

Availability

- **Availability** – Ability of a component or service to perform its required function at a stated instant or over a stated period of time.
- **Continuous Availability** – Attribute of a system to deliver non-disruptive service to the end user 7 days a week, 24 hours a day (no planned or unplanned outages)
- **Continuous Operations** – Attribute of a system to continuously operate and mask planned outages from end-users
- **High Availability** – The attribute of a system to provide service during defined periods, at acceptable or agreed upon levels and mask unplanned outages from end-users

Recovery

- **Disaster Recovery** - can you tolerate a major disruption caused by a disaster, and fully recover essential data and operation
- **Business Recovery Objective** – The desired time within which business processes should be recovered, along with the minimum staff, assets and services required within this time
- **Network Recovery Objective** – How long can you afford to take to switch over the network
- **Recovery Time Objective** – How long can you afford to be without your systems?
- **Recovery Point Objective** – When recovered, how much data can you afford to re-create?

Customer Considerations

Service Level Agreement (SLA)

Scheduled down time

- ◆ Tape backup, application upgrade, OS upgrade
- ◆ how many hours per week can the application be unavailable?

Unscheduled down time

- ◆ user error, application error, environmental problem, hardware failure
- ◆ how many hours following a failure before the application is available?

Disaster

- ◆ loss of computer room
- ◆ what is the minimum distance between production and backup locations?
- ◆ how many hours before the environment is recreated at another location?

Application Considerations

Middleware

- ◆ Websphere and WebSphere MQ
- ◆ High Availability business partner

Granularity

- ◆ customer applications which are managed independent of each other

Logical Replication

- **Benefits:**

- **Costs**

- Only **two copies** of disk required
 - Very **flexible** (any hardware or disk, various OS release, most ISV software, etc.).
 - Performed at the lowest unit of change for the object, e.g. **record level** for database i5/OS Remote Journaling provides efficient, reliable transport mechanism
 - HA Edition and Capacity Backup Edition available

- **Geographical dispersion**

- Replication on a data and object level (file, data area, program, etc.) **near real-time**

- **Concurrent access**

- Replicated data can be **concurrently accessed** for backups or other read-only apps

- **RTO/RPO**

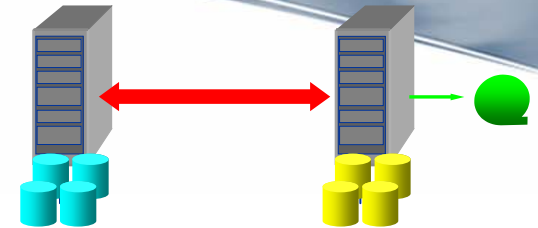
- Rapid switch or failover to production environment on backup server,
no IPL required

- **Outage type coverage**

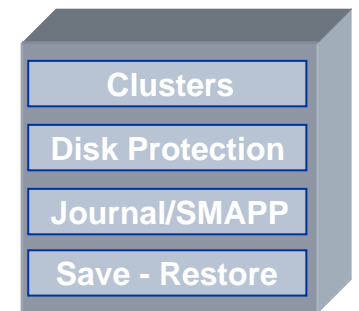
- Covers all outage types

- **Considerations:**

- Application and operations can be more complex to setup and maintain



Building Blocks



Application Resilience

Fully transparent

- ◆ Full resilience with automatic restart & transparent failover
- ◆ Users repositioned to last committed transaction
- ◆ No data loss, no sign-on required, no perceived loss of server; only delay in response

Semi-transparent:

- ◆ Automatic application restart & recovery to last transaction boundary
- ◆ The resilient data & the application restart point match exactly

Semi-automatic:

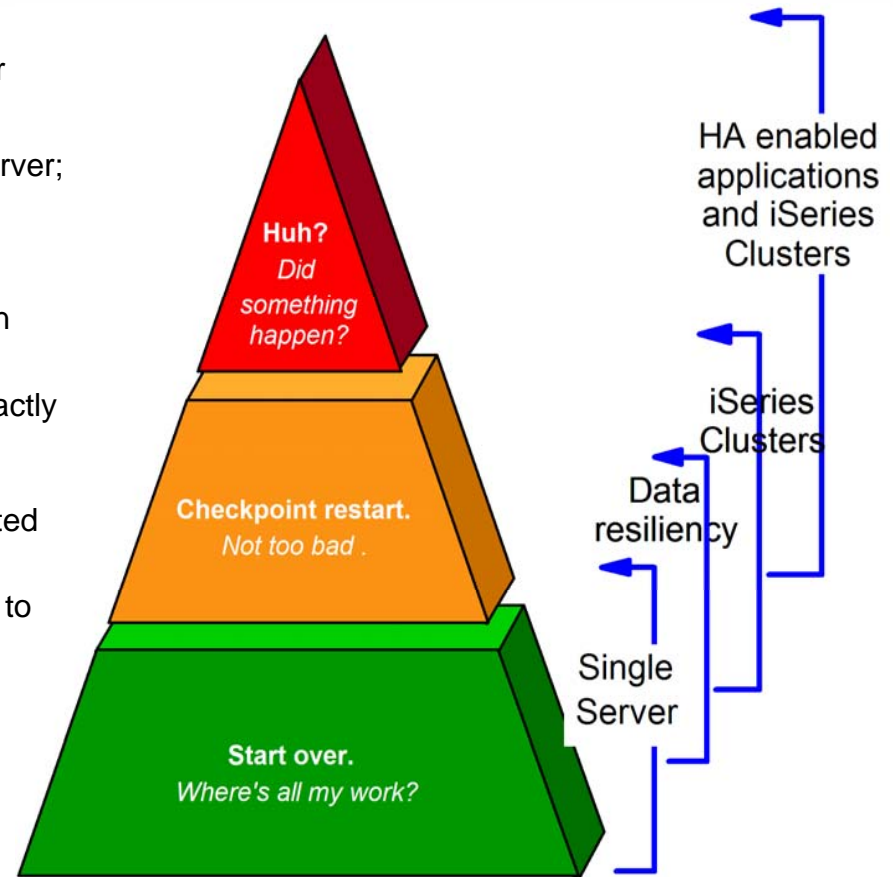
- ◆ Automatic application restart & recovery to some architected application "restart" point
- ◆ Normally consistent with state of data, but user may have to manually match application to position of data

Basic application failover:

- ◆ Automatic application restart after outage
- ◆ User manually repositions within application

No application recovery:

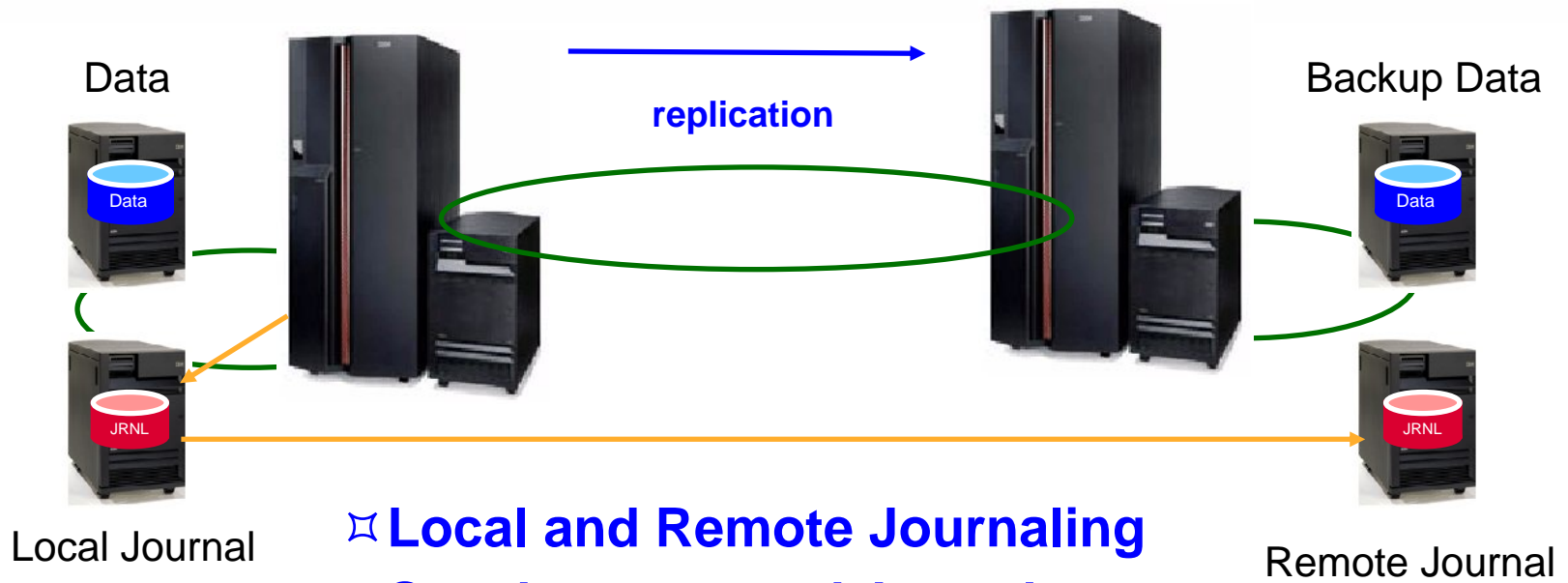
- ◆ Users manually restart application with resilient data
- ◆ User determines where to resume work



Switchover and Failover Defined

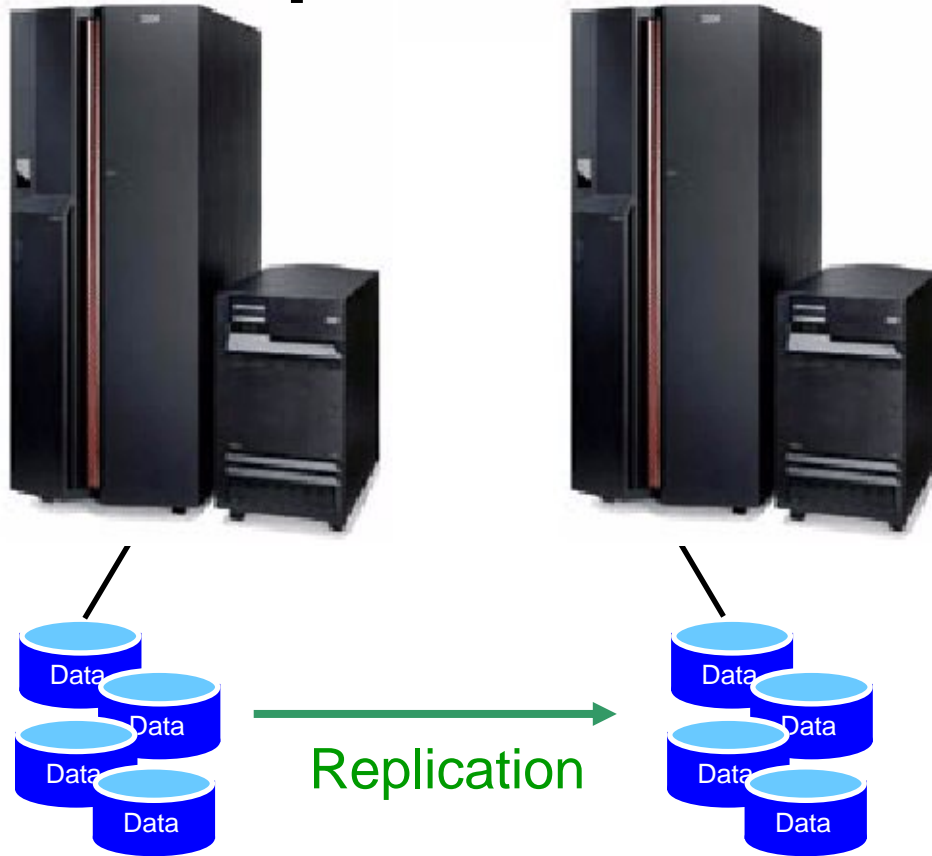
- A **switchover** is a manual process initiated by the user in order to move server functions during a planned outage of the primary server, to the backup server.
- This process involves the pausing of the application environment and system functions on the primary server and the initiation of the application and system resource on the backup system
- A **failover** is the capability to switch automatically to a redundant server upon the failure or abnormal termination of the previously active server. This happens generally without warning

i5OS Journaling



- ❖ **Local and Remote Journaling**
- ❖ **Synchronous and Asynchronous**
- ❖ **Multiple Object types**
 - ★ Database, IFS, Data Areas, Data Queues
- ❖ **OS/400 Optional Feature 42**
 - ★ Batch Journal cache
 - ★ Journal Standby Mode
- ❖ **Journal Minimal data**

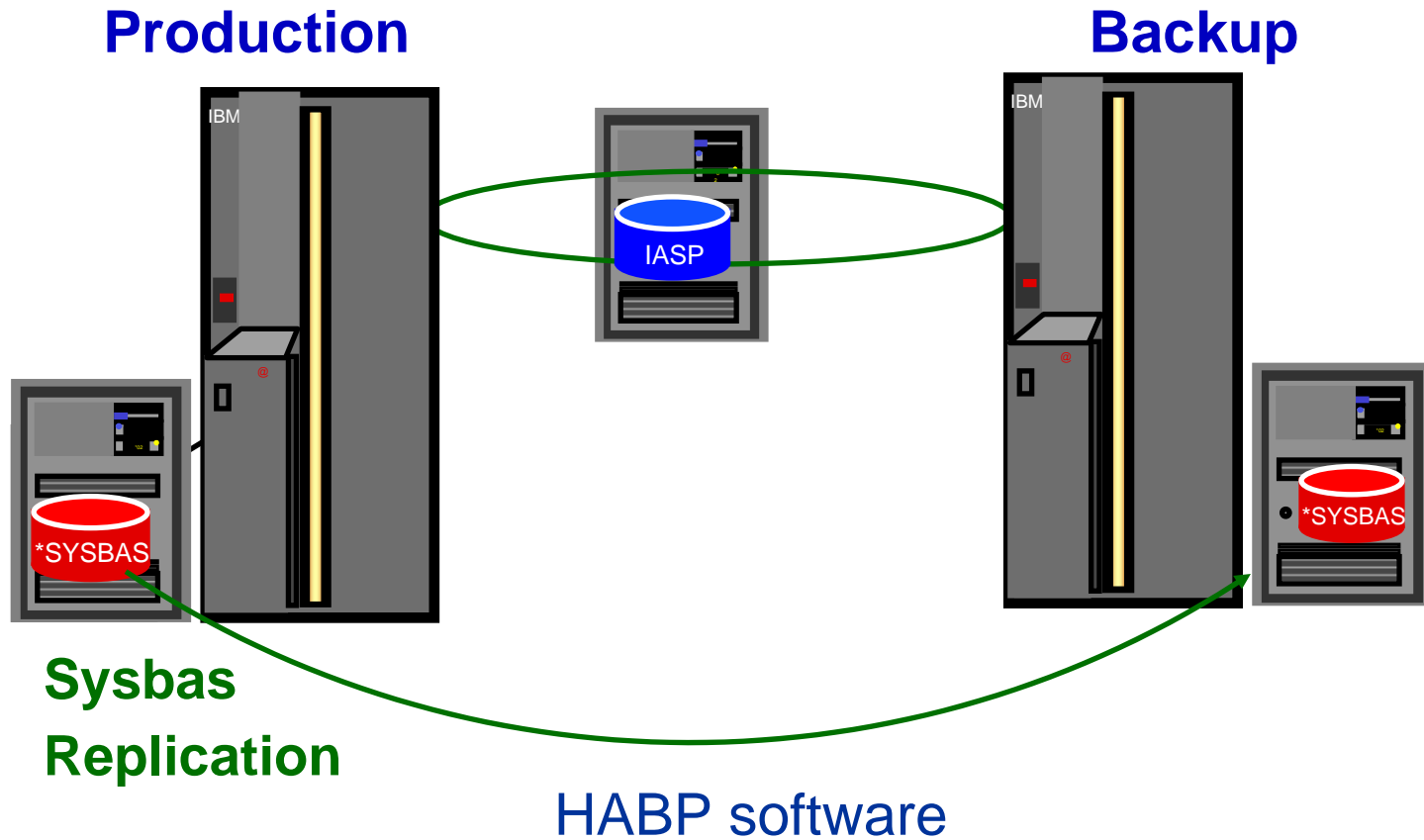
Data Replication Cluster



▪ Logical Replication

- f* High Availability Business Partner software (HABP)
- f* Second copy of data is generated logically identical to the first
- f* Replication is done on object basis near real time
- f* **backup copy available for query / saves**
- f* **Distance requirement supported**

Two Node Cluster



Two Node Cluster Logical Replication

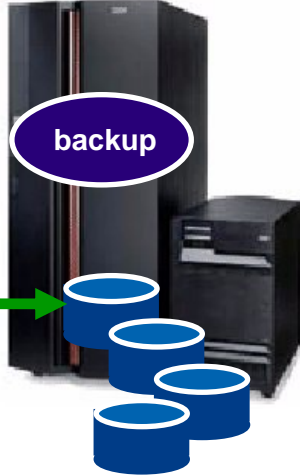
Active Production



Logical Replication

HABP software

Active Backup



Production Site

Remote

Logical Replication

- Data Integrity
- Application Integrity
- Environment Integrity
- Ease of use and manageability
- Speed and Ease of Role Swap

Logical Replication

Data Integrity : the ability to maintain synchronized data at any point between the source and its target destinations

- o Data on all systems must be equal at any point in time
- o Database synchronization must be validated continuously
- o Data serialization is essential

Why is it important ?

- o During a disaster data must be accurate and exact
- o When backups are performed on target system, data must be the same as the source system
- o When workload distribution is used all systems must ensure data consistency

Logical Replication

Application Integrity : the ability to maintain integrity at the application level ensuring the serialization of transactions and their dependencies on other objects.

- o Data integrity is assumed and dependencies between objects such as databases, data area and data queues are maintained
- o Support of referential integrity and triggers
- o Transactional integrity / Commitment control support

Why is important ?

- o Backups can be performed on the target system without downtime while keeping mirroring active
- o Continual protection of the application by dynamically transporting objects changes to the target system

Logical Replication

Environment Integrity : the ability to maintain integrity at the system level thus ensuring that any changes that occur to the source environment will be replicated to the target

- o Application integrity
- o All additional objects types such as devices, system configurations, user profiles, passwords, etc...
- o Security and ownership must also be maintained

Why is important ?

Guarantees that the environment will be in an operational state at disaster time

Enables workload balancing

Enables centralized management of environment

Logical Replication

Solution Ease of Use & Manageability

- o Menu driven operations or graphical navigator
- o Easy and automated set-up
- o Automatic management of journals and its receivers
- o Built-in self-healing technique to ensure continuous protection

Why is it important ?

- o Reduced implementation costs
- o Manageability of the product should be intuitive and cost effective

Logical Replication

Speed and Ease of Role Swap

- o Role Swap should be integrated and totally automated
- o Role Swap from production to target and, after recovery, from target to source

Why is it important ?

- o Role Swap must be executed effectively through an automated process
- o Returning to a normal state does not require reloading of the system
- o The speed of switch time reduces down time
- o Role swaps through an automated process will reduce customer risk and exposure

Common 13 december 2006

**The only disaster that
strikes,**

**Is the one you haven't
planned for !**