SAP on Google Cloud: Disaster-recovery strategies

Overview

Contents

About this document

This document is part of a series about working with SAP on Google Cloud. The series includes the following documents:

- High availability
- Migration strategies
- Backup strategies and solutions
- Disaster-recovery strategies (this document)

Introduction

This document helps architects to make smart decisions when designing disaster-recovery (DR) architectures and strategies. The document considers not just the criticality of individual solutions, but also the different components of a typical SAP system.

A good disaster-recovery strategy begins with a business impact analysis that defines two key metrics:

- **Recovery Time Objective (RTO)**: How long you can afford to have your business offline.
- **Recovery Point Objective (RPO)**: How much data loss you can sustain before you run into compliance issues due to financial losses.

For both cases, you must determine the costs to your business while the system is offline or for data loss and re-creation.

Typically, the smaller your RTO and RPO values are (that is, the faster your application must recover from an interruption), the more your application will cost to run. Because smaller RTO and RPO values often mean greater complexity, the associated administrative overhead also increases with lower RTO and RPO values.

Disaster recovery

Disaster recovery involves a set of policies, tools, and procedures to enable the recovery or continuation of vital technology infrastructure and systems following a natural or human-induced disaster. Disaster recovery focuses on the technology systems that support critical business functions, as opposed to business continuity, which involves keeping all essential aspects of a business functioning despite significant disruptive events. Disaster recovery can therefore be considered as a subset of business continuity.

When a primary site that hosts mission-critical technical infrastructure becomes completely unavailable, that is considered a *disaster*. It's important to consider the probability that a disaster might occur with or without warning because some steps, such as relying on completion of replicas, depend on whether a disaster occurs with warning. In all cases, you need to maintain full access to the disaster-recovery systems both for users and any other systems interacting with the DR system. You also need to ensure that enough capacity is available at the disaster-recovery site.

To provide business continuity during a major disaster, these IT systems need to keep functioning. In Google Cloud, zones are used to provide high availability (typically, three zones per region), and regions are used to provide disaster recovery. Google regions are dispersed across the globe with ongoing investment in newer regions. Given the diversity of these regions, it's possible to protect against a disaster bound to the local geography, but to also provide disaster recovery at a continent level.

You can help provide this protection by creating a single virtual private cloud (VPC). A VPC in Google Cloud is global by default. A VPC provides networking for your cloud-based resources and services that span multiple regions without communicating across the public internet. This means that when your systems are on Google Cloud, there is no extra setup required in order to configure network connections between regions. The ability to move your data between regions is assumed and comes at no additional cost. For more information, see global VPC.

Architecture considerations

For disaster recovery, you need to consider all layers that make up an application, including the following:

- Application servers and web servers
- File stores such as NFS or SMB links and systems
- Supporting services such as Network Time Protocol (NTP) servers, Cloud DNS, and print services
- The underlying database

Also make the following part of your disaster-recovery planning:

- Application server lifecycle management
- TLS certificates
- Management servers

For Google-supplied supporting services, such as Cloud DNS and NTP, Google Cloud provides disaster-recovery systems as a baseline. For customer-specific services such as print, you might want to consider using standby systems. Where possible, we recommend that you use

Google-supplied services, rather than defining or constructing your own environment with all peripheral services. For more information, see the Disaster recovery planning guide.

Disaster-recovery strategies

Depending on your RTO and RPO requirements, you need to consider different tools and design strategies for the primary site and the disaster-recovery site. This document considers three scenarios:

- **Scenario 1**: RTO within days, and RPO depends on business function. This scenario is meant for noncritical business applications and non-production environments.
- **Scenario 2**: RTO less than a day, and RPO within minutes. The business can function without this application for a short time as long as there is a plan to recover within a reasonable time.
- **Scenario 3**: RTO in minutes, and RPO as close to zero as possible. This scenario is intended for business-critical applications and is designed with the shortest possible recovery with near-zero data loss.

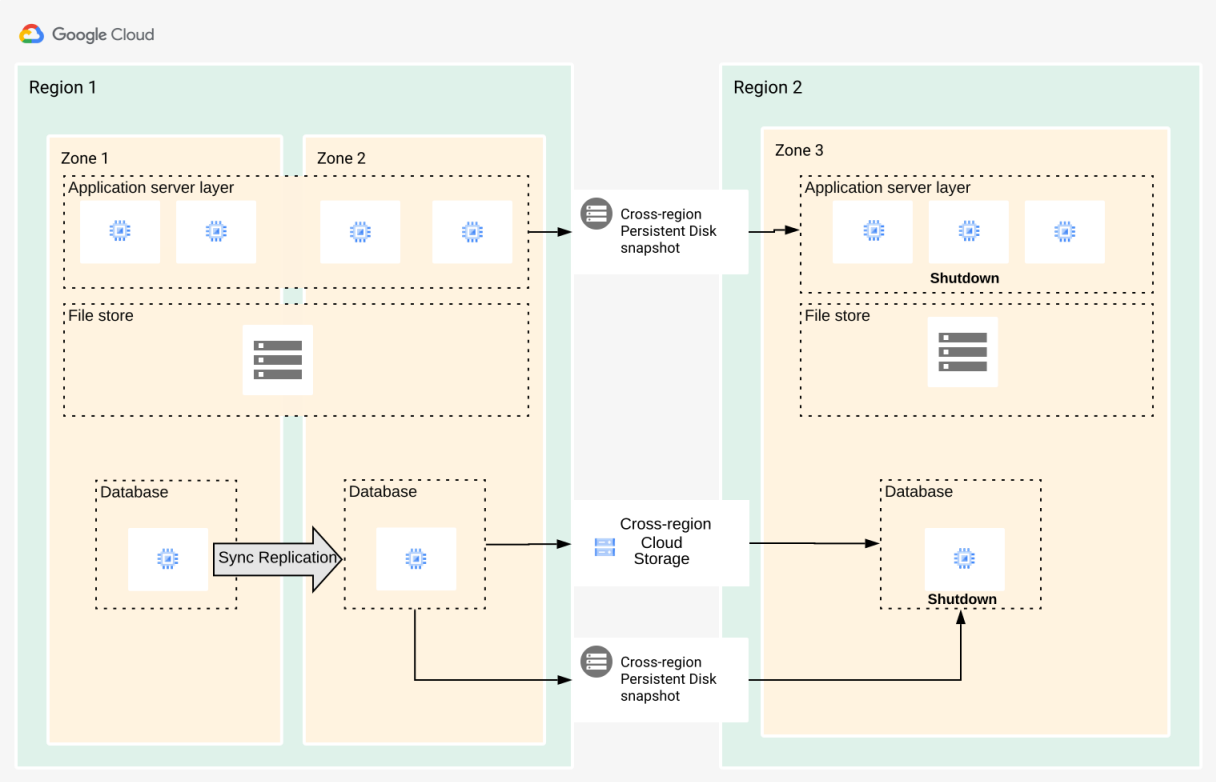
The following table provides an overview of recovery methods by component for each of the scenarios.

Scenario	RTO	RPO	Database	File shares	App servers
1	days	day	Backup/restore*	Backup/restore	Backup/restore
2	hours	minutes	Backup/restore or asynchronous replication	Backup/restore or asynchronous replication	Backup/restore
3	minutes	~0	Synchronous replication	Synchronous replication	Pre-build, restore from Google Cloud machine image or persistent disk snapshot

* Backup/restore is not limited to database technologies and can include other third-party solutions or persistent disk snapshots.

Scenario 1: RTO within days, RPO dependent on business function

This scenario has a recovery time objective in days and a recovery point objective of less than a day. Backups are restored, and the system is started as normal. For more information on different backup options, see SAP on Google Cloud: Backup strategies and solutions. In the event of a disaster, the SAP servers are recovered from persistent disk snapshots, backups, or both. One of the key advantages of this strategy is that none of the target systems incur running costs.

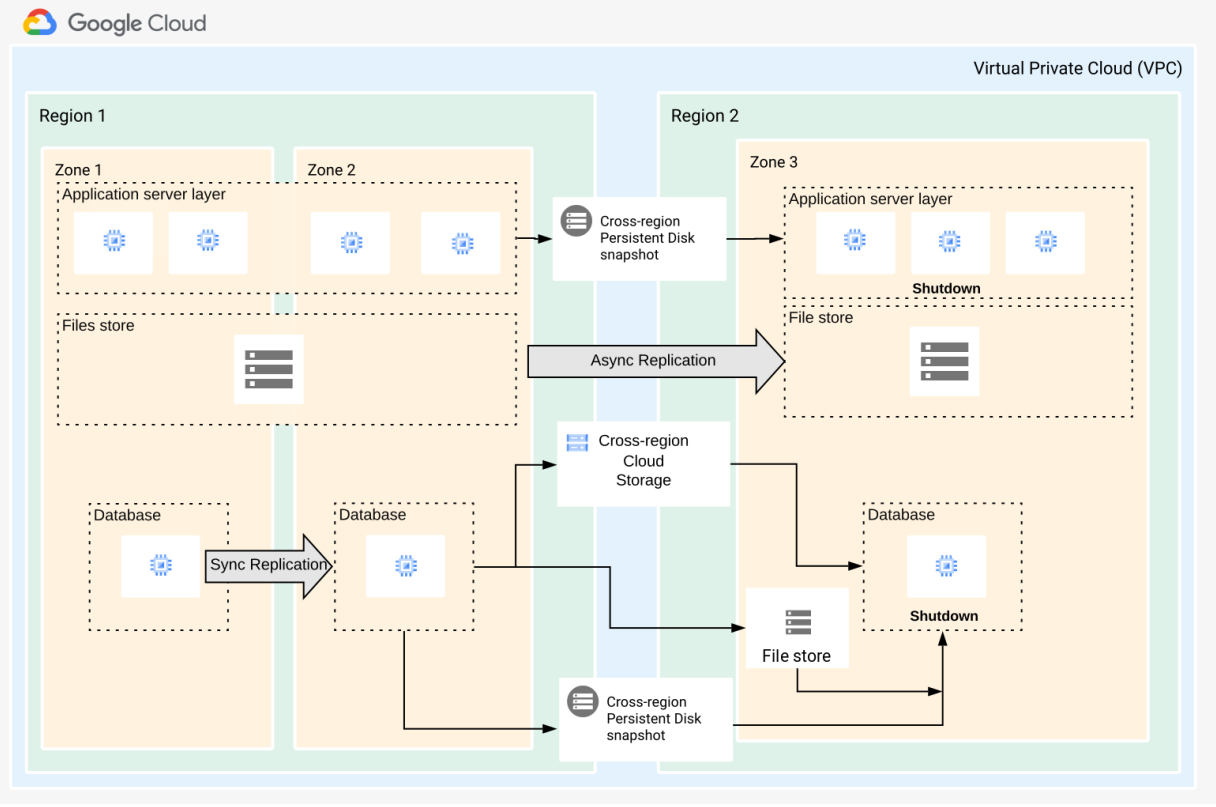


Scenario 2: RTO less than a day, RPO within minutes

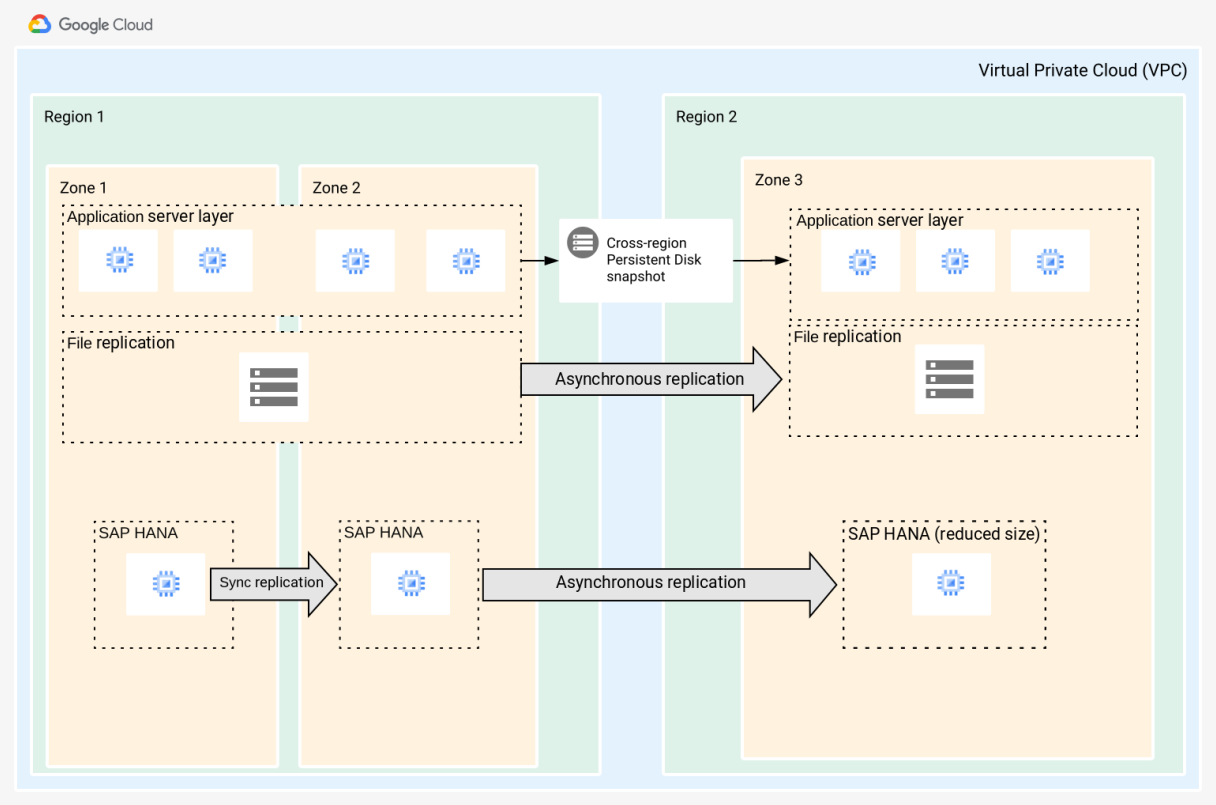
This scenario follows two different approaches to replicate data and recover an SAP system in the event of disaster. The approach that you follow differs depending on whether your SAP deployment uses an SAP HANA database or any other traditional database.

You can replicate traditional databases by shipping database change logs. During normal database recovery, these change logs are applied. In this case, the database logs are continuously copied from the primary site to the disaster-recovery site. In the event of a disaster, the last full backup is restored and all remaining logs are replayed to bring the system back to the state of the last replicated log.

As with scenario 1, none of the disaster-recovery systems run in the disaster-recovery site. You recover the SAP servers from persistent disk snapshots, restore the database from backups, and replay the logs to a point in time set by the RPO. Because you can recover the database to any point in time until the time of the last replicated log, you help to protect the system from potential user error.



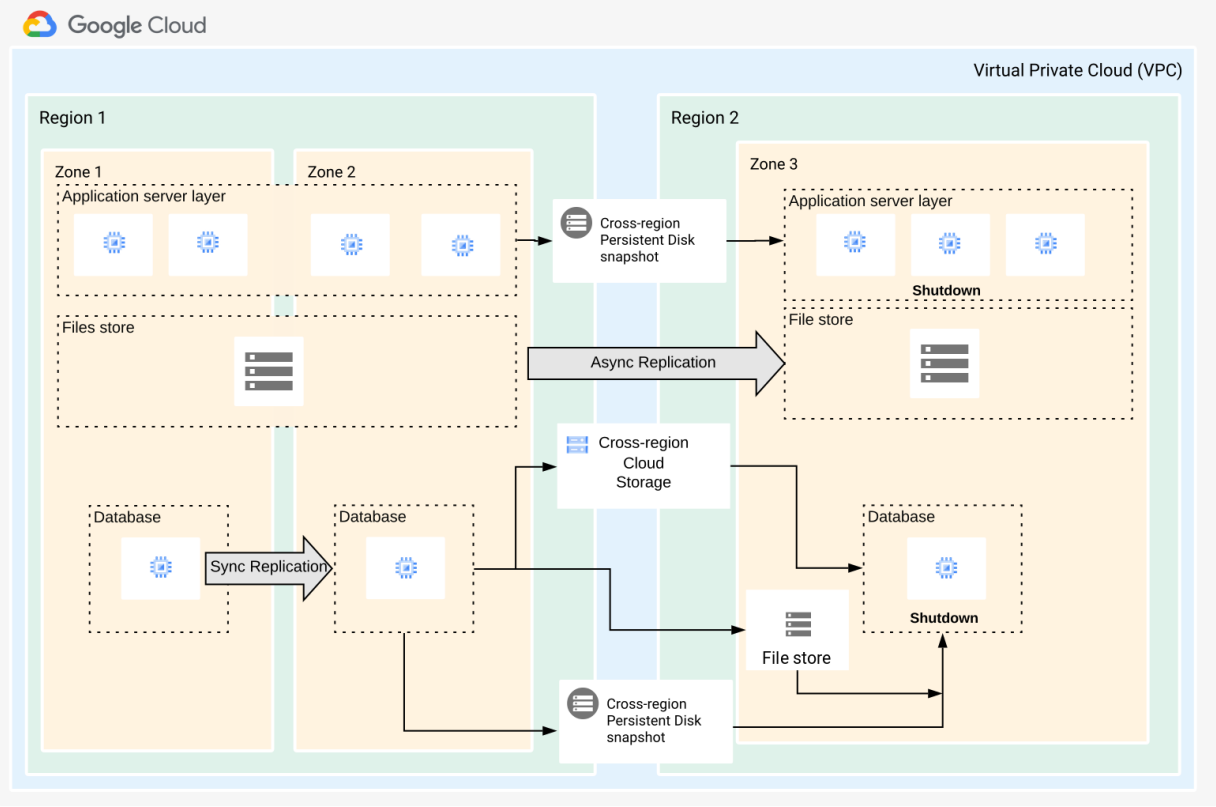
For SAP HANA, you can use a cost-optimized system replication method. In this case, the disaster-recovery virtual machine is about half the size of the source database. All data changes at the source are replicated to this system. In the event of a disaster, the disaster-recovery system restarts and is sized to match the source database.



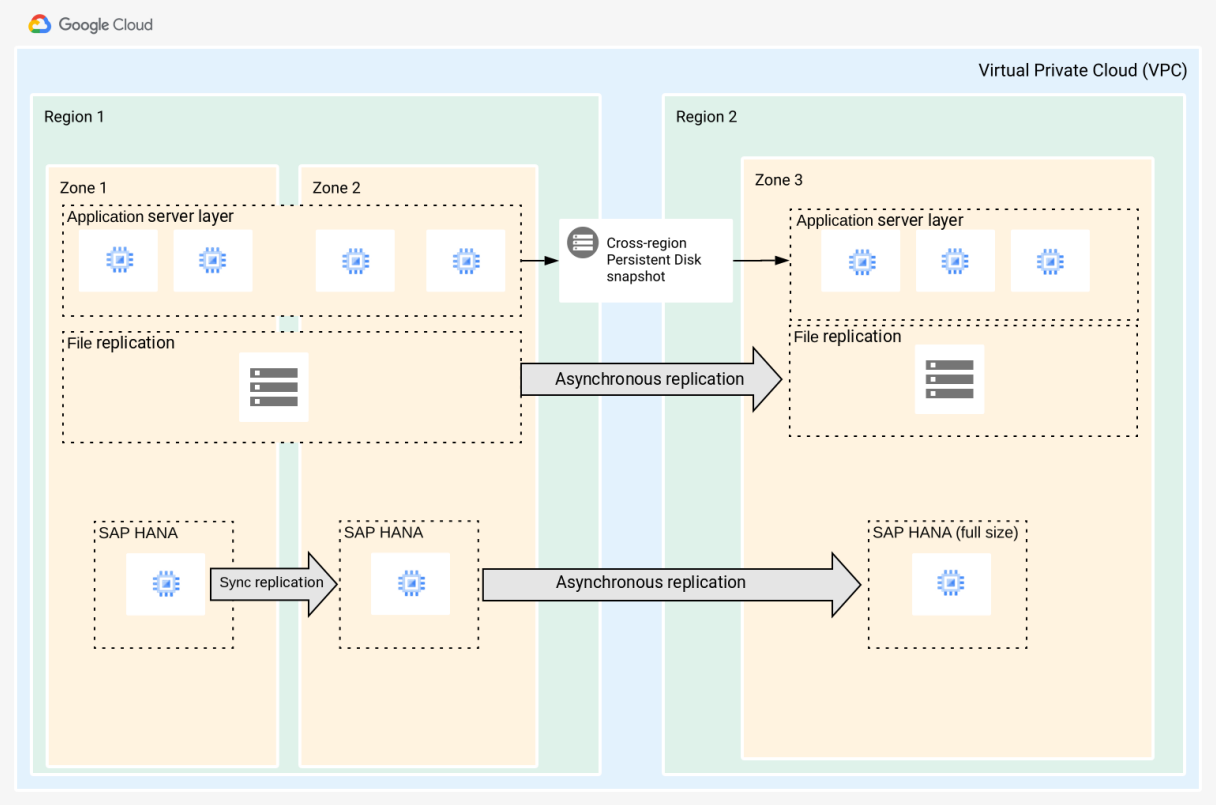
In both cases, you replicate the file shares to ensure synchronization of file-based interfaces and SAP kernel changes. You can pre-build application servers and turn off or restore them from a persistent disk snapshot. The advantage of this scenario is that it blends RTO, RPO, and cost. You can recover the systems in less than a day with minimal data loss.

Scenario 3: RTO in minutes, RPO close to zero

Although the replication methods for this configuration are similar to scenario 2, the target systems are always on and configured to the same size as the source systems. Conventional databases will have their change logs applied as soon as they arrive at the disaster-recovery system.



You can asynchronously replicate SAP HANA systems with the disaster-recovery system sized according to the primary site (replication with pre-load).



We recommend that you automate replication fail-over from the primary database to the disaster-recovery database, but that you initiate it manually. For more information about database-specific configurations, see Further reading.

The key advantage of scenario 3 is that full reservation of resources is guaranteed at the disaster-recovery site. Again as with scenario 2, you replicate the file shares to ensure synchronization of file-based interfaces and SAP kernel changes. You can pre-build application servers and either turn them off or take a persistent disk snapshot.

Disaster-recovery planning and testing

As noted earlier, you need a proper definition for RPO and RTO. We don't recommend defining these metrics on a system-by-system basis. Instead, take into account all solution components in the business process. For example, consider a system landscape where you might have designed an ERP (enterprise resource planning) system to recover according to scenario 3 (in minutes), but a middleware system that forwards messages to this ERP system might have been designed to recover according to scenario 1 or 2 (hours or days). Although the ERP system might be available within an hour, you can't begin processing messages until the middleware product is back online. So you must ensure that all system owners are aligned on requirements such that the overall RTO and RPO requirements and resulting design match the business requirements.

It's not enough just to have a plan for disaster recovery. Make sure that your plan addresses the full recovery process, from fail-over to fail-back. Planning and architecture are only part of the solution, though. Testing and rehearsing a disaster is key to a successful disaster-recovery strategy. Make sure that your testing follows a step-by-step guide (disaster-recovery blue book), which ideally doesn't assume any in-depth knowledge of technical processes.

Make sure that you test your disaster-recovery plan, not just on a system-by-system basis, but also on the full solution stack and on the communications plan and business continuity. You won't have confidence in the plan until you've successfully tested it.

Recommendations

After you choose the disaster-recovery plan that matches your RTO and RPO requirements, you must consider capacity planning and automation.

Capacity planning

If you adopt a cost-optimized strategy for your disaster-recovery systems, they are shut down during normal operation. To ensure that enough capacity is available, consider moving non-production systems to the disaster-recovery site. In the event of a disaster, you can shut down these systems, with the disaster-recovery systems using the capacity reserved by the non-production systems. From a maintenance standpoint, you must ensure that enough capacity is available to stand up a copy of a development system, so that you can control how to develop and transport any emergency SAP changes to the production system.

Automation

You usually manually initiate a disaster-recovery plan. However, after you and the relevant stakeholders have decided on that plan, you should automate the recovery and startup to help to ensure fast and error-free recovery.

With Google Cloud, infrastructure is considered to be code. This means that you can repeatably build, start, and stop landscapes, including compute, storage, and networks. Using IaC (infrastructure as code) techniques, you can automate all steps that are required to bring the system back to an operational state, reducing the cut-over time. You can build these automation scripts by using Google Cloud Deployment Manager or by using open source tools such as Terraform.

Disaster-recovery plan

When you design your disaster-recovery plan, you can take advantage of the Google Cloud architecture and possible cost savings achieved by virtualization and by laying out environments across regions to reserve capacity. Given these resources, you can deploy a disaster-recovery environment that not only matches but even exceeds current on-premises RTO and RPO designs.

Summary

Deciding on your RTO and RPO requirements is important. We recommend that you consult with the business to ensure continuity in the event of a disaster and to control costs. In agreement with your business stakeholders, you can choose the disaster-recovery strategy and architecture that best suit your needs. Keep the following in mind as you plan:

- Fully document your disaster-recovery plan, and make sure that the documentation is available in the event of a disaster.
- Regularly test your plan, and ensure that system administrators and the business are aware and trained in the procedures.
- Test any changes in the threat landscape, and ensure that your plan reflects them.

After your disaster-recovery plan is in place, test it regularly, noting any issues that arise and adjusting your plan accordingly. Using Google Cloud, you can test recovery scenarios at minimal cost—for example, by creating a replica environment for the duration of the disaster-recovery test and deleting the environment afterward.

Further reading

- Disaster recovery planning guide
- Disaster recovery for SQL Server
- Disaster recovery for SAP Adaptive Server Enterprise
- HowTo - Standby System (Recovery from Log Backup) - MaxDB
- High-availability disaster recovery (HADR)