Designing a bullet-proof Disaster Recovery Architecture for business-critical Applications

While real disaster incidents are rare, the impacts are dramatic. What if your business-critical Application becomes unavailable? How much data would you lose? How much downtime can you afford? Who would take responsibility?

A functioning Disaster Recovery Architecture will make or break the survival of a business in a worst case scenario. This White Paper is aiding companies to design a practical and comprehensive architecture.

We will outline essential project goals such as Recovery Point and Recovery Time objectives including ROI considerations and compare the pros and cons of three mainstream strategies. We close the 13 page paper with a modern-day approach to data mirroring which makes your architecture bullet-proof.

Libelle AG. Authors: Lars Albrecht, Bernd Baier
Contents

Why is Disaster Recovery still an Issue after all? ...............................................................3
Scope of a Disaster Recovery Project ..................................................................................3
Disaster Recovery Project Goals ..........................................................................................4
  Recovery Time and Recovery Point Objectives .........................................................4
  Transactional Integrity of Application Data .................................................................4
  Proof-of-Concept Objectives .........................................................................................5
  Return on Investment Goals .........................................................................................6
  Simplicity Goals .............................................................................................................6
  Bandwidth Requirements ..............................................................................................6
Three popular DR Architectures for Applications ...............................................................6
  Shipping Tapes to an off-site Location ........................................................................6
  Block-Based Storage Replication ................................................................................7
  Standby Databases .......................................................................................................8
Using Business Shadow® for Disaster Recovery .................................................................9
  Setting up a DR System with Business Shadow® .......................................................10
  Normal Mirroring Operation .........................................................................................11
  Switching to the Disaster Recovery Site .....................................................................11
The Business Shadow® Advantage ..................................................................................12
Vendor Credentials and Live Demo Invite .......................................................................13

Illustrations

Illustration 1: Recovery Point and Time Objectives .........................................................4
Illustration 2: Transactional Integrity of Application Data ..............................................5
Illustration 3: Pros and Cons of Tape Restore .................................................................6
Illustration 4: How distance reduces ROI when using block-based replication ..........7
Illustration 5: Pros and Cons of Storage Replication .....................................................8
Illustration 6: Pros and Cons of Standby Databases .......................................................9
Illustration 7: Components of Business Shadow® ..........................................................10
Illustration 8: Screenshot of Business Shadow® mirroring Landscapes .......................12
Why is Disaster Recovery still an Issue after all?

Global dependencies on local application systems, the Sarbanes Oxley Act and threats due to natural disasters, power outages or acts of terrorism are the main market drivers for today’s DR Projects.

Business needs and legal requirements are clearly demanding a functioning Disaster Recovery concept. However, many companies still don’t have adequate mechanisms in place. A non-representative sample of 300 people attending Gartner Inc.’s Storage Show affirmed that 44% of attendees expressed that DR was still a project. Only two out of 300 said that they perform change control testing as part of DR1. This says a lot about the current state of Disaster Recovery Projects.

Many companies seem to struggle in one way or another with DR. Even companies who do have architectures in place often lack proper testing. In other words: almost everybody is doing some kind of off-site backup or replication, but hardly anybody is having adequate testing procedures in place. In addition to presumably high costs and the time spent for planning and implementing a solution, the project won’t seem to deliver short-term benefits for the organization.

There are, however, a small but growing number of companies and project managers who show a strong commitment when working towards their DR goals. The priorities in which DR projects are implemented are indicating how companies take responsibility towards their customers, their investors and their employees. A functioning Disaster Recovery Architecture will make or break the survival of the business in a worst-case-scenario.

Scope of a Disaster Recovery Project

We need to keep in mind that DR projects cannot be seen from a pure technical standpoint. It is the company’s business which is kept up and running after the disaster incident. Even if done right, providing for example a disk-mirrored SAN infrastructure is only one aspect of the project. Aside from additional technical details (Application system running on a different server, different license keys, DNS failover, application interdependencies …) there are crucial business questions to be answered:

Who is in charge for DR? What constitutes a disaster? Who declares a disaster? Who updates the written DR Plan? Who is responsible for DR change management? How often per year are we conducting a DR test? What determines a “successful” DR test?

The contents of these organizational aspects are as important as the technical infrastructure, but are not further detailed in this White Paper.

---

1 Source: searchcio.techtarget.com/originalContent/0,289142,sid19_gci1195048,00.html
Disaster Recovery Project Goals

Recovery Time and Recovery Point Objectives

Any DR project starts with the question of the potential data loss in case of a disaster (Recovery Point Objective or “RPO”) and the time necessary to have the systems up and running on the DR server (Recovery Time Objective or “RTO”).

Illustration 1: Recovery Point and Time Objectives

RPO = maximum acceptable data loss in case of a disaster
RTO = maximum acceptable time until application is up on the DR system

The target RPO and RTO values need to be determined by the business and weighed against the Return of Investment of the DR project. With an unlimited budget, companies would like to see RTO and RPO values of zero. Technical and budgetary reasons are however speaking against that. Is it reasonable to invest 5m$ to reduce the RPO from 10 minutes to 1 minute with a disaster probability of 0.5% per given year? The following gives an illustration on calculating a Return on Investment with RPO and RTO in mind:

- Estimated disaster probability in a given years = 2%
- Estimated cost for data loss and downtime per incident = $15m
- Probability × cost per incident = $300,000
- Average cost of DR solution = $80,000 per year

→ Cost of Avoidance Savings = $220,000 per year

Transactional Integrity of Application Data

Another goal of Disaster Recovery projects is to have the business data in the correct state upon switch-over. The term “transactional integrity” refers to the overall integrity of transactions from the point data is entered into an application down to the lowest granularity of a data block on a disk drive. The question of transactional integrity becomes an issue especially when replicating data on a storage-level (“block-level replication”) which is a popular method for local disaster protection for many customers.

By definition, the further away we get from the initial transaction within the production system, the lower the transactional integrity. A confirmed storage-block transaction has
only little to do with a confirmed application or database transaction. Storage blocks make up files, files make up databases, and databases contain data. While differences between every stage should be non-existent, they will show up in the big picture and particularly when mirroring data on a storage level versus mirroring data on a database level as DR architecture.

As users enter data into the application, transactions move through the application system and the database server hands the transactions over to the database. The database makes sure that the transactions are logged properly for backup/recovery purposes and then hands them over to the file system. The file system then makes sure that the respective log files and data files are stored properly and can be accessed through the Operating System. Finally, the operating system hands the files to an internal or external storage controller which makes sure that the files are stored properly and can be accessed through storage blocks, the finest granularity in the whole process.

Transactional integrity from the application perspective is critical for Disaster Recovery because it determines in which state we will find our DR system after the production server becomes unavailable. It is critical because it determines if and when we get the application system up and running on the DR system after a disaster incident.

**Proof-of-Concept Objectives**

Any backup is only as good as the probability that the restore actually works. The same applies to DR. It would be a nightmare to invest hundreds of thousands of dollars in the project, only to learn that the application won’t start on the DR system in case of a disaster incident. It is imperative to insist that regular Disaster Recovery tests, which include the activation of the solution on the DR system, are being held.

Well-tested and documented procedure with a good change management control using manual tape restore (low RPO, low RTO) is often of a higher overall value than a block-based replicated architecture (high RPO, presumably high RTO) which was never tested but “should” work.
Return on Investment Goals

Any investment needs to bring returns for the business. To calculate the ROI exactly or to at least have an approximate calculation model, Disaster Recovery investments are typically compared to life insurance policies for a business. For calculating how much investment is justified, we determine how much coverage we need and how much we want to spend. The ROI calculation is then based on the cost of downtime for example on a per-day or on a per-hour basis and the actual risk of a disaster incident.

Simplicity Goals

Even small or medium sized application systems are complex. It takes only one critical component to fail for the whole system to malfunction. Redundancy of components will help a lot to cover outages of single components, but it is the complexity itself which often becomes a threat to the system. Additional components for Disaster Recovery could make systems even more complex and vulnerable. A DR solution should be as simplistic as possible while providing the most comprehensive protection.

Bandwidth Requirements

The question of how much network bandwidth is reasonable, is another factor influencing the Disaster Recovery project. It is a huge difference cost-wise if we can for example mirror a 400 GB system using an 8 MBit/s WAN link when setting up a standby database versus having to invest in a 100 MBit/s WAN link for a block-based storage replication, especially since these are recurring monthly costs and will exponentially increase with the distance between the production and the DR system.

Three popular DR Architectures for Applications

Most DR concepts fall into only three different categories:

1. Shipping Backup Tapes to an off-site Disaster Recovery Location
2. Block-based replication by storage hardware or other means
3. Setting up standby databases with native database tools

Shipping Tapes to an off-site Location

Disaster Recovery with traditional tape backups is the strategy which most companies already perform as part of their backup procedures. A tape backup is performed to create a full copy of the database and files in regular intervals. Those tapes are then shipped to the designated DR location.

<table>
<thead>
<tr>
<th>Manual Tape Restore for Disaster Recovery</th>
<th>Illustration 3: Pros and Cons of Tape Restore</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pros</td>
<td>Cons</td>
</tr>
<tr>
<td>• High transactional integrity</td>
<td>• High data loss in case of a disaster</td>
</tr>
<tr>
<td>• Additional protection against user and</td>
<td>• Immense time required to restore data at off-site</td>
</tr>
<tr>
<td>software errors (Point-in-time Recovery)</td>
<td>location</td>
</tr>
<tr>
<td>• Least expensive</td>
<td>• Manual procedures are time-intensive and bare</td>
</tr>
<tr>
<td></td>
<td>the risks of human error</td>
</tr>
</tbody>
</table>
The advantages of this strategy are the minimal investment and no changes in technology since backup to tape is done anyway. The disadvantage is the very low RPO and RTO values: we would lose a lot of data and it typically requires days to be back up and running. Further, a lot of manual interventions are necessary for the restore. Many DR tests showed that no systems’ restore ever went flawless despite the fact that all involved parties are warned and prepared well ahead of an upcoming test.

**Block-Based Storage Replication**

Another popular DR architecture is to implement block-based storage replication. In this method, a second storage system is provided and every bit of data is synchronized to a remote location directly between two storage systems. Hardware and SAN providers typically promote this concept. It is tempting: every single bit of data is copied immediately to the Disaster Recovery site. If something goes wrong, we can simply flip the switch and continue to work with the solution on the DR site.

There are of course stipulations attached to it. The multiple dedicated Fiber or ESCON lines necessary to keep the mirror in sync at any given time and proprietary hardware components will add enormous complexity to the systems. This will require a high initial and ongoing investment in bandwidth cost, maintenance and other resources.

The biggest problem however is the distance since it is necessary to replicate every single bit of any transactions at any time. With larger distances, network traffic is largely consisting of overhead due to network latency. The larger the distance, the exponentially severe problems with network latency will get. The latest improvements in storage technology will only make a marginal difference.

Finally, replicating every block on a storage system says nothing about data consistency of transactions from an application perspective. Such a replicated mirror can be consistent from a storage perspective (block-level consistency) and at the same time inconsistent from the application perspective (application-level consistency). Typical methods used to restore the application and database consistency are a database crash-recovery and files system checks.
To cover the disadvantages, we see customers making compromises on their original distance goals so that the Disaster Recovery Server is only 5, 20 or sometimes 30 miles away from the production site in order to choose the block-based replication concept.

**Standby Databases**

Using standby databases for Disaster Recovery is the manual or half-automated process of setting up a standby database at a remote location by using scripts and/or tools which are part of the database.

The popular databases such as DB2, Oracle, Microsoft SQL Server and MaxDB are providing more or less sophisticated tools enabling customers to build a basic standby database. The architecture is typically set up by assigning the project to an internal DBA or to an external consulting firm. In this scenario, database log files are continuously transferred to the DR site either with scripts or with automated tools. A mirror database would continuously receive database transactions which are then applied into a standby database.
The advantage of this method is that there seem to be only little financial investment required since scripts can be written in-house and the tools come typically free with the database. The method further provides a high level of transactional integrity especially compared to a block-based replication. Another advantage is that there are no distance limitations between the production and the disaster recovery location.

One of the numerous disadvantages of standby databases is that only the database can be mirrored. Certain flat files which are important for the application system are very often not stored in the database and would not be available after a disaster.

Another disadvantage is the lack of automation of standby tools coming with the database. The architecture depends heavily on the knowledge and availability of the database administrator. The staff dependency is not only an issue when setting up the DR mirror initially, but in particular when activating the mirror in case of a disaster incident.

### Using BusinessShadow® for Disaster Recovery

BusinessShadow is seen by many as “the middle road” between block-based replication and standby databases. Compared to block-based replication, BusinessShadow doesn’t have any of the typical downsides such as lack of transactional integrity, overcoming large distances, excessive complexity or costs. Compared to standby databases it accommodates multi-database support, flat file mirroring, additional features and full automation of mirroring complete system landscapes.

The core DBShadow® component of BusinessShadow serves as a physical standby database while covering the missing features and the lack of automation of native standby tools. Other components will take care of the network optimization, the automation of the switch-over process and the handling of the flat files. Hundreds of
customers worldwide are using BusinessShadow® for many years ranging from medium-sized to large enterprises mirroring complete landscapes with multiple terabytes.

**BusinessShadow** consists of four different components:

1. **DBShadow®** to mirror the core databases which can be Oracle, MaxDB, DB2 and MS SQL Server (or a combination of different databases)
2. **FSShadow®** to mirror flat files belonging to the application systems
3. **SwitchApplication** to switch IP addresses and hostnames to the DR server
4. Option Long Distance to optimize WAN traffic, cover network latency, compress files and ship IP packets parallel to the DR site.

**Setting up a DR System with BusinessShadow**

The only requirement to set up and operate BusinessShadow is a second server with the same operating system, the same disk space and a standard network connection. From a performance perspective, the mirror server requires enough performance to update changes during normal operation and adequate performance in case of an emergency to run the application.

Our customers often use their test or backup systems as a DR server and temporarily shut down tests during the disaster incident.
Step 1: Initial Load to the DR Site

Upon installation, we start with preparing the DR mirror by making a copy of the production system which is done during normal operation without interfering with production.

Step 2: Archiver Process continuously ships changes to the DR Site

In the next step, all committed transactions on the production system are continuously copied to the mirror server. Customers can specify the maximum archive switch time to define their Recovery Point Objective, for example every 5 or 10 minutes. Interruption due to temporary network, database, application or server outages on either site don’t require manual intervention. Enhanced WAN optimization comes with our Option Long Distance and is covering network latency and bandwidth issues.

BusinessShadow® is “the middle road” between block-based replication and standby databases. Compared to block-based replication, it doesn’t have any of the typical downsides such as lack of transactional integrity, overcoming large distances, excessive complexity or costs. Compared to standby databases it accommodates multi-database support, flat file mirroring, additional features and full automation of mirroring complete system landscapes.

Step 3: Recover Process manages the DR Server

In the next step we automatically apply the changes from the primary system to the secondary system which holds the copy of the database in loading mode. It is advantageous to wait a couple of hours before applying the changes to protect the mirror system against faulty transactions. If for example a batch job or malicious manipulation destroys the production database, the mirror database would still be valid and can be recovered to any desired time stamp before the point of destruction within minutes. Just as the initial copy, the recover process is fully automated and does not require any manual intervention or special database know-how. The mirror database is kept in a loading mode and changes are continuously applied. All mirroring parameters can be changed dynamically using the GUI or with a Command Line Interface.

Normal Mirroring Operation

Once the Initial Copy is performed the mirror activated, we are in “normal operating mode” for Disaster Recovery. Changes are now continuously shipped to the DR server by the Archiver Process. The Recovery Process is either continuously applying those changes. The DR system is ready to take over with current data at any time. Please refer to Illustration 8 for a BusinessShadow GUI screenshot with mirroring a landscape of Oracle, MaxDB and UNIX flat files.

Switching to the Disaster Recovery Site

The only step necessary in case of a disaster is to instruct BusinessShadow to start up the mirror system, restart the application, eventually switch the IP address and we are up and running on our DR site. The switch-over process is handled smoothly and automated after the process is initialized by the customer: remaining changes are recovered; the database will be renamed to the correct name before it gets restarted.
The **FSShadow®** component covers flat files and **SwitchApplication** handles the IP addresses and hostname failover.

**The BusinessShadow® Advantage**

In comparison to native standby database tools for Disaster Recovery we offer specific value in the following areas:

- One solution for heterogeneous environments including, DB2, MaxDB, MS SQL Server, Oracle and available on all UNIX and Windows platforms.
- Solution provides **multi-platform support** for heterogeneous systems on multiple database technologies such as an ECC system based on Oracle combined with an SAP APO/SAP liveCache based on MaxDB.
- **FSShadow/DBShadow®** combination support mirroring complete landscapes and not only a database.
- No specific database know-how is required to set up and maintain the mirror and to switch the system in case of a disaster.
- A dedicated “structure process” reflects new data files, new directories and no-logging transactions on the secondary server.
- The possibility of dynamic changes of all parameters via GUI keeps everything easy to operate.
- **SwitchApplication** provides an automated switch of hostnames and IP addresses.

Illustration 8: Screenshot of BusinessShadow® mirroring UNIX flat files and Oracle and MaxDB
• The Option Long Distance provides massive network optimization including parallel archive shipping, high compression and the “Very Large Package Technology” to cover network latency for distances > 50 miles.
• Automated switch-over interfaces for “Point-in-Time Recovery”, “Lossless Switching”, “Defined-Switching” and automated database rename.
• Radical simplicity of the overall approach with independent server processes will make the DR architecture easy to operate on a day-to-day basis.

Vendor Credentials

Libelle is an established solution provider for high availability and disaster recovery located in Stuttgart, Germany. A strong team of experts provides years of experience and competences with a proven track record.

Customers worldwide trust in the intelligent Libelle software solutions to protect and recover their business-critical data. Libelle customers benefit from excellent one-stop service as support, sales, and software development are all located in Stuttgart.

The solutions are supplemented by a great service portfolio, ranging from the design and implementation of holistic HA and DR concepts to 24/7 support. Further, Libelle works in close partnership with all relevant hardware and software vendors and is certified per DIN EN ISO 9001:2008.