Using cloud technologies to improve disaster recovery

Microsoft IT has always had a comprehensive disaster recovery (DR) program protecting its critical on-premises applications. However, coordinating all the existing plans was complex because of dependencies and varied platform architecture that had been developed over the years. Data sovereignty and changing regulatory compliance obligations were affecting agility.

Then there was the problem of configuration “drift.” We found that the configurations at two different sites would slowly accumulate configuration differences. For example, the SQL structure and access permissions between the primary and failover sites would diverge. This made maintaining replication between sites difficult and made failover less predictable.

By using Microsoft System Center Data Protection Manager and SQL Data Sync to improve application resilience and performance in the cloud, Microsoft IT helps ensure that business applications are always available and data is protected in case of disaster.

As part of the “mobile-first, cloud-first” initiative, our strategy is to identify critical business processes to transform and improve business continuity and disaster recovery (BCDR) plans. It’s important to note that we pay the same costs, including Azure costs, as any other large IT organization. We evaluate the costs and benefits the same as any business, and we make cloud migration decisions based on business value.

Migrating to the cloud

Migrating applications to the cloud requires a lot of thought and planning. Decisions made early in the migration process often play a significant role later in the process. We took a highly planned approach to migrating to the cloud. This approach began with selecting the correct platform for an application. Whether the platform is IaaS, PaaS, or SaaS, we wanted to understand how it would affect the business continuity and backup strategy options for the application.

In addition to the IaaS, PaaS, and SaaS cloud platforms, we reserved the option of leaving an application on-premises, when appropriate. For information about our cloud adoption strategy, see Driving cloud adoption in an Enterprise IT organization. At each level, from traditional applications that are retained on-premises to applications completely developed for the SaaS cloud platform, the cloud provider is responsible for more layers:
Also, shared responsibilities changed, depending on the platform we selected:

*Figure 2: Shared responsibilities in the cloud*

**Which applications need a highly resilient cloud DR architecture?**

Our first step in designing cloud solutions did not begin with the application; it began with the business processes. We started by analyzing all business processes within the company. The business processes were evaluated using objective measures including financial loss, legal and regulatory issues, and customer impact. Using this objective data, we were able to identify critical applications that support those processes.

The business impact analysis also enabled our team to determine quantifiable DR goals for each business process. Underlying applications inherited the recovery time for the process. For example, if the process “Pay Employees” had a recovery of four hours before hitting a legal and regulatory control, the underlying human resources and finance applications inherited that requirement.

Recovery time objective (RTO) = How much downtime?
Recovery point objective (RPO) = How much data loss?

We had the technical requirements needed to plan a successful and appropriate cloud strategy for each application. The next step was to define a taxonomy for the failure scenarios to mitigate when designing cloud solutions. For this, we began by looking at how resiliency had been defined and designed for on-premises applications.
Failure types
Our DR plan focuses on services and technologies that can survive three different types of failures: component failures, horizontal failures, and vertical failures. We have used this taxonomy for many years to describe the failure scenarios for application resilience. An important lesson learned during our cloud migration was that although the technologies to handle the scenarios are different in the cloud, the three basic scenarios described below do not change.

Most IT strategies are based around the concept of “technology stacks.” Technology stacks are the layers that make up the datacenter environment. In a technology stack, the server operating system, server hardware, storage, and network layers are at the bottom of the stack. The data layer is at the top of the stack, followed by the application and database layers.

Typical on-premises IT installations build two redundant sets of computing resources at one datacenter and a mechanism to replicate the data to a second set of two stacks at a failover location.

![Diagram of data center resilience](image)

Figure 3. Resilience in on-premises servers
With such architecture, we can survive the following three disaster scenarios:

Component failures
A component failure occurs when any individual layer of a technology stack stops working. A common example is server hardware failure. If a server hardware component fails, an application will typically failover to the duplicate stack in the same datacenter. The second stack enables an application to survive with no data loss. Most companies have processes in place to handle this common scenario.
Figure 4. Resilience in on-premises servers—component failure
Vertical Failures
A vertical failure occurs when both the primary and secondary stacks at the primary datacenter fail. An example of a vertical failure is an earthquake that affects the primary datacenter. This type of failure has a high impact on business, but occurs infrequently. We found that most critical applications already had DR plans to address these catastrophic failures even though these failures rarely occur.

Figure 5. Resilience in on-premises servers—vertical failure

Horizontal Failures
A horizontal failure occurs when any one of the layers in the stack fails across all locations. Common causes are deploying bad application code at both the primary and secondary sites, accidental data deletion that replicates immediately to the remote site, viruses, and denial of service attacks.

Figure 6. Resilience in on-premises servers—horizontal failure
Often, IT departments are least prepared to address horizontal failures. However, most organizations actually experience horizontal failures more often than vertical failures. In most cases, the solution is to roll back to a known, good backup of the application, data, or operating system.

Because an online, current copy of the data already exists at the failover site, backups are only used to restore earlier data versions in our plan. A cloud backup strategy must include this functionality. We found that backups were no longer a viable solution for recovery of critical applications, because of the time required to restore the data. However, backups still play a vital role in a roll-back scenario and in recovering non-critical applications.

Cloud requirements for data

Based on our analysis of the on-premises architectures and our resiliency taxonomy listed above, recommended the cloud technologies that maintain equal or better recovery functionality in the cloud. We needed four copies of the data to duplicate the same level of resiliency.

The four copies are listed below, along with the resilience scenarios.

*Table 1. Four copies of data that will provide resiliency*

<table>
<thead>
<tr>
<th>Data copy</th>
<th>Protects against</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCAL COPY of the data, can survive</td>
<td>Component failures</td>
</tr>
<tr>
<td>component failures</td>
<td></td>
</tr>
<tr>
<td>REMOTE mountable COPY of the data at a</td>
<td>Vertical failure</td>
</tr>
<tr>
<td>remote site</td>
<td></td>
</tr>
<tr>
<td>LOCAL roll-back COPY of the data</td>
<td>Horizontal failure in which the data needs to be rolled back as quickly as</td>
</tr>
<tr>
<td></td>
<td>possible</td>
</tr>
<tr>
<td>REMOTE roll-back COPY of the data</td>
<td>Horizontal failure in which the data needs to be rolled back from a remote</td>
</tr>
<tr>
<td></td>
<td>location. This has much slower restoration time, but offers site resiliency.</td>
</tr>
</tbody>
</table>

Data storage technology solutions in Azure

The following table shows the final Azure solutions that we chose to recommend to Business Units as solutions that duplicate or exceed current on-premises solutions. In many cases, the cloud solutions proved to be superior and more flexible than had been possible before.

For example, the level of granularity in roll-back scenarios was clearly better in Azure. Overall, not only were the current service offerings superior, the high level of investment in future BCDR features by Azure offered a very convincing roadmap for the years to come.
Table 2. Azure solutions to meet or exceed past on-premises resilience scenarios

<table>
<thead>
<tr>
<th>Data Repositories</th>
<th>Failure Type</th>
<th>Component</th>
<th>Vertical</th>
<th>Horizontal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Copy</strong></td>
<td></td>
<td>LOCAL REDUNDANCY</td>
<td>REMOTE COPY</td>
<td>LOCAL/REMOTE roll-back COPY</td>
</tr>
<tr>
<td>Azure SQL Database</td>
<td>Non-critical apps</td>
<td>3 copies maintained by the platform locally</td>
<td>Provided for all SQL SKU by default</td>
<td>Restore points: per Transaction Retention: Basic and Standard SKU (14 days), Premium SKU (35 days)</td>
</tr>
<tr>
<td></td>
<td>Critical apps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Azure Storage</td>
<td>3 copies maintained by the platform locally</td>
<td>Customer must configure GRS or RA-GRS</td>
</tr>
<tr>
<td></td>
<td>Non-critical apps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Critical apps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Windows VM Disks* (IaaS)</td>
<td>Non-critical apps</td>
<td>3 copies maintained by the platform locally</td>
<td>Customer must use Azure Backup for consistency across multiple VM disks</td>
<td>Customer must use the Azure Backup Service (VM Extension)</td>
</tr>
<tr>
<td></td>
<td>Critical apps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQL Server VM (IaaS)</td>
<td></td>
<td>Customer responsible for configuring SQL SQL Server Always-On within the region</td>
<td>Customer responsible for configuring SQL Server Always-On</td>
<td>Customer can use standard SQL backup and restore and can configure the retention as required (same as on-premise)</td>
</tr>
<tr>
<td>Azure SQL Data Warehouse</td>
<td></td>
<td>3 copies maintained by the platform locally</td>
<td>RA-GRS</td>
<td>Snapshots (7 days)</td>
</tr>
</tbody>
</table>

*GRS=geo-redundant storage; RA-GRS=read access geo-redundant storage; VM=virtual machine; VHD=virtual hard disk

Benefits

A strategy that is easier and more cost-efficient – Moving to Azure let us not only use new cloud technologies, but also use the same platform architecture throughout.

Quantifiable goals for disaster recovery – Because we established measurable goals, we were able to use testing and report cards to measure the effectiveness of the program. This also led to improved reporting and scorecard.

Improved budgeting for disaster recovery – We worked with the individual teams to select the appropriate technologies and backup strategies for their applications. Additionally, we realized lower management costs because managing a standard platform architecture was simpler and more efficient.
Best practices

- Conduct a business impact analysis to identify critical business processes.
- Establish RTO and RPO objectives for each critical business process.
- Move communications technologies (Exchange, SharePoint, and Skype) to external SaaS first to mitigate the lack of communications in case of a disaster at a datacenter.
- Consider all factors when deciding the correct platform for an application, including operating cost, future capital investment, shared responsibility, regulatory and legal requirements, performance requirements, and differences in cloud technologies.
- Deploy standard server software configurations and architectures wherever possible.

Lessons learned

- Criticality of a process is determined first, the required applications for that process inherit the same criticality.
- Cloud deployments encourage standardization of DR architecture, reducing complexity and operational cost.
- Moving applications to the cloud can result in different points of failure. Those need to be addressed by changes in architecture.
- Backing up cloud resources to on-premises servers may be time consuming and expensive. Use the in-cloud backup solutions offered by the cloud provider.
- Take advantage of massive investments being made by Azure in building DR solutions. The more stack layers that move to Azure, the more that future Microsoft investments can benefit you.
- Take advantage of the very low entry cost of deploying applications in a geo-diverse manner. Compared to a business building their own remote datacenter, Azure can be up to 100 times more cost effective.

Resources

- Azure Business Continuity Technical Guidance
- Driving cloud adoption in an enterprise IT organization
- Microsoft improves its SAP disaster recovery capabilities by upgrading to SQL Server 2014
- Overview of Windows Performance Monitor

For more information

Microsoft IT Showcase
microsoft.com/itsshowcase

© 2016 Microsoft Corporation. All rights reserved. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners. This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.