Disaster Recovery (DR) objectives:

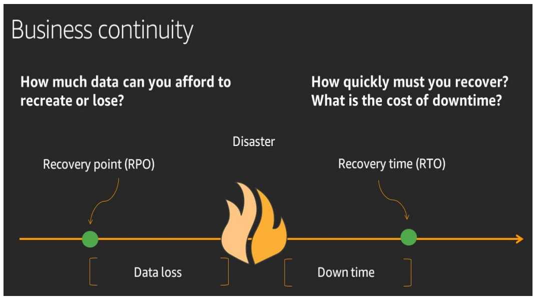
In addition to availability objectives, your resiliency strategy should also include Disaster Recovery (DR) objectives based on strategies to recover your workload in case of a disaster event. Disaster Recovery focuses on one-time recovery objectives in response to natural disasters, large-scale technical failures, or human threats such as attack or error. This is different than availability which measures mean resiliency over a period of time in response to component failures, load spikes, or software bugs.

Recovery Time Objective (RTO) Defined by the organization. RTO is the maximum acceptable delay between the interruption of service and restoration of service. This determines what is considered an acceptable time window when service is unavailable.

Recovery Point Objective (RPO) Defined by the organization. RPO is the maximum acceptable amount of time since the last data recovery point. This determines what is considered an acceptable loss of data between the last recovery point and the interruption of service.

The relationship of RPO (Recovery Point Objective), RTO (Recovery Time Objective), and the disaster event.

RTO is similar to MTTR (Mean Time to Recovery) in that both measure the time between the start of an outage and workload recovery. However MTTR is a mean value taken over several availability impacting events over a period of time, while RTO is a target, or maximum value allowed, for a single availability impacting event.



Business continuity

How much data can you afford to recreate or lose?

How quickly must you recover? What is the cost of downtime?

General Objectives of a Disaster or Contingency Plan:

★.The following is a list of general objectives departments should consider when creating an Information Disaster Prevention and Recovery Plan:

★.Ensure the safety of all employees and visitors at the site/facility

★.Protect vital information and records

★.Secure business sites and facilities

★.Safeguard and make available vital materials, supplies and equipment to ensure the safety and recovery of records from predictable disasters

★.Reduce the risk of disasters caused by human error, deliberate destruction, and building or equipment failures

★.Be better prepared to recover from a major natural catastrophe

★.Ensure the organization's ability to continue operating after a disaster

★.Recover lost or damaged records or information after a disaster

Plan Information:

A sound understanding of the nature, scope, and limitations of a disaster plan ensures that management's expectations are realistic and the plan plays its proper role in achieving the department's overall goals and objectives.

An information disaster plan is a practical program of preventive steps and actions responding to potential and actual disasters. While each department must determine its own specific objectives, several main objectives are present in all plans.

The plan should: Identify and adequately protect the department's vital records (vital records program)

Reduce the risk of disasters caused by human error, deliberate destructiveness, and building or equipment failure, as well as, the adverse consequences of all disasters by mandating specific security, maintenance and training programs (disaster prevention)

Ensure the department's ability to effectively resume operations after a disaster by spelling out management policies, procedures, and resources to be activated in disaster situations (crisis management)

Ensure the department's ability to rapidly reconstruct essential information and salvage damaged records containing information essential to establishing detailed recovery procedures, and a management directive for implementation (disaster recovery)

An information disaster plan is a written, approved, implemented, and periodically tested program to identify, protect, reconstruct or salvage an organization's vital and historical records, and establishes procedures for the immediate resumption of business operations in the event of a disaster.

The objectives of disaster management are as follows:

★.Improving tolerance.

★.Preventing losses and dangers.

★.Providing relief to the affected people.

★.Preparing for actions to be taken at the time of disaster.

★.Assessing the damage caused.

★.Arrangement of rescue for the affected.

★.Rehabilitation and rebuilding the affected area.

Enterprise Design Thinking is our approach to applying design thinking at the speed and scale that the modern enterprise demands. It’s a framework for teaming and action. It helps our teams not only form intent, but deliver outcomes—outcomes that advance the state of the art and improve the lives of the people they serve.

The field guide provides a high-level overview of Enterprise Design Thinking:

LEARNING IT: A summary of the fundamental concepts of Enterprise Design Thinking.

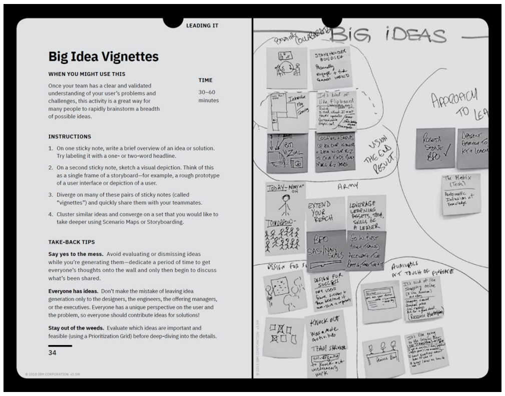
LEADING IT: A quick reference for facilitating essential Enterprise Design Thinking activities on your team.

Are you ready to see problems and solutions from a new point of view? **Embrace the principles of Enterprise Design Thinking:**

Focus on user outcomes: Our users rely on our solutions to get their jobs done every day. Success isn’t measured by the features and functions we ship—it’s measured by how well we fulfill our users’ needs.

Diverse empowered teams: Diverse teams generate more ideas than homogeneous ones, increasing your chance of a breakthrough. Empower them with the expertise and authority to turn those ideas into outcomes.

Restless reinvention: Everything is a prototype. Everything—even in-market solutions. When you think of everything as just another iteration, you're empowered to bring new thinking to even the oldest problems.



LEADING IT

Big Idea Vignettes

WHEN YOU MIGHT USE THIS

Once your team has a clear and validated understanding of your user's problems and challenges, this activity is a great way for many people to rapidly brainstorm a breadth of possible ideas.

TIME

30–60 minutes

INSTRUCTIONS

1. On one sticky note, write a brief overview of an idea or solution. Try labeling it with a one- or two-word headline.
2. On a second sticky note, sketch a visual depiction. Think of this as a single frame of a storyboard—for example, a rough prototype of a user interface or depiction of a user.
3. Diverge on many of these pairs of sticky notes (called "vignettes") and quickly share them with your teammates.
4. Cluster similar ideas and converge on a set that you would like to take deeper using Scenario Maps or Storyboarding.

TAKE-BACK TIPS

Say yes to the mess. Avoid evaluating or dismissing ideas while you're generating them—dedicate a period of time to get everyone's thoughts onto the wall and only then begin to discuss what's been shared.

Everyone has ideas. Don't make the mistake of leaving idea generation only to the designers, the engineers, the offering managers, or the executives. Everyone has a unique perspective on the user and the problem, so everyone should contribute ideas for solutions!

Stay out of the weeds. Evaluate which ideas are important and feasible (using a Prioritization Grid) before deep-diving into the details.

34

© 2018 IBM CORPORATION v3.5W

This article is the first part of a series that discusses disaster recovery (DR) in Google Cloud. This part provides an overview of the DR planning process: what you need to know in order to design and implement a DR plan. Subsequent parts discuss specific DR use cases with example implementations on Google Cloud.

The series consists of these parts:

★.Disaster recovery planning guide (this article)

★.Disaster recovery building blocks

★.Disaster recovery scenarios for data

★.Disaster recovery scenarios for applications

★.Architecting disaster recovery for locality-restricted workloads

★.Disaster recovery use cases: locality-restricted data analytic applications

★.Architecting disaster recovery for cloud infrastructure outages

Introduction

Service-interrupting events can happen at any time. Your network could have an outage, your latest application push might introduce a critical bug, or you might someday have to contend with a natural disaster. When things go awry, it's important to have a robust, targeted, and well-tested DR plan.

With a well-designed, well-tested DR plan in place, you can make sure that if catastrophe hits, the impact on your business's bottom line will be minimal. No matter what your DR needs look like, Google Cloud has a robust, flexible, and cost-effective selection of products and features that you can use to build or augment the solution that is right for you.

Basics of DR planning

DR is a subset of business continuity planning. DR planning begins with a business impact analysis that defines two key metrics:

A recovery time objective (RTO), which is the maximum acceptable length of time that your application can be offline. This value is usually defined as part of a larger service level agreement (SLA).

A recovery point objective (RPO), which is the maximum acceptable length of time during which data might be lost from your application due to a major incident. This metric varies based on the ways that the data is used. For example, user data that's frequently modified could have an RPO of just a few minutes. In contrast, less critical, infrequently modified data could have an RPO of several hours. (This metric describes only the length of time; it doesn't address the amount or quality of the data that's lost.)

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. The following graph shows the ratio of cost to RTO/RPO.

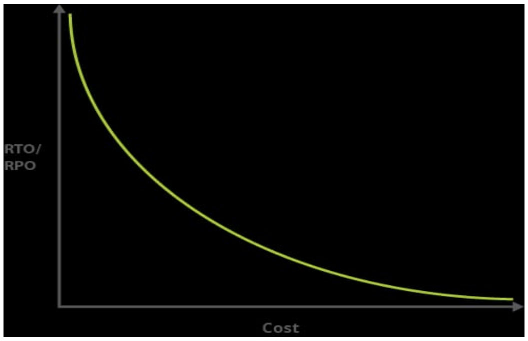
Graph showing that small RTO/RPO maps to high cost.

Because smaller RTO and RPO values often mean greater complexity, the associated administrative overhead follows a similar curve. A high-availability application might require you to manage distribution between two physically separated data centers, manage replication, and more.

RTO and RPO values typically roll up into another metric: the service level objective (SLO), which is a key measurable element of an SLA. SLAs and SLOs are often conflated. An SLA is the entire agreement that specifies what service is to be provided, how it is supported, times, locations, costs, performance, penalties, and responsibilities of the parties involved. SLOs are specific, measurable characteristics of the SLA, such as availability, throughput, frequency, response time, or quality. An SLA can contain many SLOs. RTOs and RPOs are measurable and should be considered SLOs.

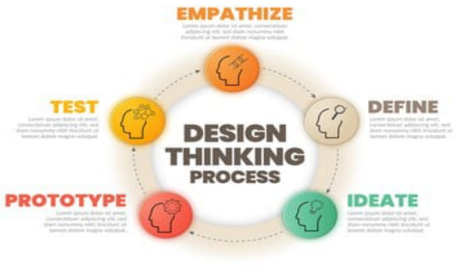
You can read more about SLOs and SLAs in the Google Site Reliability Engineering book.

You might also be planning an architecture for high availability (HA). HA doesn't entirely overlap with DR, but it's often necessary to take HA into account when you're thinking about RTO and RPO values. HA helps to ensure an agreed level of operational performance, usually uptime, for a higher than normal period. When you run production workloads on Google Cloud, you might use a globally distributed system so that if something goes wrong in one region, the application continues to provide service even if it's less widely available. In essence, that application invokes its DR plan.



Design for end-to-end recovery

It isn't enough just to have a plan for backing up or archiving your data. Make sure your DR plan addresses the full recovery process, from backup to restore to cleanup. We discuss this in the related documents about DR data and recovery.



shutterstock.com · 1926616472

Stage 1: Empathize—Research Your Users' Needs

Illustration of Empathize showing two profile heads looking at each other and overlapping about 25%.

Empathize: the first phase of design thinking, where you gain real insight into users and their needs.

The first stage of the design thinking process focuses on user-centric research. You want to gain an empathic understanding of the problem you are trying to solve. Consult experts to find out more about the area of concern and conduct observations to engage and empathize with your users. You may also want to immerse yourself in your users' physical environment to gain a deeper, personal understanding of the issues involved—as well as their experiences and motivations. Empathy is crucial to problem solving and a human-centered design process as it allows design thinkers to set aside their own assumptions about the world and gain real insight into users and their needs.

Depending on time constraints, you will gather a substantial amount of information to use during the next stage. The main aim of the Empathize stage is to develop the best possible understanding of your users, their needs and the problems that underlie the development of the product or service you want to create.

Stage 2: Define—State Your Users' Needs and Problems

Illustration of a target with an arrow in the center to represent the Define stage of the Design Thinking process.

Define: the second phase of design thinking, where you define the problem statement in a human-centered manner.

In the Define stage, you will organize the information you have gathered during the Empathize stage. You'll analyze your observations to define the core problems you and your team have identified up to this point. Defining the problem and problem statement must be done in a human-centered manner.

For example, you should not define the problem as your own wish or need of the company: "We need to increase our food-product market share among young teenage girls by 5%."

You should pitch the problem statement from your perception of the users' needs: "Teenage girls need to eat nutritious food in order to thrive, be healthy and grow."

The Define stage will help the design team collect great ideas to establish features, functions and other elements to solve the problem at hand—or, at the very least, allow real users to resolve issues themselves with minimal difficulty. In this stage, you will start to progress to the third stage, the ideation phase, where you ask questions to help you look for solutions: "How might we encourage teenage girls to perform an action that benefits them and also involves your company's food-related product or service?" for instance.

Stage 3: Ideate—Challenge Assumptions and Create Ideas

Illustration of three light bulbs going off as a representation of the Ideate part of the design process.

Ideate: the third phase of design thinking, where you identify innovative solutions to the problem statement you've created.

During the third stage of the design thinking process, designers are ready to generate ideas. You've grown to understand your users and their needs in the Empathize stage, and you've analyzed your observations in the Define stage to create a user centric problem statement. With this solid background, you and your team members can start to look at the problem from different perspectives and ideate innovative solutions to your problem statement.

There are hundreds of ideation techniques you can use—such as Brainstorm, Brainwrite, Worst Possible Idea and SCAMPER. Brainstorm and Worst Possible Idea techniques are typically used at the start of the ideation stage to stimulate free thinking and expand the problem space. This allows you to generate as many ideas as possible at the start of ideation. You should pick other ideation techniques towards the end of this stage to help you investigate and test your ideas, and choose the best ones to move forward with—either because they seem to solve the problem or provide the elements required to circumvent it.

Stage 4: Prototype—Start to Create Solutions

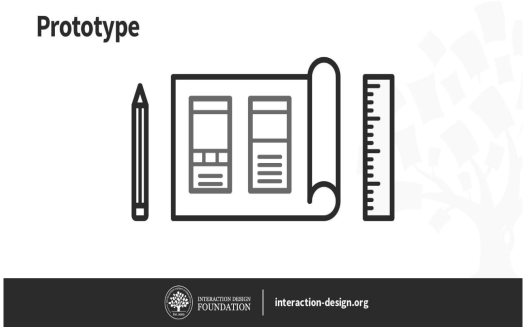
Illustration of the Prototype phase of the design process showing a pencil, wireframes on paper, and a ruler.

Prototype: the fourth phase of design thinking, where you identify the best possible solution.

The design team will now produce a number of inexpensive, scaled down versions of the product (or specific features found within the product) to investigate the key solutions generated in the ideation phase. These prototypes can be shared and tested within the team itself, in other departments or on a small group of people outside the design team.

This is an experimental phase, and the aim is to identify the best possible solution for each of the problems identified during the first three stages. The solutions are implemented within the prototypes and, one by one, they are investigated and then accepted, improved or rejected based on the users' experiences.

By the end of the Prototype stage, the design team will have a better idea of the product's limitations and the problems it faces. They'll also have a clearer view of how real users would behave, think and feel when they interact with the end product.



Stage 5: Test—Try Your Solutions Out

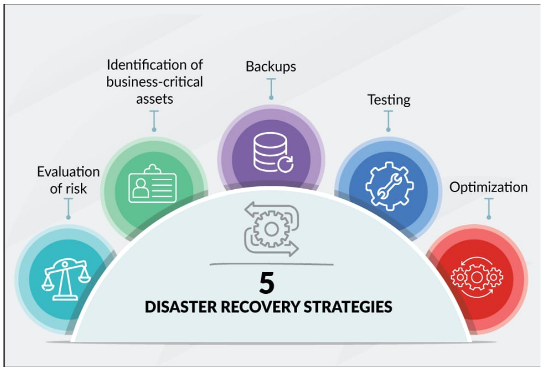
Illustration of the Test phase of the design process showing a checklist on a clipboard.

Test: the fifth and final phase of the design thinking process, where you test solutions to derive a deep understanding of the product and its users.

Designers or evaluators rigorously test the complete product using the best solutions identified in the Prototype stage. This is the final stage of the five-stage model; however, in an iterative process such as design thinking, the results generated are often used to redefine one or more further problems. This increased level of understanding may help you investigate the conditions of use and how people think, behave and feel towards the product, and even lead you to loop back to a previous stage in the design thinking process. You can then proceed with further iterations and make alterations and refinements to rule out alternative solutions. The ultimate goal is to get as deep an understanding of the product and its users as possible.

Disaster recovery strategy:

Disaster recovery strategies are plans and processes designed to restore and resume normal business operations after a disruptive event.


Identification of business-critical assets
Backups
Testing
Evaluation of risk
Optimization
5
DISASTER RECOVERY STRATEGIES

Some common strategies include:

1.Data Backups: Regularly backing up data to off-site or cloud locations to ensure its availability in case of system failure or data loss.

2.Redundancy and Failover Systems: Employing redundant hardware, software, or systems to ensure continuity in case of a primary system failure. This can involve failover mechanisms that automatically switch to backup systems.

3.Disaster Recovery as a Service (DRaaS): Utilizing third-party services that specialize in disaster recovery to ensure quick recovery and minimal data loss in case of a disaster.

4.Business Continuity Planning (BCP): Developing a comprehensive strategy that outlines procedures and protocols to keep essential business functions running during and after a disaster.

5.Virtualization: Employing virtual machines or cloud-based services to replicate systems and data, enabling rapid recovery and continuity.

6.Testing and Training: Regularly testing the disaster recovery plan to ensure its effectiveness and training employees on their roles during a disaster.

7.Geographic Diversification: Spreading operations across different geographic locations to reduce the impact of regional disasters.

8.Incident Response Plans: Developing specific procedures to address various types of disasters (natural, cyber attacks, human error) to mitigate their impact and respond effectively.

Combining these strategies in a comprehensive disaster recovery plan can help organizations better prepare for and recover from unexpected events.

Configuration of Disaster Recovery (DR) Backups:

Disaster Recovery (DR) Backups are executed using the following phases:

Export

★.During the export phase, the software copies a metadata backup of the CommServe database to the default staging directory that is in software_installation_directory\CommserveDR.

★.You can specify a local path or a network path as a destination location to export the metadata.

You can set up the following additional export destinations:

1.Commvault Cloud Services portal

2.Cloud storage library

Disaster recovery in a replication setup:

Disaster recovery in a replication setup involves implementing strategies to ensure data and system availability in case of a catastrophe. Replication creates duplicate copies of data across different locations. To enable disaster recovery, these copies must be maintained and utilized if the primary site fails due to natural disasters, cyber attacks, or other unforeseen events. This involves regularly synchronizing data between sites, defining recovery point objectives (RPOs) and recovery time objectives (RTOs), establishing failover mechanisms, and having a comprehensive plan to switch operations to the secondary site swiftly and efficiently to minimize downtime and data loss.

Synchronous and asynchronous replication

During synchronous replication, data is written to a target data object while simultaneously being written to the corresponding source, allowing you to attain the lowest possible RTO and RPO objectives. This type of disaster recovery replication is preferred for high-end transactional applications and high-availability clusters requiring instant failover. The software client that writes the data receives the confirmation of the writing only after the data is committed to both the primary and secondary storage.

Although an object and its replica are kept synchronized, this creates latency in, and slows down, the app being synchronized, taking up bandwidth, and creating general overhead. If an alternative storage location is used, there is also the possibility that it could be disconnected. Yet synchronous replication allows you to fail over to the secondary site almost instantly and without data loss.

During asynchronous replication, data is written to a target data object only some time after it has been written to the corresponding source. The disaster recovery replication of the data occurs in set intervals (once a minute, ten minutes, an hour, etc.), according to a set schedule. This is a good choice if your network bandwidth cannot support the pressure of synchronous replication, that is, if the change rate of your mission-critical data constantly exceeds its rate of transfer to the failover site.

File-based and block-based replication

A file system stores files on certain disk blocks. One file may be stored on blocks scattered all across the disk. That is why when a file-based replication process reads the file, it has to “run” about the disk to find the read file’s scattered pieces. This “running about” takes considerable time. This time loss can be avoided via block-based replication, which transfers to a target location, not changed files, but changed blocks, reading blocks in the order in which they are situated on the disk. Therefore, other conditions being equal, it is preferable to opt for a DR solution performing block-based replication.

Full-size replication and incremental replication

To continuously replicate the whole volume of your data is unreasonable and impractical. However, one full-size replication needs to be done at first. As a result of this full-sized replication, an exact replica of the source object is created. Then, incremental replication can start, which means that only the data changes are copied over to the failover site (changes on the block level, if block-based replication is used). At present, all advanced DR solutions, like NAKIVO Backup & Replication, allow you to perform incremental disaster recovery replication.

Application-aware replication

If disaster recovery replication is application-aware, it captures the state of in-memory application data and I/O operations. This allows you to avoid data loss on the application. Replicated applications remain transactionally-consistent, meaning they won’t crash when run on the DR site.

Disaster Recovery testing procedures:

Disaster recovery testing involves several procedures to ensure preparedness for various scenarios.

These procedures typically include:

1.Tabletop Exercises: Simulated scenarios discussed among team members to evaluate the response and decision-making processes without executing the recovery plan.

2.Functional Testing: Testing specific components or functions of the recovery plan to verify their proper operation, often in a controlled environment.

3.Data Recovery Testing: Verifying the backup systems' ability to restore data accurately and timely in case of a disaster.

4.Simulation Testing: Replicating disaster scenarios to assess the efficiency and effectiveness of the recovery plan and the team's response.

5.Full-Scale Testing: Conducting a complete run-through of the disaster recovery plan to test its execution, often involving system shutdowns or other disruptive measures.

6.Communications Testing: Verifying the communication procedures, both internal and external, ensuring all stakeholders are informed and the necessary communication channels are functional.

7.Documentation Review: Assessing the disaster recovery plan's documentation, ensuring it's updated and comprehensive.

Regularly performing these tests helps organizations identify weaknesses, improve response procedures, and ensure their readiness in the event of a disaster.

A disaster recovery plan (DRP) in cloud application development is designed to ensure business continuity in unforeseen events. It typically involves:

1.Risk Assessment: Understanding potential risks, such as system failures, cyberattacks, or natural disasters that could disrupt cloud services.

2.Backup and Recovery Strategies: Implementing regular backups of data and systems, often stored in multiple locations or on different servers to ensure redundancy. This allows for quick recovery in case of data loss or system failures.

3.Redundancy and Failover: Designing the infrastructure to have redundancy - multiple servers or data centers that can take over if one fails. This ensures the continuous availability of services.

4.Continual Monitoring and Testing: Regularly monitoring systems to detect any issues or vulnerabilities. Testing the disaster recovery plan periodically to ensure its effectiveness and make necessary adjustments.

5.Documentation and Communication: Having a detailed plan that outlines steps to be taken during an emergency. Communication strategies to notify key personnel and users about the status and recovery process are crucial.

By having a comprehensive disaster recovery plan in place, businesses can mitigate the impact of unforeseen events, minimize downtime, and maintain the continuity of their cloud-based services. This ensures that even in the face of disasters, the business operations can continue as smoothly as possible.