

- Analytics and Business Intelligence
- Tools provided as a service with PaaS allow organizations to analyze and mine their data finding trends and patterns and predict outcomes to improve forecasting, product demand decisions, investment returns and other business decisions.

Software as a Service

- Using ready a fully developed app such as: Email, financial system, messaging apps, etc...
- less flexible but easier to use.

Scenarios

- Email and messaging
- Business productivity apps.
- Finance and expense tracking.

Review Azure Resource Manager Benefits

- The infrastructure is made by many components: VM, storage, VN, web app database, database servers and third-party components/services.
 - These components are not independent, they are related and interdependent parts of a single utility. You want to deploy, manage and monitor them as a group.
- Azure Resource Manager enables you to work with the resources in your solution as a group
- You can deploy, update or delete all the resources for your solution in a single coordinated operation.
- You use a template for deployment and that template can work for different environments such as testing, staging and production. Azure Resource Manager provides security, auditing and tracing.

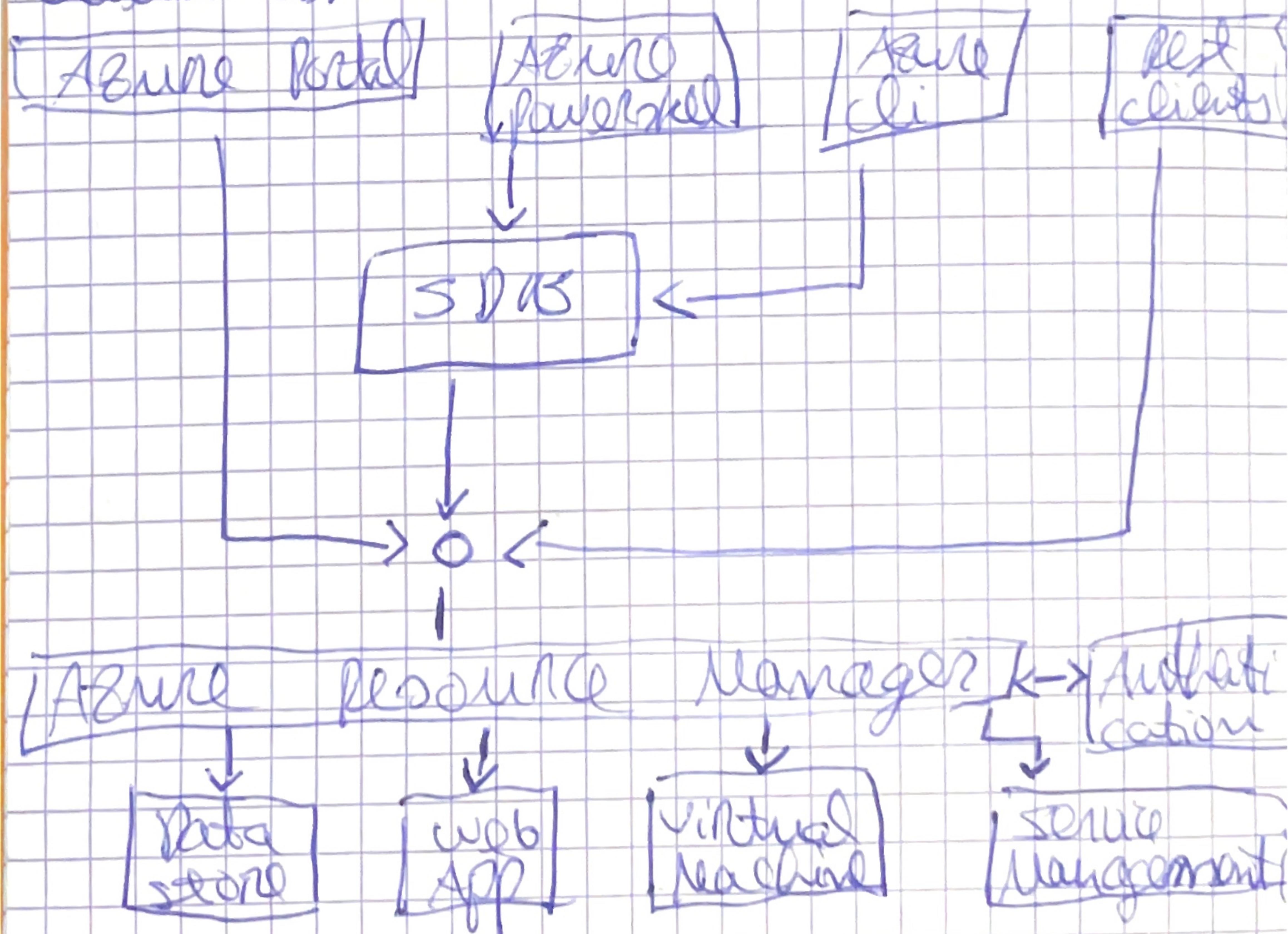
Features to help you manage your resources after deployment.

consistent Management Layer

- Azure Resource Manager provides a consistent management layer to perform tasks through Azure PowerShell, Azure CLI, Azure portal, REST API, and client SDK's. Use the tools and APIs that work best for you.

consistent Management Layer

- Azure Resource Manager provides a consistent management layer to perform tasks through PowerShell, Azure CLI, Azure portal, REST API and client SDK's.



~~Benefits~~ Benefits

AZURE RESOURCE Manager provides several benefits:

- Reuse, manage and monitor all the solution resources for your application as a group, rather than individually.
- Repeatedly deploy your solutions through the development lifecycle and have confidence your resources are deployed in a consistent state.
- Manage your infrastructure through declarative templates rather than scripts.
- You can define dependencies between resources so they're displayed in the correct order.
- You can apply access control to all services in your resource group because Role-Based Access Control (RBAC) is natively integrated into the management platform.
- Apply tags to resources to logically organize all the resources in your subscription.
- ↓ You can clarify your organization billing by using costs for a group of resources sharing the same tag.

Guidance

- Suggestions to take full advantage of Azure Resource Manager:
 - Define and deploy your infrastructure through the declarative syntax in Azure Resource Manager templates, rather than using imperative commands.
 - - Define all deployment and configuration steps in the template. You should have no manual steps for setting up a

your solution.

- Run imperative commands to manage your resources such as to start or stop an app on machine.
- Manage resources with the same lifecycle in a resource group. Use tags for all other organizing of resources.

Review Azure Resource Terminology

- Resource

- Resource available through Azure (VM, storage account, web app, database, virtual network and more).

- Resource Group

- Container that holds related resources for a Azure solution.

- This group can contain all the resources of a given solution or only some of those that we wish to manage as a group.

I recommend groups according to what makes sense for your organization.

- Resource provider

- Service that supplies the resources you can deploy and manage through resource manager.

- Each resource provider offers operations for working with the resources that were deployed.

- Common resources are storage account, Microsoft web, providers are: Microsoft Compute (virtual machines), Microsoft Storage (storage account resources), Microsoft Web (Web apps).

- Template

- JSON object that defines one or more resources to deploy to a resource group. It defines the dependent

ties between the deployed resources

- we can use the template to deploy the resources consistently and repeatedly.

Declarative Syntax

- syntax that lets you state what we want and where it is instead of writing a sequence of programming commands to reach it.

Resource Providers

- they are written in file format
resource providers / resource types.

Ex: Microsoft.KeyVaultVaults

- before deploying resources we should firstly check the existing ones and their valid locations and versions.

Create Resource Groups

- we can deploy in new or existing groups.

- we can see the lifecycle of it and in case it fails we get the output.

- deployments are incremental (if there is something there already then you will increment to it).

Considerations

- resources can only exist in one resource group.

- resource groups cannot be renamed.

- resource groups can have resources of many different types (services).

- resource groups can have resources from many different regions.

Creating Resource Groups

- The resources in the same group share the same life cycle and share the same update and delete
- You deploy, update and delete them all together!
- Each resource can only exist in one resource group.
- You can add or remove a resource to a resource group at any time.
- You can move resources between groups but there are some limitations.
- Different region resources can be in the same group.
- A resource group can be used to scope access control for administrative actions.
- A resource can interact with resources in other resource groups.
- This happens when having related resources but with different lifecycles.
- It's important to take into consideration the region where you are applying the resource because different regions mean different policies.

Using Azure Resource Manager Locks

- Deleting a resource without wanting it is a huge concern so Azure created the locks to make it harder to happen.
- We can lock subscriptions, resource groups and resources.
- Locking parents will lock the children.

Lock Types

- Read-only
- Prevent changes to the resource.
- Delete access
- Prevent deletes

Only the owner or user admin can create/delete locks.

Organize Azure Resources

- When we are moving resources to a new group or to a new subscription the source and the target get locked (this does not mean they get unlinked they are just locked for changes).
- We should read the docs for more operations to know what all the limitations.

Implementation

- To move we should select a group click in move and then select the opt and select the resource to move.
- You can still move components that have limitations so be carefull.

Previous Resources and Resource Groups

- think twice when deleting groups because it deletes all resources within and there can be dependencies of another group.
- you can delete using PowerShell.
- you can delete by moving.

Determining Resource Limits

- Azure lets us see usage and limits.

- we can request increases on a given resource.
- all resources increases have limits
- these limits cannot be increased.