Below is a **comprehensive and professionally formatted sentence-by-sentence breakdown** of the document titled **“254. Backup and Recovery Notes”**. Each statement has been meticulously transformed into **exam-focused study notes**, aligning with **CompTIA A+ 220-1102 (Core 2) Objective 4.3: Implement Backup and Recovery Methods**.

**Backup and Recovery Study Notes – CompTIA A+ 220-1102 (Objective 4.3)**

**1. Concept Overview**

* Backup and recovery are essential to data integrity and business continuity.
* Backups are used in scenarios like accidental deletion, hardware failure, or disasters.
* Recovery involves restoring the data from a backup to bring systems back online.

**2. Sentence-by-Sentence Bullet Notes and Detailed Explanation**

**General Backup Principles**

* Any workstation or server with critical data must have proper backups.
* Backups are essential for restoring operations after:
  + Accidental deletion
  + Hard drive failure
  + Other data loss events
* Offsite backups are necessary for disaster recovery scenarios.

**Types of Backups**

**1. Full Backup**

* A **full backup** copies **everything** from the source or system: OS, apps, files, configs.
* Example: If a technician is doing a full backup of your laptop, it’s going to copy all the files from your hard drive onto some kind of external device that I can then use as that backup copy.
* There are some drawbacks:
  + Pros:
    - Fast recovery—only one file set to restore
    - Complete snapshot of the system
  + Cons:
    - **High storage** use (e.g., 1 TB backup needs 1 TB space)
      * **If I have an eight-terabyte external hard disk that I’m using for those full backups, I can really only do about eight days of backups with those eight terabytes.**
      * **If I’m backing up one terabyte’s worth of data each time I do a full backup.**
    - **Slow process**—copies every file even if only one changed
      * **They take a long time because it must copy every single file**.
      * **1 single file change can cause a scheduled backup to happen causing to copy every single file from that hard drive**.
* Example scenario:
  + Let’s say that you just saved a new Word document on your hard drive and I do a backup tonight. When I do a full backup, I’m not going to copy just that Word document but I’m going to copy everything on that hard drive including all of your applications, OS, configurations, and everything else.

**2. Incremental Backup**

* This is a really good backup to use if you’re really concerned about speed and saving the least amount of information.
* Saves **only files changed since the last backup** (full, incremental, or differential).
* An **incremental backup** is designed to **efficiently capture only the data that has changed** since the last backup, whether that last backup was a **full** or **another incremental**.
  + It does **not** copy all the data on the drive.
  + It **only backs up new or modified files** since the previous backup event.
* Pros:
  + **Fast backup process**
  + **Low storage usage**
* Cons:
  + **Slow recovery**—requires full backup + **all** subsequent incremental’s
* Example backup schedule:
  + Sunday: Full backup
  + Monday: Incremental (files changed since Sunday)
  + Tuesday: Incremental (changes since Monday), etc.
* Crash on Friday:
  + Must restore Sunday’s full + Monday–Thursday incremental = 5 jobs

**3. Differential Backup**

* Saves **all files changed since the last full backup** (ignores previous differential/incremental).
* **Differential backups = “What has changed since the last full back up.”**
  + They **do not include the full backup**, but they **grow larger each day** because they include **more changes**.
  + The **differential** backup **grows** each day — not because it’s storing the full back up again, but because **more files have changed** since the last known good backup.

| **Backup Type** | **Daily Size Trend** | **Why?** |
| --- | --- | --- |
| **Incremental** | Stays **small** daily | Only includes that day’s changes |
| **Differential** | **Grows larger** daily | Includes **all changes since full backup** |

* Midpoint between full and incremental backups in terms of time and storage.
* Pros:
  + **Faster recovery**—only need the last full + latest differential
* Cons:
  + Larger backup size each day as week progresses
* Crash on Friday:
  + Restore Sunday’s full + Thursday’s differential = 2 jobs
* Ideal when a balance between recovery speed and backup size is needed.

**Absolutely — here’s a professional, exam-aligned, side-by-side 7-day example showing how incremental and differential backups work, including:**

* **What happens during each day’s backup**
* **What files are stored**
* **What’s required to restore the system after a crash**
* **A clear breakdown of storage vs. restore time**
* **The difference in backup logic**

**🔄 7-Day Backup Strategy Comparison: Incremental vs. Differential**

**🗓️ Scenario Setup:**

* **Full backup occurs on Sunday**
* **Backups from Monday to Friday**
* **We simulate daily file changes**
* **We simulate a server crash on Friday**
* **Goal: Restore system with minimal data loss**

**✅ DAY-BY-DAY CHANGES (Simulated File Activity):**

| **Day** | **File Changes Made** |
| --- | --- |
| **Sunday** | **No changes (full system snapshot)** |
| **Monday** | **File A edited** |
| **Tuesday** | **File B added** |
| **Wednesday** | **File C edited** |
| **Thursday** | **File D added** |
| **Friday** | **Server crashes — recovery needed** |

**📦 INCREMENTAL Backup Workflow**

| **Day** | **Backup Type** | **What It Saves** |
| --- | --- | --- |
| **Sunday** | **Full Backup** | **Entire system (all files, apps, configs, OS)** |
| **Monday** | **Incremental** | **Only File A (changed since Sunday)** |
| **Tuesday** | **Incremental** | **Only File B (changed since Monday)** |
| **Wednesday** | **Incremental** | **Only File C (changed since Tuesday)** |
| **Thursday** | **Incremental** | **Only File D (changed since Wednesday)** |
| **Friday** | **❌ Crash** | **Must restore from Sunday + all incrementals** |

**🔧 Recovery on Friday:**

**You must restore the following backups:**

1. **Sunday (Full)**
2. **Monday (File A)**
3. **Tuesday (File B)**
4. **Wednesday (File C)**
5. **Thursday (File D)**

**✅ Pros: Fast backups, small size**

**❌ Cons: Slow recovery (must apply each incremental in order)**

**📦 DIFFERENTIAL Backup Workflow**

| **Day** | **Backup Type** | **What It Saves** |
| --- | --- | --- |
| **Sunday** | **Full Backup** | **Entire system (all files, apps, configs, OS)** |
| **Monday** | **Differential** | **File A (changed since Sunday)** |
| **Tuesday** | **Differential** | **File A + File B (changes since Sunday)** |
| **Wednesday** | **Differential** | **File A + B + C (since Sunday)** |
| **Thursday** | **Differential** | **File A + B + C + D (since Sunday)** |
| **Friday** | **❌ Crash** | **Must restore from Sunday + latest differential** |

**🔧 Recovery on Friday:**

**You must restore:**

1. **Sunday (Full)**
2. **Thursday (Differential: Files A, B, C, D)**

**✅ Pros: Faster recovery, fewer restore points**

**❌ Cons: Larger backup files as the week progresses**

**📊 Comparison Table: Incremental vs. Differential**

| **Feature** | **Incremental** | **Differential** |
| --- | --- | --- |
| **Daily Backup Size** | **Small** | **Grows daily** |
| **Backup Time** | **Fast** | **Moderate** |
| **Recovery Time** | **Slower (multiple restores)** | **Faster (only 2 restores)** |
| **Files Needed to Restore** | **Full + all incrementals** | **Full + latest differential** |
| **Archive Bit Cleared?** | **✅ Yes** | **❌ No** |
| **Storage Used Over 5 Days** | **Small overall** | **Medium to large (accumulates)** |

**📌 Key Takeaways for Exam and Real Life**

* **Incremental is best when:**
  + **Storage space is limited**
  + **Backup windows must be short**
  + **You’re okay with longer recovery**
* **Differential is best when:**
  + **You want simpler, faster recovery**
  + **You can tolerate larger daily backups**
  + **You’re focused on minimizing restore points**

**4. Synthetic Backup**

* An additional type of backup that can create full backups with lower data transfer requirements.
* Synthetic Backup is not generated directly from the original data like a full backup is.
  + Instead, its actually being assembled from other backup types.
* **Offline-generated full backup** created from a prior full backup and a set of incremental.
  + **Requires less amount of time.**
* Not pulled from the live system; built from backup server.
* Benefits:
  + **Reduces impact on production servers**
  + **Minimizes data transfer**
* Process:
  + Initial full backup → daily incrementals
  + On full backup day: last full + 6 incrementals + latest incremental = new synthetic full
* Used to avoid taxing production systems during full backups
* Not widely adopted in some environments due to preference for direct-from-source full backups

**✅ Definition (In Simple Terms):**

A **synthetic backup** is a type of **full backup** that is **not created directly from the original data** on the system.

Instead, it is **built (“synthesized”) from existing backup files** — usually a **previous full backup plus several incremental backups**.

Think of it as a “**backup of backups**” — assembled offline on the backup server, **not by reading the actual system again**.

**🛠️ How It Works – Step-by-Step**

1. **Sunday:** Full backup is performed as usual from the source server.
2. **Monday–Thursday:** Incremental backups are taken daily.
3. **Friday:** Instead of backing up the full server again, the system:
   * Takes the **Sunday full backup**
   * Adds the **Monday–Thursday incremental backups**
   * **Combines** them into a **new full backup** = Synthetic Full Backup

All of this happens **on the backup storage server**, not on the original system.

**3. Backup Strategy Implementation**

* Best practice: Use a mix of **full, incremental, and differential backups** to optimize:
  + **Backup duration**
  + **Storage efficiency**
  + **Recovery time**
* Common enterprise schedule:
  + Sunday: Full
  + Mon–Thu: Incrementals or Differentials
  + Friday recovery depends on backup method used

**4. Archive Attribute (Archive Bit) – How Systems Track Backups**

* **Archive attribute**: A file-level flag used to determine if the file needs to be backed up
* **Set to ON** when a file is created or modified
* Backup behavior:
  + **Full backup**: Copies all files, **clears archive bit**
  + **Incremental backup**: Copies only changed files, clears archive bit
  + **Differential backup**: Copies all changed files since the last full, **does not clear** archive bit
* Implication:
  + Differential backups keep growing in size as archive bits are not cleared
  + Incremental backups reset archive tracking to isolate each backup window

**5. Real-World Implementation Examples**

* **Enterprise Backup Example:**
  + Sunday: Full backup at 1 AM → saves all server data to tape/cloud
  + Mon–Thu: Incremental backups at 10 PM
  + Friday Crash Scenario:
    - Restore process:
      * Option 1: Full (Sunday) + Incrementals (Mon–Thu) → 5 steps
      * Option 2: Full (Sunday) + Differential (Thursday) → 2 steps
* **Synthetic in Practice:**
  + Backups assembled by backup appliance to create new full
  + Reduces downtime on critical production servers
* **Archive Bit Use:**
  + Ensures backup software knows which files have changed
  + Prevents redundant backup of unchanged data

**6. Exam Relevance**

✅ **Included in the CompTIA A+ 220-1102 Curriculum – Objective 4.3**

**Why it matters:**

* Backup types (full, incremental, differential, synthetic) are specifically referenced in the **official exam objectives**.
* Candidates must understand:
  + **When to use each backup method**
  + **Trade-offs in backup speed, storage, and recovery complexity**
  + **How to count restore jobs in scenario-based questions**
  + **Behavior of the archive bit with each method**

**Common exam scenario:**

* “A full backup occurs Sunday. Incrementals Mon–Thu. Server crash on Friday. How many restores are required?”

Answer: **5** (Full + 4 Incrementals)