Microsoft®
tech·ed
India | 2009

Implementing High-Availability Solutions for the right problem with SQL Server 2008

Amit Bansal
CTO
eDominer Systems P Ltd |
Peopleware India

msdn Microsoft® Developer Network

Microsoft TechNet

Who is Amit Bansal?

- CTO, eDominer Systems & Peopleware India
- Conducted more than 200 workshops on SQL Server & BI for top notch IT companies world wide
- Microsoft MVP for SQL Server
- Microsoft Certified Trainer Advisory Council member
- Speaker at TechED India, TechED US & TechED Europe
- SME – SQL Server 2008 certifications

Agenda? --- TODO

- Intro to High-Availability/DR
- Key Terms
- Always On Technologies
  - Overview, Planned / Unplanned downtime
  - When to use
  - Enhancements in SQL Server 2008
- Combining Technologies
- Summing it up

What is Availability?

- Availability is defined by an implicit or explicit application specific Service Level Agreement
- Example
  - The ordering system is considered available when orders can complete in less than one minute. The maximum downtime acceptable is five minutes per month.
  - Downtime = the amount of time when the SLA is not met
  - Typically uptime is tracked (i.e. 99.993%), which doesn't include the "planned" downtime
- The SLA should be used during design, deployment, and production monitoring phases of the application lifecycle

Intro to Disaster Recovery

- Processes and procedures designed to restore business operations due to a natural or human-induced disaster
  - Typically involves providing redundancy spanning multiple sites or across geographic regions

Key Terms

- SLA Metrics:
  - Recovery Point Objective (RPO) = data loss
    - How much data can you lose?
    - guided by criticality of application data
  - Recovery Time Objective (RTO) = downtime
    - How much downtime can you tolerate?
    - guided by availability requirements
- Local High-Availability (Local HA)
- Site Disaster Recovery (Site DR)

Always On Technologies

- Unplanned Downtime
  - SAN/RAID
  - Backup and Restore
  - Log Shipping
  - Database Mirroring
  - Failover Clustering
  - Replication
- Planned Downtime
  - Rolling Upgrades and Patching
  - Online Operations
  - Resource Governor
  - Database Snapshots

Always On Availability Scenarios

- System Administration
  - Upgrades, Index rebuilds…
- Logical Errors (User or Application)
  - Incorrect data modifications, accidental changes…
- Site Disaster
  - Hurricanes, fires…
- Hardware Failures
  - Memory, network, disks…
- Software Failures
  - Bugs

Categorizing Availability On Solutions

High Availability Requirement	Solution
System Administration	Online management
Recovery from application or user errors	Logical recovery
Site disaster protection	Database or storage mirroring
Database workload scale out with redundancy	Replication

Online Management

- Downtime Scenarios
  - Applying a patch or upgrade
  - Moving a table as part of system maintenance
  - Deploying a new version of an application
  - Recovering from a corrupted data file
- Always On Solutions
  - Rolling Upgrades utilizing Database Mirroring, Log Shipping, or Replication
  - Fast recovery
  - Online alter table, index create, index rebuild
  - Online granular restore

Categorizing Availability On Solutions

High Availability Requirement	Solution
System Administration	Online management
Recovery from application or user errors	Logical recovery
Site disaster protection	Database or storage mirroring
Database workload scale out with redundancy	Replication

Recovery from User or Application Errors

- Downtime Scenarios
  - A table is accidentally dropped
  - An application makes incorrect updates to data
  - A user possibly made unauthorized changes to data
- Always On Solutions
  - View point in time snapshots of the system
    - Take database snapshots at intervals
    - Log ship to another site using restore delay/standby options
    - Compare table to older version using tablediff utility
  - Track history of DML statements
    - Use SQL Trace or customized replication

Log Shipping

How it works

Update orders set….

Primary Instance

Primary DB

Log Backup

Transaction Log Backup

Transaction Log Backup

Transaction Log Backup

Log Backup Copy

Secondar

Log Restore

Secondary DB

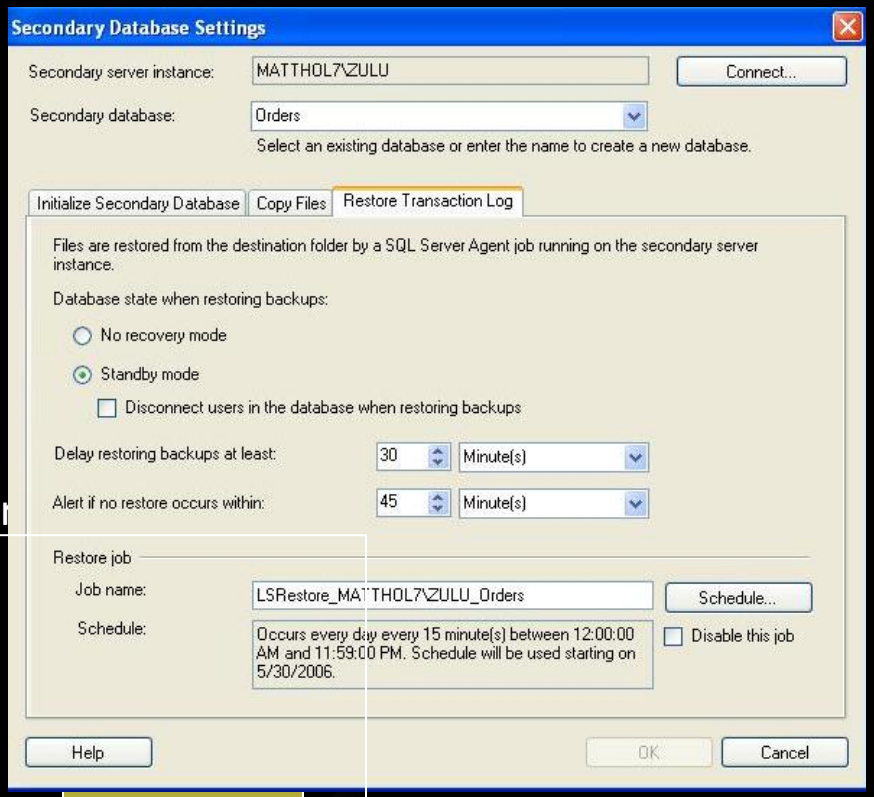
Transaction Log Backup

Transaction Log Backup

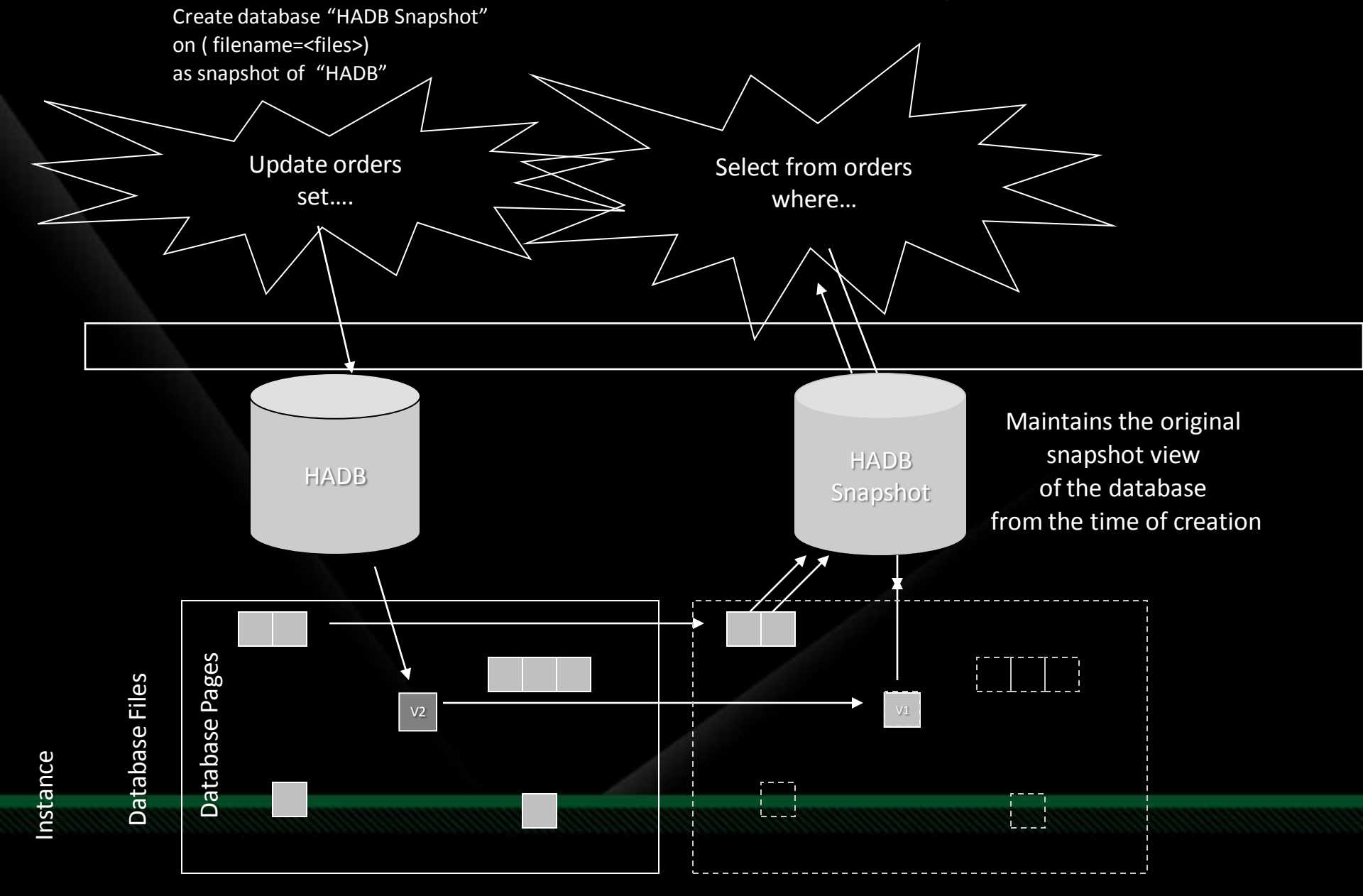
Transaction Log Backup

Transaction Log Backup

Transaction Log Backup



SQL Server - Database Snapshots



Tablediff command line utility

\Program Files\Microsoft SQL Server\90\COM\tablediff.exe

- Tablediff identifies the data differences between two tables
- Tables must have same schema but can be on different servers
- Use the –f option to generate a SQL Server script to fix the out of sync destination version
- The diff algorithm uses an efficient hash technique

```
usage: tablediff
        -- Source Options --
-sourceserver                   Source Host
-sourcedatabase                 Source Database
-sourceschema                   Source Schema Name
-sourcetable                    Source Table or View
-sourceuser                     Source Login
-sourcepassword                 Source Password
-sourcelocked                   Lock the source table/view durring tablediff

        -- Destination Options --
-destinationserver              Destination Host
-destinationdatabase            Destination Database
-destinationschema              Destination Schema Name
-destinationtable               Destination Table or View
-destinationuser                Destination Login
-destinationpassword            Destination Password
-destinationlocked              Lock the destination table/view durring tablediff

        -- Misc Options --
-t                              Timeout
-c                              Column Level Diff
-f                              Generate Fix SQL (You may also specify a file name and path)
-q                              Quick Row Count
-et                             Specify a table to put the errors into
-dt                             Drop the error table if it exists
-o                              Output file
-b                              Number of bytes to read for blob data types (Default 8000)
-strict             Strict compare of source and destination schema
-rc                             Number of retries
-ri                             Retry interval
```

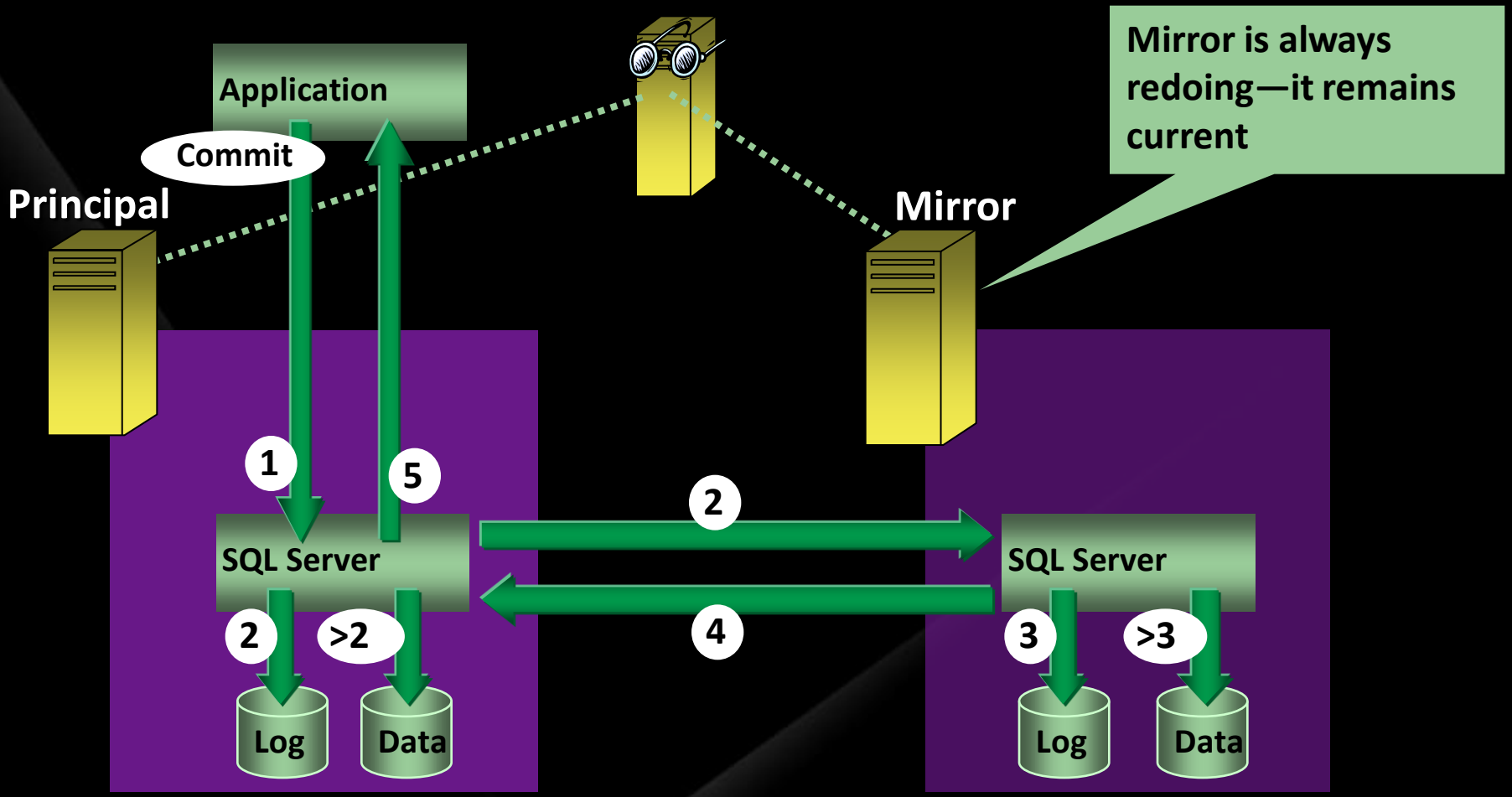
Categorizing Availability On Solutions

High Availability Requirement	Solution
System Administration	Online management
Recovery from application or user errors	Logical recovery
Site disaster protection	Database or storage mirroring
Database workload scale out with redundancy	Replication

Site Disaster Protection

- Downtime Scenarios
  - Earthquake, fire, or flood causes datacenter outage
- Always On Solutions
  - Database Mirroring to a secondary site
    - Optimized solution - allows very fast failover times to the secondary site
    - Optionally add log shipping for additional site protection
  - Third party geo-clustering solutions for data center storage level redundancy
    - Find SQL Server *Always On* reviewed solutions at the Microsoft *Always On* website: www.microsoft.com/SQL/AlwaysOn

Database Mirroring
Synchronous mode



Database Mirroring – trade offs

- Hot-standby
  - + Fastest failover (best RTO)
- Synchronous
  - + No data loss (RPO=0)
  - - Performance impact for commit

- Limited to one mirror per database
- Performance impact increases with number of mirrored databases
- No read access to mirror

demo

Automatic Page Repair in DB Mirroring

Name
Title
Company

msdn Microsoft Developer Network

Microsoft TechNet

Categorizing Availability On Solutions

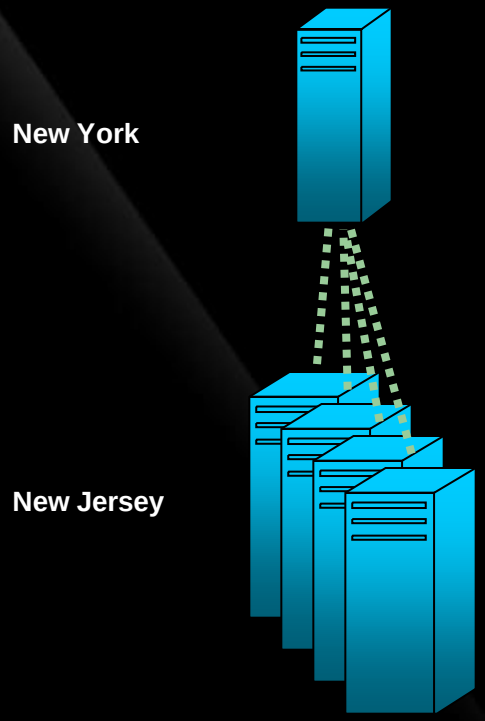
High Availability Requirement	Solution
System Administration	Online management
Recovery from application or user errors	Logical recovery
Site disaster protection	Database or storage mirroring
Database workload scale out with redundancy	Replication

Database Scale Out with Redundancy

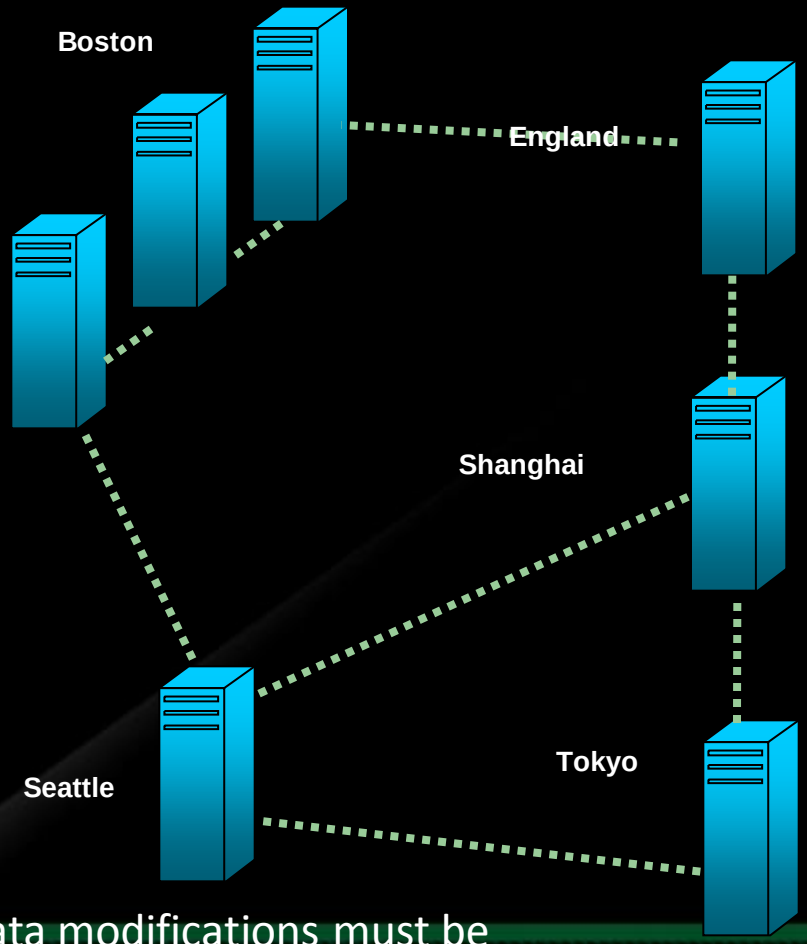
- Scenario
  - Real time reporting on one or more secondary servers that can also be used for disaster recovery
  - Tier of identical databases for scaling out applications which are partitioned by site
- Solutions
  - Transactional Replication
  - Peer to Peer Replication

Database Scale Out with Redundancy

Transactional Replication
Reporting + Redundancy



Peer to Peer Replication
Scale Out* + Redundancy



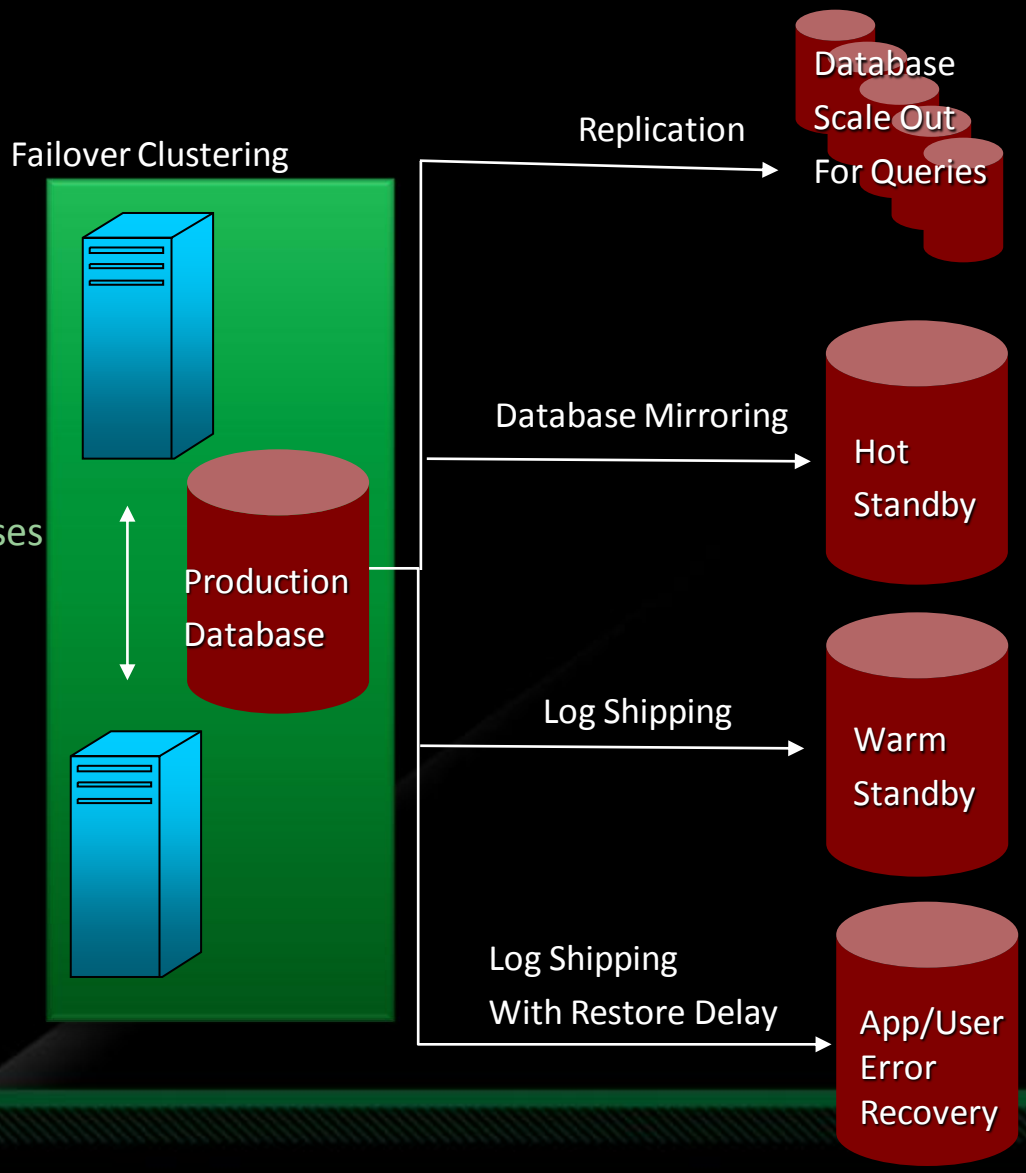
*Application data modifications must be partitioned by site to prevent conflicts

demo

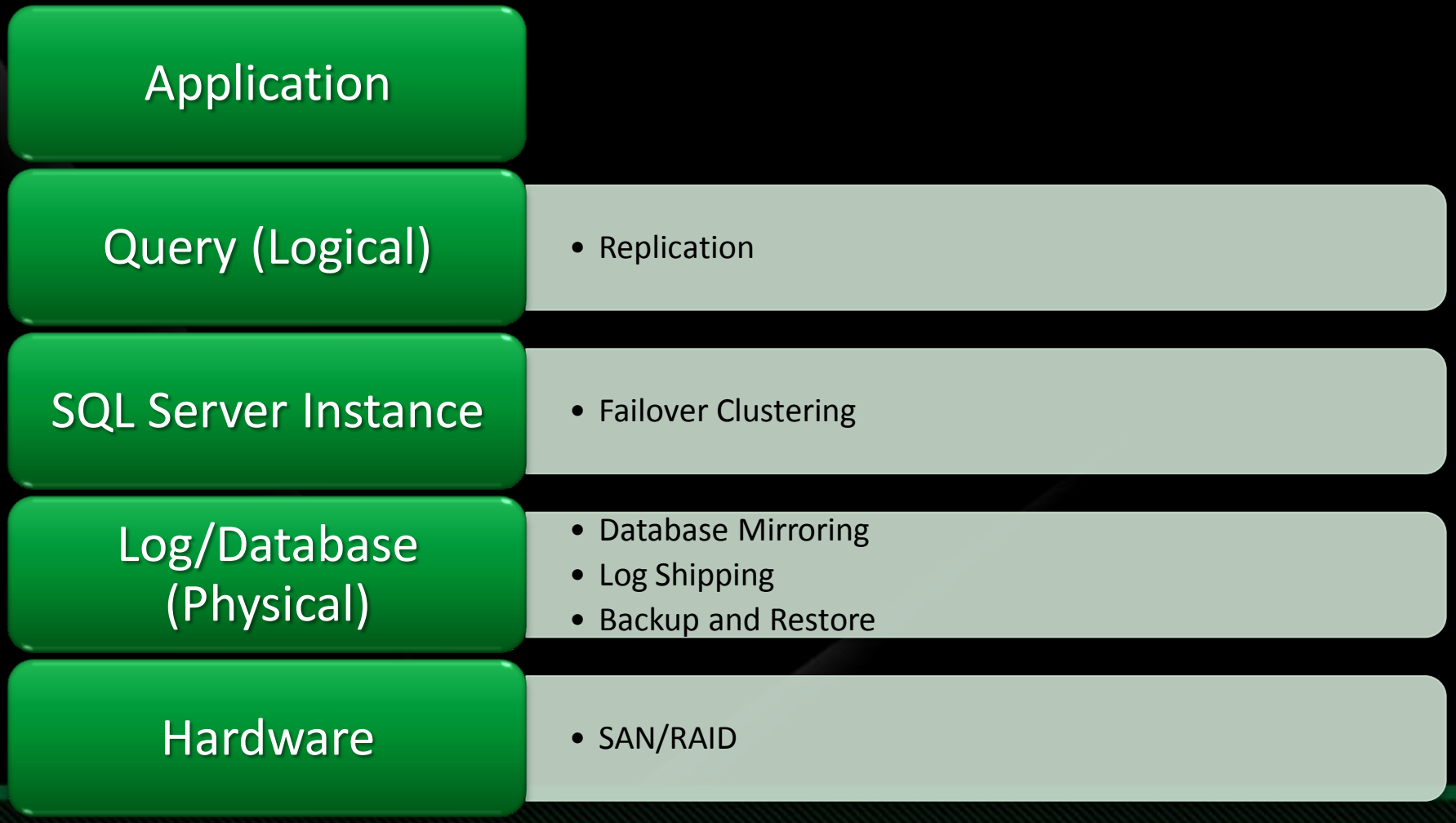
Peer-to-Peer Replication – Enhancements in SQL Server 2008

Combining Technologies

- Failover Clustering
  - Local server redundancy
- Database Mirroring
  - Primary disaster site for databases
- Log Shipping
  - Additional disaster sites for databases
  - App/user error recovery
- Replication
  - Database reporting and read scale out with redundancy
- Always On Partner Solutions
  - Highest hardware reliability



Summary - Unplanned Downtime Redundancy Stack



આભાર

ধন্যবাদ

நன்றி

धन्यवाद

Thank You!

ధన్యవాదాలు

ಧನ್ಯವಾದಗಳು

ধন্যবাদ

ਧੰਨਵਾਦ

നിങ്ങൾക്ക് നന്ദി

question & answer

msdn Microsoft Developer Network

Microsoft TechNet

Microsoft®

Your potential. Our passion.™