



# How Scale-Up and Scale-Out Flash Based Databases Can Provide Both Breakaway High Performance and Breakaway High Availability for Enterprise and Cloud Datacenters

*Dr. John R. Busch*  
*Founder and CTO*  
*Schooner Information Technology*  
[\*\*John.Busch@SchoonerInfoTech.com\*\*](mailto:John.Busch@SchoonerInfoTech.com)  
[www.SchoonerInfoTech.com](http://www.SchoonerInfoTech.com)

*Storage Developer Conference : September 19, 2011*

# How Scale-Up and Scale-Out Flash-Based Databases Can Provide Both Breakaway High Performance and Breakaway High Availability for Enterprise and Cloud Datacenters”

## ABSTRACT

We present emerging storage and database software technologies providing optimal scale-up through ultra-high flash and multi-core parallelism and optimal scale-out through synchronous replication, exploiting commodity hardware advances to yield 10x performance and 90% reduction in downtime, and providing key new building blocks for greatly improving data center QOS and TCO.

## LEARNING OBJECTIVES

- Information and Data Management Technologies : understand how highly parallel and concurrent software coupled with advanced hierarchical storage management exploit flash memory and multi-core processors for optimal scale-up and cluster-wide synchronous replication for optimal scale-out
- Scalable and Distributed Storage Systems : understand the scalability, consistency, and availability trade-offs in scale-up and scale-out architectures, and the key enabling technologies to concurrently optimize them
- Large Data Storage and Management: understand how cluster-wide synchronous replication simplifies large data multi-node cluster management by providing fully consistent data, eliminating data loss, and enabling automatic and transparent fail-over and recovery



# Industry Trend: The Mission-Critical Imperative

# Business' Most Valuable Asset: Its 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 hurt the bottom line

# The Mission-Critical Imperative



the social network



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

# The Mission-Critical Imperative

- Maintaining data availability and low response time is critical for key classes of businesses
  - Web-facing applications and high-volume web sites
  - eCommerce, social networking, gaming
  - Finance
  - Telecommunications
  - Enterprise
- These applications and websites are now mission-critical
  - Require mission-critical databases

# Requirements for Mission-Critical Databases

# Mission-Critical Database Requirements



High  
Availability



High  
Performance  
and  
Scalability



Simple and  
Powerful  
Administration



Data Integrity



Cost Effective



Standards  
and  
Compatibility

Mission Critical



# Mission-Critical Database Goals and Metrics

## Goals

- **High Availability**
- **High Data Integrity**
- **High Performance and Scalability**
- **Simple and powerful administration**
- **Cost effective**
- **Standards and Compatibility**

## Metrics

- Service unavailability (minutes/year) from failures, during planned administration, or from disasters
- Probability of data loss or corruption; data consistency levels
- Transaction throughput, response time; performance scalability; performance stability
- Ease of cluster administration; fail-over automation; monitoring and optimization tools
- Total cost of ownership (TCO); return on investment (ROI)
- Level of standards compliance

# Database High Availability and Scalability Architecture and Ability to Exploit Commodity Technologies: Fundamental Impact

Database HA and Scalability Architecture

Flash Memory

Multi-core

Cloud

## Profound Impact on Every Mission-Critical Dimension

- Service Availability
- Data Integrity
- Performance and Scalability
- Ease of Administration
- Cost

# Database HA Goal

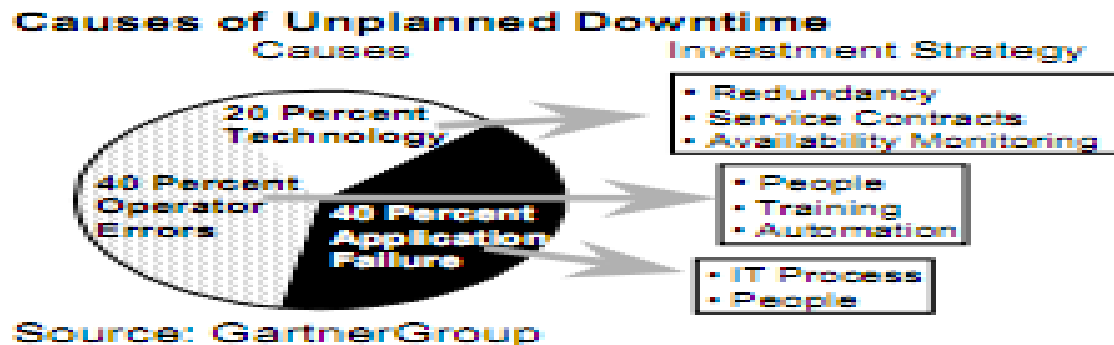
**Database HA Goal** : Maximize Service Continuity during database unplanned or planned downtime

## **Unplanned Downtime**

Caused by hardware, software failures or operator errors

## **Planned Downtime**

Administrator makes the database unavailable to users and processes at a scheduled time (for upgrades, migrations, etc)



## Basic HA Approach to Cover Database Unplanned and Planned Downtime

- ➔ Switch users to a redundant copy
- ➔ Repair offline
- ➔ Bring back into operation in a consistent state

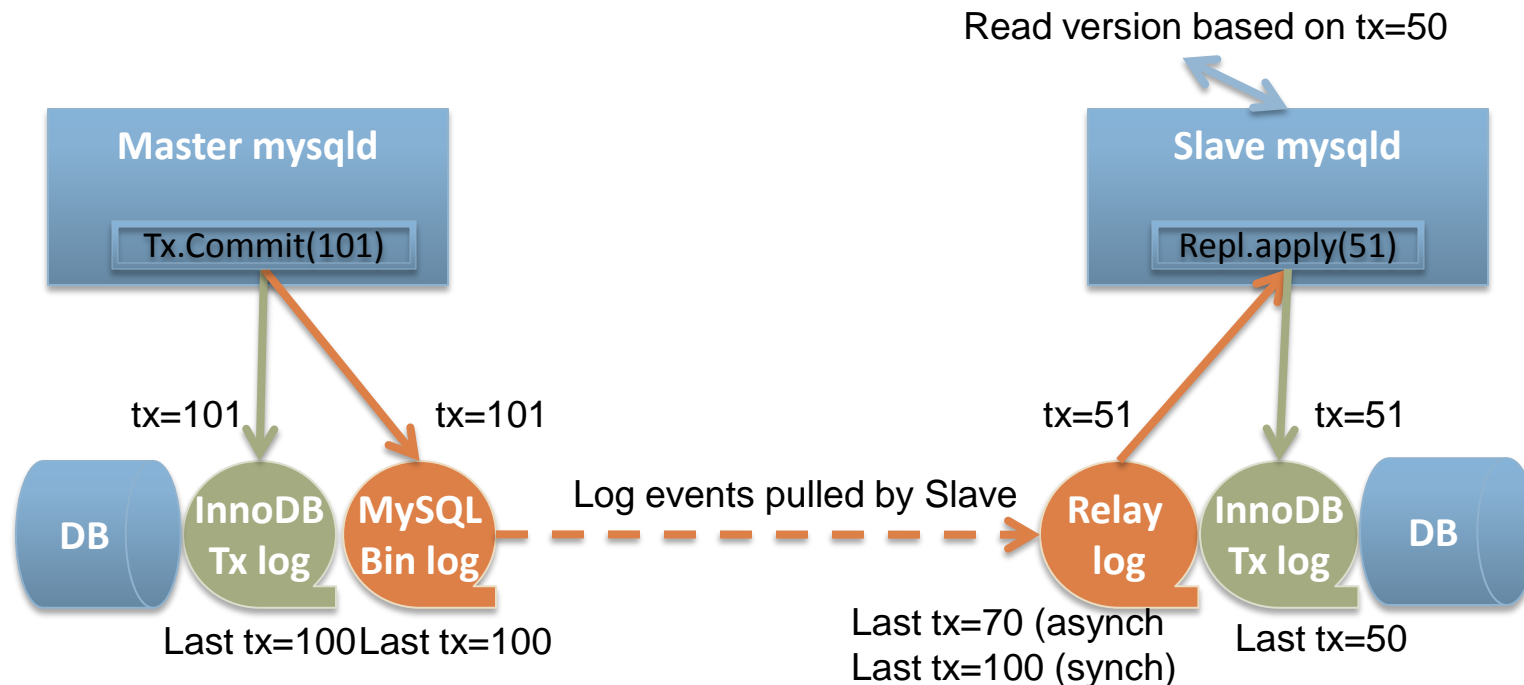


# Alternative Database HA and Scalability Architectures and their Mission-Critical Impact

# MySQL Database Replication and Fail-Over Technologies

- Database-Specific Replication Technologies
  - Loosely-coupled
    - Asynchronous
    - Semi-synchronous
  - Tightly-coupled
    - Synchronous replication
- Database-Independent Replication

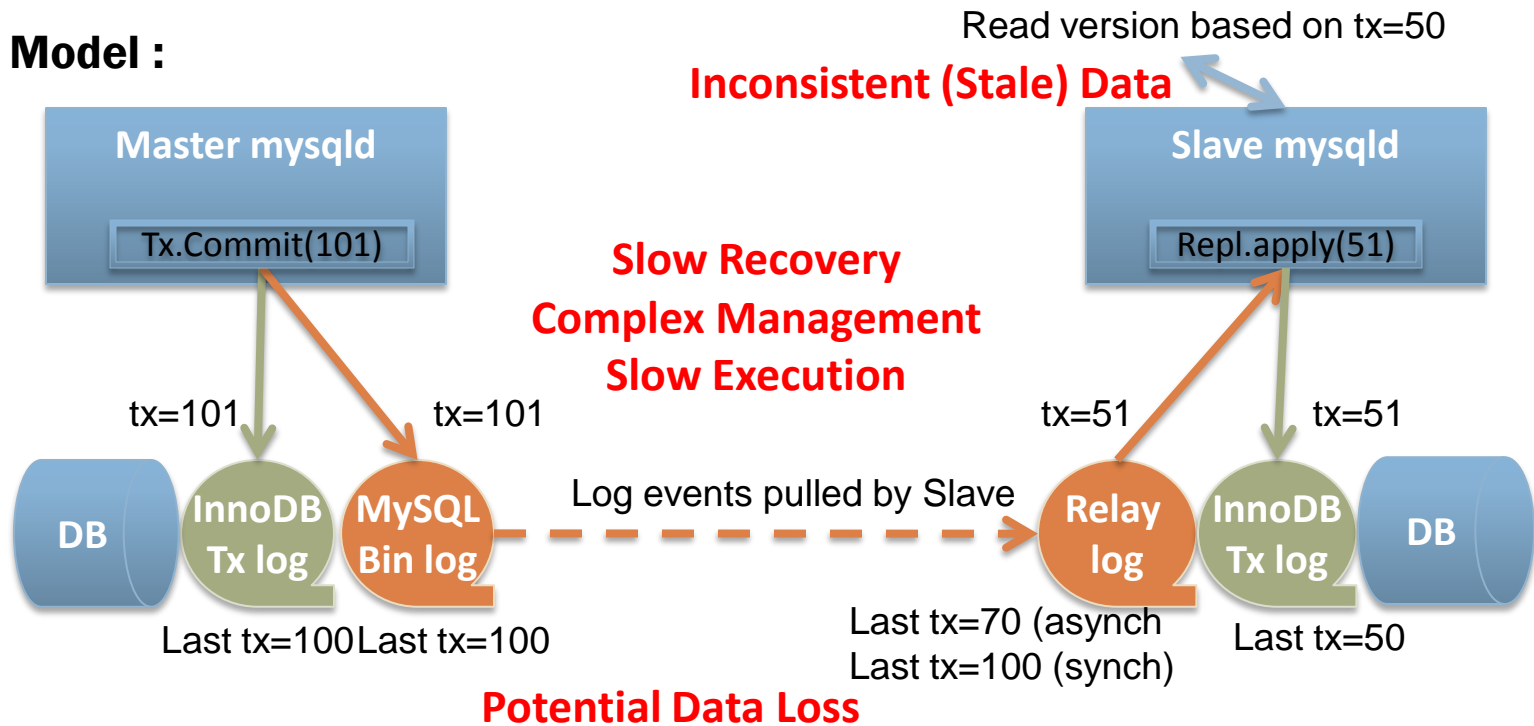
# Loosely-Coupled Asynchronous and Semi-Synchronous Replication



**Example Products : MySQL Enterprise 5.1 Asynchronous and 5.5/5.6 Semi-Synchronous Replication**

# Loosely-Coupled Asynchronous and Semi-Synchronous Replication

## HA Model :



## Scaling Model :

- **Low throughput, low utilization, server sprawl**
- **Frequent master sharding due to low update throughput**



# Mission-Critical Impact

## Limited service availability

- No master fail-over, requires re-synch of slaves

## Limited data integrity

- Lost data; inconsistent data

## Limited performance, utilization, scalability

- Low throughput, low utilization,
- Hard write scaling limits forcing frequent sharding; inconsistent read scaling

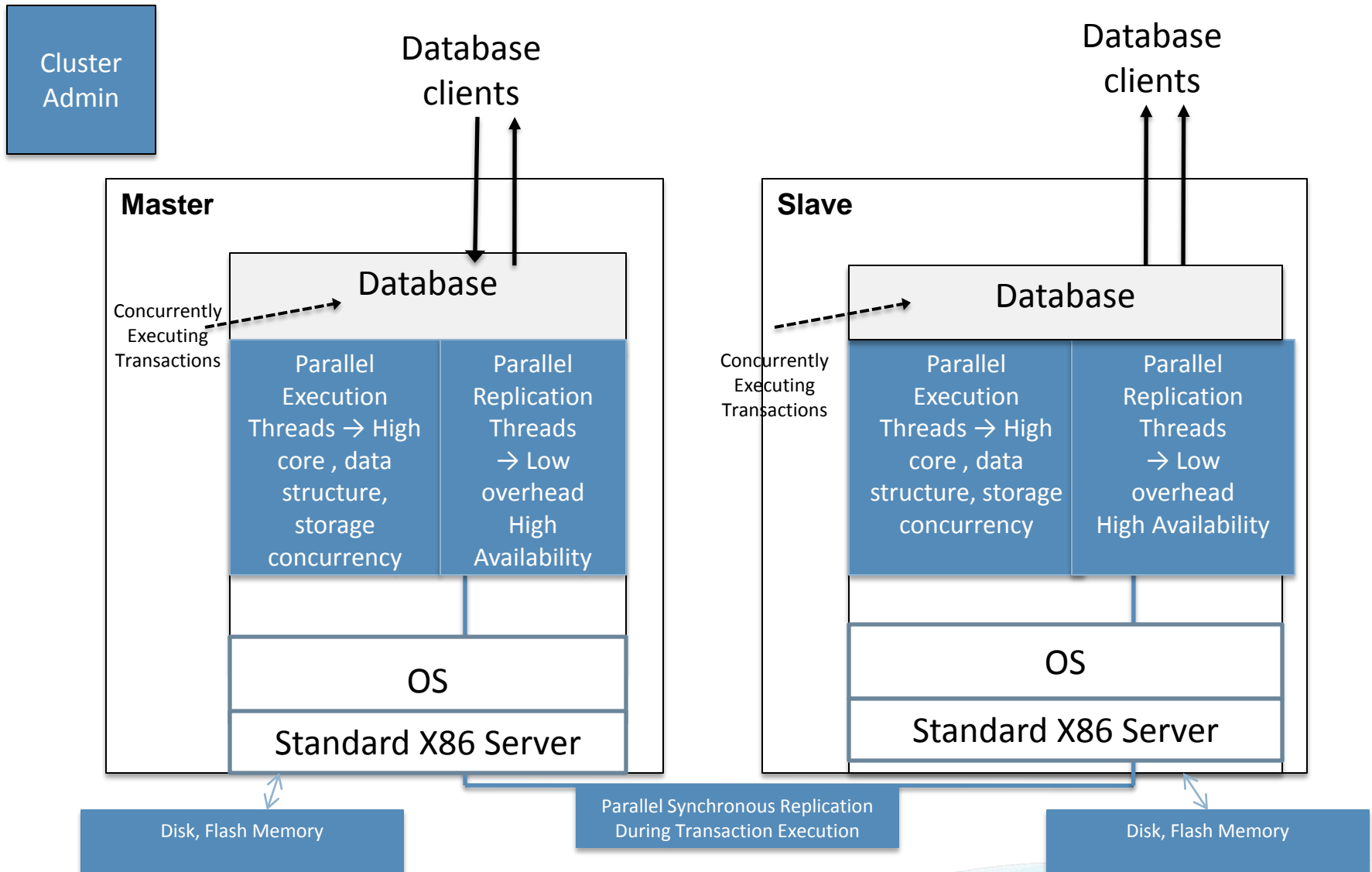
## Complex administration

- Manual processes, slave re-synch, sharding

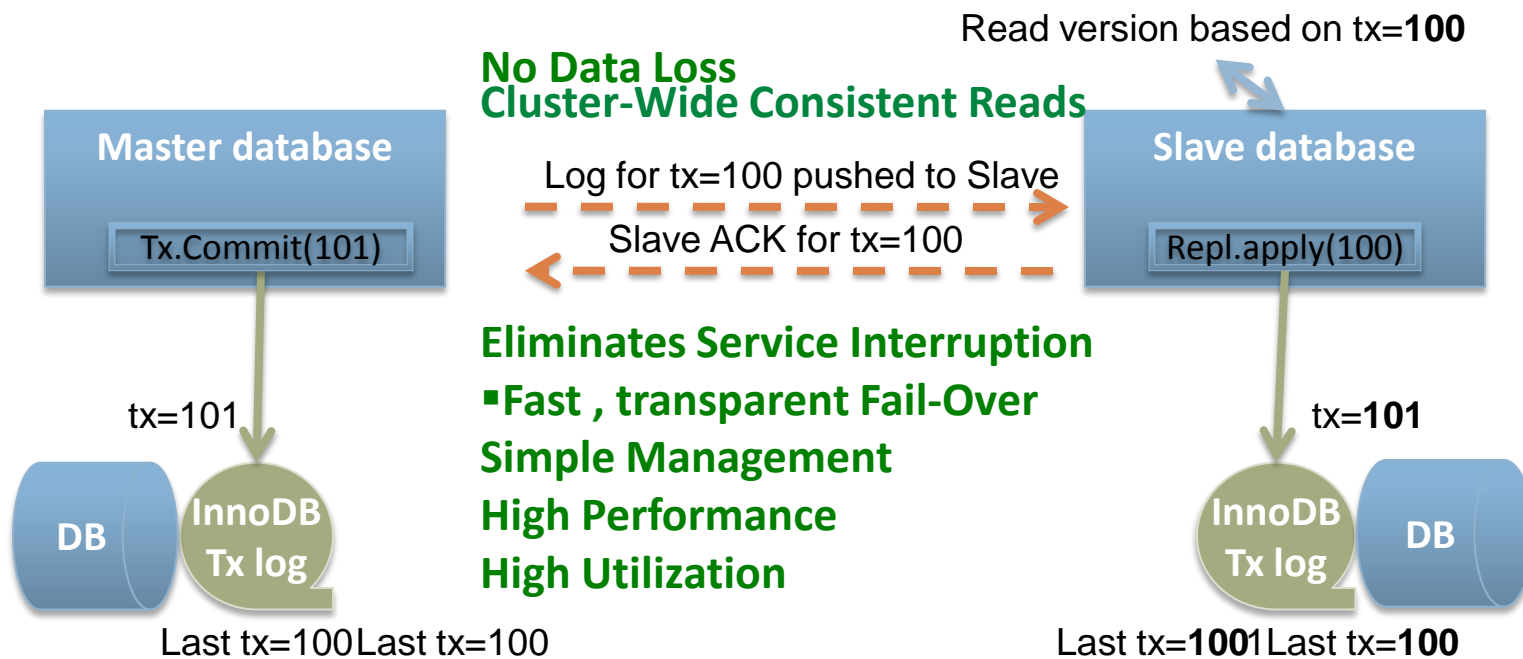
## High cost of ownership

- High capital expense from server sprawl
- Increased operating expense from power, space, admin
- Reduced revenue and customer satisfaction from service downtime

# Tightly-Coupled Storage Optimized with Fully Synchronous Replication

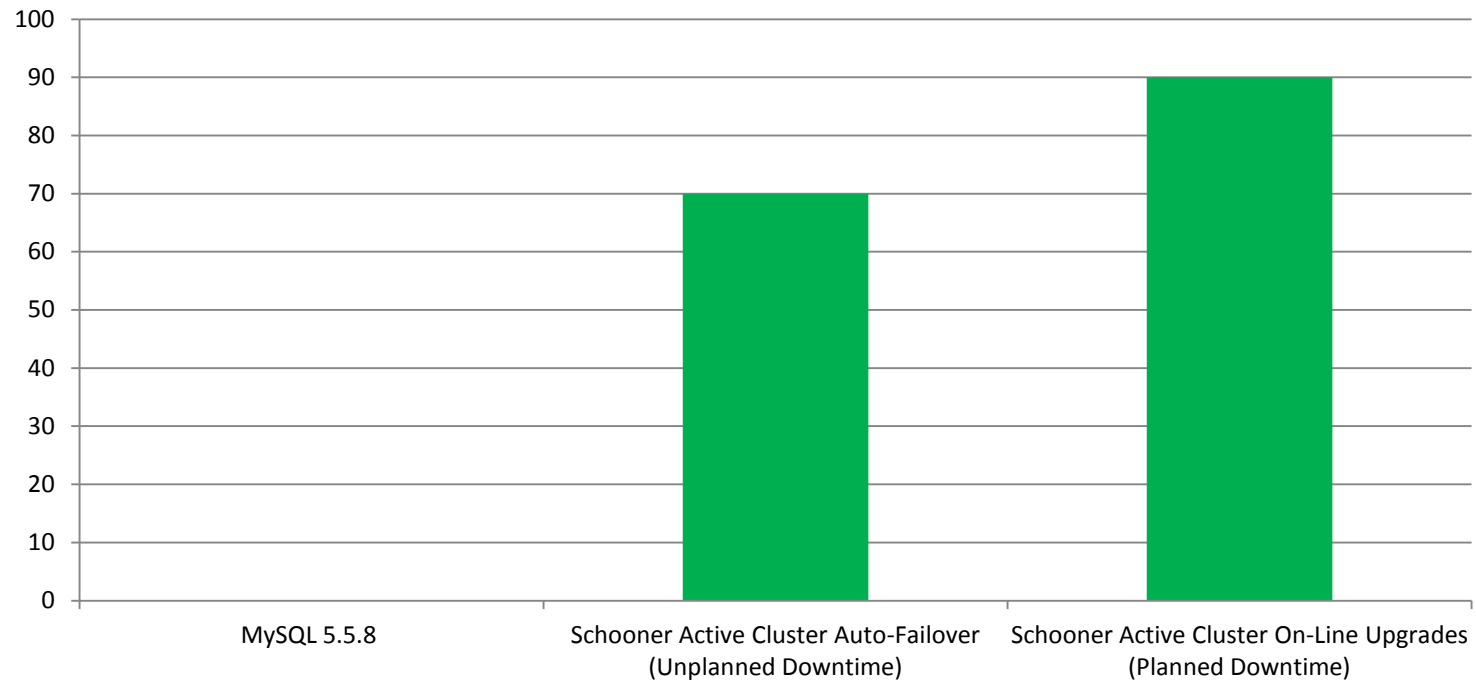


# Tightly-Coupled Storage Optimized with Fully Synchronous Replication



# Increased Service Availability and Data Integrity

## Availability Improvement from Synchronous Replication (% Cumulative Down Time Reduction)



# Key Resource Management Algorithm Design Requirements

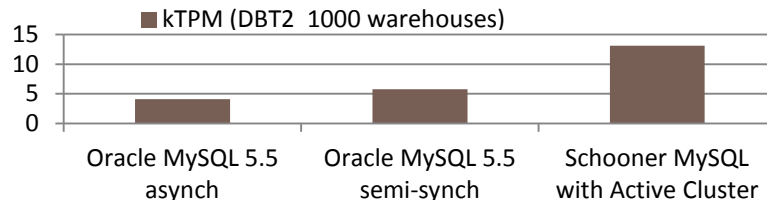
- Processor
  - Multi-core scalability
    - fine-grained locking, concurrent data structures
- Flash Storage Optimization
  - Concurrent DRAM buffer-pool management algorithms
    - Multi-threaded background write of dirty blocks so clean on misses
  - Highly-parallel multi-threaded flash-memory access
    - 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
  - Batched commits
  - Log files on HDD with persistent DRAM controller
    - Fast, saves flash for high access data
- Network
  - Memory to memory multi-threaded parallel synchronous replication
    - No asynchronous transmission of persisted log files
  - Connection Pools

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

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

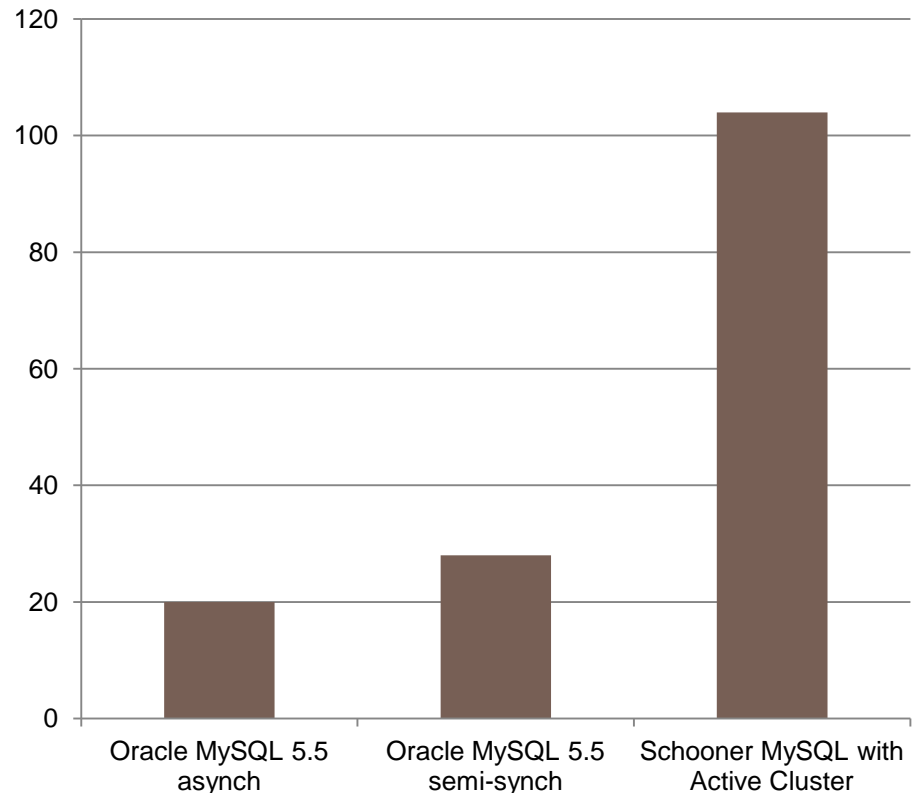
Measurement Configuration  
2 node Master-Slave configuration  
2 socket Westmere  
72GB DRAM

## Transaction Throughput with Hard Disc Drives



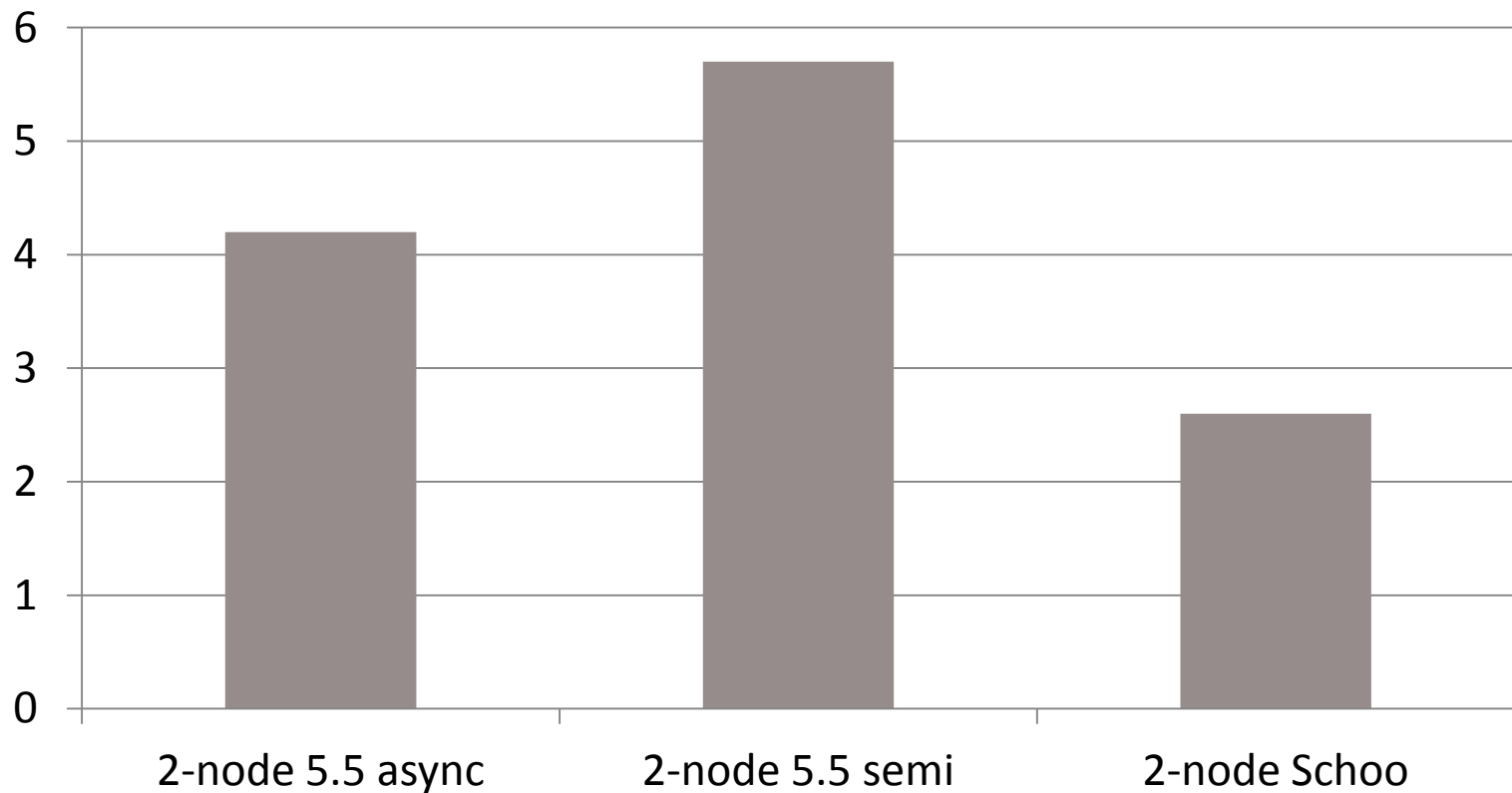
## Transaction Throughput with Flash

■ kTPM (DBT2 1000 warehouses)



# Lower Response Times

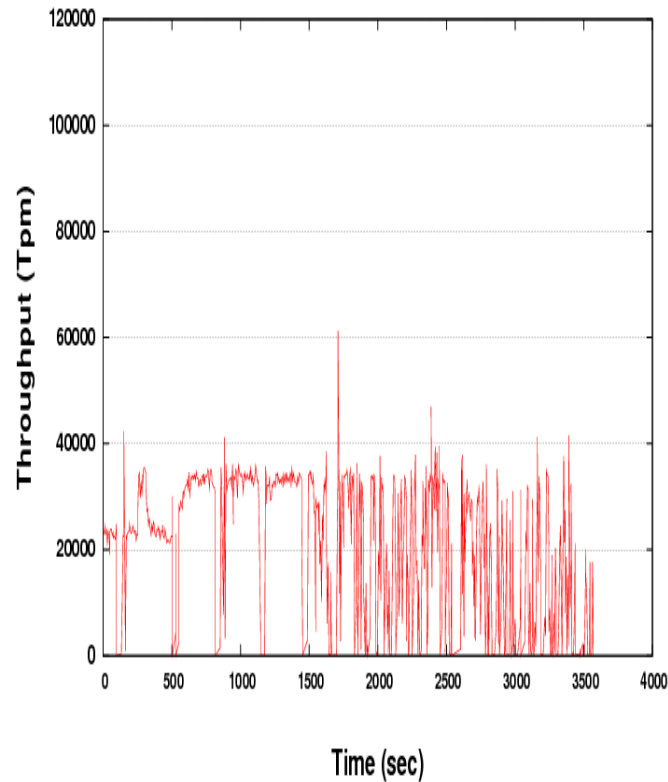
## Response Time (ms)



# Higher Performance Stability

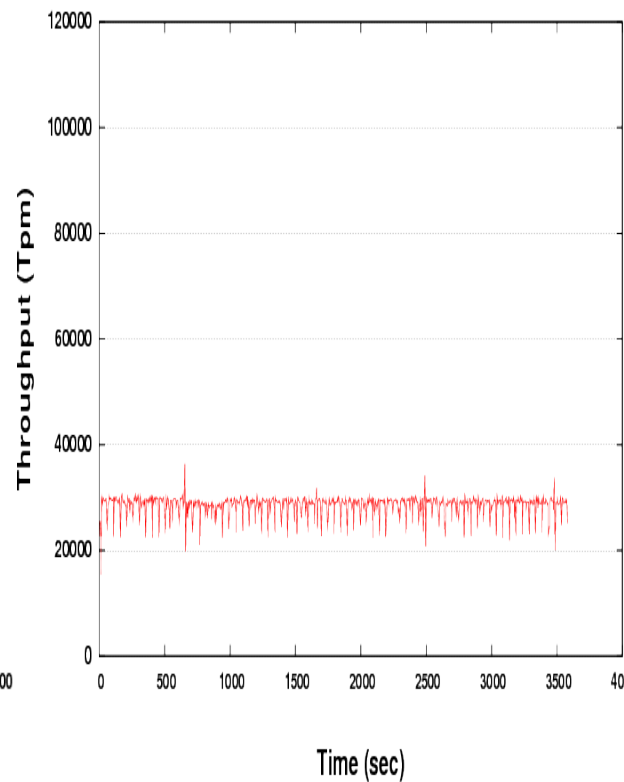
## 5.5 Async

Master Throughput vs. Time



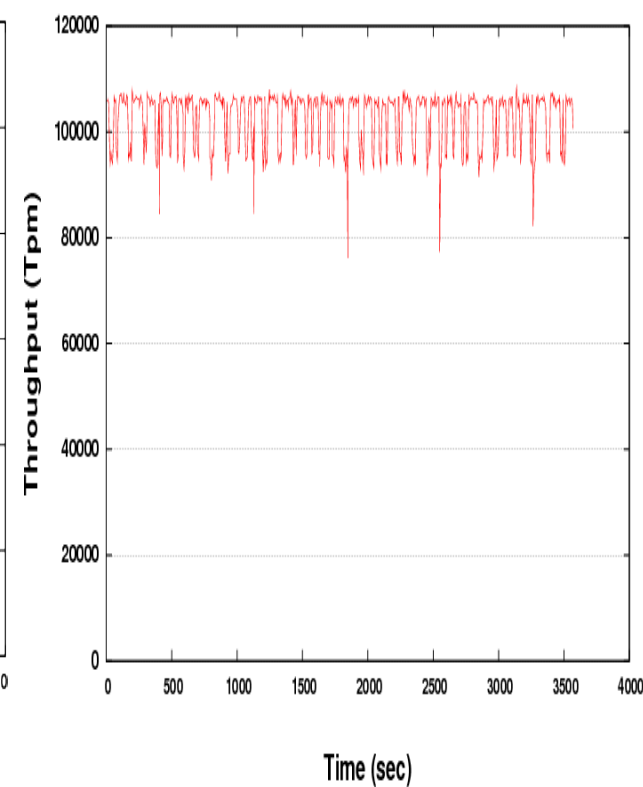
## 5.5 Semi-sync

Master Throughput vs. Time



## SAC

Master Throughput vs. Time

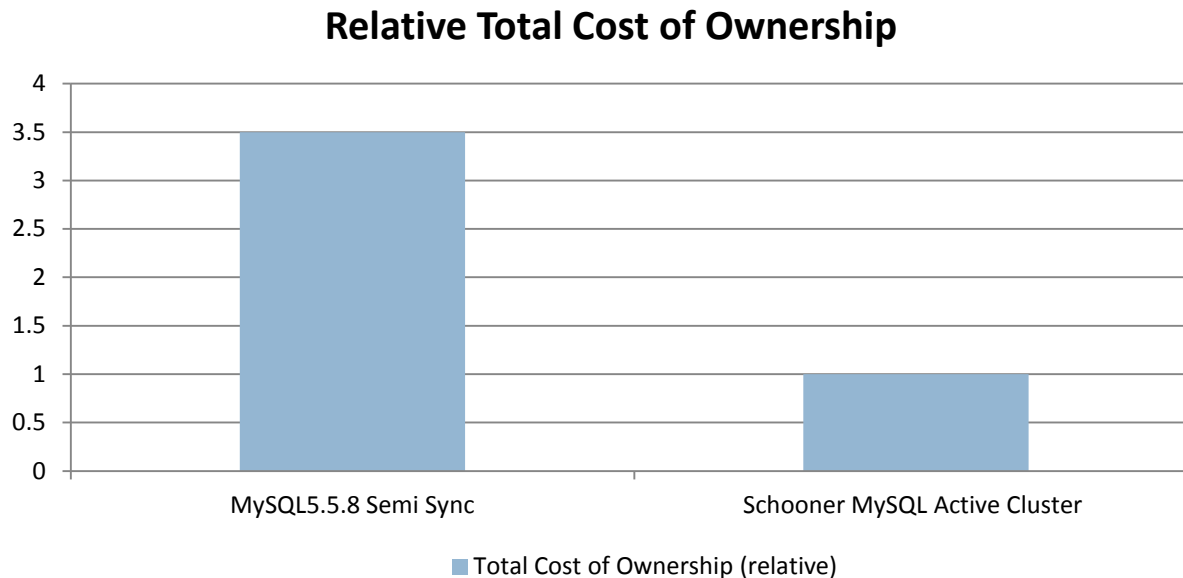




# Lower Total Cost of Ownership

## 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

The screenshot displays the Schooner MySQL Cluster Administrator GUI. The interface includes a sidebar with a tree view of the cluster hierarchy, a main content area with tabs for Overview, Setting, and Attach Instance, and a bottom section for Tasks.

**Group Metric**

Type	Synchronous	VIP Policy	Balanced
User	admin	Read VIPs	10.1.137.3, 10.1.136.3
Interface	eth4	Write VIPs	10.1.137.2
Async Slave	0	Schooner Data Format	Disabled

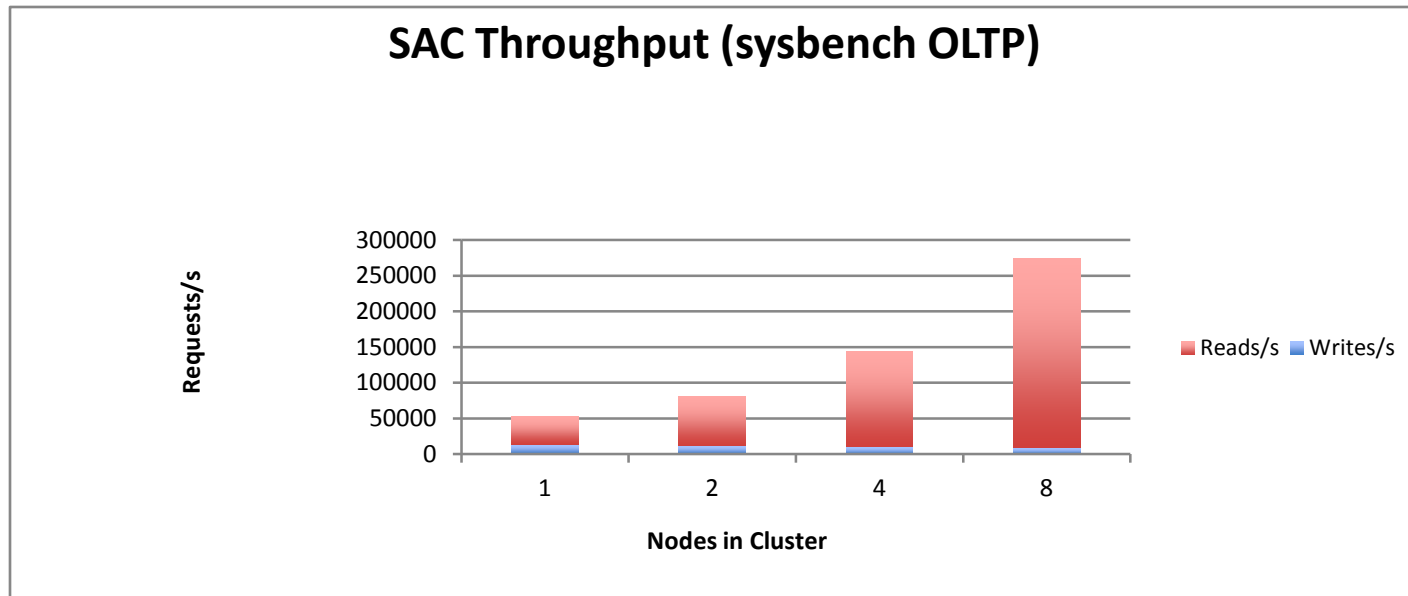
**Instance Members**

Name	Host	Version	Role	Progress	State	Commit/s	Select/s	Status
mysqld1	lab137.schoonerinfotech.net	5.1.52-3.1.547.393	Master	N/A	MYSQL_READY	0.00	0.20	up
mysqld1	lab136.schoonerinfotech.net	5.1.52-3.1.547.393	Slave	N/A	MYSQL_READY	0.00	0.00	up

**Tasks**

Status	Name	Node	Instance	Group	Time(start)	Time(end)	Description
✓	Add Backup	lab137.schoonerinfotech.net	mysqld1	N/A	4:46:21 PM Apr/08/2011	4:46:22 PM Apr/08/2011	Add backup task successful.

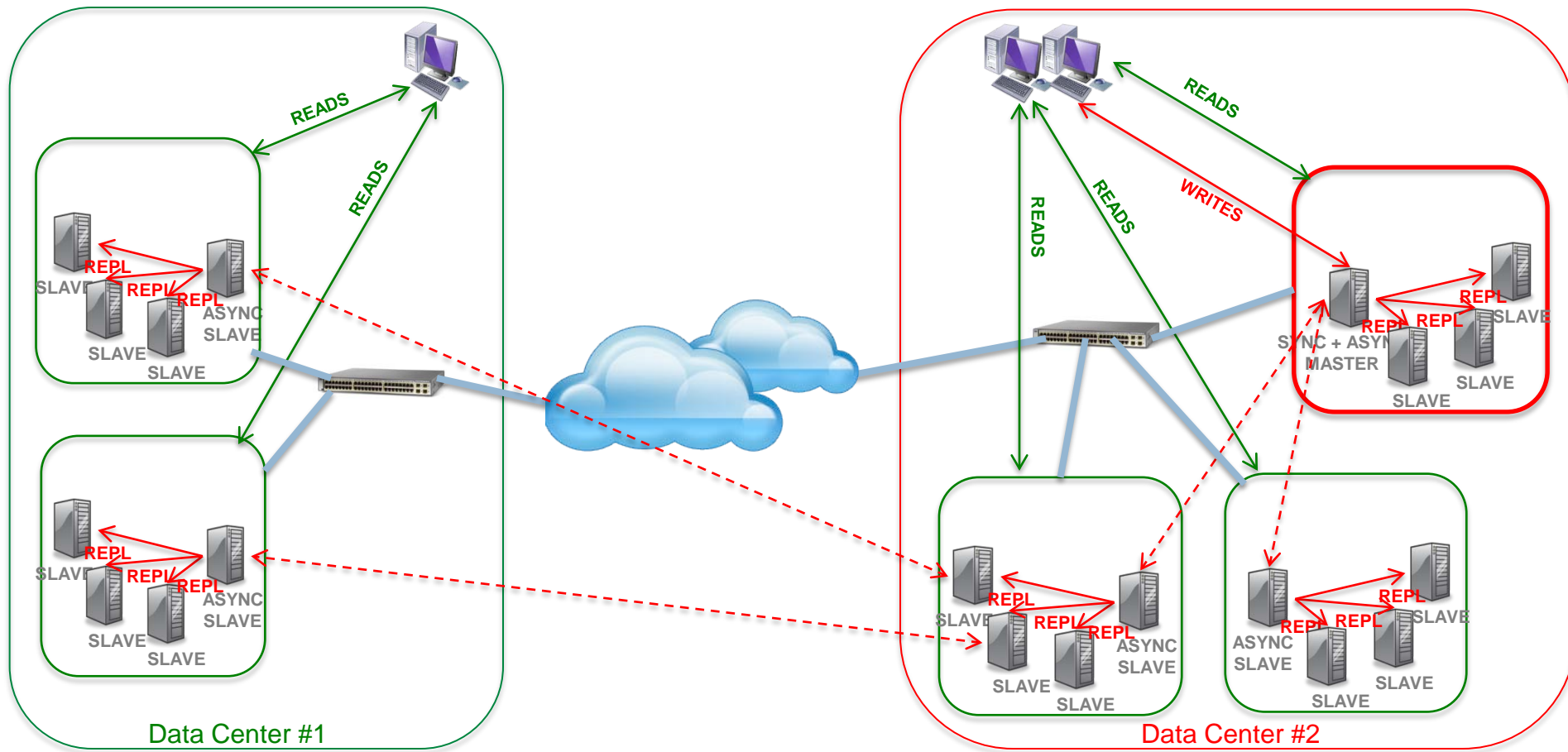
# Linear Horizontal Scaling of Queries with Full Consistency and HA



## Query Scaling in a Synchronous, Fully Consistent 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

# Parallel Asynchronous Replication: Unlimited Read Scaling on LAN/MAN/WAN, and Automated Failover



Parallel Slave Appliers with sync and async replication ensure that a geographically distributed MySQL database cluster runs at high throughput (no slave lag), the same as within a datacenter, and with high availability (auto-failover)

# Unlimited Update Scaling with Transparent Sharding

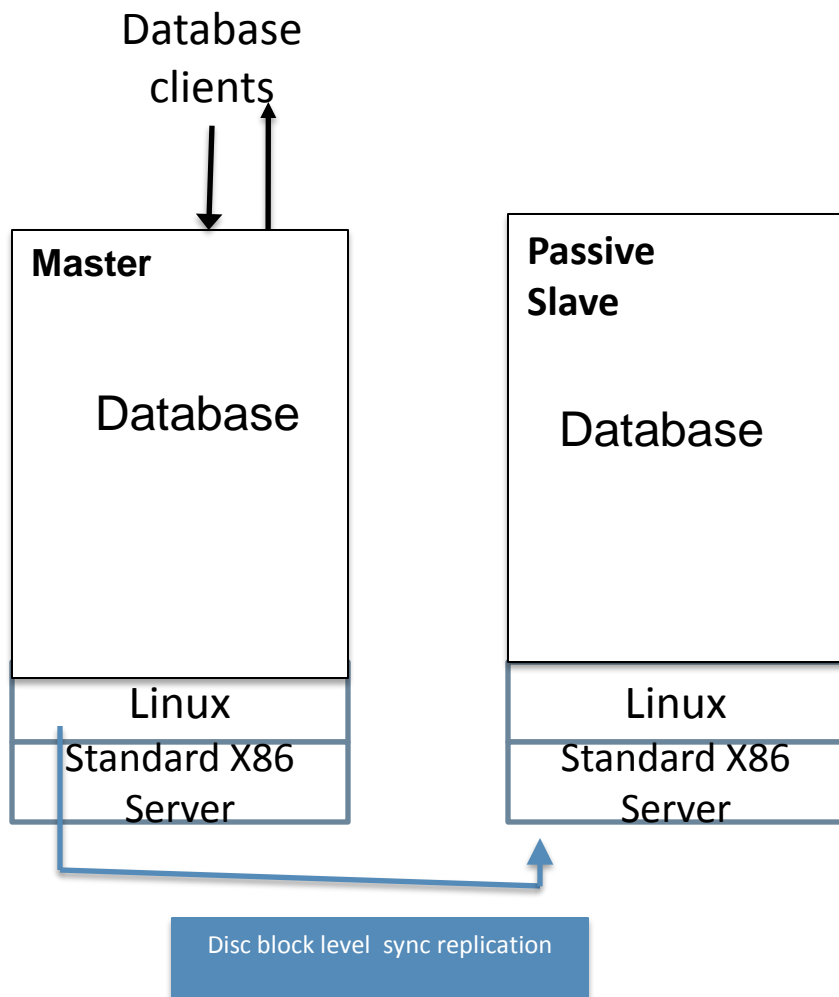
- Database update scalability

...After optimal vertical scaling:

Horizontally scale through Schooner MySQL 4.0 transparent sharding

- Application transparent with optimal layout
  - Administrator analysis and set-up tools
    - » Schooner tightly-coupled DB Shards
    - » Layout and query data access optimization tools
  - 2 Million DBT2 (1000 warehouse) Transactions per Minute with 16 servers
    - » Places Schooner in top 10 TPC-C results
    - » Note that ScaleBase only achieves 50k TPM with 8 shards

# DatabaseL-Independent Replication: Storage Level Replication : 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 asynchronous replication => same issues as MySQL 5.5/5.6

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

## Distributed Replicated Block Device

# Database-Independent Replication for Heterogeneous Database Interoperability

**Very loosely-coupled external replication services based on Database asynchronous replication log**

**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

**Examples :**

**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

# Evaluating the Options and Tradeoffs for your Data Center



# Example: 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
Cost (TCO)	High	Med	High	High	High	High	Low
Heterogeneous Databases	No	No	No	No	Yes	Yes	No

# 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
  - limits ability to exploit flash memory for database performance

# 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

# Mission-Critical Database Best Practices

## Goal

- **High Availability**
- **High Data Integrity**
- **Excellent Performance and Scalability**
- **Simple and powerful administration**
- **Cost effective**
- **Standards and Compatibility**

## Best Practice

- Replication (synch local and asynch parallel WAN); automation of failure detection and recovery
- Synchronous replication to eliminate data loss and fully consistent data; combined with parallel asynchronous replication for WAN disaster recovery
- Effective vertical and horizontal scaling for exploiting flash and multi-core
- Centralized management; automation; visibility (statistics); alerts
- Leverage commodity hardware and software; achieve high hardware utilization; leverage cloud if workload permits
- 100% standards compliance and certification

**Thank You!**