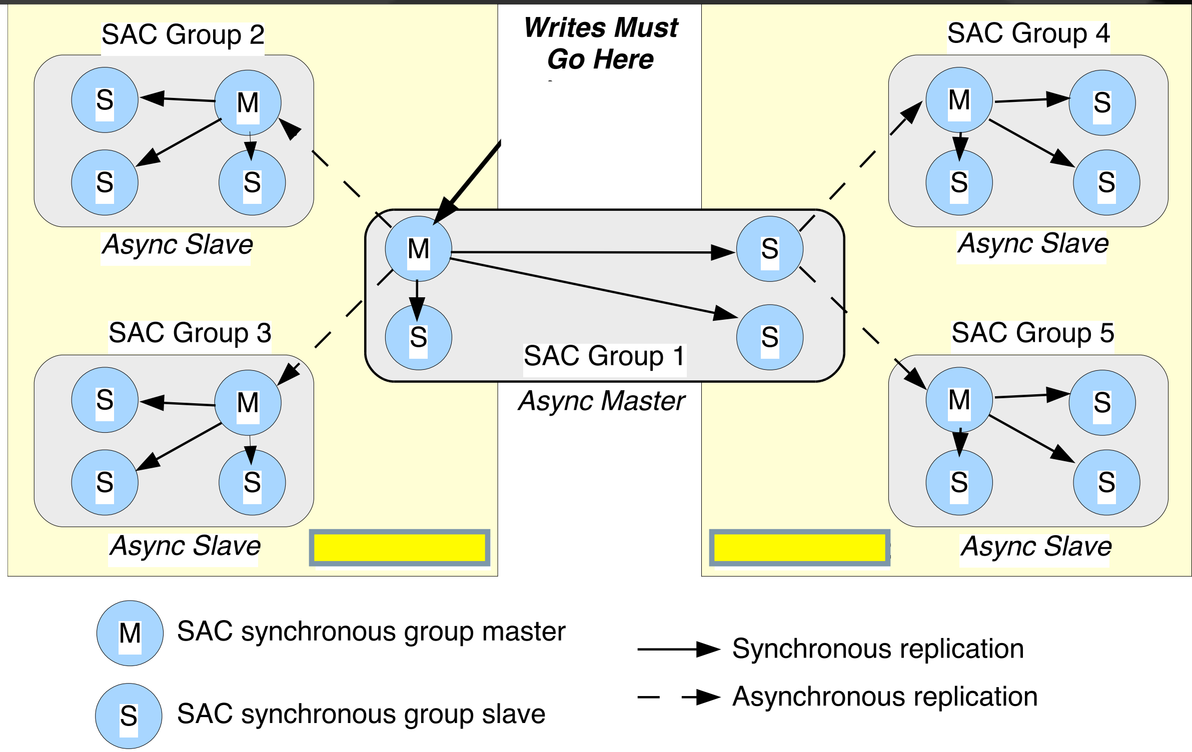
Limitations of Synchronous Replication: How Best to Scale Database Queries



Query Scaling in a Synchronous Replication Group

- Fully replicated Master/Slave cluster
  - No cluster overhead for adding queries to a slave
  - Can add synchronous query nodes linearly
  - Update synchronization and cluster management eventually limit
    - workload dependent
- With partitioned databases, scaling is sub-linear with severe cross node query degradation

Limitations of Synchronous Replication: How Best To Scale Queries (ctd)

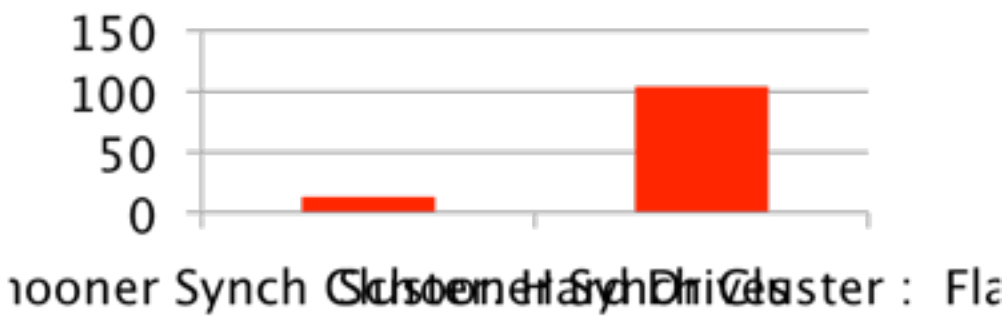


1 Synch Replication Group + Multiple Integrated Asynchronous Replication Group

- Can infinitely scale reads
- No data loss, auto-failover
- near zero slave lag requires asynchronous parallel update slave replication

Limitations of Synchronous Replication: How Best to Scale Updates

- Database Update Scalability
  - Vertically scale with commodity : flash memory, more cores, higher frequency



- Compelling option exploiting low cost, high performance commodity technology

Limitations of Synchronous Replication: How Best to Scale Updates (ctd)

- Database Update Scalability

  …After Optimal Vertical Scaling:

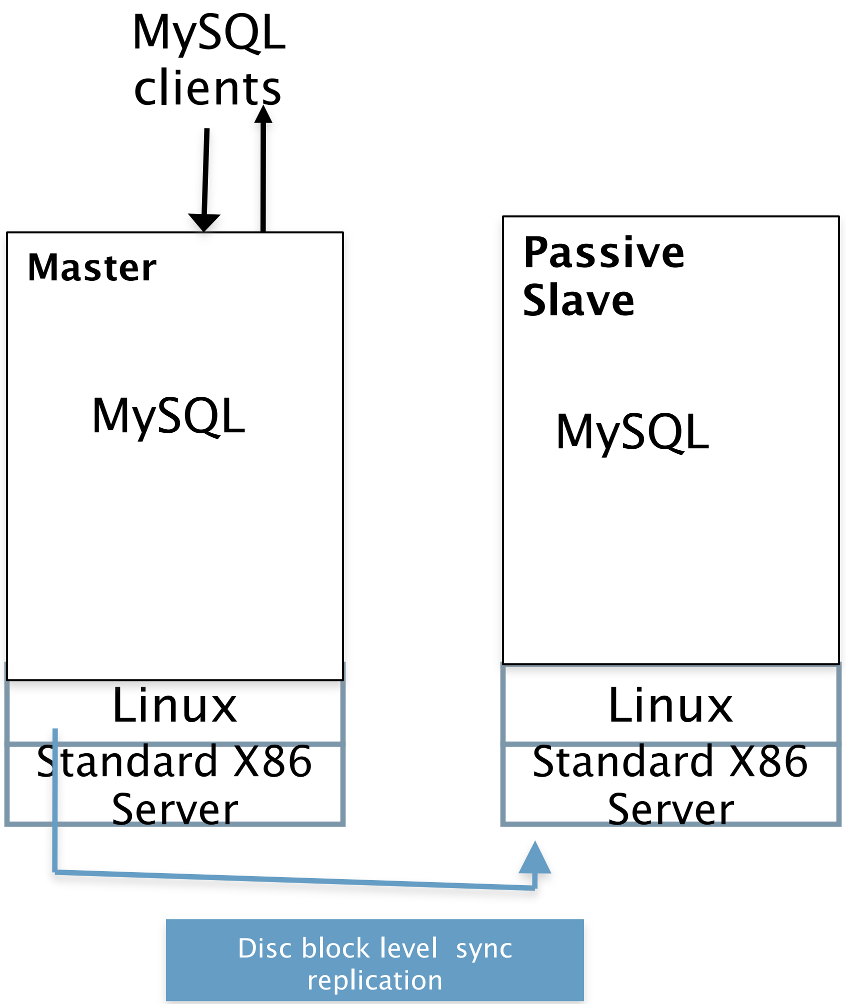
  Horizontally Scale Through Sharding

  - Application Transparent
    - Automated internal partitioning (MySQL NDB Cluster and Clustrix)
      - High query performance sensitivity (very slow cross partition queries)
    - Administrator analysis and set-up tools (Schooner MySQL Active Cluster)
      - DB Shards allows layout and query data access optimization
  - Application Aware

MySQL Specific Database Alternatives for Mission-Critical Deployments

	MySQL 5.5	MySQL NDB Cluster	Clustrix	Schooner MySQL Active Cluster
Fail-Over Downtime	Minutes-hours	seconds	seconds	seconds
Automated Fail-over	No	Yes	Yes	Yes
Data Loss	Yes	No	No	No
Data Consistency	No	Yes	Yes	Yes
Performance	Med	Med/Low	Med/Low	High
Scalability	Low	Med/Low	Med/Low	High
Ease of Management	Low	High	High	High
WAN perf and fail-over	No	No	No	Yes
InnoDB Compatible	High	Med/Low	Med/Low	High
Custom Hardware	No	No	Yes	No
Cost (TCO)	High	Med	High	Low

MySQL-Independent Replication: Linux DRBD Passive Standby Master



- Eliminates data loss
- Master failure-over to standby in minutes

But not transitionally consistent:

- Stand-by master cannot service load
- No warm re-start => hours for full service
- Can propagate corruptions(no log checksums)

- Slaves are still operating with asynchronou replication => same issues as MySQL 5.5/5

  - high administrative complexity
  - reduced service availability
  - inconsistent slave data
  - poor performance
  - high TCO

Linux DRBD (Distributed Replicated Block Device)

MySQL-Independent Replication for Heterogeneous Database Interoperability

Very loosely-coupled external replication services based on MySQL's asynchronous replication Bin log

Oracle Golden Gate:

- Converts MySQL Asynchronous Bin log to a common log format
- Heterogeneous database replication interoperability: Oracle, IBM DB2, and Microsoft SQL Server

Continuent Tungsten Replicator

- Converts the MySQL asynchronous Bin log to a transaction history log
- Uses JDBC through a client proxy to access MySQL indirectly
- Heterogeneous database replication interoperability: PostgreSQL

If used in MySQL Master - Slave deployments:

- Performance is significantly worse than MySQL 5.5/5.6
- Same issues as all loosely coupled asynch Bin log approaches
    - reduced service availability
    - poor data integrity
    - high administrative complexity
    - high TCO

MySQL Database Alternatives for Mission-Critical Deployments

	MySQL 5.5	MySQL NDB Cluster	Clustrix	Linux DRDB	Continuent Tungsten	Golden Gate	Schooner MySQL Active Cluster
Fail-Over Downtime	Minutes-hours	seconds	seconds	minutes	seconds	Minutes-hours	seconds
Automated Fail-over	No	Yes	Yes	No	Yes	No	Yes
Data Loss	Yes	No	No	No	Yes	Yes	No
Data Consistency	No	Yes	Yes	No	No	No	Yes
Performance	Med	Med/Low	Med/Low	Med	Low	Low	High
Scalability	Low	Med/Low	Med/Low	Low	Low	Low	High
Ease of Management	Low	High	High	Low	Med	Med	High
WAN perf and auto- fail-over	No	No	No	No	No	No	Yes
InnoDB Compatible	High	Med/Low	Med/Low	High	High	High	High
Custom Hardware	No	No	Yes	No	No	No	No