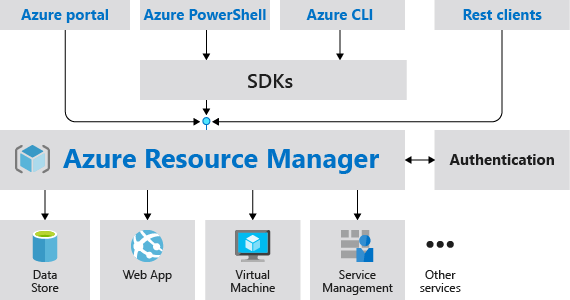
**Azure Resource Manager**

* The infrastructure for your application is typically made up of many components – maybe a virtual machine, storage account, and virtual network, or a web app, database, database server, and third-party services.
* These components are not separate entities, instead they are related and interdependent parts of a single entity. 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.
* Azure Resource Manager provides security, auditing, and tagging features to help you manage your resources after deployment.
* Azure Resource Manager provides a consistent management layer to perform tasks through Azure PowerShell, Azure CLI, Azure portal, REST API, and client SDKs.



## Benefits

Azure Resource Manager provides several benefits:

* You can deploy, manage, and monitor all the resources for your solution as a group, rather than handling these resources individually.
* You can repeatedly deploy your solution throughout the development lifecycle and have confidence your resources are deployed in a consistent state.
* You can manage your infrastructure through declarative templates rather than scripts.
* You can define the dependencies between resources so they're deployed 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.
* You can apply tags to resources to logically organize all the resources in your subscription.
* You can clarify your organization's billing by viewing costs for a group of resources sharing the same tag.

# Azure resource terminology

* **resource** - A manageable item that is available through Azure. Some common resources are a virtual machine, storage account, web app, database, and virtual network, but there are many more.
* **resource group** - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization.
* **resource provider** - A service that supplies the resources you can deploy and manage through Resource Manager. Each resource provider offers operations for working with the resources that are deployed. Some common resource providers are Microsoft.Compute, which supplies the virtual machine resource, Microsoft.Storage, which supplies the storage account resource, and Microsoft.Web, which supplies resources related to web apps.
* **template** - A JavaScript Object Notation (JSON) file that defines one or more resources to deploy to a resource group. It also defines the dependencies between the deployed resources. The template can be used to deploy the resources consistently and repeatedly.
* **declarative syntax** - Syntax that lets you state "Here is what I intend to create" without having to write the sequence of programming commands to create it. The Resource Manager template is an example of declarative syntax. In the file, you define the properties for the infrastructure to deploy to Azure.

## Resource providers

Each resource provider offers a set of resources and operations for working with an Azure service. For example, if you want to store keys and secrets, you work with the **Microsoft.KeyVault** resource provider. This resource provider offers a resource type called vaults for creating the key vault.

The name of a resource type is in the format: **{resource-provider}/{resource-type}**. For example, the key vault type is **Microsoft.KeyVault/vaults**.