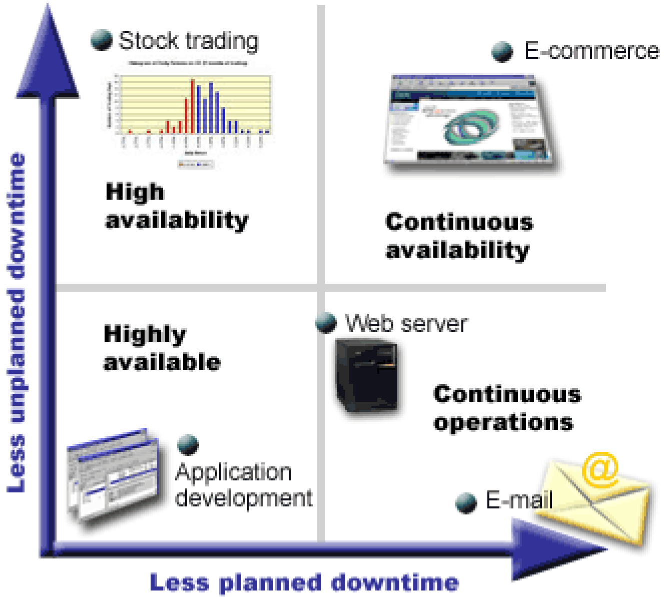
High Availability

Poornima Pochana
August 23 2011

High Availability



IBM:HighAvailabilityConcepts

Availability	Downtime/year	Downtime/month	Downtime/week
90%	36.5 days	72 hrs	16.8 hrs
95%	18.25 days	36 hrs	8.4 hrs
99%	3.65 days	7.2 hrs	1.68 hrs

Wikipedia:HA

High Availability

- Components
- Solutions
- Environments

Components

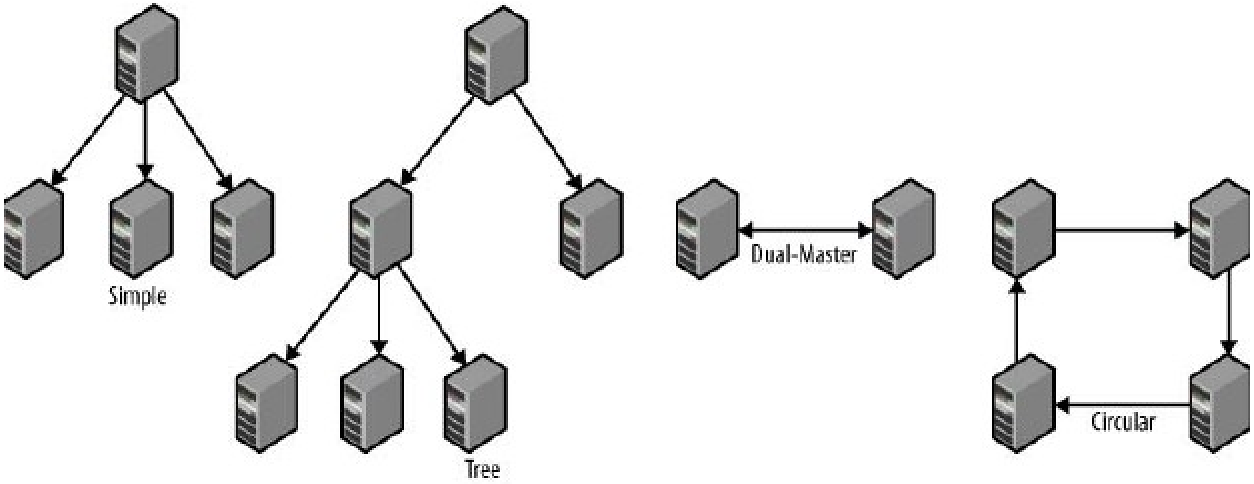
- Redundancy
- Contingency
- Procedure
- Monitoring
- Backup & Recovery
- Disaster Recovery

Redundancy

- Replacements
- Hardware
- Software
- MW
- Criteria
- Alternatives

Contingency

- Planning
- Topologies
- Failures
- Example - Replication



MySQL High Availability by Charles Bell, Mats Kindahl, and Lars Thalmann.
Copyright 2010 Charles Bell, Mats Kindahl, and Lars Thalmann, 9780596807306
(chapter2 - pg22)

Procedure

- Steps
- Execution
- Automation
  - Custom tools
  - MySQL-Replicant-Python
  - Flipper
  - Multi-Master Replication Manager

Monitoring

- Load
- Performance
- Proactive vs. Reactive
- Monitoring Solutions
    - System
    - Application
    - Network

Backup & Recovery

- Recovery Point Objective (RPO)
- Recovery Time Objective (RTO)
- Automation
- Tests

Disaster Recovery

- Extreme HA
- Plan
- Physical/non-physical threats
- Level of operational capability
- Restore/wait time
- Available resources

Solutions

- Scale-up
- Heartbeat/DRBD/SAN
- Virtual IP/DNS switchover
- MySQL Replication for
  - High Availability
  - Scale-out/Load Balancing – {MySQL Proxy, Memcached, Sharding}
  - Backup & Recovery
  - Reports & Analysis
- PostgreSQL

Environments

- Amazon EC2
- MySQL Cluster
- RedHat Cluster Suite

Acknowledgements

Wikipedia on High Availability

http://en.wikipedia.org/wiki/High_availability

MySQL High Availability by Charles Bell, Mats Kindahl, and Lars Thalmann. Copyright 2010 Charles Bell, Mats Kindahl, and Lars Thalmann, 9780596807306

MySQL White Papers on Replication

http://www.mysql.com/why-mysql/white-papers/mysql-wp-replication.php

http://downloads.mysql.com/docs/mysql-ha-drbd-en.a4.pdf

http://forge.mysql.com/w/images/5/52/FOSDEM-MySQL-HA-2009-02-08.pdf

Backup and Recovery Considerations for Websphere

http://publib.boulder.ibm.com/infocenter/iseries/v5r3/.../snrmain.htm

MySQL HA at Yahoo by Jay Janssen

http://mysqlguy.net/blog/2010/08/03/mysql-master-ha-yahoo

The following slides are notes regarding the High Availability Concepts and MySQL High Availability

Redundancy

- Do you have the replacements?
- Identify every potential Single Point of Failure (SPOF)
- Questions to consider in identifying SPOFs
  - Cost of duplicating components
  - Probability of failure of differing components
  - Time to replace a component
- What are the Hardware replacements and Software replacements?

Contingency

- Do you know what to do?
- This depends on which component failed and how it failed?
- Contingencies to plan for
  - Slave failures
  - Master failures
  - Relay failures
  - Disaster recovery failures

Contingency

Solutions to handle failures: Design topologies that fits your system/operations. Some of the replication topologies are:

1. Asynchronous
    - Simple (aka non-hierarchical, hot-standby)
    - Tree (hierarchical with/without relay servers)
    - Dual-Masters
        - Active-Active
        - Active-Passive (symmetrical hot-standby)
    - Slave promotion
    - Circular

Contingency

1. Semi-synchronous
2. Synchronous – two phase commit, achieved with DRBD
3. Statement
4. Row

Technologies to implement the above topologies are:

- Shared disks (SAN)
- Replicated disks using DRBD
- DRBD + Heartbeat
- Heartbeat + Virtual-IP switchover

Procedure

- Do you know what steps to take to detect and handle the failure and then execute them swiftly and efficiently?
- Solution: Automation for atleast the following tasks -
  - Adding new slaves
  - Removing slaves from the topology
  - Switching the master (slave promotion)
  - Handling slave failures
  - Handling master failures
  - Upgrading slaves
  - Upgrading masters

Monitoring

- Do you know how to keep track of the load?
- Do you know how to measure performance of your systems/applications/database?
- Proactive vs. Reactive monitoring

Scale-out/Load-Balancing

Balancing the load for applications/websites – how to handle increased load?

- Scaling up – add more memory, expensive (RAM/Hard disk)
- Scaling out – add more servers, less expensive

Scaling out/load-balancing ie., dividing up queries between the servers can be done in two ways:

1. Dividing read queries
2. Dividing write queries

Scale-out/Load-Balancing

Scaling out for reads: Two ways in which load is balanced

1. Application asks for a server based on the query
2. Proxy/Intermediate server analyzes query and directs to correct server

Application Level Load Balancing

1. Mediawiki: provides load-balancer settings to add master/slave servers. Writes go to masters and reads got to slaves.
   - Mediawiki:LoadBalancer
2. Wordpress: has a plugin HyperDB, to provide the load balancer settings (vanilla wordpress does not have provision for load balancing)
   - Wordpress:HyperDB

Proxy Level Load Balancing

MySQL Proxy: Two scenarios

1. When the client application is aware of the db servers to switch to if one of them fails
    - Only mediawiki and wordpress (with hyperdb) fit into this category (implicitly), but still the back-end needs to implement replcation, ie., multiple masters/slaves topology etc.
1. When the client is shielded from what goes on in the back-end
    - All the above replication solutions come into this.

Backup & Recovery

- Recovery Point Objective (RPO)
  - How much can I afford to lose/skip/never recover in the event of a total loss?
  - Different levels of RPOs determine how much you need to backup and how often
- Recovery Time Objective (RTO)
  - How long can you afford to have your data unavailable while it is being recovered?
- Naming convention
- Automate backup
- Test backups by practicing data recovery

Backup & Recovery

Backup must have the capability to:

make copies of data in a form fit for restoring

make consistent backup copy

support monitoring

Several forms of backup:

Full

Differential

Incremental

Types of backup:

Logical - slow

Physical – fast

Two terms are found in almost all references to data recovery and backup solutions, so it’s important to understand them:

Recovery Point Objective (RPO): The state of the system that is acceptable for use. Thus, RPO represents the maximum acceptable degradation of services (operational capability) or maximum acceptable data loss. Use these goals for measuring the success of a recovery action.

Recovery Time Objective (RTO): The maximum amount of time permitted to endure a loss of capability (also called downtime).

Backup/Archival plan:

pick a naming convention for your backup archives

How much data can I afford to lose in the event that I need to recover from scratch? How much can I skip (never recover) in the event of a total loss?

This becomes your RPO, or the level of operational capability that your company must have in order to conduct business.

You may have different levels of RPOs

Having different levels of RPOs determines how often you need to backup and how much to backup

How long can you afford to have your data unavailable while it is being recovered? - weigh in the cost of recovery to your company.

Determine several time frames

Automate backups

Test the backups periodically by practicing data recovery

Disaster Recovery

- Extreme HA
- HA for the unexpected
- Do you have plans for handling major and unexpected disasters/failures?

HA & DR

- High Availability options - intended to
- permit real-time or near real-time recovery without loss of any data (or to minimize the data loss to very small segments, such as the last changed record).
- High availability solutions can address the minor disasters and even form one layer of defense for major disasters.
- High availability protects against the known or expected
- Disaster Recovery - allows you to plan for the unexpected.