2008 IBM Rational
软件开发高峰论坛

Rational. software

开发的智慧 协作的力量

Informix Cheetah高可用性的技术特点及其实现

IBM公司/高级技术顾问
张积斌
zhangjb@cn.ibm.com
IBM软件部

Informix Cheetah的产品定位

Capitalize on its strengths

On line Transaction Processing

V.11 - new features increase the performance of IDS for OLTP. Multi-Active node Cluster with High Availability

Integrated Solutions

IDS is IBM's leading data server for industrial-strength integrated solutions.

Low DBA effort

Low DBA requirements are a strong selling point. Express and Workgroup Editions available.

Key Industries

IDS has significant market presence within key industries including Telco, Government, Retail & Banking

IDS is a "lead-with" data server for ANY OLTP oppty

IDS的发展

IDS的研发计划已经至2012年

11

专注于关键业务OLTP的企业级数据库系统
共享磁盘的数据库集群技术
99.999%的高可用性
并发访问的优化
基于标签的安全访问控制

10

灵活的数据页面设置
动态的优化器指引
字段级的数据加密
内存访问的优化

9.40

性能更上一层楼
高达128PB的扩展能力
更高的数据安全性
灵活的管理能力

9.30

共享SQL语句缓存
模糊检查点
动态日志技术
动态锁管理
数据刀片技术

7.x

高端数据库系统…
- 数据分片, 并行数据库查询, 高可用数据复制和企业级数据复制, 支持大量并发用户等.

6.x

增强的功能和性能…
- 多线程动态可伸缩体系结构(DSA), 异步 I/O, 完善的备份管理等.

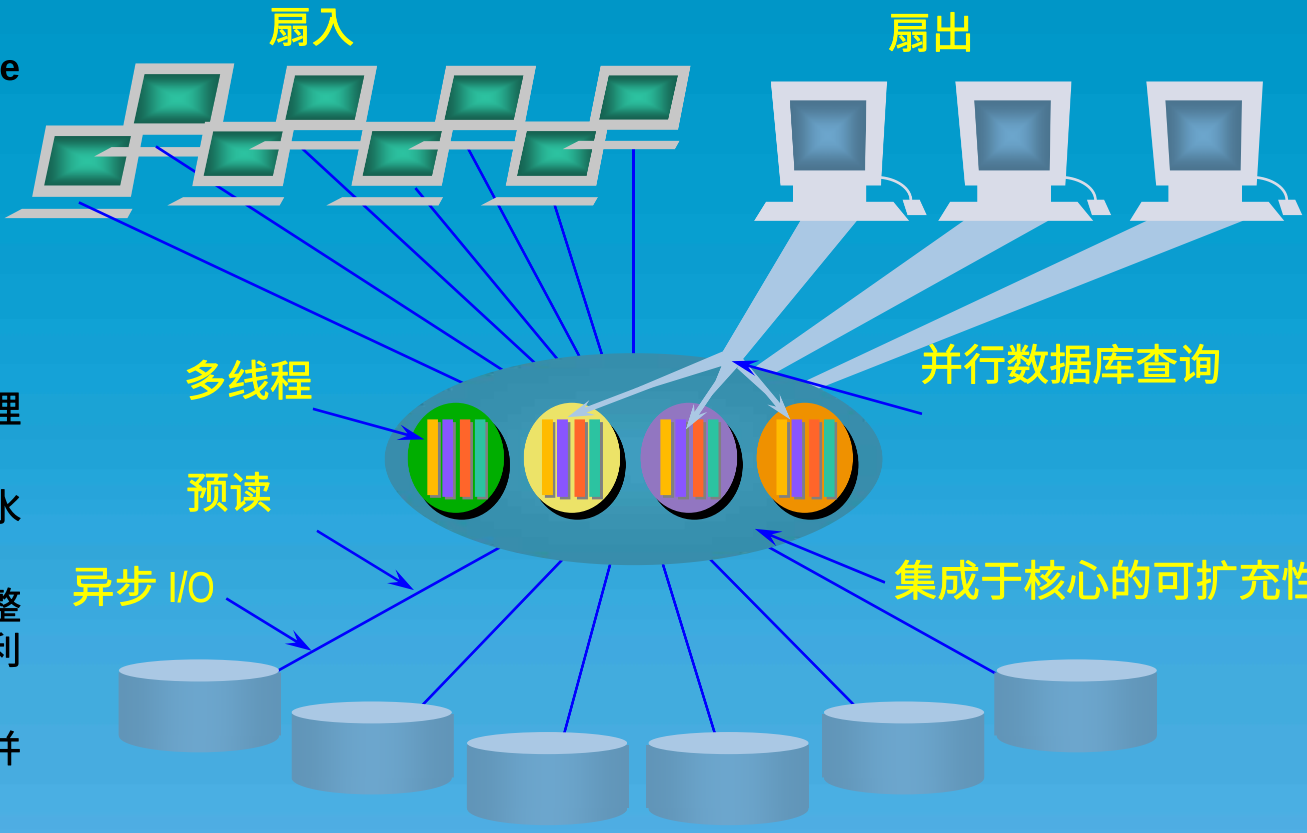
5.x

低端数据库系统,具有…
- 基于成本的优化器,两阶段提交,参照完整性等.

IDS并行的动态可伸缩体系架构-Parallel Everything

Modern parallel-everywhere base RDBMS

- 核心并行能力
- 动态调谐和管理
- 不断进行性能提升
- 不断进行功能扩展
- 多线索的进程构成虚处理机
- 分类的虚处理机组成流水线的工作方式
- 虚处理机数量可动态调整以适应系统负载及充分利用硬件资源
- 在多处理机性能扩展和并行处理方面有突出表现



为海量数据管理提供多种数据分割方案

Availability….可用性的定义

- **什么是可用性?**

  The proportion of time that an application can be used for productive work, measured against the time that it must be functional.

- **决定可用性的两个因素**
  - The reliability of the components that comprise the application: namely, how often they fail.
  - How long it takes for the application to be restored once a failure has occurred.

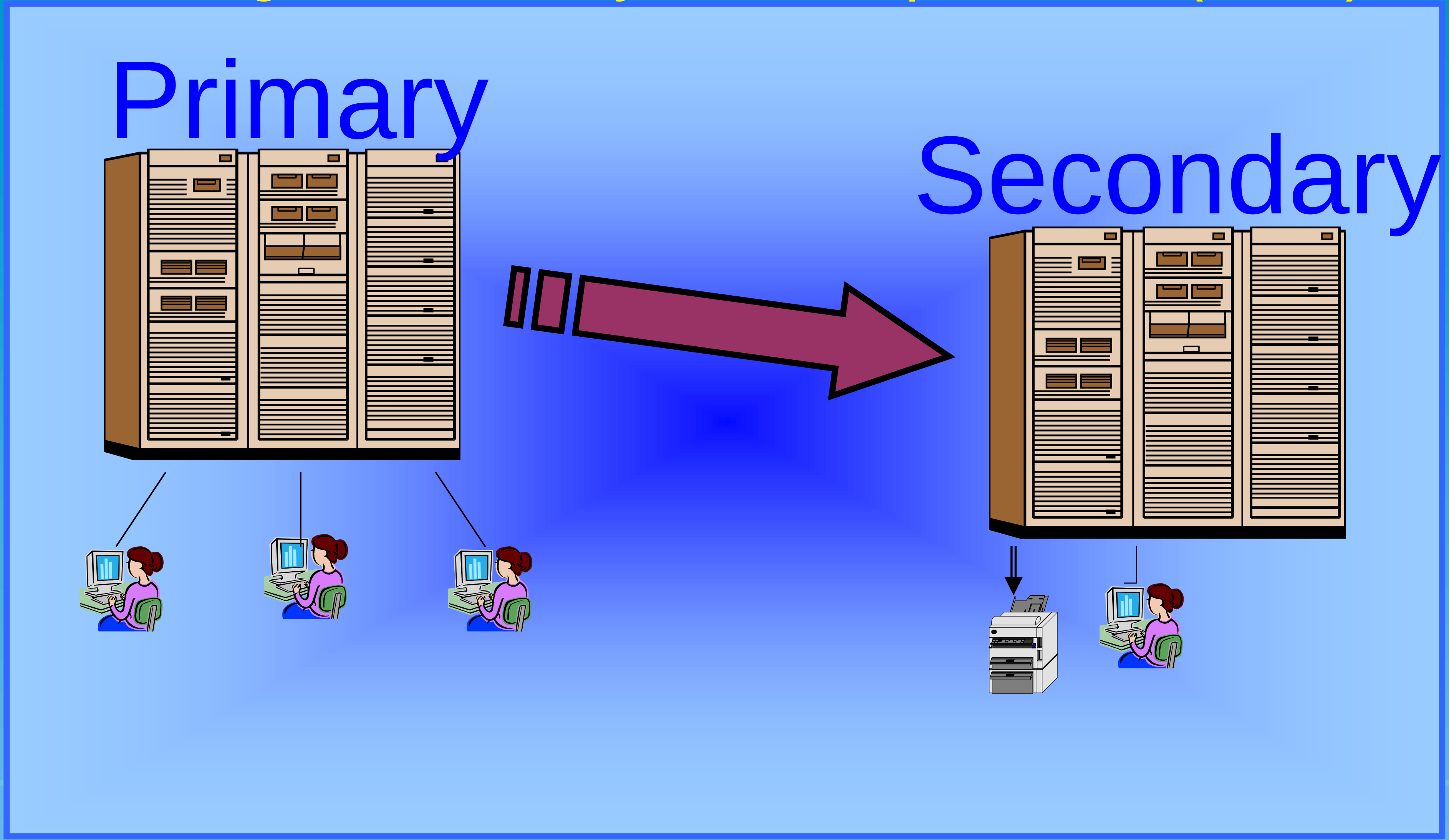
可用性的层次

Availability: The accessibility of a system resource in a timely manner; for example, the measurement of a system's uptime. Availability can be measured relative to "100% operational“ in terms of the number of “9s” that the system is available.

Availability Level Class	Uptime	Downtime Limit Per Year
2	99%	4 Days
3	99.9%	9 Hours
4	99.99%	1 Hour
5	99.999%	5 Minutes
6	99.9999%	30 Seconds
7	99.99999%	3 Seconds

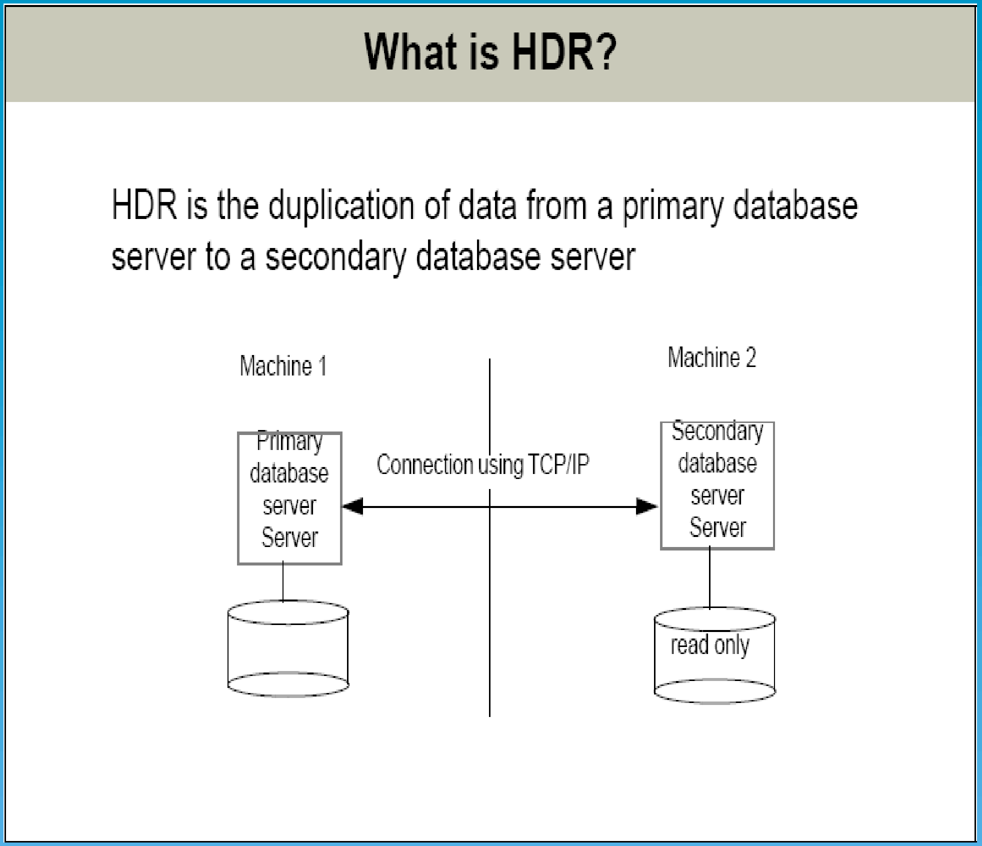
Source: “High Availability: A Perspective,” Jane Wright, Ann Katan (Gartner Group), 11/24/2004

Informix内置的高可用特性 --High Availability Data Replication（HDR)



What is HDR?

- Two identical servers on two identical machines
  - Primary server
  - Secondary server
- Primary server
  - Fully functional server
  - **<u>All database activity</u>** – insert/update/deletes, are performed on this instance
  - Sends logs to secondary server



What is HDR?

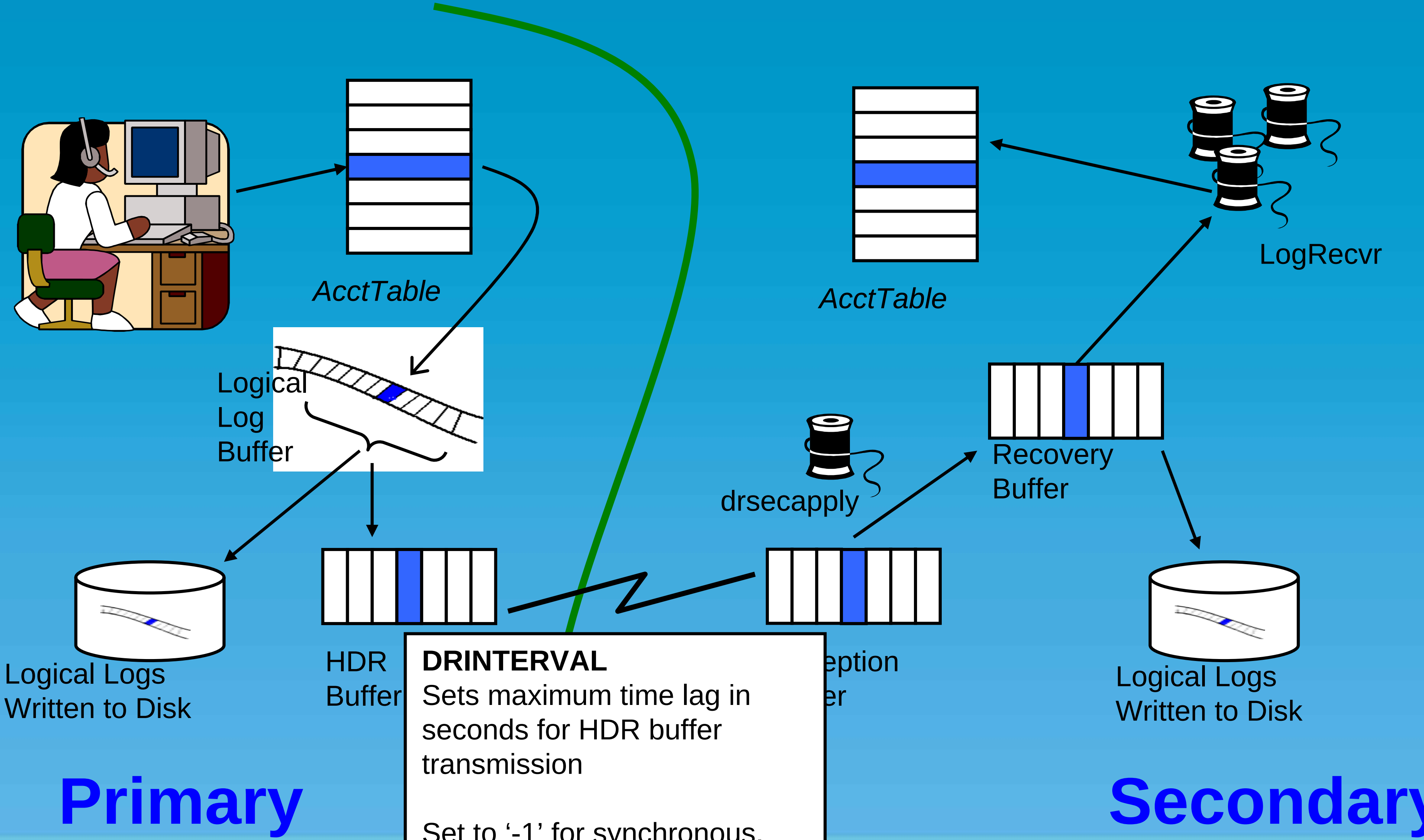
HDR is the duplication of data from a primary database server to a secondary database server

Machine 1
Machine 2
Primary database server Server
Connection using TCP/IP
Secondary database server Server
read only

HDR (cont…)

- Secondary server
  - **Read only server** : allows read only query (Pre cheetah)
  - Always in recovery mode
  - Receives logs from primary and replay them to keep in sync with primary
  - **Cheetah allow read/write operations**
- When Primary server goes down, secondary server takes over as Standard server

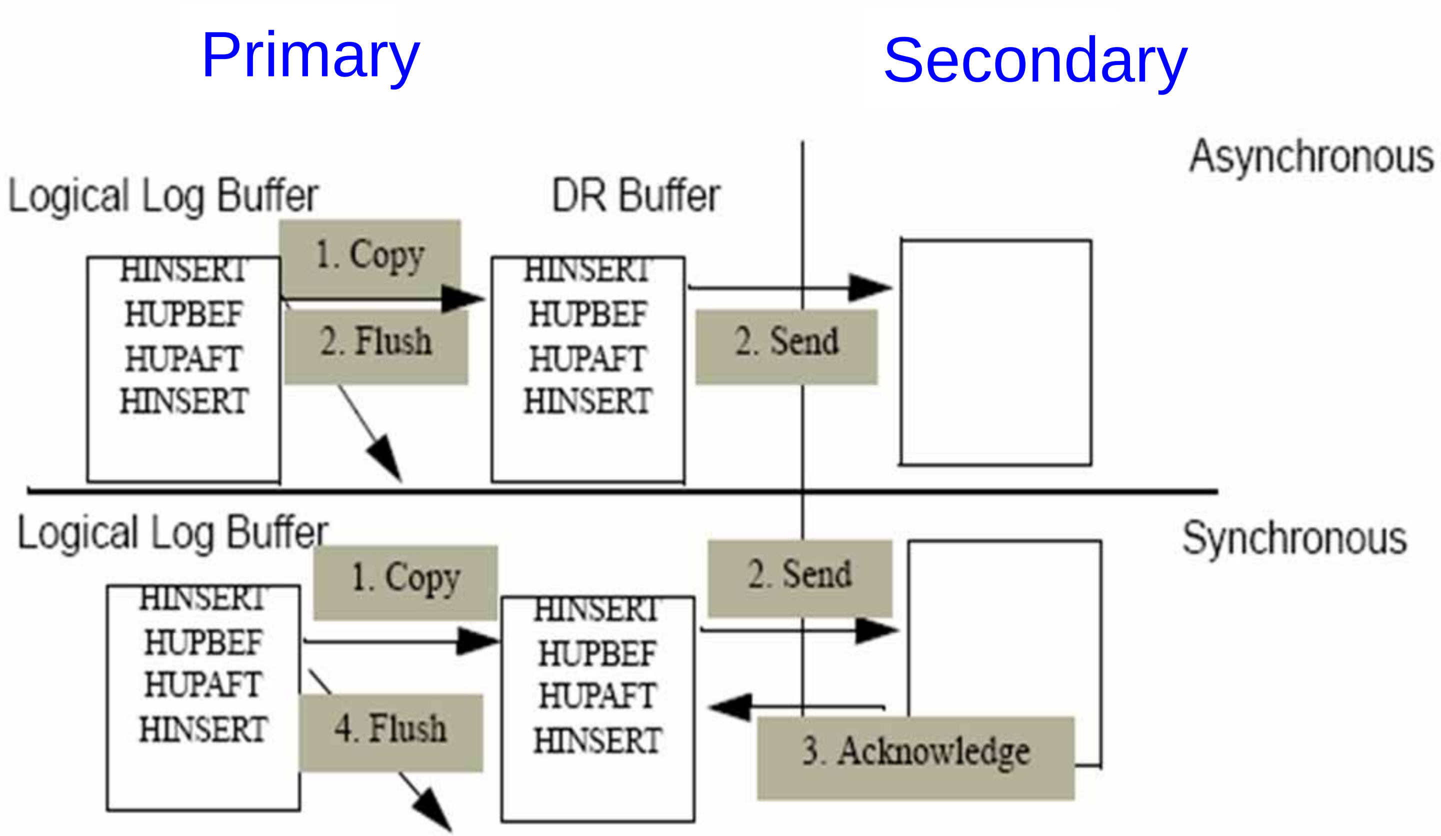
HDR – How it works



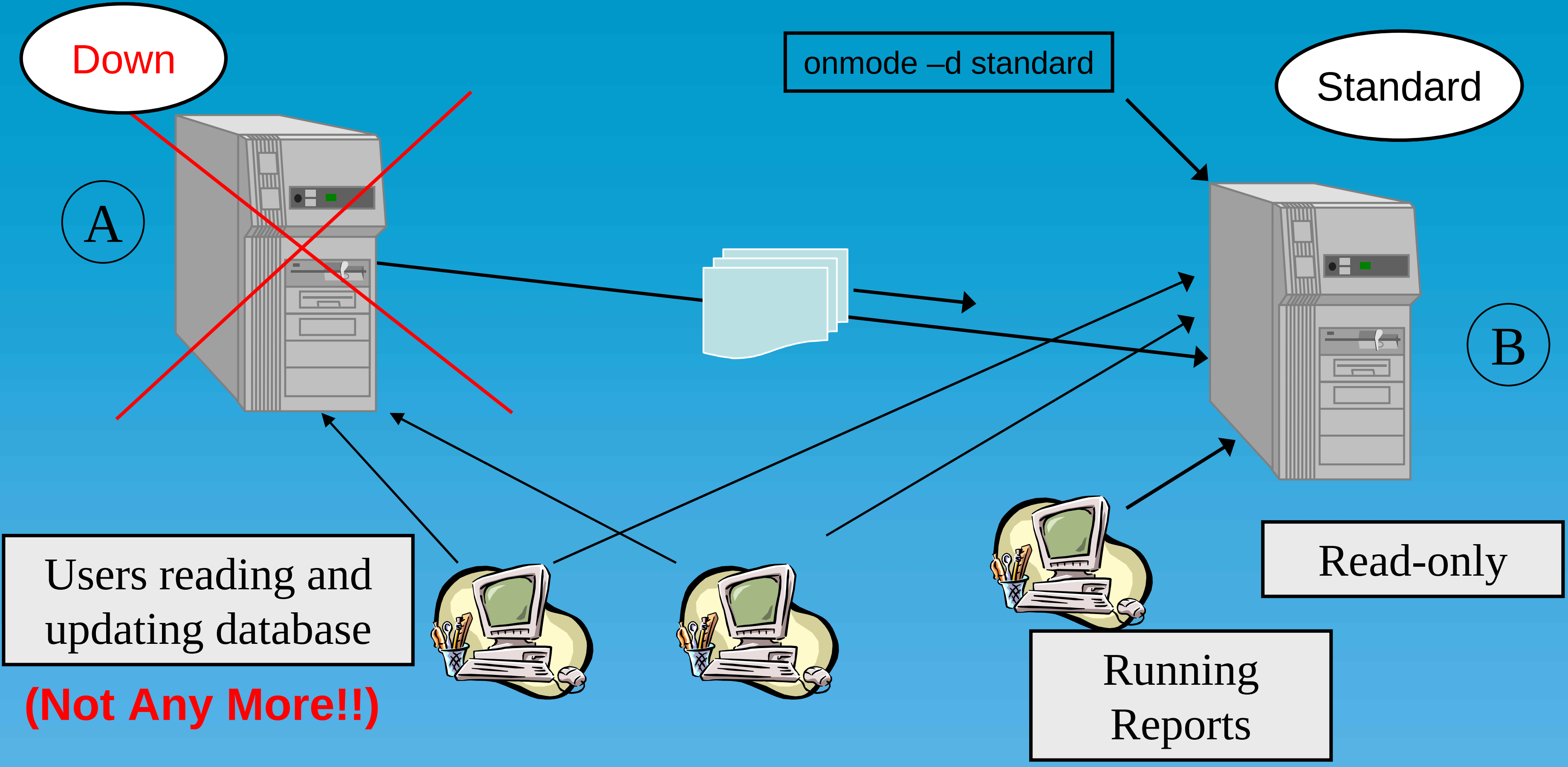
DRINTERVAL
Sets maximum time lag in seconds for HDR buffer transmission

Set to '-1' for synchronous.

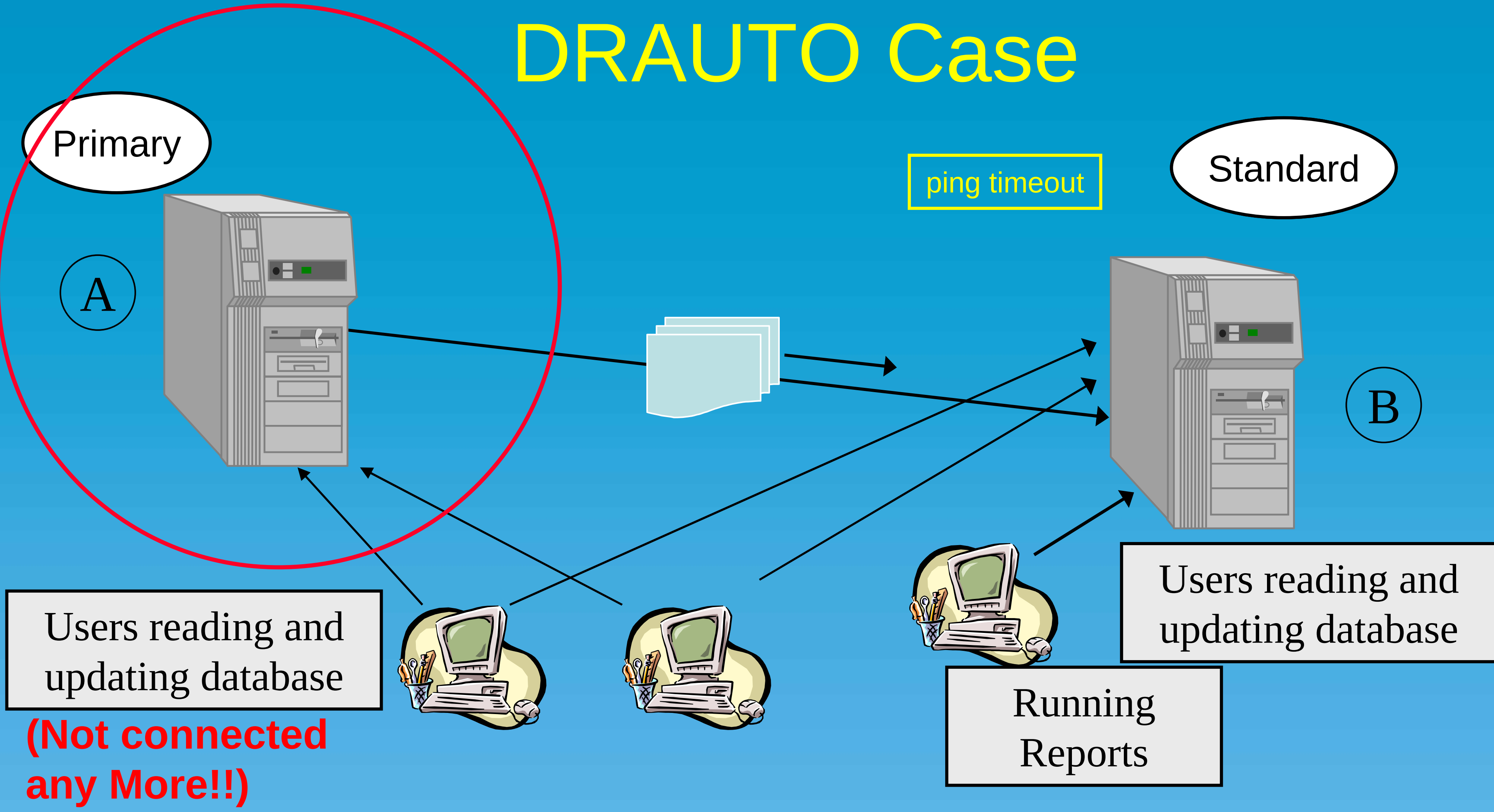
HDR -- Asynchronous vs Synchronous



HDR Availability - Failover



What If Network Is Down: DRAUTO Case



Users reading and updating database

(Not connected any More!!)

Users reading and updating database

Running Reports

HDR Reference Sites

• Walmart	(HPUX)
• PZU	(AIX)
• Wells Fargo Bank	(Sun)
• Rakuten	(Sun)
• German Border Patrol	(Sun)
• Phonehouse.DE	(Sun)
• Huawei	(AIX,HPUX,Sun)

Availability with IDS is platform independent

Factors that Affect HDR Availability

- Failure Detection Time
  - Network speed + OS tuning
- Failover Time
  - HDR takes 3 to 5 seconds
- Recovery Time for Open Transactions
  - When failover occurs, open transactions must be rolled back
  - Time for rollback is dependent on length of transactions
  - OLTP transactions should be short
  - Application may need to be changed to reduce recovery time
  - IDS Cheetah’s new Recovery Time Objective feature

Frequency of failures is the main determining factor

Cheetah – RTO Configuration Parameter

- RTO_SERVER_RESTART allows users to specify a target amount of time the server is allowed for fast recovery.

- RTO_SERVER_RESTART values:
  - 0 – off, uses CKPTINTVL to trigger checkpoints (<= 10.0 functionality).
  - 60 to 1800 seconds (1 – 30 minutes).

- Server will automatically monitor current workload and adjust checkpoint frequency to meet RTO policy.

- Server will fine tune with each fast recovery to improve predictability.

- Dynamically updatable with onmode –wm and –wf.

HDR vs Hardware HA Solutions

- AIX HACMP
- HP ServiceGuard
- Sun Cluster PDB

Choice of hardware vs. software solution

- Hardware solution is dependent on hardware vendor
- Disk Array is a single point of failure.
- Additional time needed for database initialization.
- Secondary database is not started until failover.
- Ultimate choice is cost dependent

Informix Cheetah’s new “MACH-11”

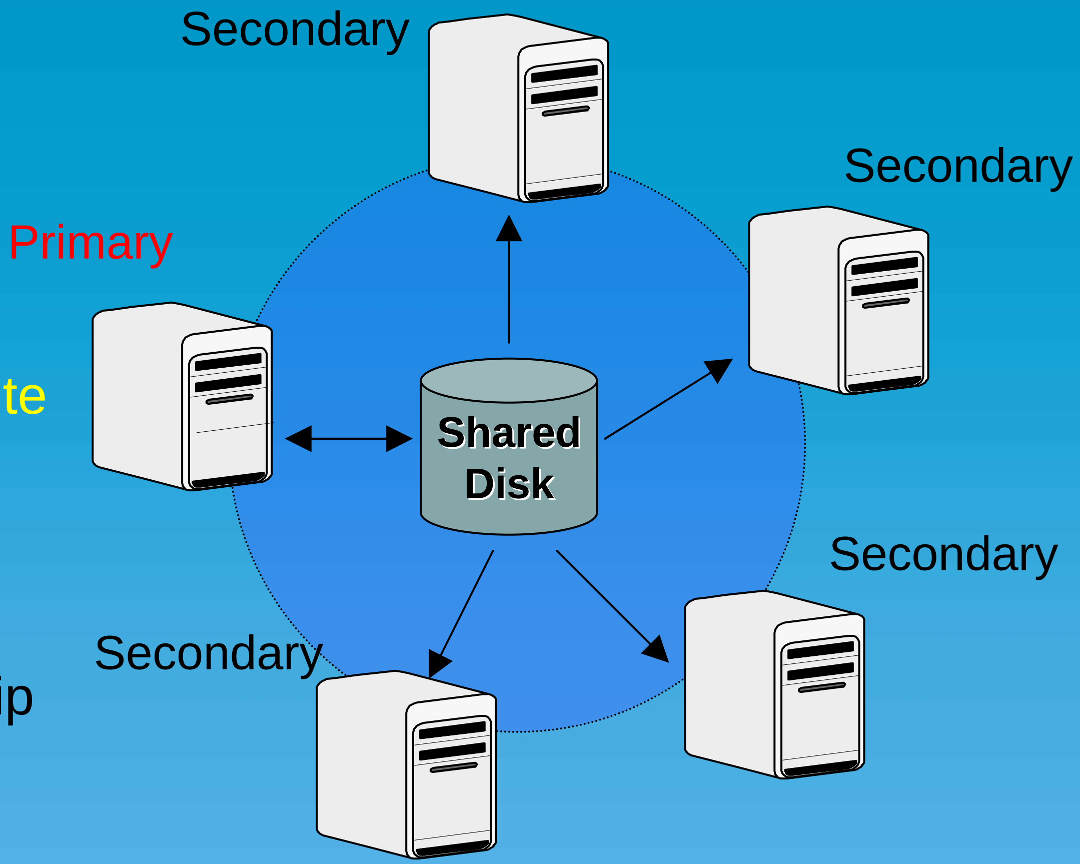
NEW!

- Multi-instance Active Cluster for High Availability.
  - Extends HDR to support more than a single primary with a single secondary instance.
  - Three new types of secondary instances:
    - Shared Disk Secondary (SDS)
    - Remote Standalone Secondary (RSS)
    - Continuous Log Restore (CLR) or “near-line” standby**
  - HDR, RSS & SDS technology can be used in any combination.

- “MACH-11” is not:
  - Just 1-to-N HDR.
  - “MACH-11” is the treating of all three forms of secondary as a multi-tiered availability solution.

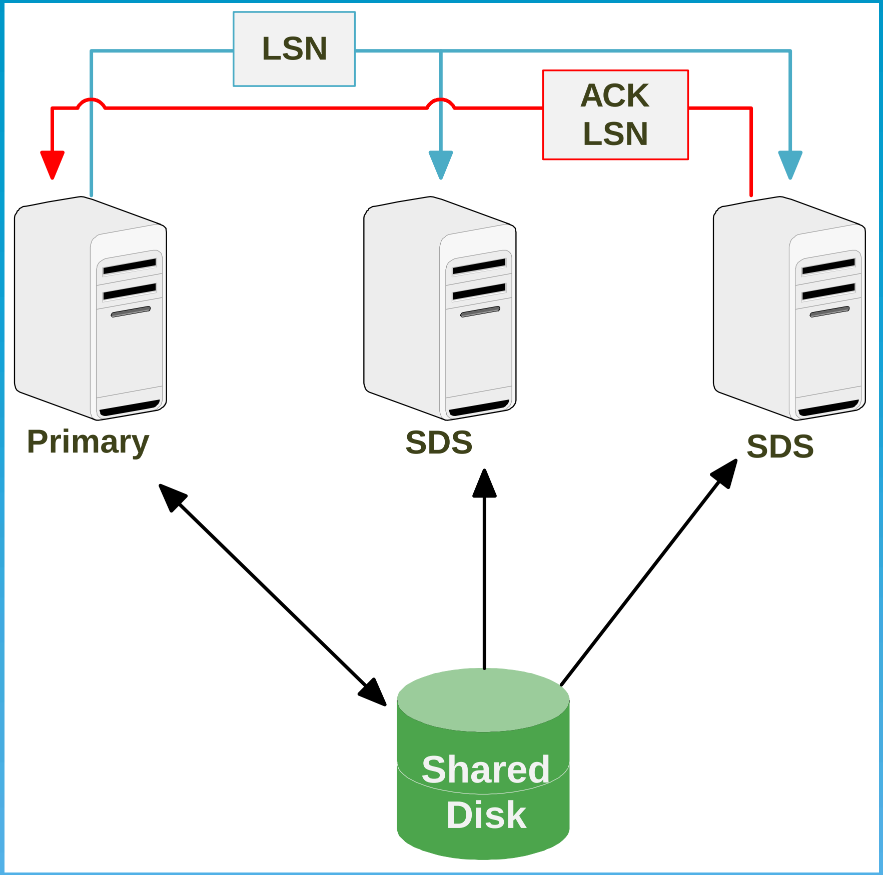
Shared Disk Secondary

- Share data
  - Single source of data
  - Multiple servers
  - Save data storage
- Optimize Capacity
  - Distributed read/write
  - Flexible
  - Easy scalability
- Multiple Redundancy
- Lower cost of ownership



Shared Disk Secondary instances

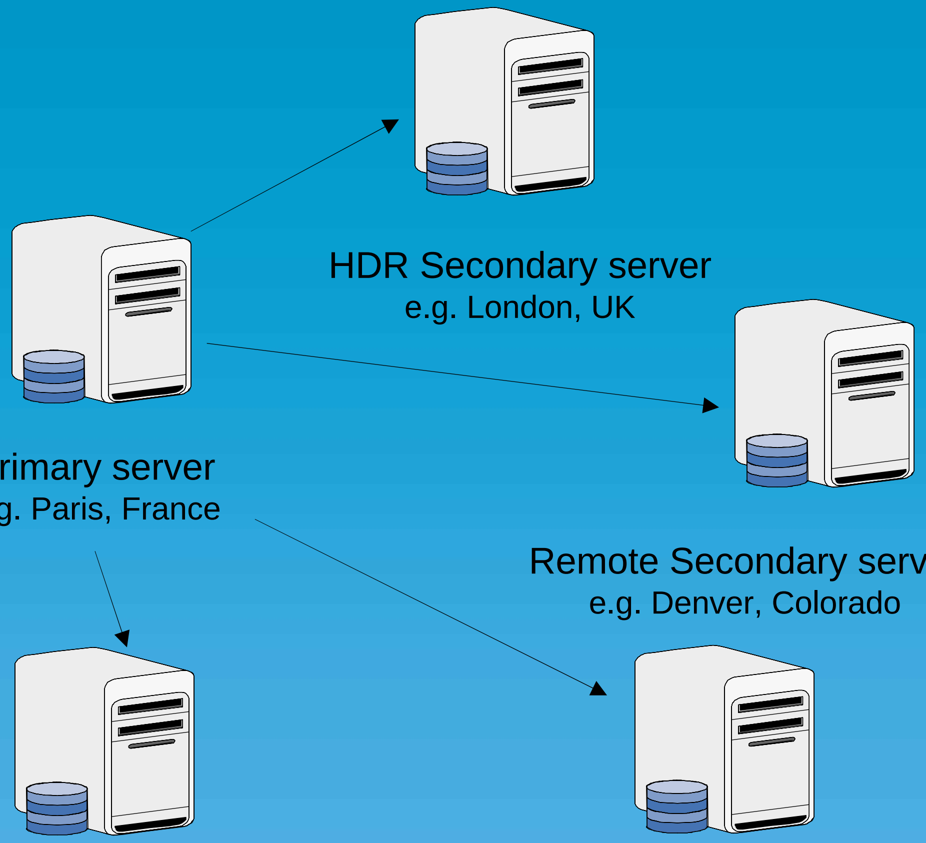
- Primary transmits the current Log Sequence Number (LSN) as it is flushing logs.
- SDS instance(s) receives the LSN from the primary and reads the logs from the shared disks.
- SDS instance(s) applies log changes to its buffer cache.
- SDS instance(s) resynch processed LSN to primary.



Remote Standalone Secondary

Evolution of HDR

- Multiple hot backups
  - Additional Secondaries
  - Global
  - Extends HDR
- Optimize Capacity
  - Distribute workload
  - Read/write locally
  - Improve performance
  - Increase capacity



HDR Secondary server
e.g. London, UK

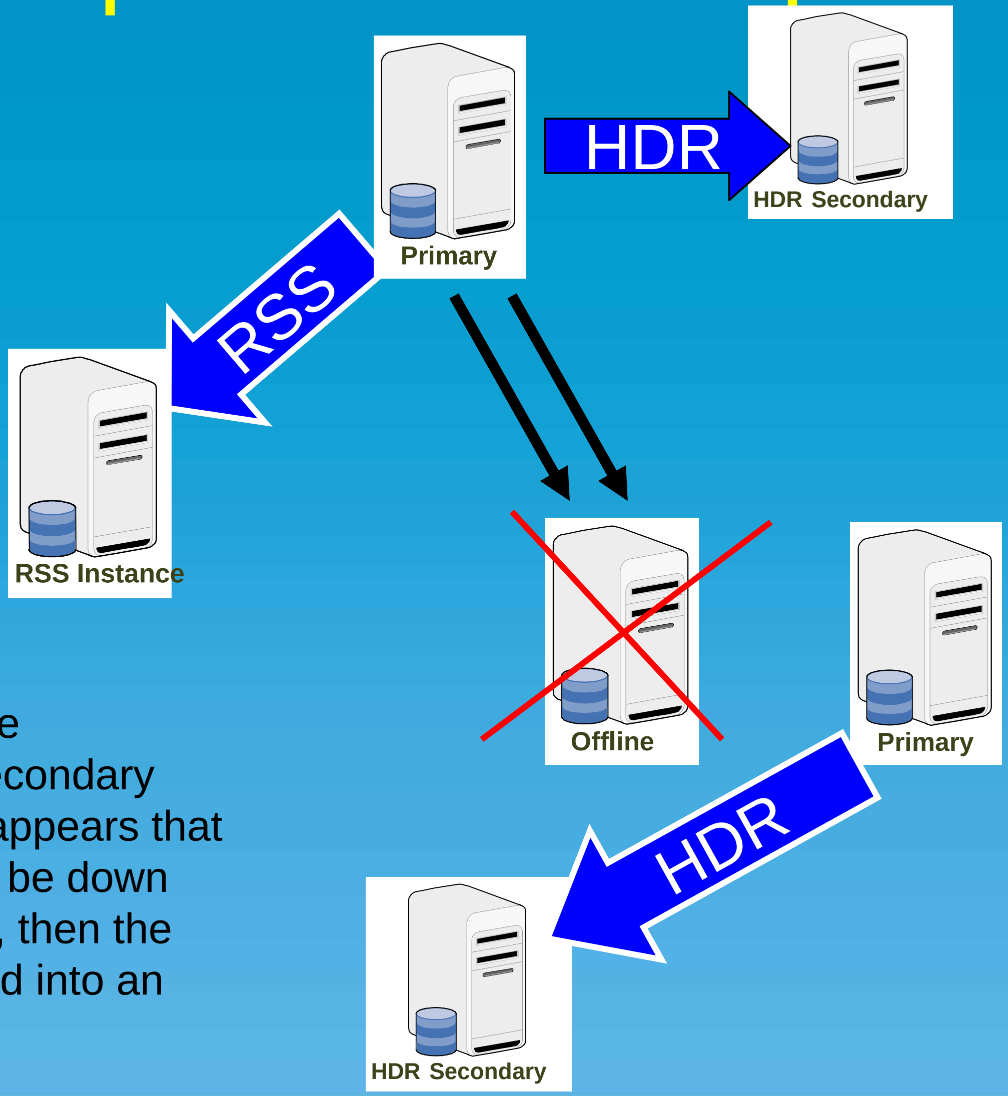
Primary server
e.g. Paris, France

Remote Secondary server
e.g. Denver, Colorado

Remote Secondary server
e.g. Mexico City, MX

Remote Secondary server
e.g. Shanghai, China

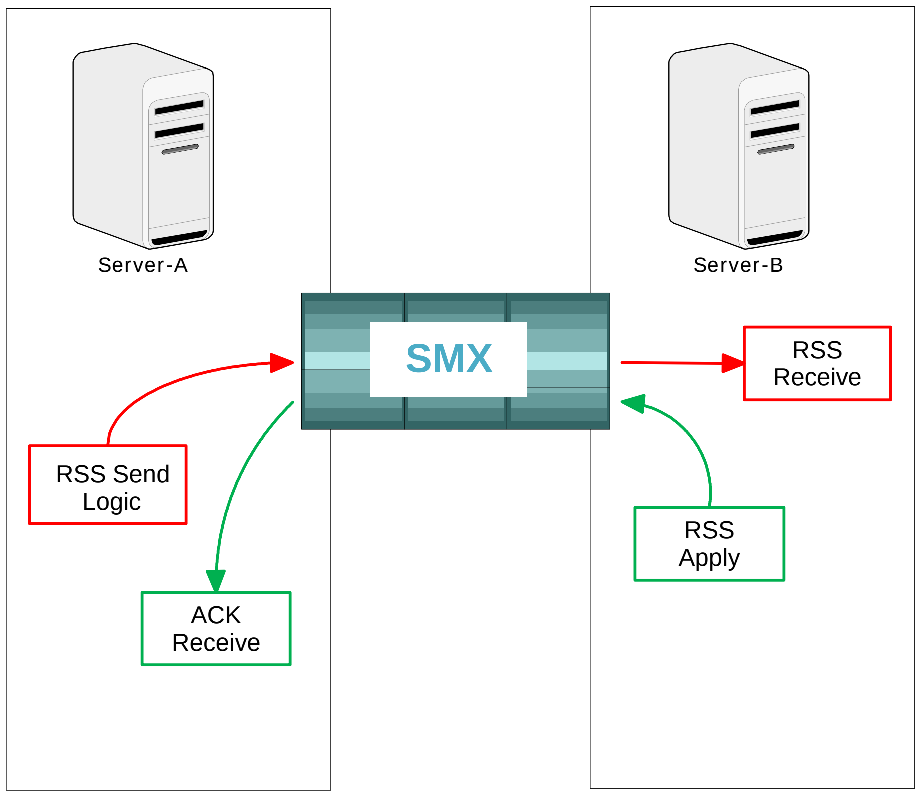
RSS – Backup to the Backup



If the primary fails, it is possible to convert the existing HDR secondary into the primary instance. If it appears that the original primary is going to be down for an extended period of time, then the RSS instance can be converted into an HDR secondary instance.

Server Multiplexer (SMX)

- Multiplexed network connection.
- Uses full duplex protocol
  - Sends packets without waiting for “ack” back:
  - HDR uses half-duplex so primary knows secondary received transaction before committing.
- Supports encryption.
- Automatically activated.
- Requires no configuration other than encryption.

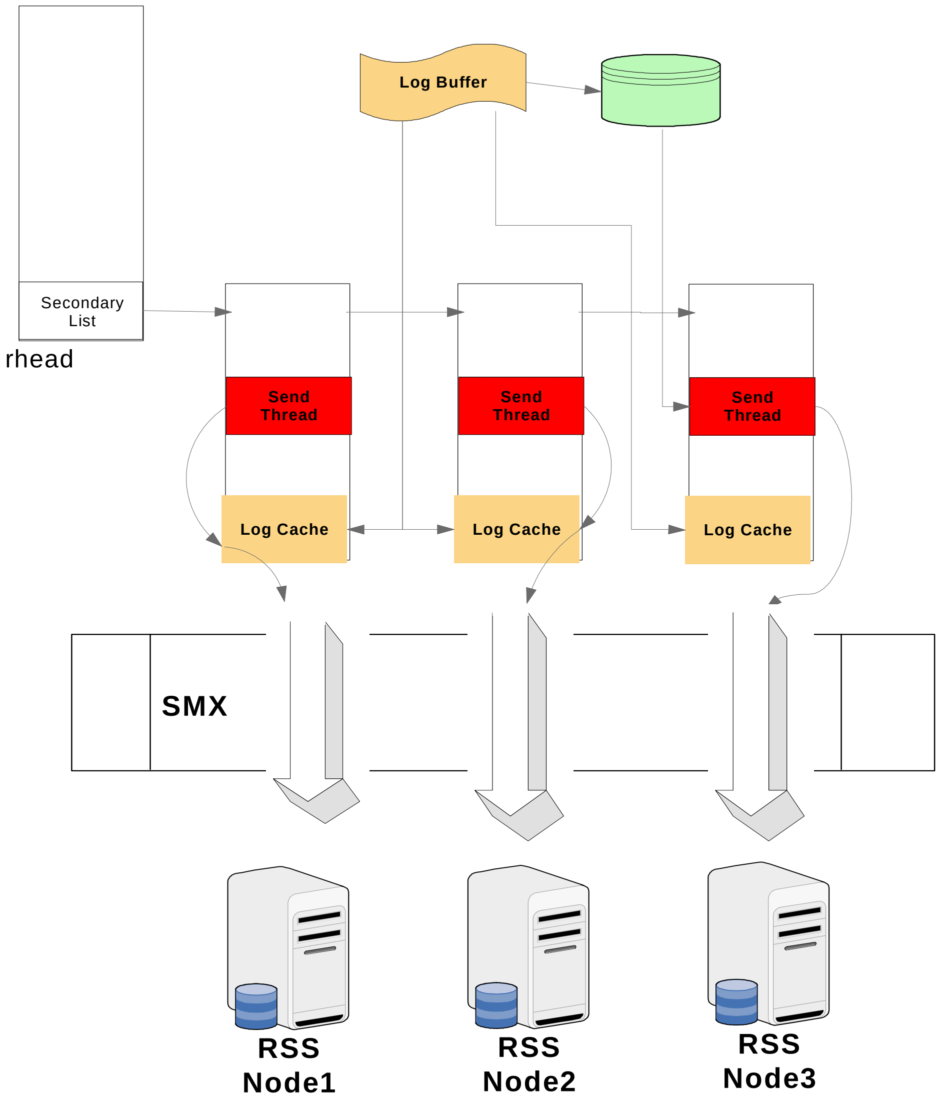


RSS: Primary – How It Works

In the current IDS engine, when we flush a log buffer to disk, we also copy that log page to the HDR log buffer and to the ER buffer cache. The primary now also checks to see if there are RSS secondary nodes and will copy that page to a log cache which is used to send that page to the remote node. If the send thread is currently sleeping, it will wake up the RSS_Send thread which will transmit that log page to the remote server.

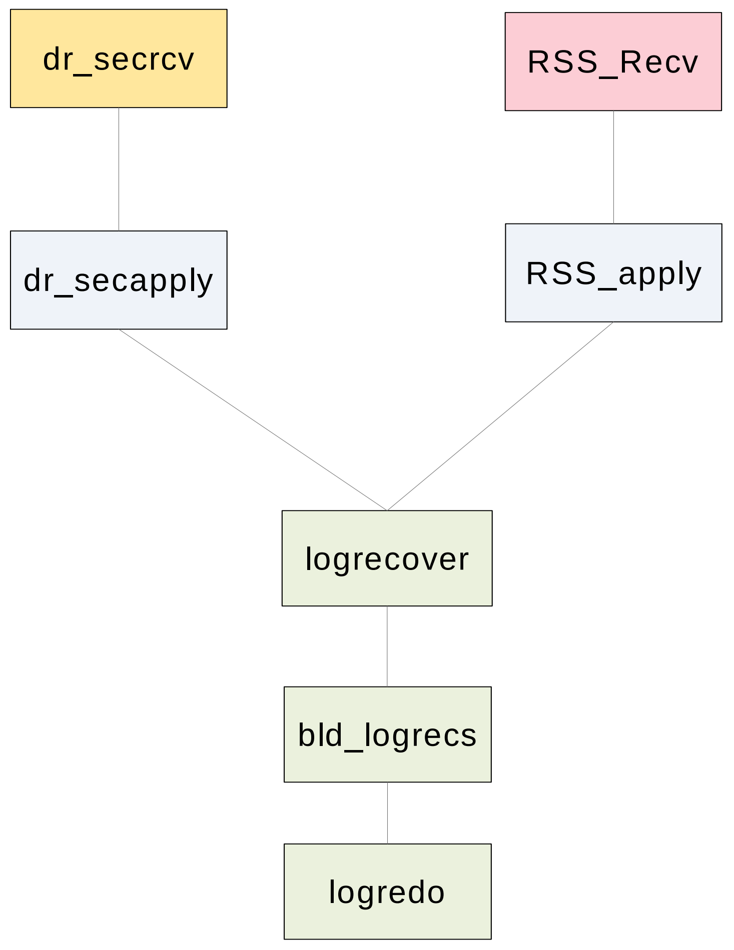
It is possible that the next page which needs to be sent is not in the log cache. In that case, the RSS_Send thread will read the log pages directly from disk.

The RSS_Send thread works in a fully duplexed network model. That means that the thread does not wait for an ACK before it sends the next buffer. Flow control is implemented in that up to 32 buffer transmissions will be sent before it requires an acknowledgement from the secondary node. If that limit is reached, then the send thread will block and wait for the RSS_Recv thread to receive an ACK from the peer node.



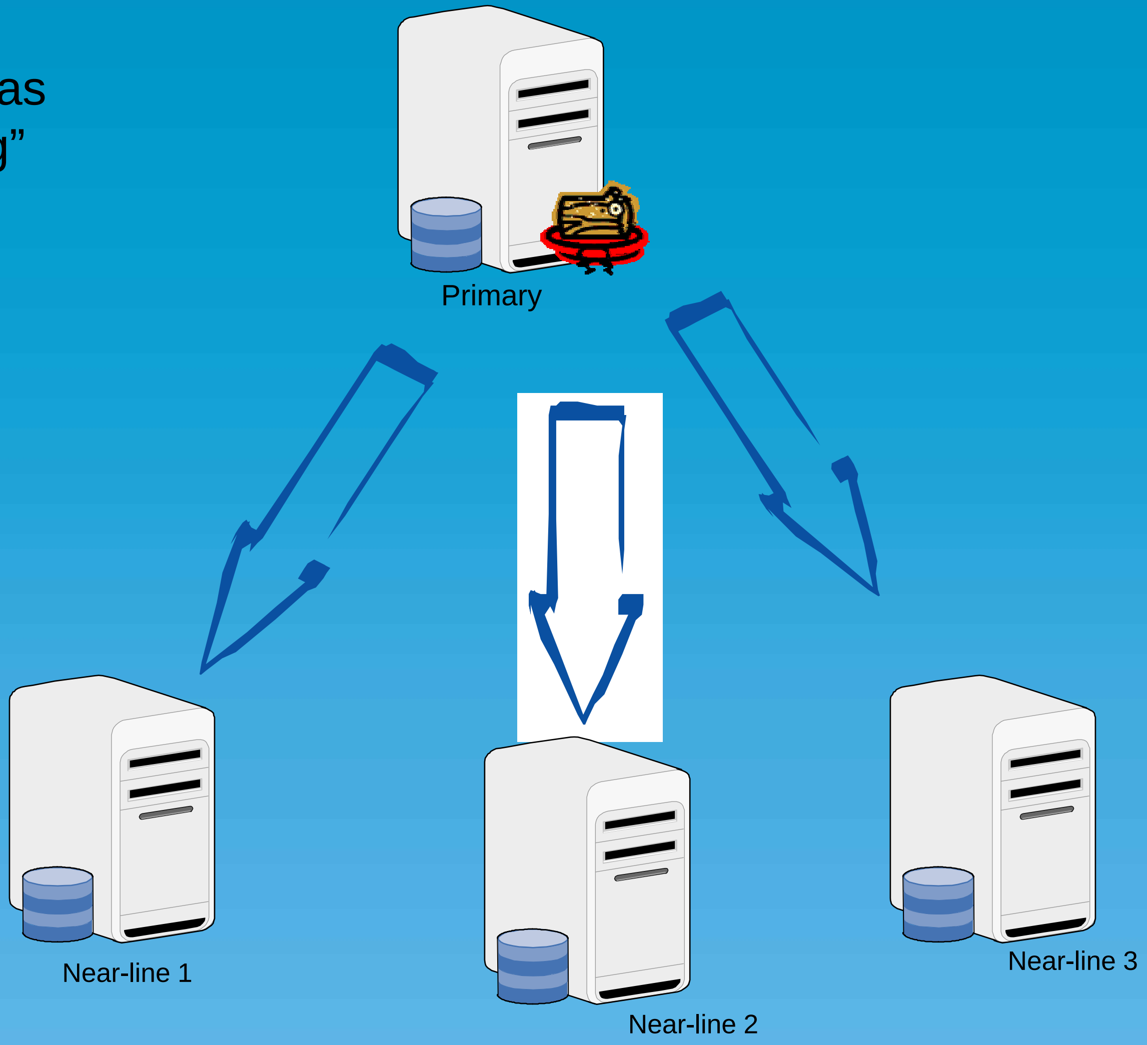
RSS: Secondary – How it Works

Existing HDR has a dr_secrcv thread which is responsible for receiving log pages from the HDR primary and interfaces with the HDR buffer pool on the secondary node. This is replaced by RSS_Recv which interfaces with SMX to receive the log pages from the primary node. RSS_apply is a minor modification of dr_secapply and is 98% common code with the HDR code which is used by dr_secapply. The rest of the recovery logic is common with existing HDR code.

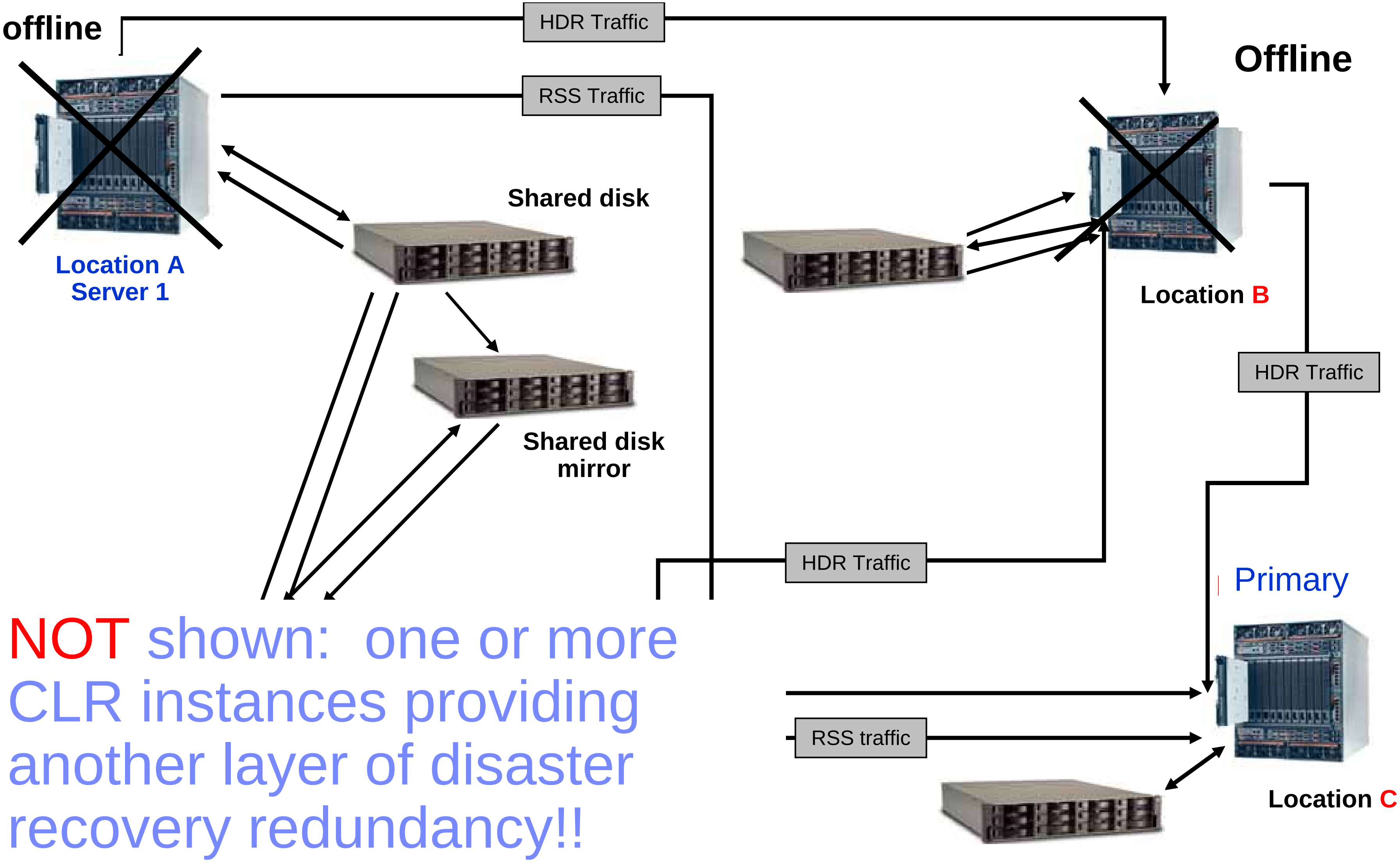


Near-line Standby

Also known as
“log shipping”



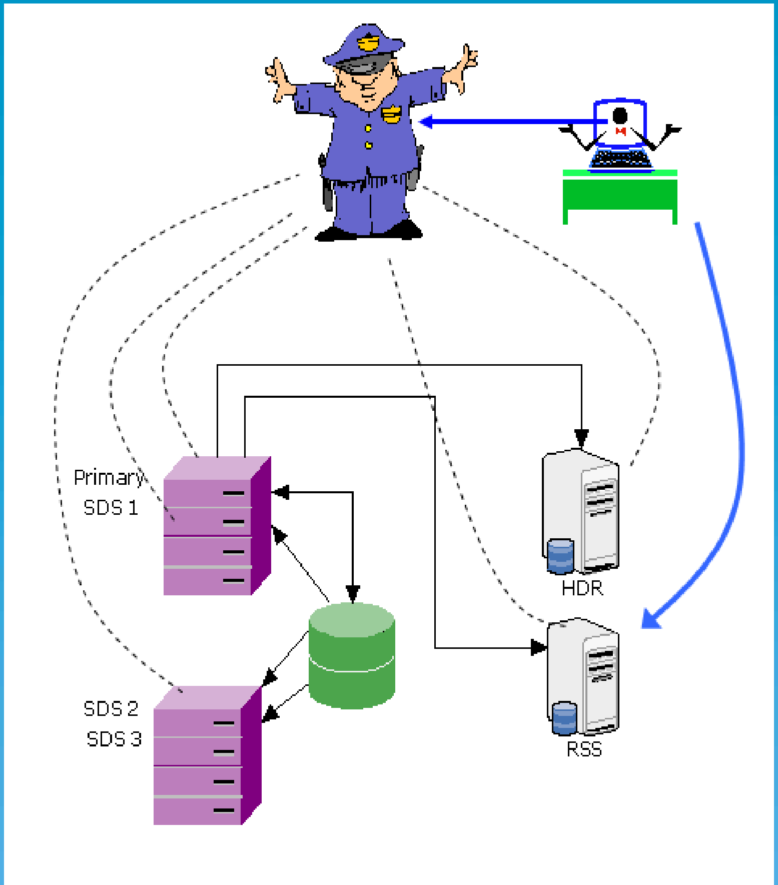
The server at location B fails



NOT shown: one or more CLR instances providing another layer of disaster recovery redundancy!!

Connection Manager

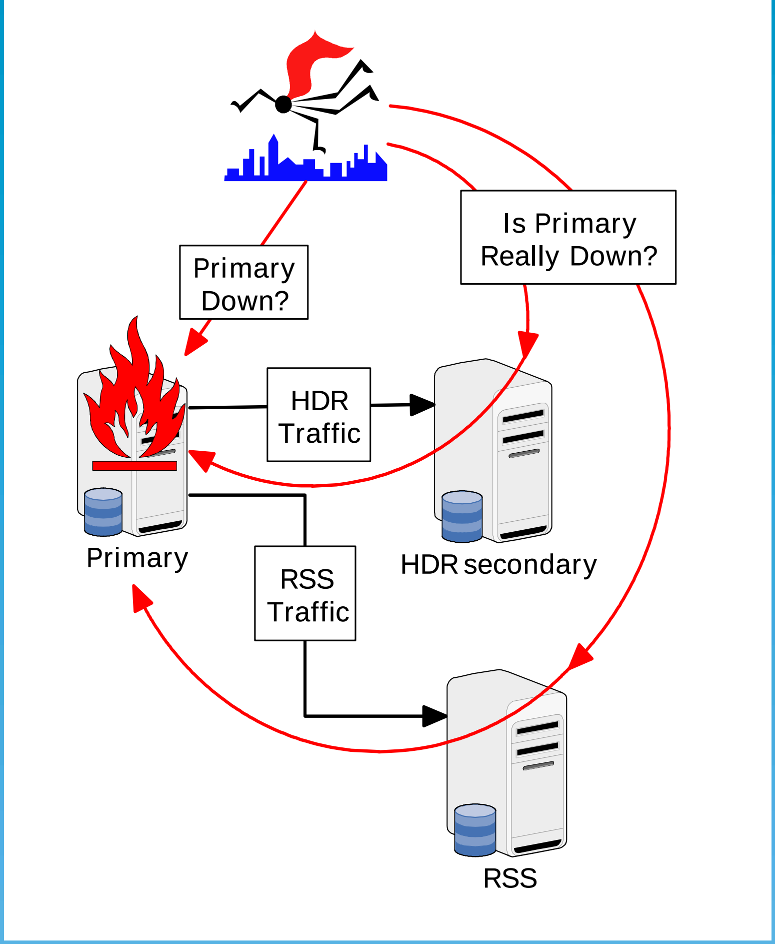
- Maintains knowledge of all nodes within the cluster
- Records adding/removal of nodes
- Monitors type of node
- Monitors workload of nodes
- Routes the client application to target node
- Automatic failover
- Works on the concept of class of service by resolving the following requirements



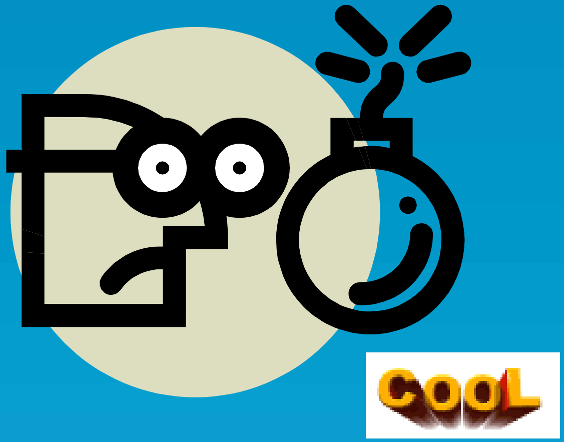
•Connect to the best possible secondary
•Connect to the current primary
•Connect to the SDS node or primary with the most free CPU cycles
•Connect to either the HDR primary or the HDR secondary
•Connect to an SDS node or HDR secondary, if any are currently active, otherwise connect to the primary

Arbitrator

- Purpose is to quickly detect the failure of the current primary.
- Part of the connection manager
- Gets confirmation through alternate paths before performing a failover
- Performs a failover using admin API interface
- There is failover for the arbitrator just as there is failover for the connection manager.



Failover



- SDS can be used with HDR and RSS.

- SDS can be promoted directly to primary:

  onmode –d set primary SDS <sds_instance_to_promote>

  When the primary is changed, other instances ‘follow the change’.

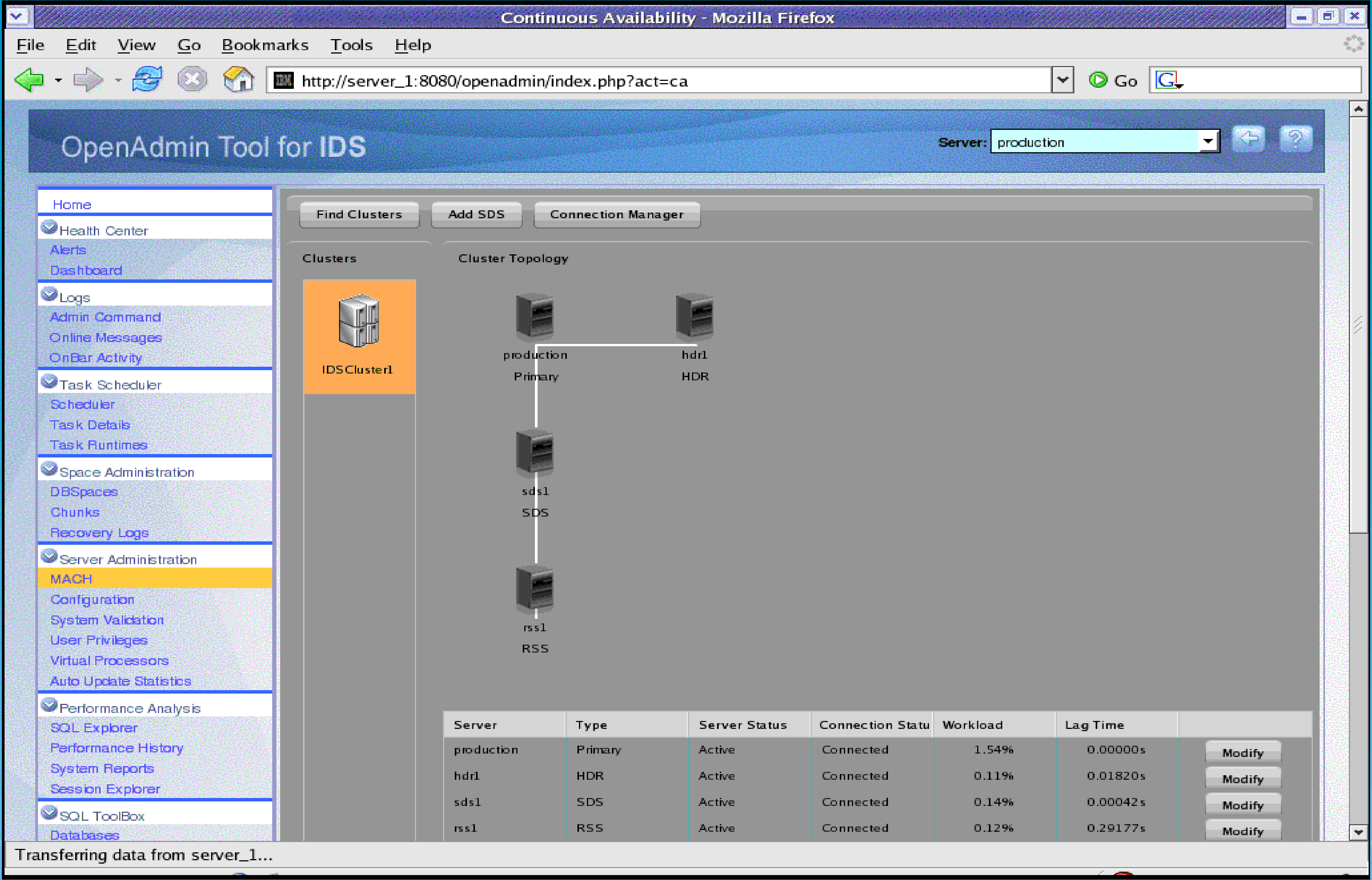
- Order of failover should be:

1. To an SDS instance.
2. To the HDR secondary.
3. To an RSS instance.

MACH11 – Database Cluster

- Redirected Writes
  - Allow HDR,SDS,RSS to handle queries which modify data
    - Update processing spread between SDS node and Primary
- Network Services
  - Runs as a separate middle layer outside IDS
  - Clients connect to this middle layer rather than directly to an SDS or Primary node
  - Detects when an RSS or SDS node fails
    - Provides application redirection
  - Provides automatic failover to RSS or SDS nodes
- Administration via IDSAdmin
  - Configure and start new nodes
  - Better graphical representation of clusters

IDSAdmin Tool



Continuous Availability - Mozilla Firefox

http://server_1:8080/openadmin/index.php?act=ca

Server: production

Server	Type	Server Status	Connection Statu	Workload	Lag Time	
production	Primary	Active	Connected	1.54%	0.00000s	Modify
hdr1	HDR	Active	Connected	0.11%	0.01820s	Modify
sds1	SDS	Active	Connected	0.14%	0.00042s	Modify
rss1	RSS	Active	Connected	0.12%	0.29177s	Modify

Transferring data from server_1...

IDS MACH11 LifeDemo

- Informix MACH11 Database cluster
- Informix MACH11 High Availability

Mach11.wrf

Thank
You