Cloud Computing

CHAPTER 5

HOW ARE CLOUDS MANAGED?

Dr. JAMIL S. ALAGHA

How are Clouds Managed?

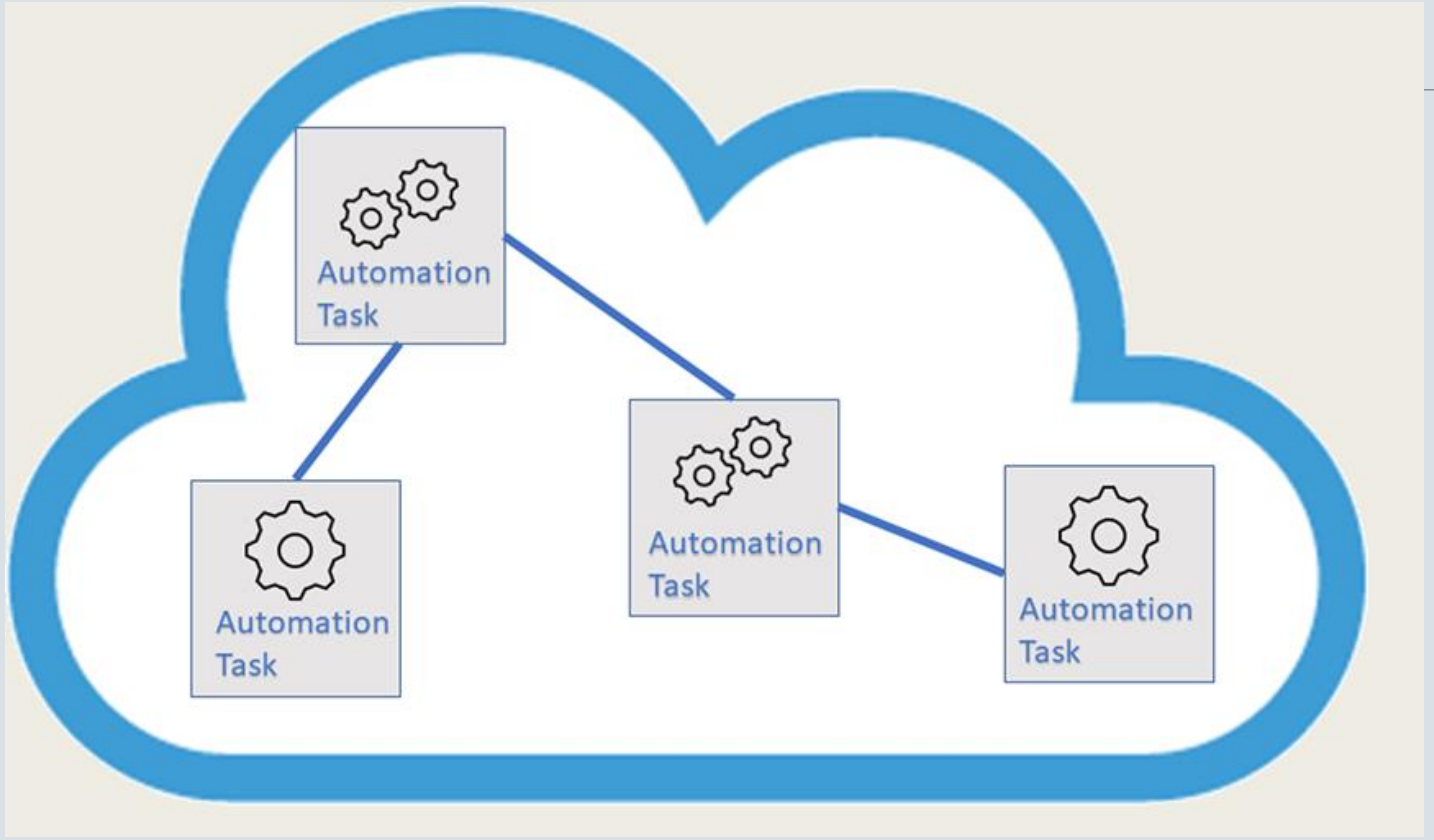
Facilitating tools and techniques are used by cloud vendors to enable implementation and day-to-day operations

Among management features are automation, orchestration, replication, and disaster recovery as a service (DRaaS)

Cloud Automation

Catch-all phrase to describe processes and tools that replace manual tasks related to managing organizational computing infrastructure

An IT revolution occurred to replace the artisan IT worker with machine-driven processes that would be self-documenting, replicable, and understandable



Cloud Orchestration

Orchestration coordinates and organizes multiple automation tasks.

Terms Related to Automation

Automation: Replacing manual tasks with automated solutions.

Automation Task: A specific element of automation. One item.

Orchestration: Organizing automation tasks to ensure the tasks work together.

Automation Tasks and Orchestration

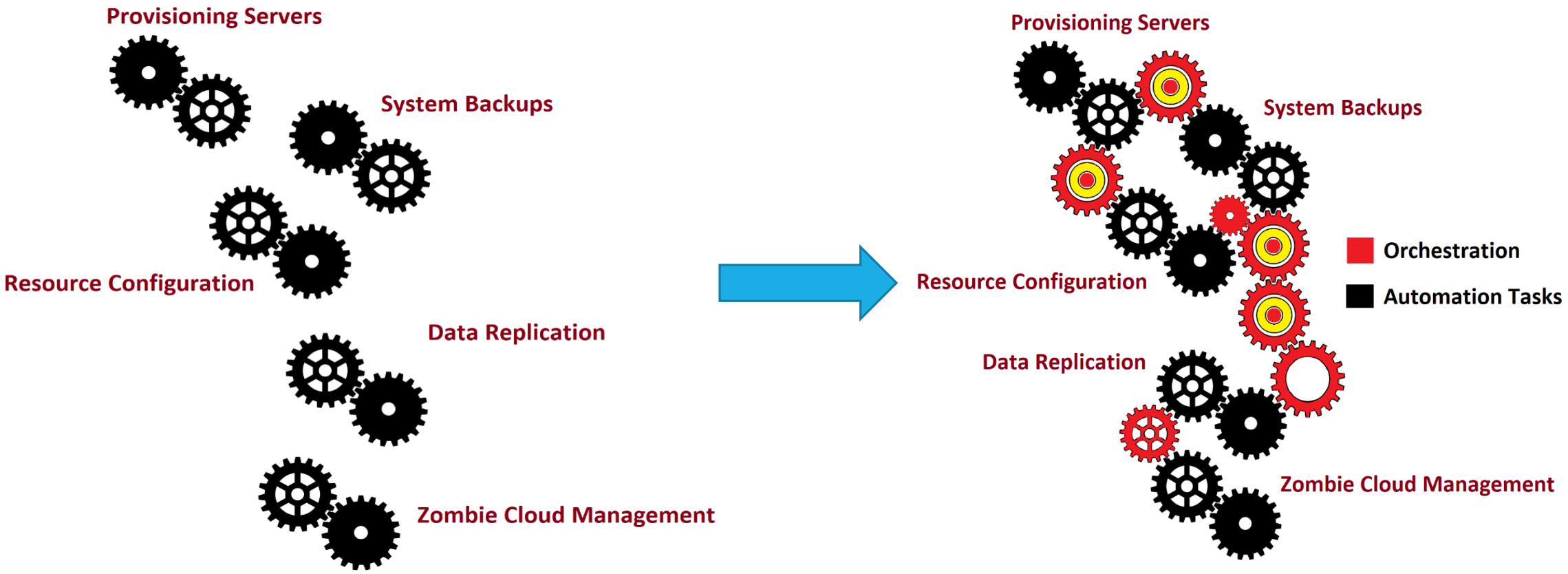
Examples: Spinning up VM, installing desktop image, deploying application.

Use tools ranging from scripts to configuration management tools

Orchestration has a goal of end-to-end task workflow automation

Automation and orchestration have a common goal: reduce manual IT management tasks

Automation to Orchestration



Holy Grail of Automation

‘Holy grail’ of automation and orchestration is completely self-managed cloud environment that intuitively anticipates resource needs and uses AI to keep costs low, utilization high, and user needs balanced

Infrastructure as Code (IaC) uses program scripts to automate mundane tasks but takes to next level

IaC removes issues with documentation and scripting.

Uses higher level programming languages to adaptively modify and reuse code across servers and locations to implement automation

General IaC Approaches

Declarative tools focus on end configuration or the ‘what’ and describe desired state and let system execute steps needed to reach that state. SQL is an example.

Imperative approaches describe the ‘how’. These are procedural steps needed to reach a desired end-state. An example is a programming language such as C# or Java.

Intelligent approaches focus on the ‘why’. Why should the configured system operate a certain way? This approach often implements best practices through automation.

```
Start
    If time=03:00
        Back up all database
    If error_code
        Start recovery
        Notify ITStaff@myBusiness.com
        Bring up backup database
End
```

Example IaC Script

Common IaC Uses

Resource Deployment: Setting up new resources as needed.

Configuration Management: Configuring new resource instances in complex environments.

Updates: Deploying changes to multiple servers.

Provisioning: Changing resource capacity of VMs as needed by ongoing demand.

Load Balancing: Automatically balancing server loads in ways that ensure optimal system usage.

Security Updates: Applying security patches across various resources

Common IaC Uses

Problem Identification: Monitoring ongoing activity and preemptively finding and correcting problems.

Alerts: Sending out alerts based on automated monitoring.

Collecting Information and Monitoring: Tracking key metrics about system use and performance from multiple machines or VMs.

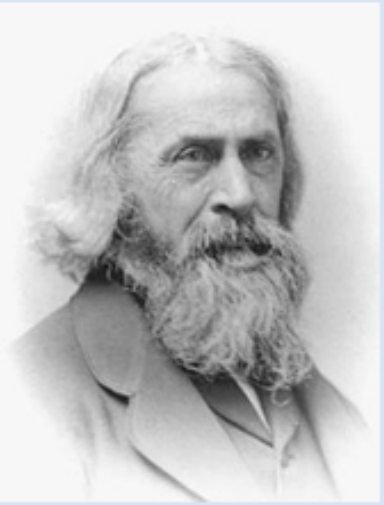
Backups: Ensuring backups occur according to a schedule and automating related tasks.

Rollback: Taking servers back to configurations prior to a selected date or time.

Disaster Recovery: Ensure production servers are continually synced with off-site backups in case of disaster.

Idempotence

Benjamin Peirce, a Harvard mathematician in the 19th century, developed the term *idempotent.* This term originally described algebraic elements that did not change when raised to a positive integer power. In cloud computing, it describes the goal of IaC: ensuring code produces the exact outcome each time it is used. This contrasts with people performing tasks manually in ways that could have slightly different results. So, IaC ensures resources are idempotent.



Professor Peirce

IaC Tools

Configure and automate infrastructure-related tasks, particularly mundane, time-consuming ones

Provide task automation and orchestrate deployment and integration of tasks

Include roll back features which allow infrastructure changes to be undone should unexpected problems arise due to the update

Help stop environment drift in the resource release pipeline

Stops IT specialists from sprinkling in custom features and becoming a snowflake

IaC Approaches

Push Approach: Uses centralized server to push out infrastructure changes and allows administrators to control production environment changes and deployments

Pull Approach: Client resources contact a central server to request updates. Central administrator has less control over configuration of resources because client users have expertise to manage their own systems.

Example IaC Tools

Puppet: Uses a declarative language to describe desired system configuration

Chef: Widely used to setup, configure, deploy, update, manage, and remove servers and applications in various environments

SaltStack: Developed by Thomas S. Hatch to manage large scale infrastructure through a Python-based open-source software system intended to support the IaC approach to enterprise IT resource management

Terraform: IaC tool used to create, change, and update IT resources in cloud environments using a series of code-based configuration files to drive infrastructure deployment

Chef Concepts

recipes	Recipes are at the heart of Chef. They define resource configurations and are used to install, update, and deploy IT infrastructure elements.
cookbook	Cookbooks contain many recipes and define scenarios to automate installation and deployment of resources. Think of cookbooks as orchestration of recipes.
workstation	This is the station where Chef recipes are developed and deployed to the Chef server for use by the nodes.
Chef-server	This is the central controller for the Chef system that serves out recipes to nodes.
node	Any physical, virtual, or cloud machine served by the Chef system.
Chef-client	This is an agent running on every node to assist with Chef-server interaction.
Chef Infra	An automation platform used to transform infrastructure into code.

IaC Solutions by Cloud Vendors

AWS CloudFormation: Configuration orchestration tool developed and released by Amazon that permits IT specialists to codify infrastructure and automate cloud resource deployments using templates based on YAML, JSON, or that are visually developed in a drag and drop environment called AWS CloudFormation Designer. CloudFormation is declarative

Google Cloud Deployment Manager: Facilitates IT shop needs for declaring, configuring, deploying, and managing cloud resources through JSON and YAML to describe deployment rules.

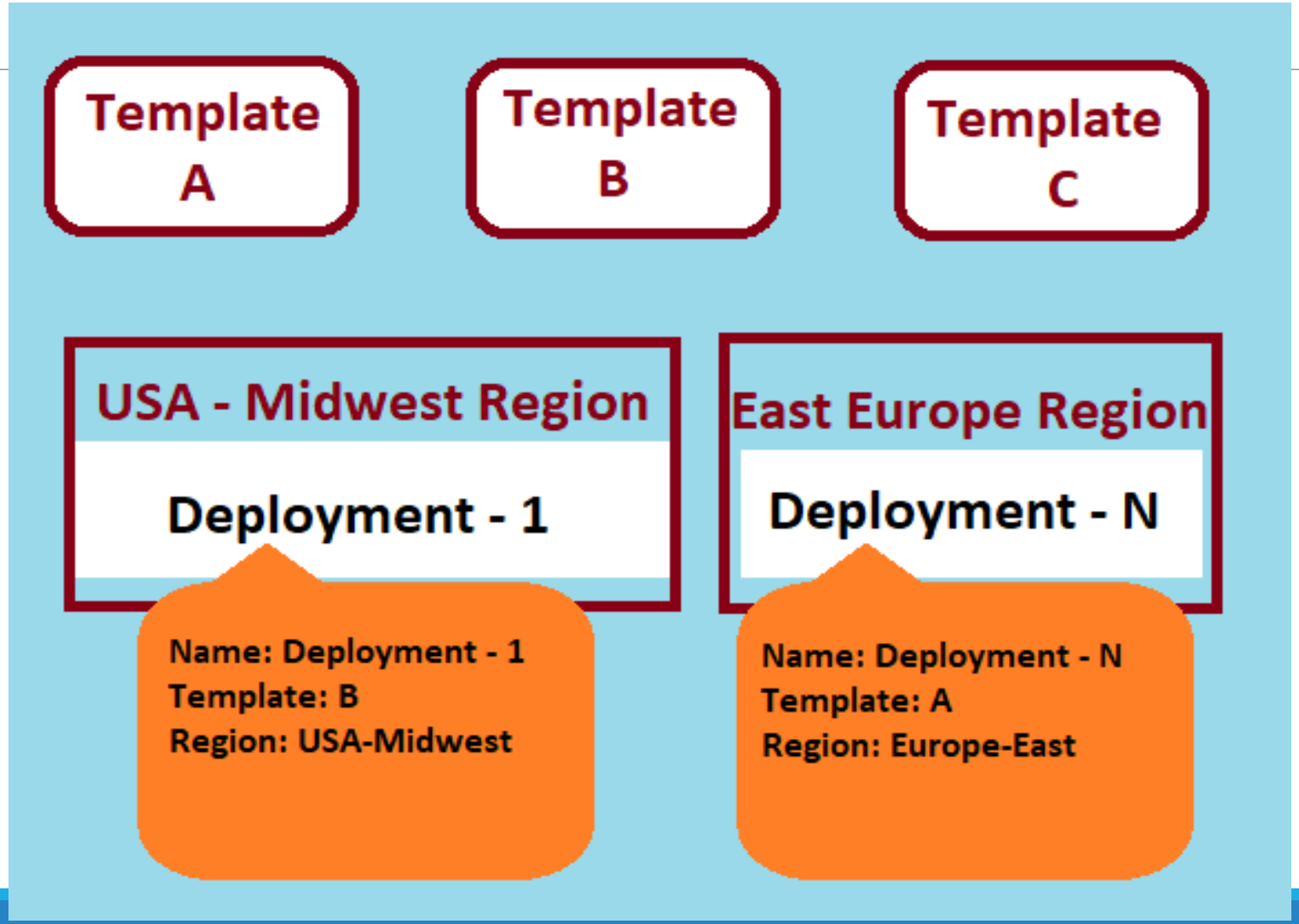
Azure Resource Manager: Microsoft's native infrastructure management tool for Azure cloud which considers complex environment that Azure may be asked to deliver replete with VMs, networks, storage, databases, webservers, microservices, third-party software installations, and so forth.

What is YAML?

YAML, which rhymes with camel, is a markup language with data-oriented use and intention. The acronym YAML originally meant, “Yet Another Markup Language”. Later, the developers suggested its name was recursive and really stood for: YAML Ain't Markup Language. To best define the use of YAML, the inventors, Clark Evans, Ingy döt Net, and Oren Ben-Kiki provide these YAML design goals in decreasing priority[2]:

1. YAML is easily readable by humans.
2. YAML data is portable between programming languages.
3. YAML matches the native data structures of agile languages.
4. YAML has a consistent model to support generic tools.
5. YAML supports one-pass processing.
6. YAML is expressive and extensible.
7. YAML is easy to implement and use.

Templates used to create Deployments with Google's Cloud Deployment Manager



Azure Resource Manager Terminology

Resources: Any item available through Azure such as VMs, virtual networks, Web Apps, databases, web servers, blobs, storage networks, and so forth.

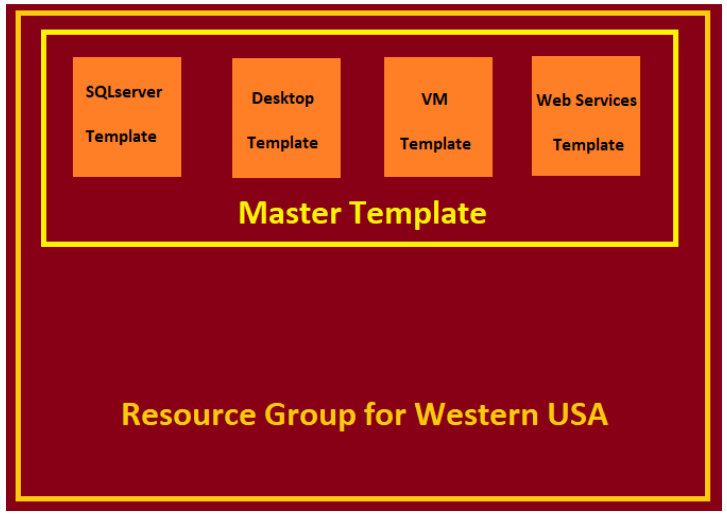
Resource Groups: Represent orchestration through containers that hold related Azure resources for each group or set of Azure solutions.

Resource Providers: Service that helps IT specialists determine what resources can be managed within the software.

Resource Manager Templates: Provides access to JSON template files that describe how resources are deployed for a group or resource.

Template deployment: Allow an IT specialist to determine which template to deploy as a blueprint for an instantiated instance of the resource on Azure.

Microsoft best practice suggests all resources in each group be synced to share the same lifecycle.



Nested templates are often used with Azure Resource Manager

Example JSON Script for Azure Resource Manager

```
"resources":
[
  {
    "apiVersion": "2019-06-15",
    "type": "Microsoft.Compute/VMAccounts",
    "name": "VMAccountWUSA”,
    "location": "westernusa",
    "sku":
    {
      "name": "Standard_VM"
    },
    "kind": "VM","properties":
    {
    }
  }
]
```

Access Control for Resource Management Tools

Allows business to determine which people and groups should have access to actions regarding resources. Most tools integrate role-based access into management platform to ensure resource deployment, changes, updates, provisioning and deprovisioning are done by correct owners.

Role-based access based on two primary concepts: role definitions and role assignments.

- Role definitions describe a permission set which can be hierarchical and inherit definition sets.
- Role assignments attach definitions to users or user groups with predefined roles
- Users are generally assigned to predefined roles.
- Some tools allow setting programmatic access levels to enable scripts to access resources.

Customized Policies

Most resource manager software used for automation and orchestra provides the ability to create customized resource management policies

Cover many areas including resource usage, scope of governance, and naming conventions

May relate to security and ensure checks and balances for user role assignments

Naming conventions may consider region, resource type, user group, and other items

May require resources be tagged with standard identifiers to perform searches and logically track usage

May also specify the business owners of various resources

Default Roles in Azure

Azure Resource Manager provides several default roles regarding resource access and management. These can be grouped into two general categories: platform roles and resource-type roles. The following lists summarize examples of each.

Platform Roles

1. Owner – manages all resources and able to assign access to others.
2. Contributor – manages all resources but may not assign access to others.
3. Reader – cannot change resources but can see everything.
4. User Access Administrator – Can manage resource access settings.

Resource-Type Roles

1. VM Contributor – manages VMs but may not assign access to others.
2. Network Contributor - manages virtual network resources but may not assign access to others.
3. Storage Account Contributor – manages storage accounts but may not assign access to others.
4. Website Contributor - manages websites but may not administrate plans or authorize users.

Microsoft provides more Azure role information at these two sites:

https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles

https://docs.microsoft.com/en-us/azure/role-based-access-control/role-assignments-portal

APIs

Application Programming Interface

Interface that allows one piece of software to talk to another

Translator that takes conventions of one system and transforms it to conventions of another

Types of APIs

SaaS APIs: Best-known use of APIs in application layer. End-user software connected using APIs. Extends the software package into capabilities used in cloud computing. Example is Microsoft's Power BI with sharing visualizations and reports enhanced via APIs.

PaaS APIs: Platform-based APIs provide integration with database services, messaging systems, communication interfaces and other areas.

*IaaS APIs: Re*levant to automation and orchestration services and help control cloud resources and implementations. APIs can control resource distribution and interaction at a high level. Can integrate and link multiple cloud providers.

Example API Market Leaders

Apache CloudStack

Amazon Web Services API

Google Compute Engine

OpenStack API

VMware vCloud API

Web APIs

These have become popular for allowing data from different systems to be accessed and shared via the internet. Many social media systems permit extraction of data based on APIs that limit users to those with an assigned API key. Other systems, for example Twitter, use an API for nearly all user operations. Web APIs really are instruction sets that permit users from the web to access and interact with a system according to predefined use cases.

SDKs

Software development kits

Permit use of software packages or resources on different platforms

Include tools, samples, pre-written code, processes, documentation, guides, and other resources to make implementation easy.

Facilitates recreation of resource in a new space

Many are open source and provided without charge through GitHub

Often created by the software developer

Facilitate cloud deployments

SDKs and APIs

Common for SDKs to include APIs as part of their toolset

If someone creates new software that needs to ‘talk’ to a web service or other resource, an API provides the communication piece

SDK contains building blocks for an application

API is a very important block in the set

SDKs usually include APIs, but APIs do not normally contain SDKs

SDKs aid in application creation while APIs enable communication between resources

Cloud Backup and Replication

Cloud backup stores resources at points in time so systems can be returned to prior states should the need arise.

Replication concerns data protection, integrity, and transaction speeds and occurs because of application migration processes, data warehousing, data mining, and analytics operations to enable more uses of existing data without interfering with critical operations.

Both processes benefit from cloud's economies of scale and elasticity.

Cloud Backup

Insurance against events that can prevent a business from maintaining business continuity (BC)

BC ensures operations remain as normal as possible under all conditions such as catastrophic weather events, equipment failures, cyber security failings, or other terrible things

Can backup software, data, and infrastructure elements due to cloud's capabilities

Types of Cloud Backup

Most vendors market two types of solutions

- (1) backup/recovery of on-premise resources to the cloud
- (2) backup/recovery of cloud-based resources and systems to cloud-based storage

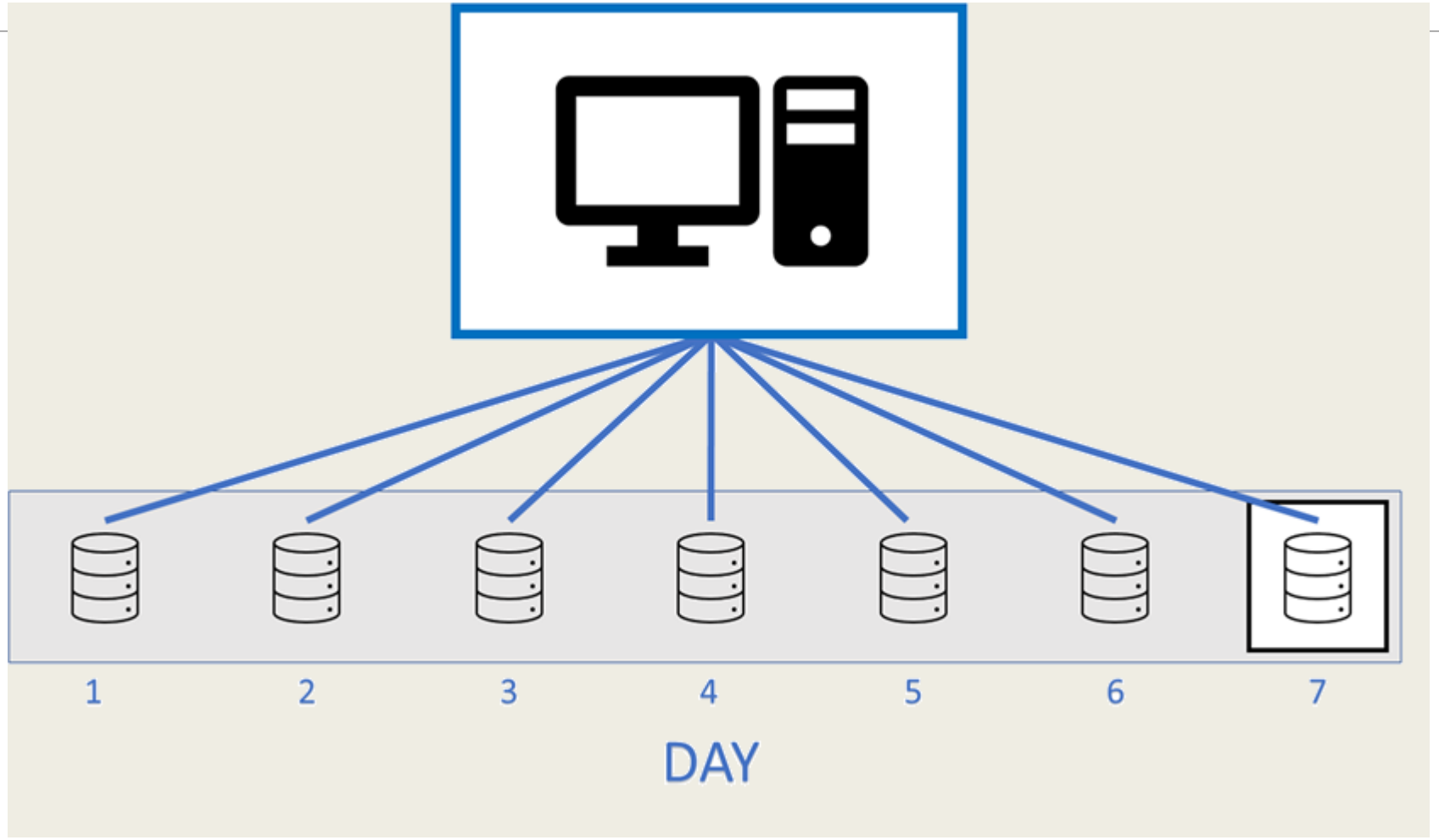
Usually rely on point-in-time triggers driven by business policies

Costs of backup versus the potential costs of loss/recovery time considered in formulating policies.

Policies govern how long to keep backups (influenced by government regulations and laws), how many backup copies to keep, and form of backup

Business rules and organizational needs should govern backup policies

Example backup policy that specifies partial backups for 6 days, then a full back up on day 7



3-2-1 Rule

Example best practice for backups

Suggests an organization:

Should always maintain (3) copies of its data.

Should back up its data on at least two different forms of storage.

Always keeps at least one (1) copy of its data offsite.

Cloud Backup Challenges

If organizational IT infrastructure is located on-premise, backing up to cloud can be slow and take over bandwidth and resources

Laptops or mobile devices present issues trying to complete a backup on a regular schedule

Recovery times impacted by whether system is on-premise system, has massive data amounts, and other factors

Custody of sensitive data

SSL and AES

SSL and AES are two forms of security for data. Both are important when considering cloud backup processes but for different reasons. First, SSL, short for Secure Socket Layer, focuses on encryption during transmission. Essentially this ensures data cannot be deciphered if a wire is tapped or if data is intercepted wirelessly. SSL relies on a technology where the sender and receiver have exchanged a unique session key required to decrypt the data. AES, short for Advanced Encryption Standard, is technology developed for securing secret government documents. Often, storage vendors use this technology so neither they nor any intruders can access client data. This eliminates the possibility of insider as well as outside hackers. AES is not meant for use during transmission. The encryption keys are in possession of the data owner and are never transmitted. The data user must have an encryption key to decrypt and use the data. AES security can be used by the client before sending data to the cloud vendor or it can be done on the vendor side. Of course, if a client encrypts their own data, chances of compromise are reduced even further. If the cloud storage vendor never possesses the encryption key, the data always remains fully secure. This stops the possibility that an employee of the cloud storage company will steal the encryption key and access client data.

Example Cloud Vendors	Overview	Comments
Arcserve UDP Cloud Direct	Useful with all major business platforms	A fully featured business solution vendor for recovery, backup, and BC. Developed for Microsoft environments. No bare metal recovery. Protects against ransomware.
Box for Business	Reliable, widely used vendor	Offers intuitive interface with easy access and many features. Supports many environments and offers read-only options and varying levels of access.
Carbonite	Good small business solution	Useful to recover networks or computers that have become disabled. Widely used.
Dropbox Business	Reliable, widely used vendor	Integrates with many software vendors and provides non-intrusive synchronization.
Egnyte	Focuses on solutions utilizing a combination of local and cloud storage	Solutions for secure data from local storage. On-premise servers add security.
iDrive	Robust small business solution	Provides excellent interface and good encryption. Lacks some software support such as Office 365 email.
MSP360	Customizable backup provider	Writes images to Amazon Cloud or Azure. Lacks some management features found in competing packages.
SpiderOak	Collaboration tool with online hosting and cloud storage services	‘Zero knowledge’ privacy environment---client is the only one who can view all stored data.
Tresorit	Secure cloud-storage with 2-party authentication	‘Zero knowledge’ privacy environment---client is the only one who can view all stored data.

Cloud Replication

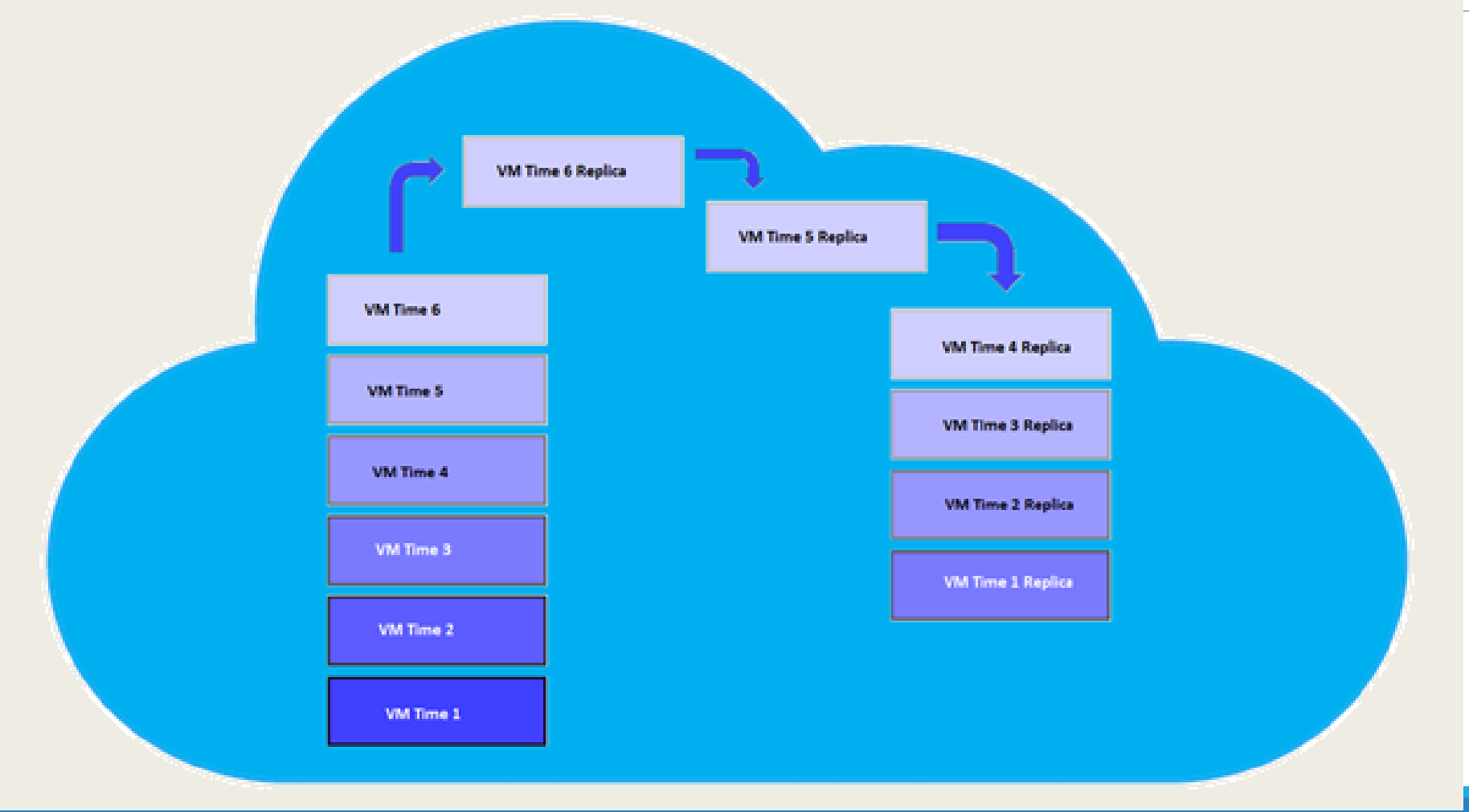
Generally part of IaaS functionality

Backup helps ensure the long-term safety of organizational data and enables compliance with governmental regulations but may not be well suited to BC and disaster recovery needs

Traditional backup can be slow to restore and does not always include the infrastructure side of an IT site

Replication enables a faster restart

Ongoing replication of a cloud-based system seeks to reduce latency and ensure rapid recovery times



Cloud Replication Goals

Designed to protect IT infrastructure and data against a catastrophic event or disaster by continually replicating (this means creating an exact copy) data and/or VMs, servers, and other virtual resources in diverse locations. If a failure occurs, a working version of the system can be started in the new location quickly. Time to restart is termed system latency.

Qlik Replicate

Qlik Replicate (formerly Attunity) is an example platform that focuses on replication. Its unique approach is to provide any-to-any services that permit users to backup any database system and other resources to any replicated platform. Among their features are:

- Universal solution for any type of source: relational database, file systems, mainframes, PCs, mobile devices, SAP, data warehouse, or Hadoop – to any cloud platforms including Amazon, Azure, and Google.
- Fast transfers of data using WAN optimizations.
- High levels of security with data protected by AES and other integrity checks.
- Monitor and user console to control and observe progress and status of tasks.

Disaster Recovery as a Service

Vendors now market Disaster Recovery as a Service (DRaaS)

Elastic and uses cloud resources

Virtual machines, data, containers, and so forth all have a virtual footprint that can be duplicated and put into motion very quickly

Burden is removed from an on-premise IT group as is the expense of having backup hardware/software waiting in case a problem occurs

Leading cloud providers have explored the DRaaS space and many offer products in this area

What should DRaaS include?

A good DRaaS implementation needs to consider a few important aspects. These include:

- What platforms does the DRaaS product work with? Considerations for mission critical systems, data, platforms, and infrastructure are particularly important.
- Can the DRaaS vendor manage all organizational data storage needs?
- What processes are used to back up organizational systems? How often does this occur? Is it disruptive? And is data synchronized to backed up on an interval basis?
- Is a local backup being created or is all done to the cloud?
- Does the system interact specifically with applications that may be difficult to replicate?
- How long does system restoration take?
- How long has the vendor been in business and who are among their clients?
- What type of recovery testing is recommended?
- Will the backed-up version of the system perform as well?
- How long will the vendor host the recovered system and what costs are involved with the hosting?

Chapter 5 Summary

Software and hardware solutions permit organizations to automate many routine functions and then link these automations together to orchestrate a reproducible IT architecture and platform.

Good automation can be integrated into an overall orchestration approach to ensure platform stability and that organizational knowledge is captured and not reliant on one expert individual.

One method for ensuring stability is IaC. This means using a script-based approach to implementation of infrastructure to ensure all processes are documented and are idempotent (e.g. can be exactly recreated). Vendors provide toolsets such as Chef and Puppet to make this easier and more manageable by IT specialists.

Another management concern is rapid recovery should a system become compromised. Business continuity planning includes disaster recovery elements for IT infrastructure.

Backups of data, software, and application configurations are common but often take longer to reinstall or implement should something bad occur.

Replicated system are updated as the production system runs and usually can be implemented in much shorter time frames.

Goal for disaster recovery is that an alternate system can be immediately started if needed.

Many cloud vendors offer DRaaS to ensure nearly seamless operation should an unexpected event occur.