



OpenQM

Data Resilience in a QM System

Martin Phillips
Ladybridge Systems Ltd

Data Resilience

Protection against loss of data...

...not theft of data.

What We Are Going To See Today....

Causes of data loss

Simple steps to protect your data

Data replication

Journalling

Causes of Data Loss

**User
Error**

**Software
Error**

**Hardware
Failure**

**Environmental
Issues**

Causes of Data Loss - Alternative View!

**User
Fault**

**Software
Fault**

**Hardware
Fault**

**San Andreas
Fault**

Simple Steps To Protect Your Data

Set appropriate permissions

Use QM user security

Backup policies

Deferred resize

Directory file update protection

Set Appropriate Permissions

Permissions are mostly used to prevent unauthorised viewing of data.

They can also be used to avoid unauthorised overwriting or deletion of data by users.

Use QM User Security

Security subroutines can restrict who can execute specific commands.

Use account restrictions to keep users out of areas they should not see.

Use ADMIN.SERVER to restrict which QMNet server users can access.

Backup Policies

Probably everyone in this room today knows how to do reliable backups.

If you are a reseller, it is your end user who may get it wrong.

Unfortunately, it is your phone that will ring when they lose their data.

Backup Policies

How often do you need to backup?

How will this fit into your business routine?

Do you need to backup everything?

Who will perform the backups?

Where do you keep the backups?

How long do you keep backups?

How quickly can you bring back lost data?

What plans do you have for total system loss?

Deferred Resize

Dynamic hashed files resize automatically to respond to changes in the stored data.

There is a risk that a file could be damaged by a system failure that doesn't flush data to disk.

Suspending resize significantly reduces this risk but will cause overflow to be created.

Deferring resize until after daily backup provides a good solution.

This is much faster than static file resize.

Directory File Update Protection

Overwriting a directory file record could run out of disk space, losing data.

The SAFEDIR configuration parameter writes the data with a temporary name, deletes the original, and renames the temporary item.

Data Replication

Replicates updates from selected files to copies on other systems.

The publisher system exports data to one or more subscriber systems.

QM imposes few restrictions, leaving application designers much freedom.

A subscriber can publish the file onwards (cascade replication).

A file can be published in both directions (symmetrical replication).

Data Replication

Uses...

Standby systems

Reporting servers

Merging remote data at a central point

Distributing central data to remote systems

Data Replication

PUBLISH *filename server:account:file ...*

PUBLISH.ACCOUNT *server:account*

SUBSCRIBE *server username password*

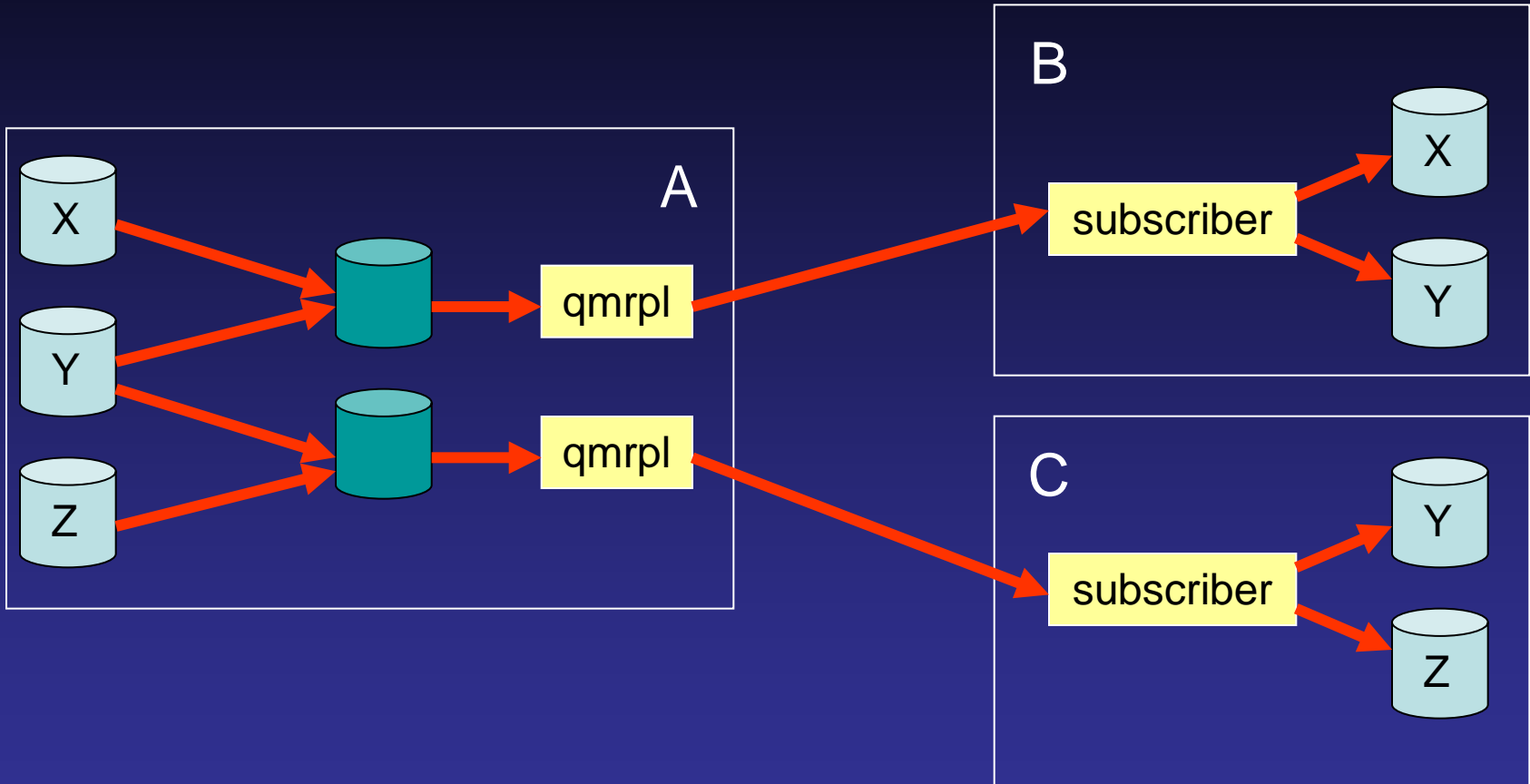
Data Replication

QM can publish directory files.

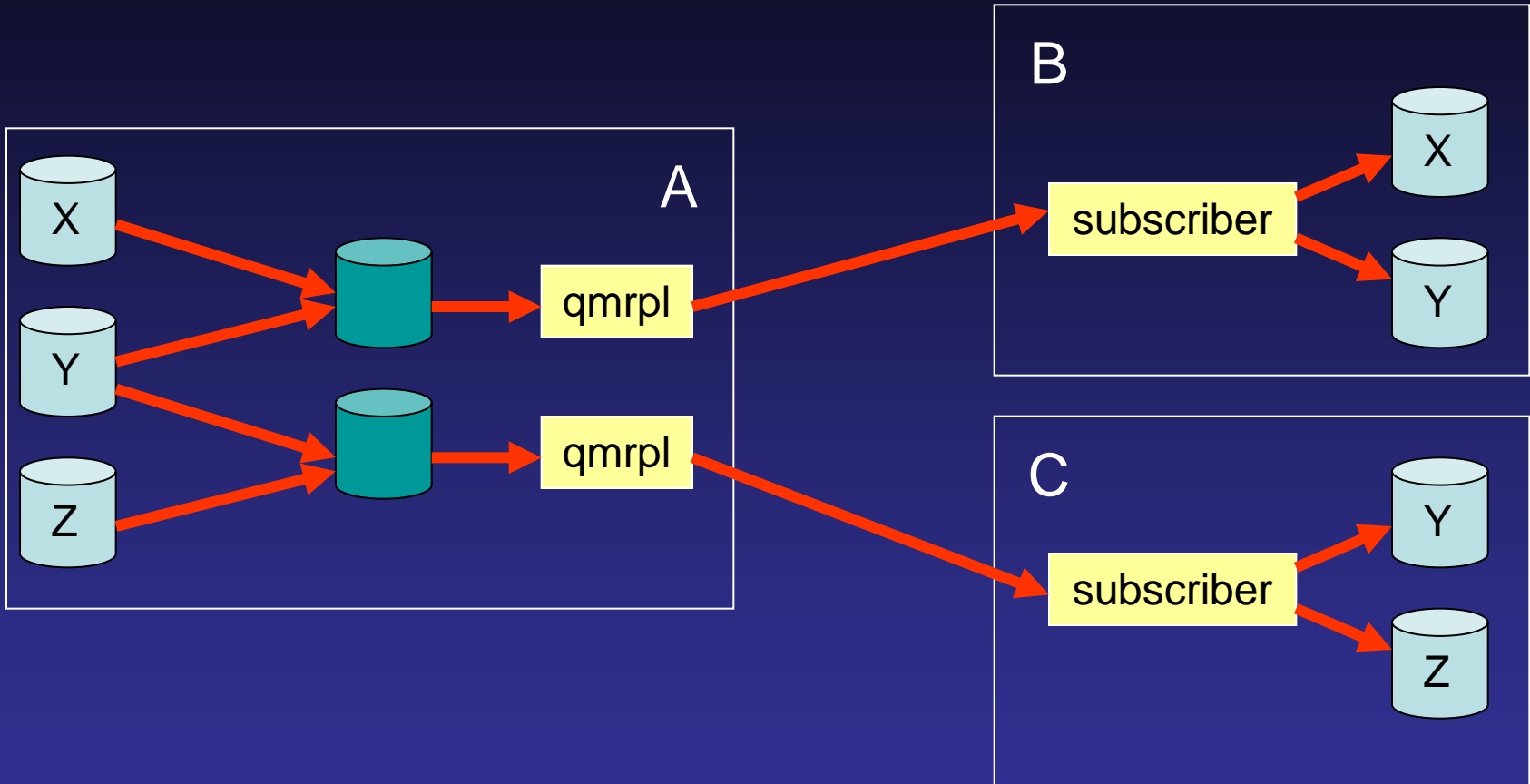
Only updates made from within QM will be exported.

Replication can be used with encryption and indices.

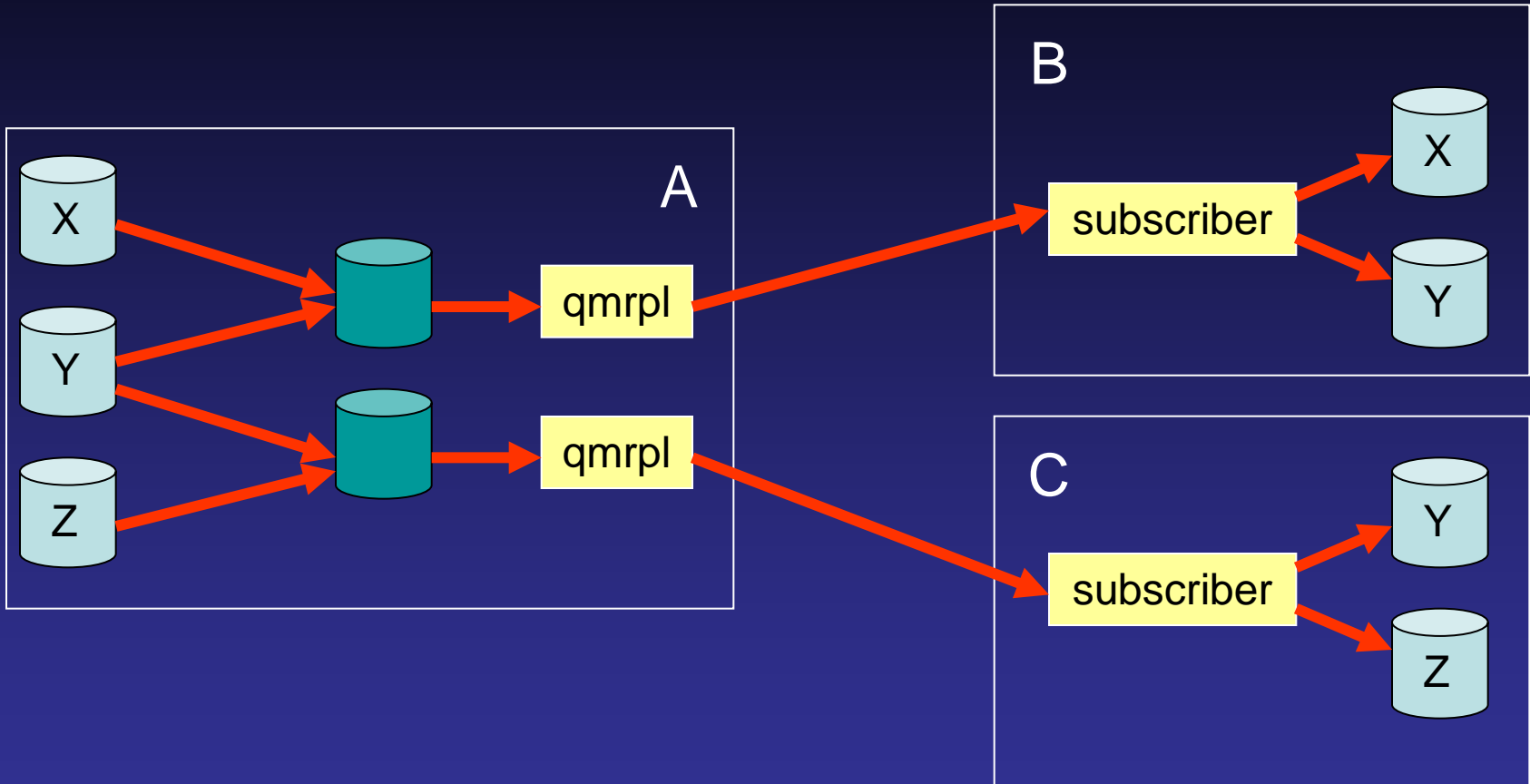
Data Replication



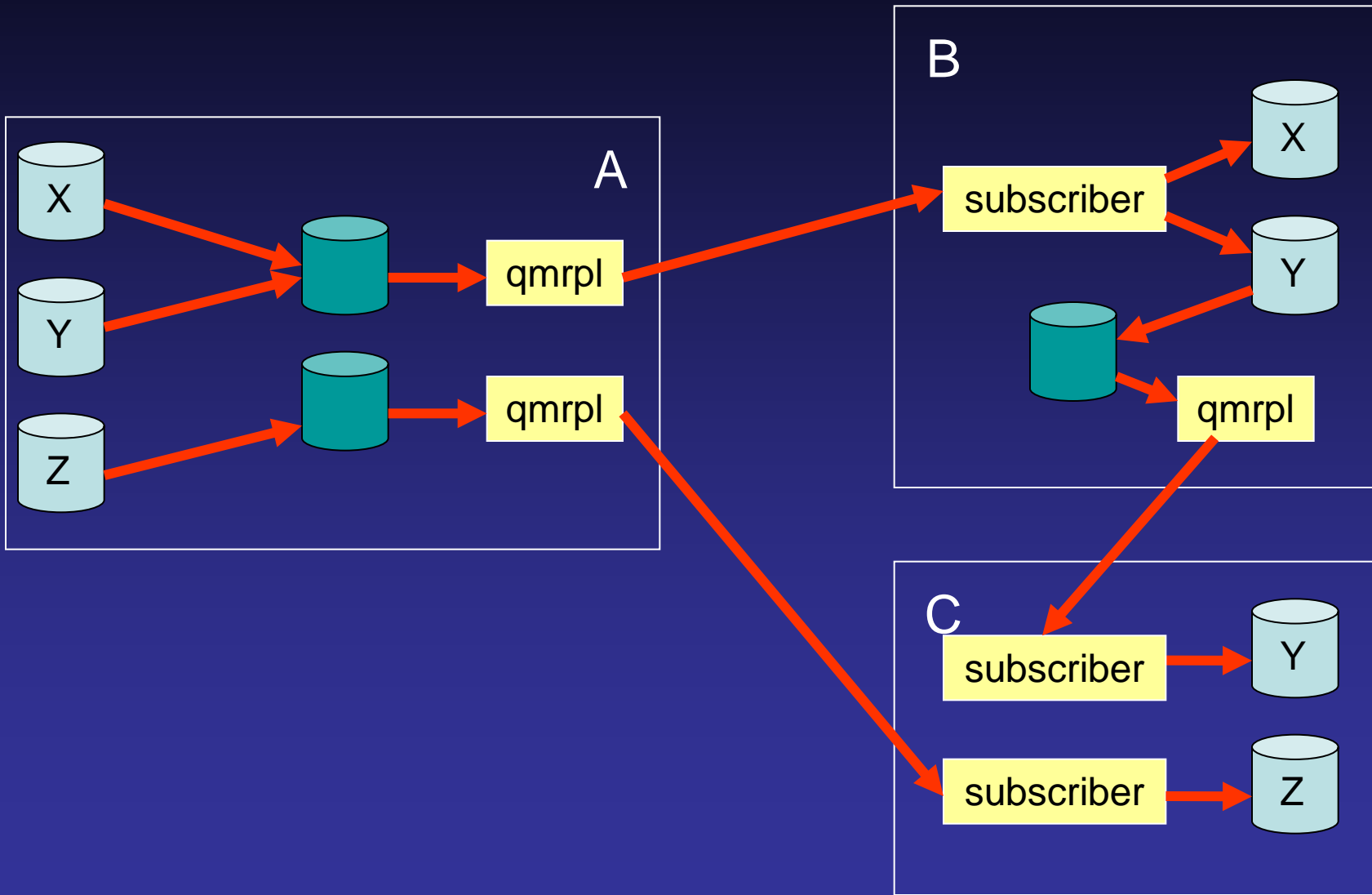
Data Replication



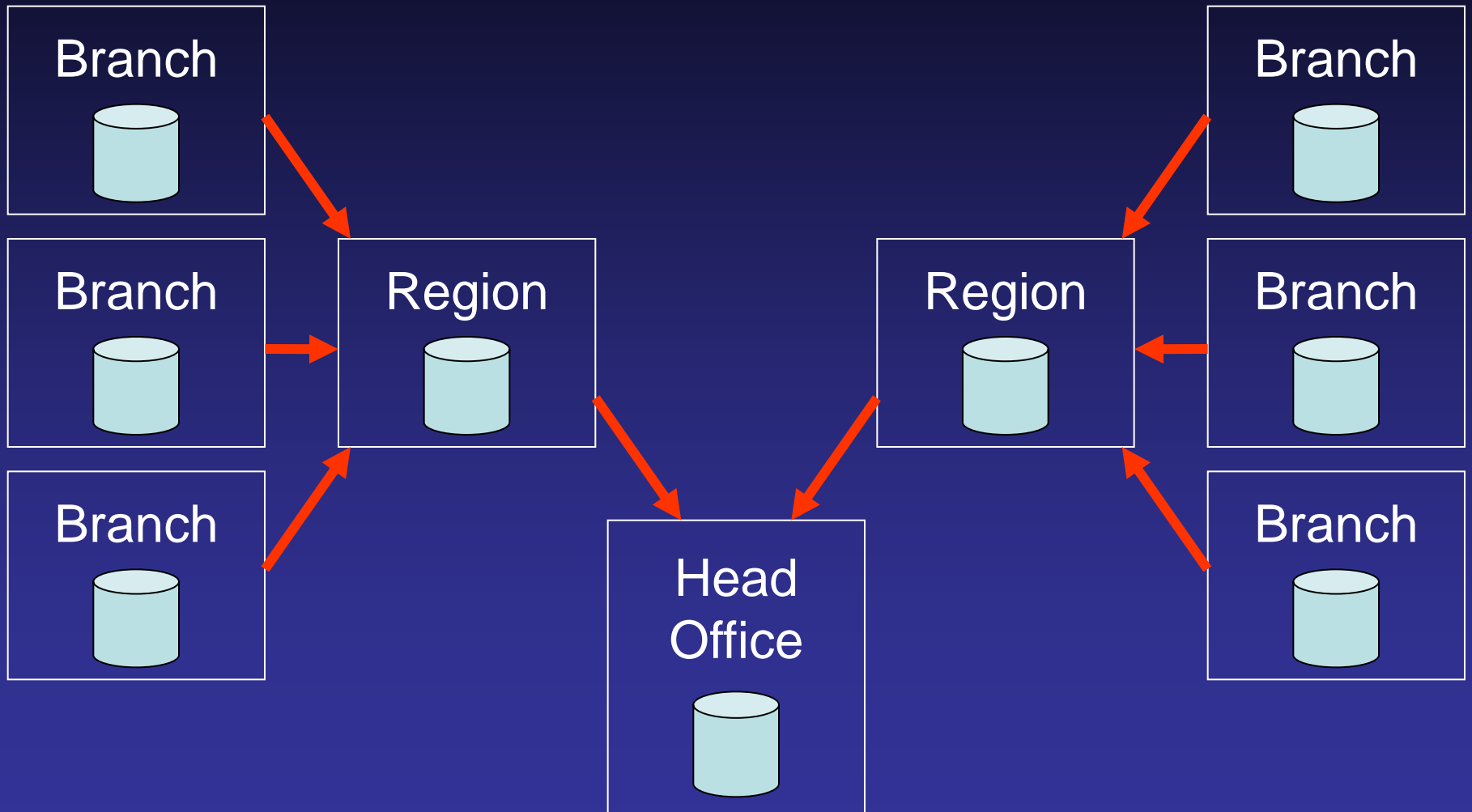
Data Replication



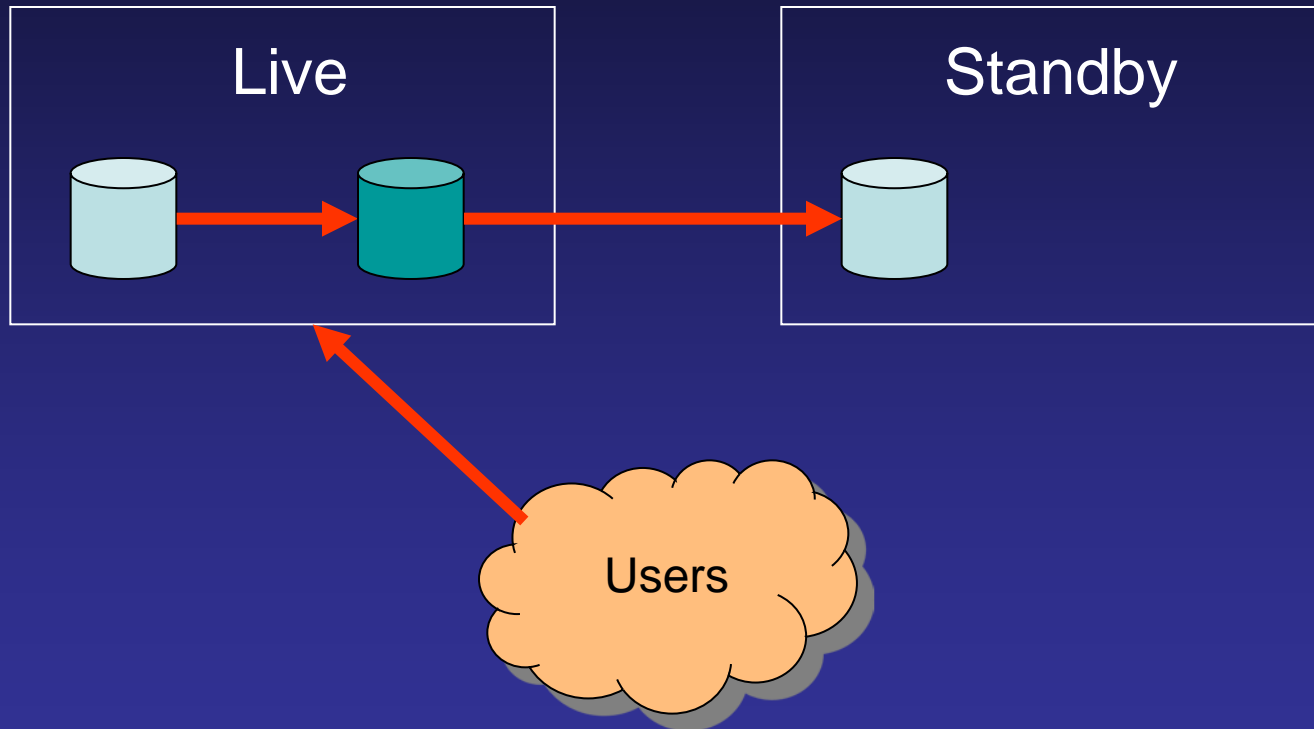
Data Replication - Cascade



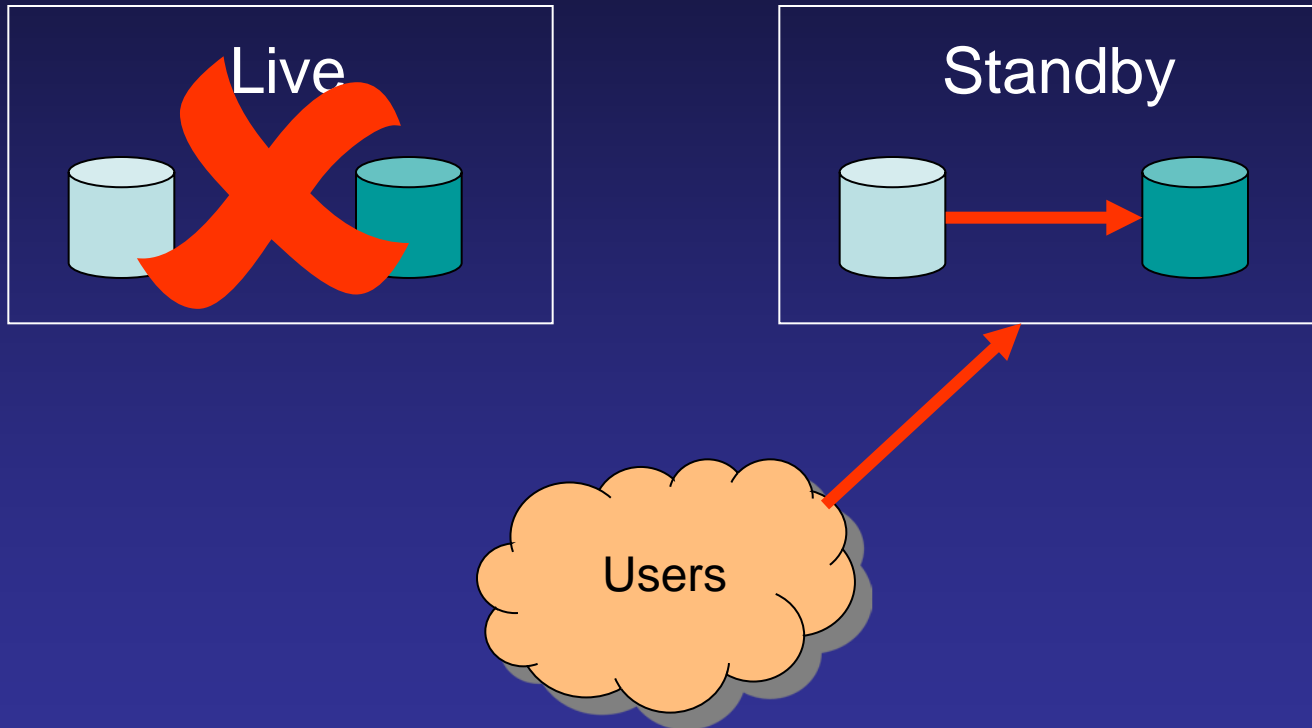
Data Replication - Cascade



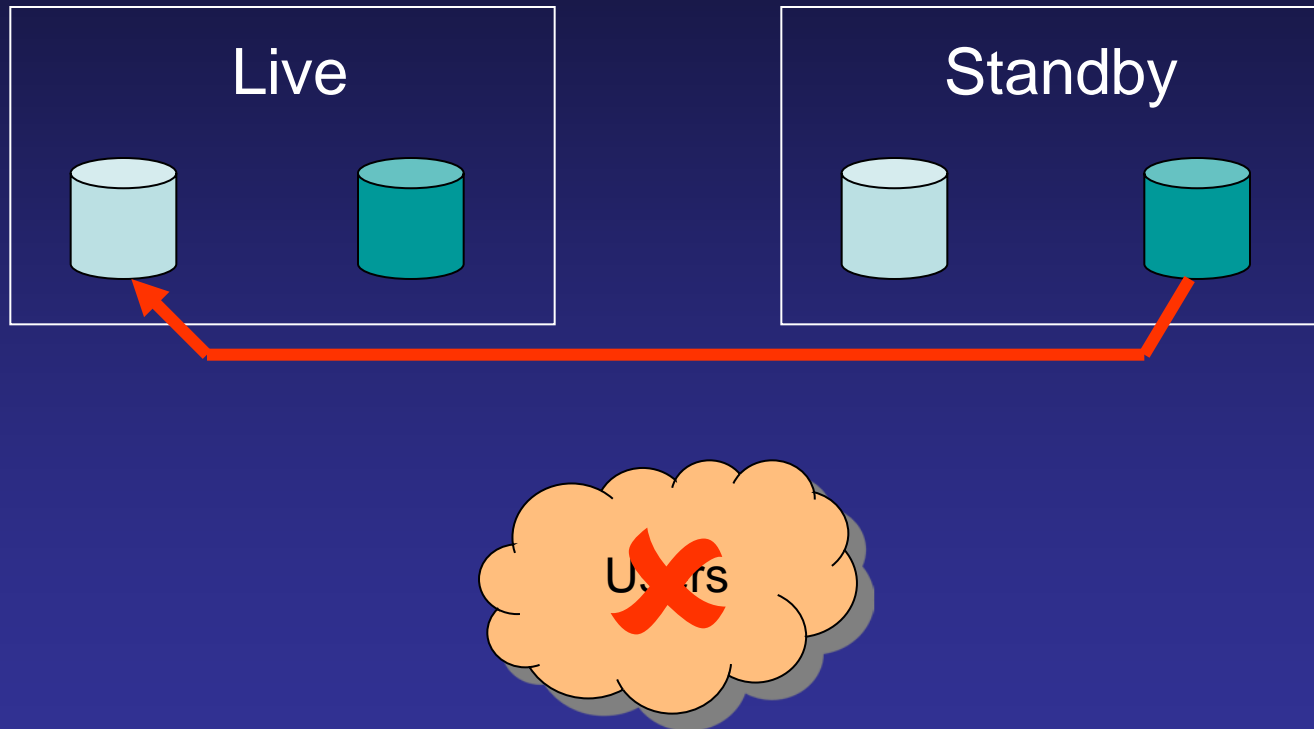
Data Replication – Standby Systems



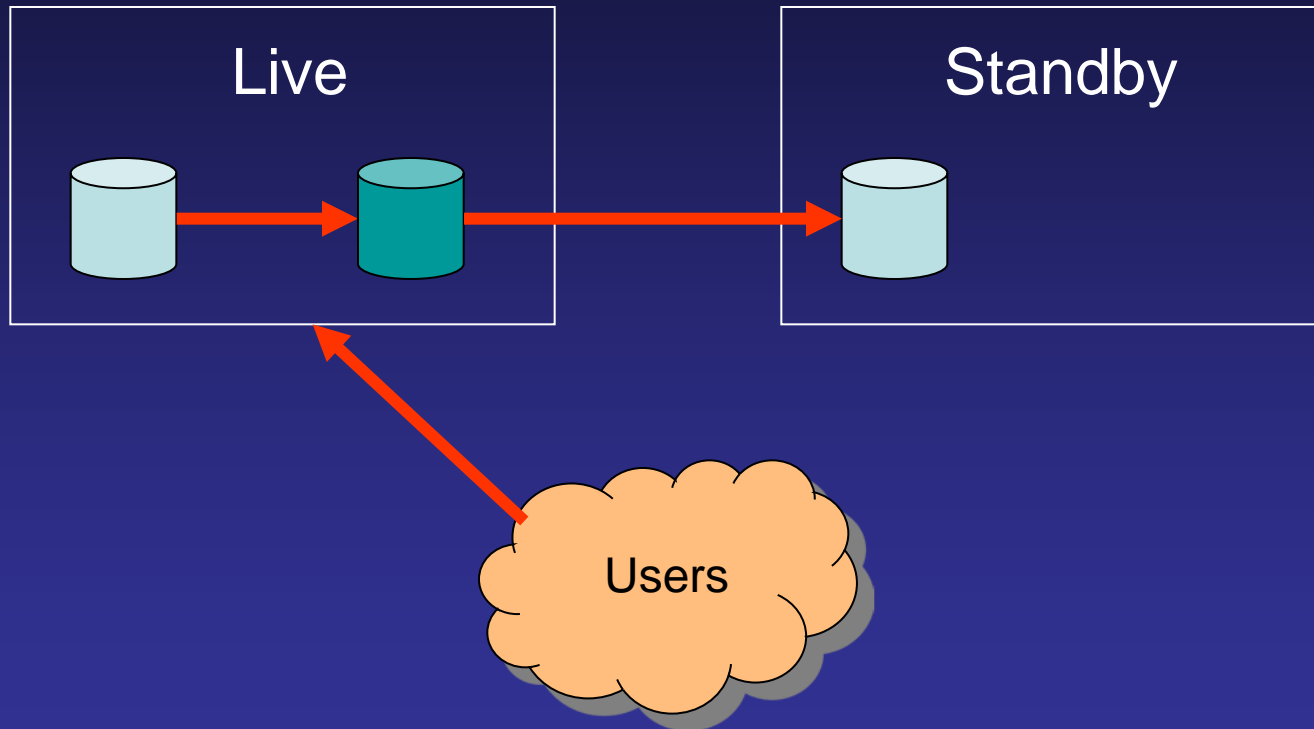
Data Replication – Standby Systems



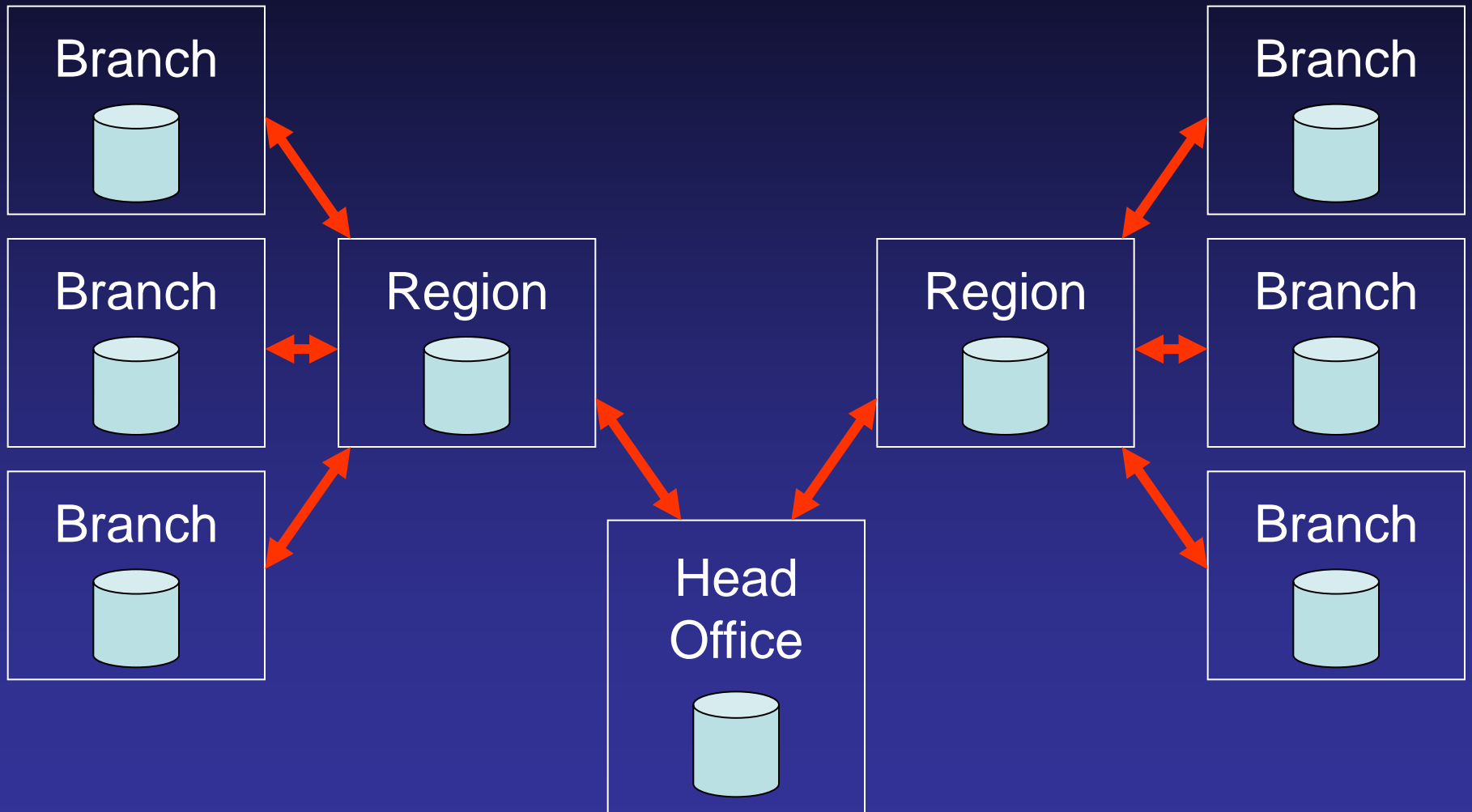
Data Replication – Standby Systems



Data Replication – Standby Systems



Data Replication - Symmetrical



Configuring Replication

Replication is part of the Resilience Features option to QM.

It is a separately licensed option on the publisher.

Dual system standby licences get replication on both systems for one option payment.

Configuring Replication

REPLDIR=*path*

Sets the pathname of the replication directory.

Each replication target server (subscriber) has a further subdirectory under this location to hold the log files.

SRVR\$ Control file

SRVR\$1 (etc) Log file

Configuring Replication

`REPLMAX=n`

Sets the maximum number of concurrent replication target servers (subscribers).

Configuring Replication

`REPLPORT=n`

Sets the port number on which replication subscribers will connect to start the qmrpl process.

Defaults to 4244 if omitted.

Configuring Replication

`REPLSIZE=n`

Sets the approximate maximum size in Mb of a replication log file. A new file will be started when this size is reached.

Defaults to 10 if omitted.

Log files are automatically deleted when no longer needed.

Account Replication

Selected commands that affect the file system can automatically be replicated on subscriber systems.

Create / delete file

Create / build index

Catalogue program

Add / remove distributed file

Create a Q-pointer (SET.FILE)

Journalling

Records a log of all updates to selected files.

QM supports roll forward from backup and roll back from current state.

Can recover individual files or entire system.

Can recover to a specific point in time.

Currently available only under a beta test agreement.



OpenQM

QUESTIONS?



OpenQM