Cognixia®

Reliability, Availability and Fault Tolerance

By - Ahmed

www.cognixia.com

1

Cognixia®

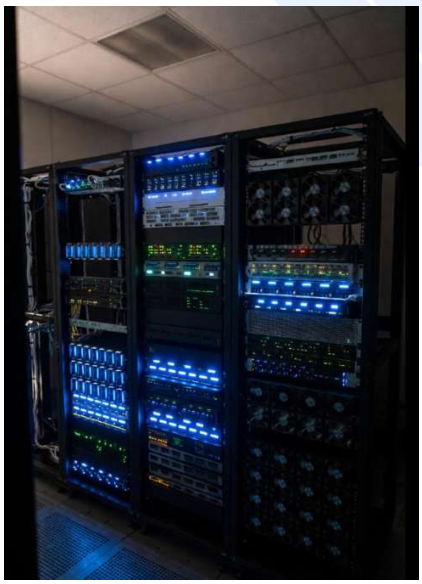
Availability

www.cognixia.com

2

What is Availability

- **Availability** is a measure of **how consistently a system, application, or service remains operational and accessible when users need it**.
- It is usually expressed as uptime percentage (**99%, 99.9%, 99.99%, 99.999%**).
- **Formal Definition**
  - **Availability = Ability of a system to remain functional despite failures or disruptions**.
- A **system** with **high availability** has:
  - **Minimal downtime**
  - **Quick recovery from failures**
  - **No single point of failure**
- **Example:** If an online payment system is available **99.99%**, it means it can only be down for **52 minutes per year**.



www.cognixia.com

3

Availability … continue

- **Calculation**
  - **Formula:**
    - ✓ Availability = (Total Time – Downtime) / Total Time
  - **Alternatively, system availability** can be **calculated using maintenance metrics** like **Mean Time Between Failures (MTBF)** and **Mean Time To Repair (MTTR)**
    - ✓ Availability = MTBF / MTBF + MTTR
- **Goal of Architects:** Design systems so they remain operational even if components fail.
- Availability is typically **expressed** as a **percentage** over a **specified period** (*e.g., a month or a year*).
- The industry **often uses** the term **"Nines"** to **describe high levels** of **availability**.

Nines	Percentage	Total Allowed Downtime Per Year (Approx)
Two Nines	99%	3 days, 14 hours, 24 minutes
Three Nines	99.9%	8 hours, 45 minutes, 56 seconds
Four Nines	99.99%	52 minutes, 35 seconds
Five Nines	99.999%	5 minutes, 15 seconds

4

Achieving High Availability (HA)

Cognixia

- High Availability **accomplished through**:
  - **Redundancy**
    - Having **duplicate**, **failover components** (*like backup servers, redundant power supplies, or mirrored data storage*) so that if **one component fails**, the **backup immediately takes** over.
  - **Fault Tolerance**
    - The **ability** of a **system** to **continue operating without interruption** even when one or more of its **components fail**.
  - **Disaster Recovery (DR)**
    - Having a plan and **systems in place** to **quickly restore service** after a **major**, catastrophic **outage** (*like a natural disaster or major data center failure*).

www.cognixia.com

5

Cognixia

Reliability

www.cognixia.com

6

What is Reliability

- **Reliability** is the **probability** that a **system**, **component**, or **product** will perform its **intended function** **correctly** and **without failure**, under **specified conditions**, for a **specified period of time**.
- In simpler terms, reliability answers the question: **"Can the system consistently do its job without breaking down?"**
- **Key Characteristics**
  - **Focus on Consistency**
    - **Reliability measures** the **system's ability** to **maintain** its **functionality** and **quality** of performance **over time**, **preventing failures from** occurring in the first place.
  - **Time-Dependent**
    - **Reliability** is typically **expressed** as a **probability** that **decreases** as the **time period considered increases**.
    - The **longer** you need a **system** to **run without failure**, the **lower** the mathematical **probability** of it achieving that.
  - **Failure Definition is Crucial**
    - A **system** is **only reliable** if it **meets** its **defined performance specifications**.
    - For **example**, a **reliable server must not only stay on** (*Availability*) but **also process transactions within a required time frame** (*Reliability*).

www.cognixia.com

7

Reliability Metrics

- Reliability is **quantified using time-based metrics** that **help predict** and **measure how often a failure occurs**.
- **Mean Time Between Failures (MTBF)**
  - **MTBF** is the most **common metric** for **repairable systems** (*like servers, aircraft engines, or software applications*).
  - It **represents** the **average time elapsed between one failure and the next**.

    MTBF = Total Operational Time\Total Number of Failures

  - Goal: A **higher MTBF indicates** a **more reliable system**, as it **operates longer before** a **failure occurs**.
- **Mean Time To Failure (MTTF)**
  - **MTTF** is **similar to MTBF** but is **used for non-repairable systems** or **components** that are **typically discarded** upon **failure** (*like a lightbulb or a non-redundant hard drive*).
  - It **represents** the **average expected time** until the **first failure**.
- **Failure Rate**
  - The **failure rate** is the **frequency of failures per unit of time**.
  - It is **generally** the **inverse of MTBF** for systems operating in their "useful life" period.

    Failure Rate = Total Number of Failures\Total Time in Service

www.cognixia.com

8

Cognixia

Redundancy

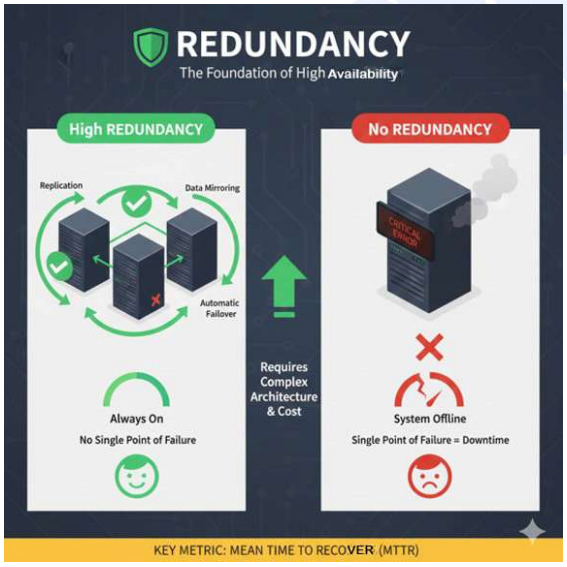
www.cognixia.com

9

What is Redundancy

Cognixia

- **Redundancy** in **computing and engineering** is the **practice** of **duplicating critical components** or functions of a system to **serve as a backup in case** the **primary component fails**.
- Its main purpose is to prevent a **Single Point of Failure (SPOF)**, thereby increasing the system's **reliability** and **availability** (*uptime*) by allowing the backup system to seamlessly take over operations.
- **Why Redundancy is Important**
  - **Redundancy ensures**:
    - Continuous service
    - Automatic failover
    - High Availability
    - Business continuity
- **How Redundancy Helps**
  - Prevents downtime
  - Automatically takes over if a component fails
  - Maintains availability even during failures
  - Supports load balancing and scaling



www.cognixia.com

10

Types of Redundancy

Type of Redundancy	Description	Examples	Where Used (Industry)
Hardware Redundancy	**Duplicate physical components** to avoid single point of failure	**Multiple servers**, dual power supplies, RAID disks	Data centers, manufacturing plants
Software / Application Redundancy	**Multiple instances of the same service** running to ensure continuity	Multiple **microservice replicas**, app clusters	SaaS, e-commerce applications
Network Redundancy	**Multiple network paths**, ISPs, routers, switches	**Dual routers**, MPLS + Internet, multi-AZ networks	Telecom, banks, enterprise networks
Data Redundancy	**Storing/replicating data in multiple locations**	**Database replication**, multi-AZ DB, cross-region copy	Financial systems, healthcare records
Geographic Redundancy	**Deploying systems across different physical locations**	**Multi-region deployments**, DR sites	Banking, aviation, global apps
Functional Redundancy	**Multiple components performing the same function but differently**	**Two monitoring tools, dual authentication systems**	Security, monitoring-heavy industries
Power Redundancy	**Backup power sources to ensure uninterrupted operation**	**Dual UPS**, generators, dual power feeds	Data centers, hospitals
Storage Redundancy	**Multiple storage units** working together to avoid data loss	RAID 1/5/10, **replicated storage**	Enterprise storage, cloud storage providers

www.cognixia.com

11

Minimizing Downtime

www.cognixia.com

12

Redundancy – The Foundation of Uptime

Cognixia

- **Redundancy** means having **multiple copies** of **critical components** so the system does not rely on any single point.
- *"If one fails, another is already ready."*
- **Types of Redundancy**
  - **Hardware redundancy** (multiple servers, disks, network devices)
  - **Network redundancy** (multiple paths/ISPs)
  - **Database redundancy** (replica nodes, multi-AZ DB)
  - **Geographic redundancy** (multi-location deployments)
- **Benefit:**
  - **Service continues** even when components fail
  - **No downtime** during maintenance or patching
  - **Supports scaling** and load distribution
- Redundancy **reduces the chance** of downtime.

13

Cognixia

Failover – Mechanism That Keeps You Running

- **Failover** is the **automatic process** of **switching** to a **healthy component** when a failure occurs.
- *"If something breaks, the system instantly moves to a backup."*
- **Failover can be**
  - **Automatic** (*preferred*)
  - **Manual** (*slow, used in low-cost DR setups*)
- **Examples of Failover**
  - **Load balancer sends traffic** to **healthy servers** only
  - **DNS failover** from **primary** to **secondary site**
  - **Database failover** from **primary** to **read replica**
  - **Kubernetes** reschedules **pods on healthy nodes**
- **What failover ensures**
  - Zero or minimal interruption
  - Quick recovery
  - No user impact in many cases
- Failover **minimizes the impact** of downtime.

14

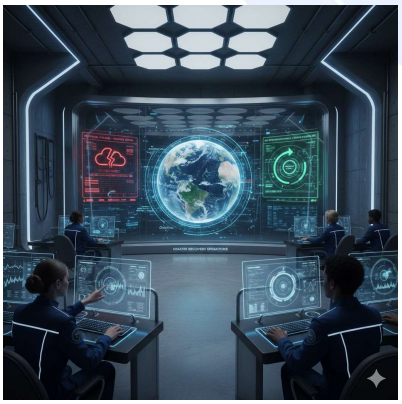
Disaster Recovery (DR)

15

What is Disaster Recovery (DR)

- **Disaster Recovery (DR)** is the **strategy, process, and technology** used to **restore IT systems, applications, and data** after a **major outage or disaster**.
- **DR activates** when **normal redundancy** or **high availability cannot protect** you.
- **Key Metrics for DR Planning**
  - An **effective DR plan** is **defined** by **two critical**, **measurable objectives**:

Metric	What it Measures	Goal
Recovery Time Objective (RTO)	The **maximum acceptable time** for systems and applications to be offline after a disaster.	To **restore operations bef**ore the **business impact becomes significant** (*e.g., RTO of 4 hours*).
Recovery Point Objective (RPO)	The **maximum acceptable age of data** that can be lost from an IT service due to a major incident.	To **minimize data loss** (*e.g., an RPO of 15 minutes means backups must occur at least that often*).



16

Core Components of a DR Strategy

Cognixia

- Disaster Recovery **relies** on having **redundant infrastructure** and documented procedures in a location separate from the primary site (often called **Geo-redundancy**).

1. **Data Backup and Replication**
   - **Backups**: Regular copies of data are taken and stored **off-site** (either physically or in the cloud) so they are unaffected if the primary data center is destroyed.
   - **Replication**: Critical data is continuously or frequently copied to a secondary, standby location.

2. **Disaster Recovery Sites**
   - These are **alternate locations** where **systems can be brought online** after a **disaster**:

Site Type	Description	RTO/Cost
Hot Site	**Fully equipped**, ready-to-run data center with real-time data replication.	**Fastest RTO** / Highest Cost
Warm Site	**Partially equipped**, requiring some time to configure and load recent data.	**Moderate RTO** / Moderate Cost
Cold Site	**Basic facility with power and cooling**; requires all hardware, software, and data to be delivered and installed.	**Slowest RTO** / Lowest Cost

17

Cognixia

Backup & Replication

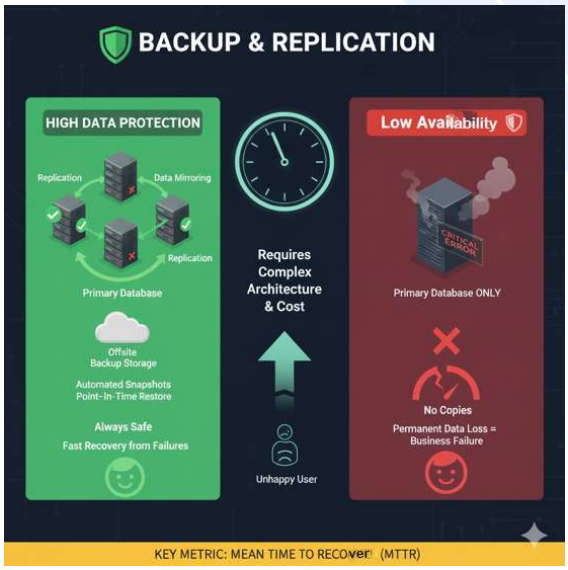
www.cognixia.com

18

What is Backup and Replication

Cognixia

- **Safeguarding data** requires a **multi-layered approach** that **combines both Backup Strategies** and **Replication** to **protect** against different types of threats, ranging from small-scale file **corruption** to **large-scale disaster**.
- **In essence**:
  - **Backup** is for **point-in-time recovery** and protection **against data corruption** or human error.
  - **Replication** is for **real-time business continuity** and minimizing downtime during a system failure.



www.cognixia.com

19

Data Backup Strategies

Cognixia

- **Data Backup** involves **creating copies** of data at specific, **scheduled intervals** and **storing them separately**.
- Its primary objective is to **facilitate the restoration of data** to a specific, uncorrupted point in the past.

A. Types of Backups

Type	Description	Pros & Cons
Full	Copies all selected data every time.	Simplest to restore but takes the most time and storage.
Incremental	Copies only the data that has **changed since the last backup** (of any type).	Fastest backup process, uses least storage, but restoration is complex (requires the last full backup plus *all* subsequent incremental).
Differential	Copies all data that has **changed since the *last full backup***.	Better restore time than incremental (only needs the last full and one differential) but takes more storage than incremental.

B. The 3-2-1 Rule

- This is the **standard** for a robust backup strategy:
  - **3 Total Copies of your data** (the original data plus two backup copies).
  - **2 Different Media Types** (e.g., local disk and cloud storage or tape). This protects against a single media type failure.
  - **1 Offsite Copy** (e.g., in the cloud or a geographically separate data center). This protects against site-wide disasters like fire or flooding.

www.cognixia.com

20

Data Replication

- Data Replication is the **process** of **creating and maintaining multiple**, **synchronized copies** of data **across different systems** or locations, often in real-time.

A. Types of Replication

Type	Synchronization	Best For
Synchronous	**Data is written** to both the primary and secondary locations **simultaneously**. The primary system waits for confirmation from the replica before completing the operation.	Mission-critical apps requiring **zero data loss (RPO = 0)**. Only suitable for short distances due to latency concerns.
Asynchronous	Data is **written to the primary first** and then copied to the **secondary** location with a **slight delay**.	High-performance apps over long distances. Tolerates a small amount of data loss (RPO > 0) but avoids performance impact.

B. Failover Capability

- **Replication** systems are typically **used with failover** mechanisms.
- If the **primary system fails**, the **secondary replicated** copy, often a **Hot Site** (ready-to-run system), can **instantly take over**, ensuring continuous operation with minimal downtime.

www.cognixia.com

21

DR Approaches

www.cognixia.com

22

Disaster Recovery Approaches

- **Building Robust Disaster Recovery** (DR) **Approaches** means **creating a comprehensive**, **resilient plan** and **infrastructure** that ensures the **rapid and reliable restoration** of **critical business functions** and IT systems after a major, **disruptive event**.
- A robust **approach moves beyond** simple **data backups** to focus on minimizing downtime and data loss across the entire organization.

www.cognixia.com

23

DR Approaches … continue

- **Define Clear Recovery Objectives**
  - The foundation of a robust DR plan lies in **establishing measurable targets based** on **business priorities**, which are determined during a **Business Impact Analysis** (BIA).
  - **Recovery Time Objective (RTO)**
    - The **maximum tolerable downtime** before the business suffers unacceptable consequences.
    - A robust system often targets RTOs measured in minutes or hours, requiring automated failover systems.
  - **Recovery Point Objective (RPO)**
    - The **maximum tolerable period** in which **data** might be lost due to a disaster.
    - A robust system uses continuous or near-real-time replication to achieve an RPO of seconds or zero.
  - **Prioritize Critical Systems**
    - **Identify** the handful of **Tier 1 or Tier 2 applications** (*e.g., trading platforms, core databases*) that **must** be **recovered first** and allocate the fastest, most expensive DR resources to them.
    - **Classify Applications by Criticality**

Tier	Type	DR Need
Tier-1	Mission-critical (Payments, customer portal)	Hot/Warm standby
Tier-2	Important (HR, CRM)	Pilot-light / Daily backup
Tier-3	Non-critical (reports, batch jobs)	Cold backup

24

25