

AZ-305T00A

Designing Microsoft
Azure Infrastructure
Solutions

Design a governance solution

<https://learn.microsoft.com/training/modules/design-governance/>

Learning Objectives

- Design for governance
- Design for management groups
- Design for Azure subscriptions
- Design for resource groups
- Design for resource tagging
- Design for Azure Policy and RBAC
- Design for Azure Landing Zones
- Case study
- Learning recap

AZ-305: Design Identity, Governance, and Monitoring Solutions (25-30%)

Design Governance

- Recommend a structure for management groups, subscriptions, and resource groups, and a strategy for resource tagging
- Recommend a solution for managing compliance
- Recommend a solution for identity governance

Design for governance

Why is governance important? -

The term **governance** describes the general process of **establishing rules and policies**. Governance ensures those rules and policies are **enforced**.

A good governance strategy helps you **maintain control over the applications** and resources that you manage in the cloud. Maintaining control over your environment **ensures that you stay compliant with**:

- Industry standards, such as information security management.
- Corporate or organizational standards, such as ensuring that network data is encrypted.

Governance is most beneficial when you have:

- Multiple engineering teams working in Azure.
- Multiple subscriptions to manage.
- Regulatory requirements that must be enforced.
- Standards that must be followed for all cloud resources.



How to govern the cloud?

Cloud governance is a continuous process. It requires ongoing monitoring, evaluation, and adjustments to adapt to evolving technologies, risks, and compliance requirements. The CAF(Cloud Adoption Framework) Govern methodology divides cloud governance into five steps.

Complete all five steps to establish cloud governance and regularly iterate on steps 2-5 to maintain cloud governance over time:

1.Build a governance team: Select a team of individuals to be responsible for cloud governance. The cloud governance team defines and maintains cloud governance policies while reporting on the overall progress of cloud governance.

2.Assess cloud risks: Evaluate and prioritize potential risks associated with the use of the cloud. The risk assessment should identify risks unique to your organization. Consider all categories of risk, such as regulatory compliance, security, operations, cost, data, resource management, and AI risks. Use Azure tools to help [assess cloud risks](#).

3.Document cloud governance policies: Define the cloud governance policies that dictate the acceptable use of the cloud. These cloud governance policies set out the rules and guidelines for cloud usage to minimize the identified risks.

4.Enforce cloud governance policies: Enforce compliance with the cloud governance policies using automated tools or manual procedures. The goal is to ensure that the use of cloud services is in line with the established cloud governance policies. Use Azure tools to help [enforce cloud governance policies](#).

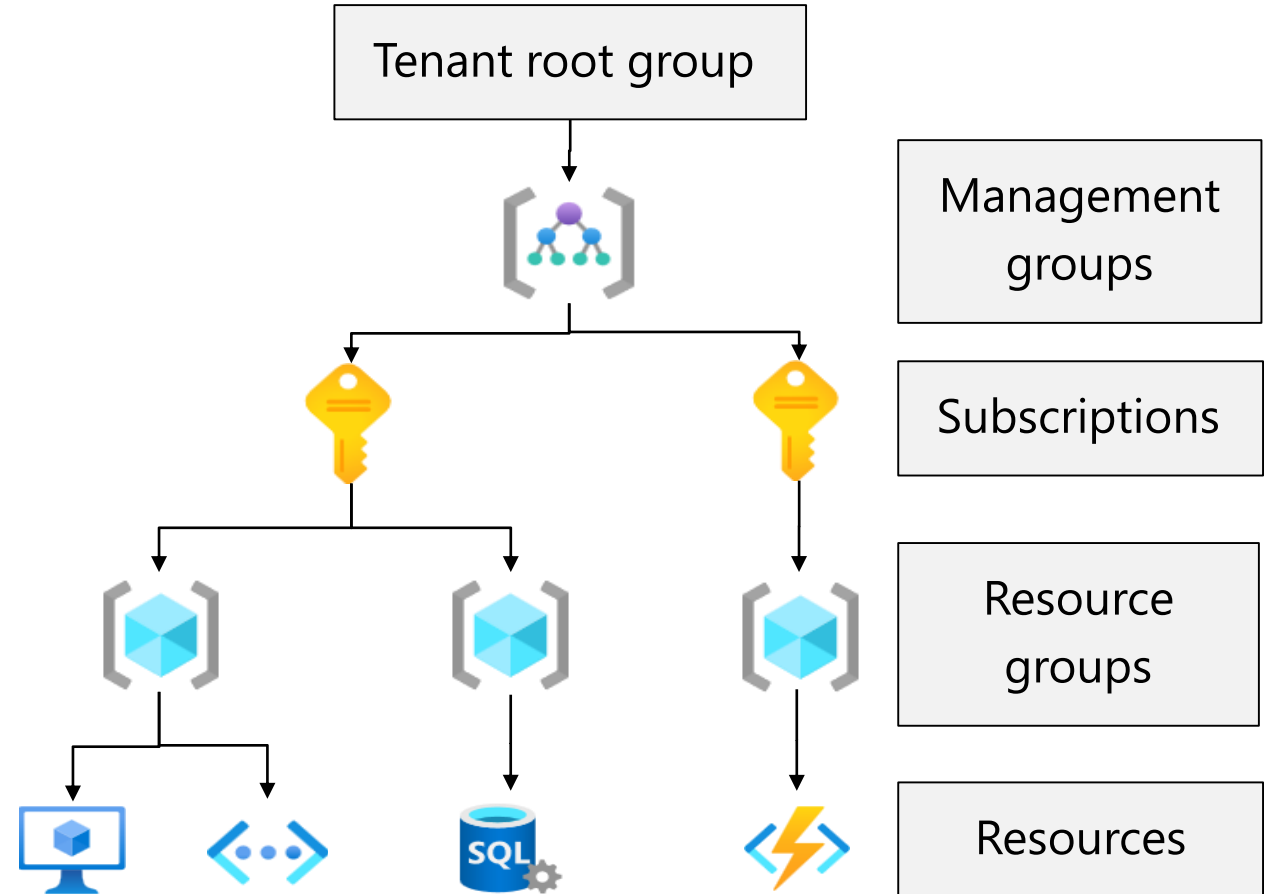
5.Monitor cloud governance: Monitor cloud use and teams responsible for governance to ensure they're compliant with the cloud governance policies. Use Azure tools to help [monitor cloud governance](#) and [set up alerts for noncompliance](#).

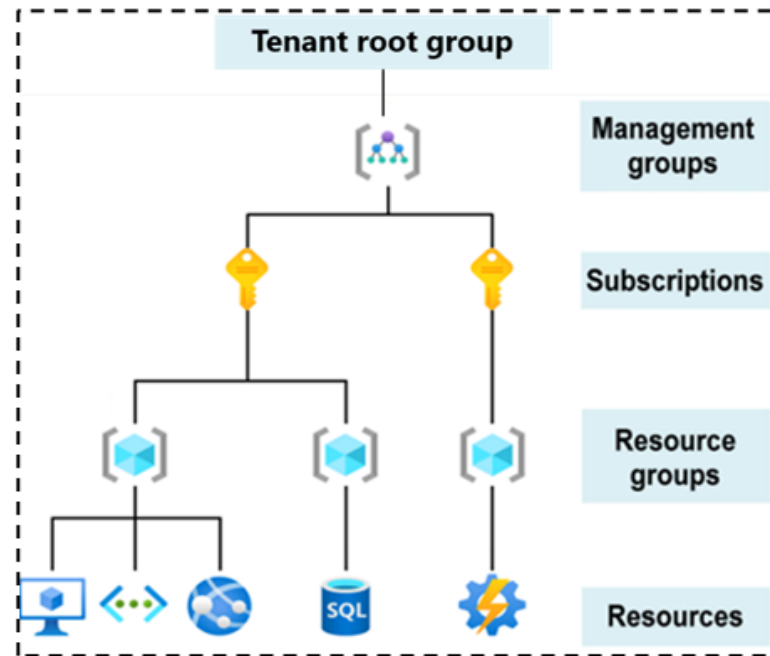
Govern resources in Azure

Governance provides mechanisms and processes to maintain control over your applications and resources in Azure.

- Determine your requirements, plan your initiatives, and set strategic priorities
- Plan for governance at every level
 - Management groups
 - Subscriptions
 - Resource groups
 - Resources

To effectively apply your governance strategies, you must first create a hierarchical structure for your organizational environment. This structure lets you apply governance strategies exactly where they're needed. The governance strategies we cover in this module are Azure policy and resource tags.





Governance strategies

- Azure policies
- Resource tags

A typical Azure hierarchy has four levels: management groups, subscriptions, resource groups, and resources. We examine the details of these levels later in this module.

- **Management groups** help you manage access, policy, and compliance for multiple subscriptions.
- **Subscriptions** are logical containers that serve as units of management and scale. Subscriptions are also billing boundaries.
- **Resource groups** are logical containers into which Azure resources are deployed and managed.
- **Resources** are instances of services that you create. For example, virtual machines, storage, and SQL databases.

ⓘ Note

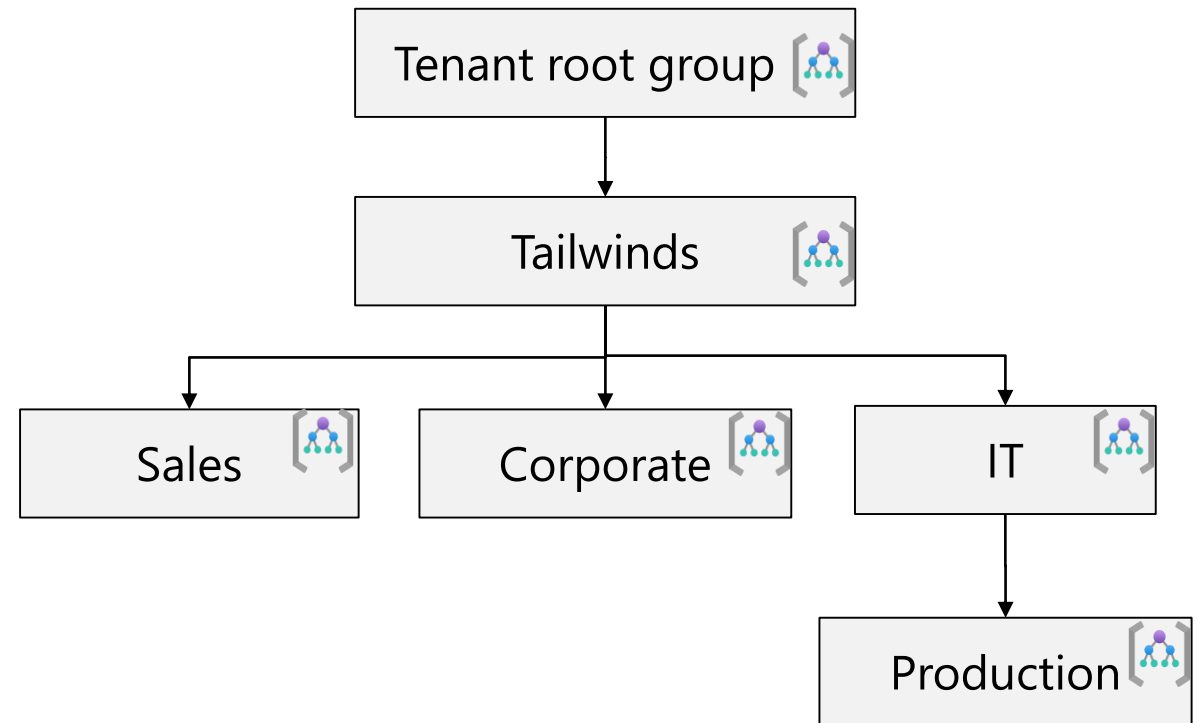
The **tenant root group** contains all the management groups and subscriptions. This group allows global policies and Azure role assignments to be applied at the directory level.

Design for management
groups

Plan your management groups

Management groups manage access, policy, and compliance for multiple subscriptions.

- Keep the management group hierarchy reasonably flat
- Consider a top-level management group
- Consider an organizational or departmental structure
- Consider a geographical structure
- Consider a production management group
- Consider a sandbox management group
- Consider isolating sensitive information in a separate management group



Design for management groups

3 minutes

Management groups are containers that help you manage access, policy, and compliance across **multiple subscriptions**.

You can use management groups to:

- Limit the regions where virtual machines can be created, across subscriptions.
- Provide user access to multiple subscriptions by creating one role assignment that's inherited by other subscriptions.
- Monitor and audit role and policy assignments, across subscriptions.

Things to know about management groups

As you plan the governance strategy for Tailwind Traders, consider these characteristics of management groups:

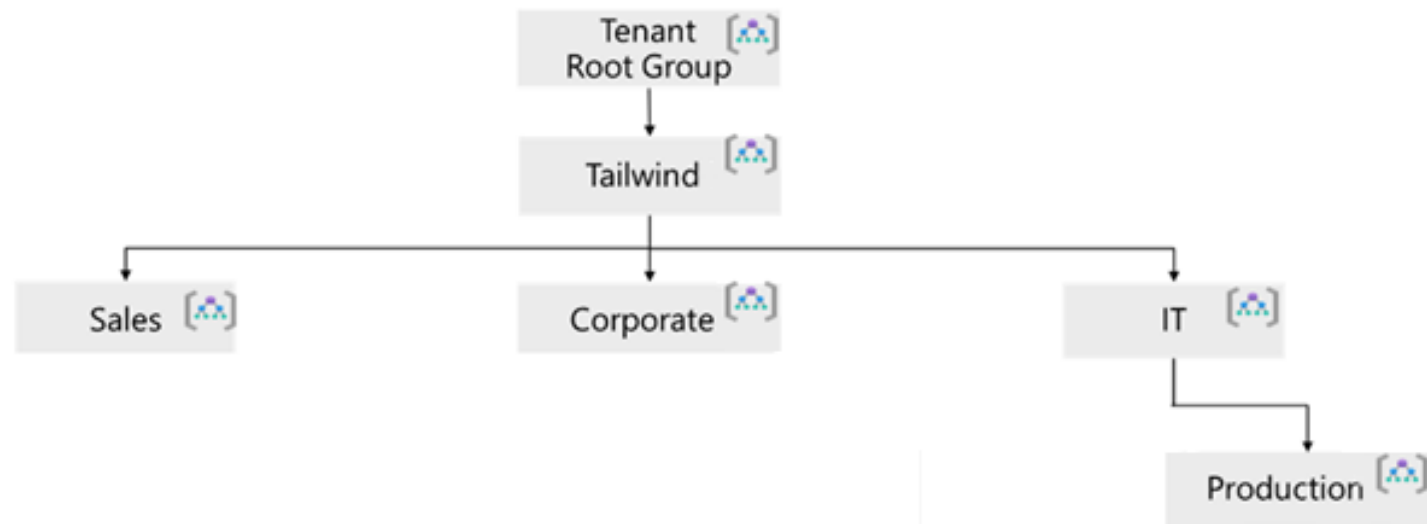
- Management groups can be used to aggregate policy and initiative assignments via Azure Policy.
- A management group tree can support up to **six levels of depth**. This limit doesn't include the tenant root level or the subscription level.
- Azure role-based access control authorization for management group operations isn't enabled by default.
- By default, all new subscriptions are placed under the root management group.



Things to consider when creating management groups

Tailwind Traders has Sales, Corporate, and Information Technology (IT) departments. The Sales department manages offices in the West and in the East. The Corporate main office includes Human Resources (HR) and Legal. The IT department handles research, development, and production. There are currently two applications hosted in Azure.

Here's a proposed management group hierarchy for your organization:


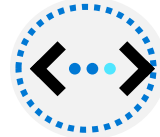






- **Design management groups with governance in mind.** Use Azure policies at the management group level for all workloads that require the same security, compliance, connectivity, and feature settings.
- **Keep the management group hierarchy reasonably flat.** Plan the Tailwind Traders hierarchy to have no more than three or four levels of management groups. A hierarchy that's too flat doesn't provide flexibility and complexity for large organizations. A hierarchy with too many levels can be difficult to manage.
- **Consider a top-level management group.** Implement a top-level management group to support common platform policy and Azure role assignments across the entire organization. A Tailwind Traders management group can be a top-level management group for all organizational-wide policies.
- **Consider an organizational or departmental structure.** Design your management groups based on the organizational structure, to make it easy to understand. Separate the management groups for each Tailwind Traders department like Sales, Corporate, and IT.
- **Consider a geographical structure.** Build your management groups by using a geographical structure to allow for compliance policies in different regions. Allocate unique management groups for governance in the West and East sales regions for Tailwind Traders.
- **Consider a production management group.** Institute a production management group to create policies that apply to all corporate products. A production management group for Tailwind Traders can provide product-specific policies for corporate applications.
- **Consider a sandbox management group.** Offer a sandbox management group for users to experiment with Azure. The sandbox provides isolation from your development, test, and production environments. Users can experiment with resources that might not yet be allowed in official Tailwind Traders production environments.
- **Consider isolating sensitive information in a separate management group.** Secure sensitive data by using a corporate management group for Tailwind Traders. The separate management group provides both standard and enhanced compliance policies for the main office.

Design for Azure subscriptions

Designing for multiple subscriptions

Azure subscription are logical containers for management and billing.

-  Align your subscriptions with business needs and priorities – consider billing and cost reporting
-  Consider subscription scale limits – specialized workloads, IoT, SAP
-  Consider administrative management – centralized or decentralized
-  Consider a dedicated shared services subscription – common services everyone shares
-  Group subscriptions together under management groups – apply common policies and role assignments.
-  Make subscription owners aware of their roles and responsibilities



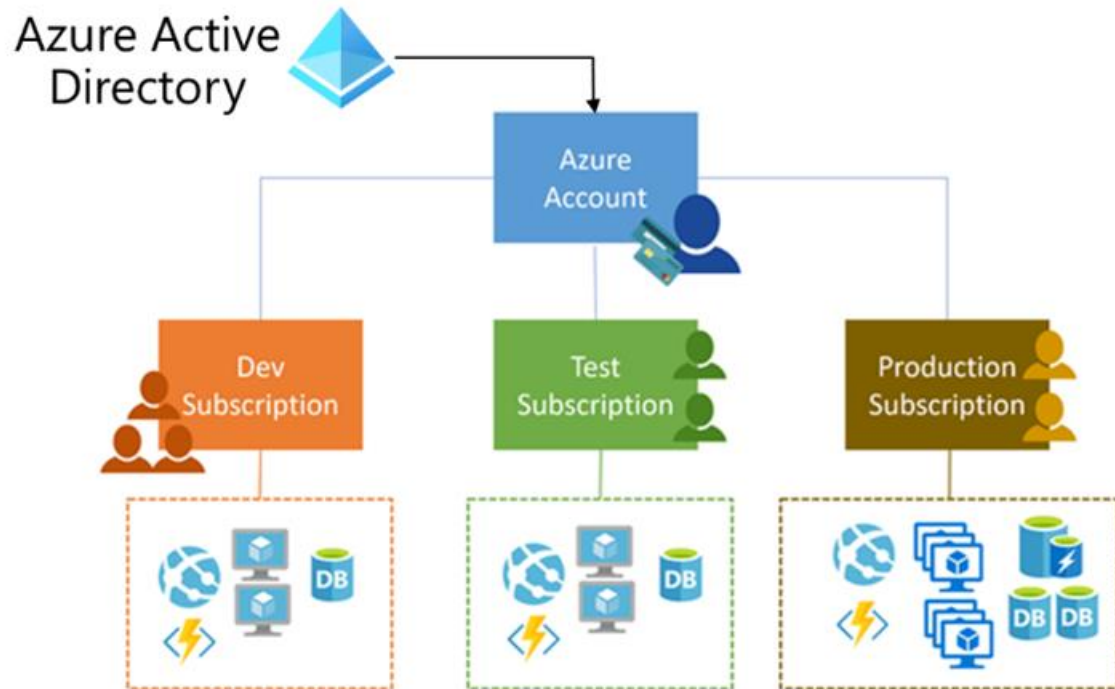
Design for subscriptions

3 minutes

Azure Subscriptions are logical containers that serve as units of management and scale and billing boundaries. Limits and quotas can be applied, and each organization can use subscriptions to manage costs and resources by group.

Things to know about subscriptions

To use Azure, you must have an Azure subscription. A subscription provides you with a logical container to create and pay for Azure products and services. There are [several types of subscriptions](#), such as Enterprise Agreement and Pay-as-You-Go.



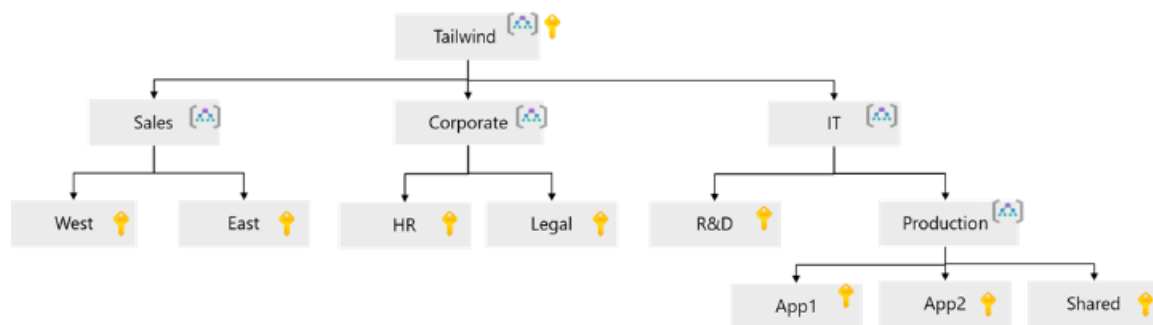
As you plan the governance strategy for Tailwind Traders, consider these characteristics of subscriptions:

- Subscriptions can provide separate billing environments, such as development, test, and production.
- Policies for individual subscriptions can help satisfy different compliance standards.
- You can organize specialized workloads to scale beyond the limits of an existing subscription.
- By using subscriptions, you can manage and track costs for your organizational structure.



Things to consider when creating subscriptions

You defined your strategy for the Tailwind Traders management group structure. Now you need to determine where to assign subscriptions. Here's one possible solution:



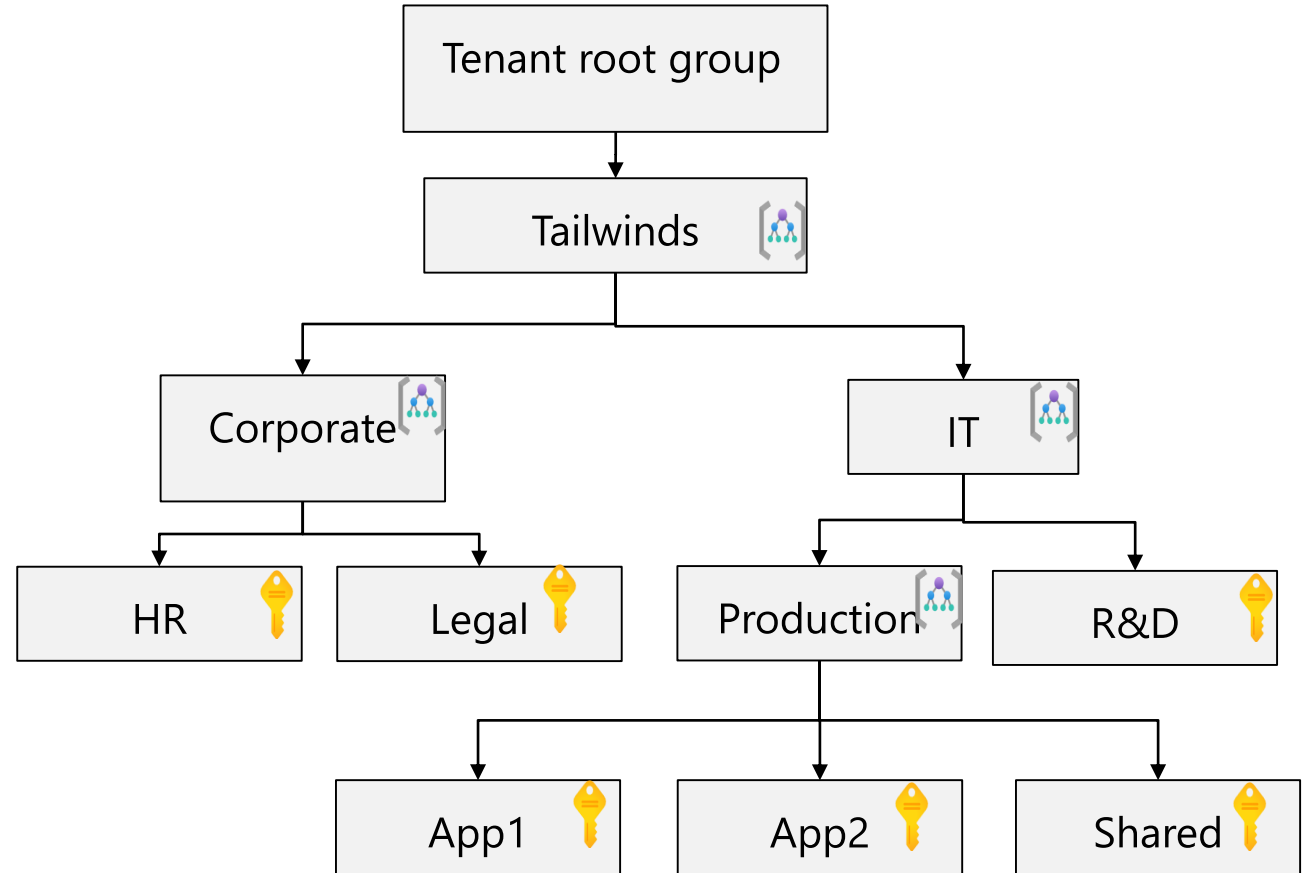
- **Treat subscriptions as a democratized unit of management.** Align your subscriptions to meet specific Tailwind Traders business needs and priorities.
- **Group subscriptions together under management groups.** Group together subscriptions that have the same set of policies and Azure role assignments to inherit these settings from the same management group. For Tailwind Traders, both the West and East subscriptions can inherit policy settings from the Sales management group.
- **Consider a dedicated shared services subscription.** Use a shared services subscription to ensure all common network resources are billed together and isolated from other workloads. Examples of shared services subscriptions include Azure ExpressRoute and Virtual WAN.
- **Consider subscription scale limits.** Subscriptions serve as a scale unit for component workloads. Large, specialized workloads like high-performance computing, IoT, and SAP are all better suited to use separate subscriptions. By having separate subscriptions for different Tailwind Traders groups or tasks, you can avoid [resource limits](#) (such as a limit of 50 Azure Data Factory integrations).
- **Consider administrative management.** Subscriptions provide a management boundary, which allows for a clear separation of concerns. Will each subscription for Tailwind Traders need a separate administrator, or can you combine subscriptions? The Corporate management group could have a single subscription for both the HR and Legal departments.
- **Consider how to assign Azure policies.** Both management groups and subscriptions serve as a boundary for the assignment of Azure policies. Workloads like those for the Payment Card Industry (PCI) typically require extra policies to achieve compliance. Rather than using a management group to group workloads that require PCI compliance, you can achieve the same isolation with a subscription. These types of decisions ensure you don't have too many Tailwind Traders management groups with only a few subscriptions.
- **Consider network topologies.** Virtual networks can't be shared across subscriptions. Resources can connect across subscriptions with different technologies, such as virtual network peering or Virtual Private Networks (VPNs). Consider which Tailwind Traders workloads must communicate with each other when you decide if a new subscription is required.
- **Consider making subscription owners aware of their roles and responsibilities.** Conduct a quarterly or biannual access review by using Microsoft Entra Privileged Identity Management. Access reviews ensure privileges don't proliferate as users move within the Tailwind Traders customer organization.

Note

When it comes to subscriptions, one size doesn't fit all. A solution that works for one business unit might not be suitable for another. Explore your options.

When to use subscriptions - example

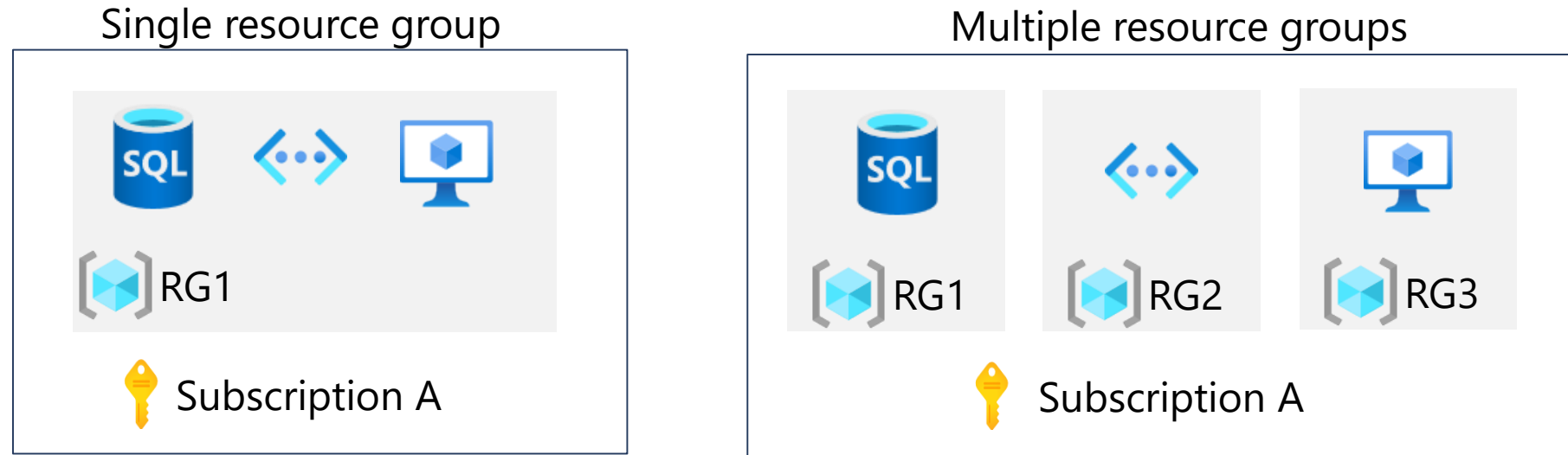
- Secure workloads that require additional policies and role-based access control to achieve compliance
- Specialized workloads and the need to scale outside the subscription limits
- Manage and track costs for your organizational structure
- Identify different environments such as development, test, and production that are often isolated from a management perspective



Design for resource groups

Plan your resource groups

A resource group is a container that holds related resources for an Azure solution.



- Group resources that share the same life cycle
- Group by type, app, department, location, or billing
- Apply RBAC and policies to a group of resources
- Use resource locks to protect individual resources from deletion or change



Design for resource groups

✓ 100 XP

3 minutes

Resource groups are logical containers into which Azure resources are deployed and managed. These resources can include web apps, databases, and storage accounts. You can use resource groups to:

- Place resources of similar usage, type, or location in logical groups.
- Organize resources by life cycle so all the resources can be created or deleted at the same time.
- Apply role permissions to a group of resources or give a group access to administer a group of resources.
- Use resource locks to protect individual resources from deletion or change.

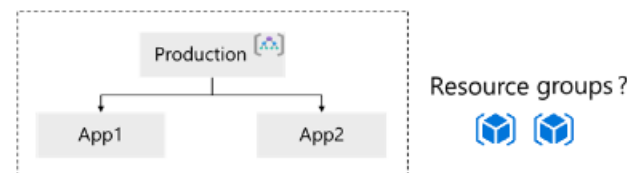
Things to know about resource groups

As you plan the governance strategy for Tailwind Traders, consider these characteristics of resource groups:

- Resource groups have their own location (region) assigned. This region is where the metadata is stored.
- If the resource group's region is temporarily unavailable, you can't update resources in the resource group because the metadata is unavailable. The resources in other regions still function as expected, but you can't update them.
- Resources in the resource group can be in different regions.
- A resource can connect to resources in other resource groups. You can have a web application that connects to a database in a different resource group.
- Resources can be moved between resource groups with some exceptions.
- You can add a resource to or remove a resource from a resource group at any time.
- Resource groups can't be nested.
- Each resource must be in one, and only one, resource group.
- Resource groups can't be renamed.

Things to consider when creating resource groups

Tailwind Traders has two Azure-based applications (App1 and App2). Each application has a web service with SQL database, virtual machines, and storage. You need to decide how to organize the resource groups for Tailwind Traders.



- Consider group by type. Group resources by type for on-demand services that aren't associated with an app. For Tailwind Traders, you can have a resource group for the SQL databases (SQL-RG) and a separate resource group (WEB-RG) for the web services.

SQL-RG



WEB-RG



- Consider group by app. Group resources by app when all resources have the same policies and life cycle. This method can also be applied to test or prototype environments. For Tailwind Traders, App1 and App2 can have separate resource groups. Each group can have all the resources for the specific application.

App1



App2



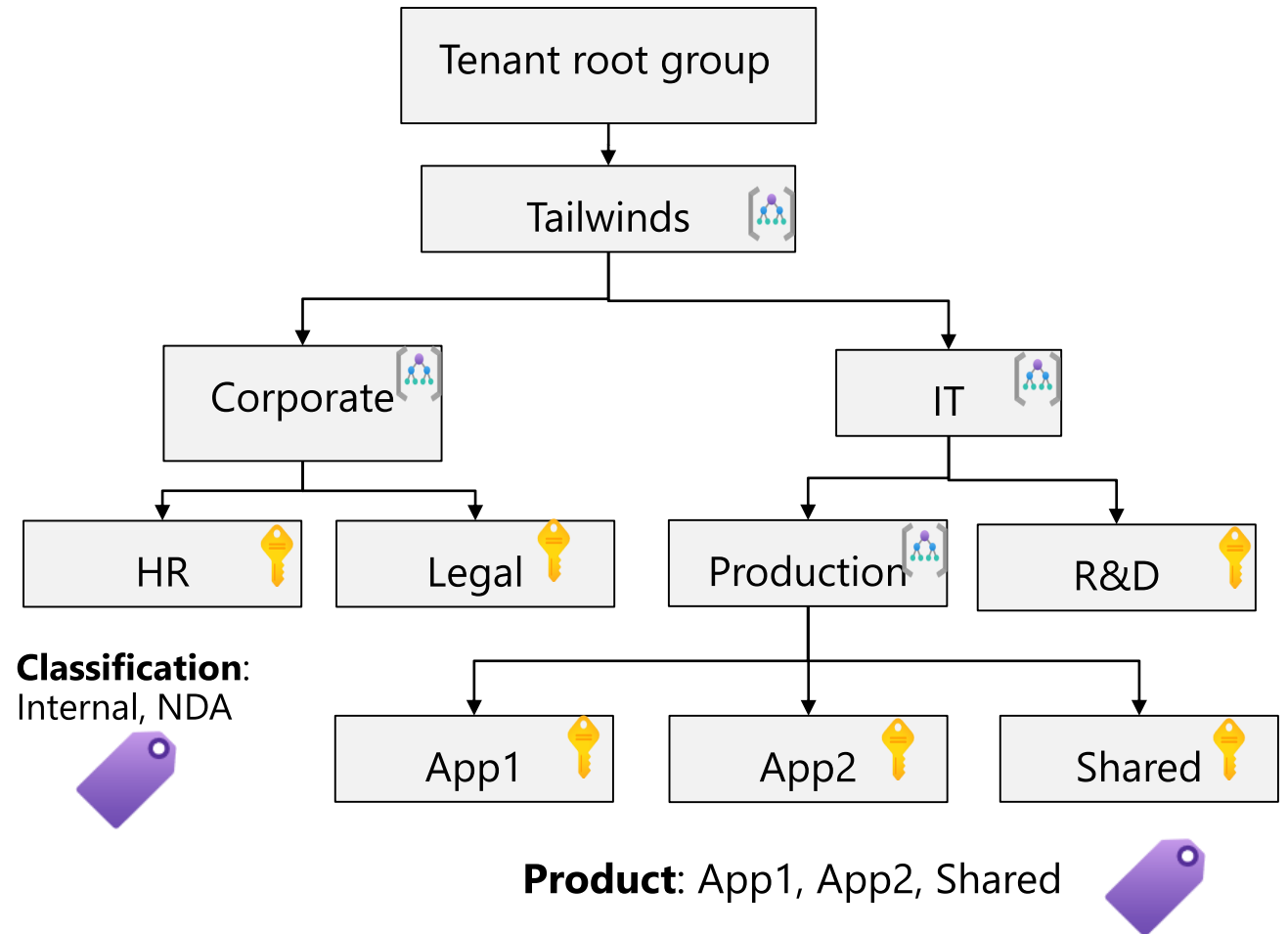
- Consider group by department, group by location (region), and group by billing (cost center). Review other grouping strategies that aren't common but might be useful in your situation.
- Consider a combination of organizational strategies. Don't restrict your Tailwind Traders strategy to using only a single resource group option. A combination of options is best.
- Consider resource life cycle. Design your resource groups according to life cycle requirements. Do you want to deploy, update, and delete certain resources at the same time? If so, place these resources in the same resource group.
- Consider administration overhead. Include overhead planning in your strategy. How many resource groups would you like to manage? Does Tailwind Traders have centralized or decentralized Azure administrators?
- Consider resource access control. Implement access control for your resource groups. At the resource group level, you can assign Azure policies, Azure roles, and resource locks. Resource locks prevent unexpected changes to critical resources.
- Consider compliance requirements. Plan to build in support for compliance in your Tailwind Traders strategy. Do you need to ensure your resource group metadata is stored in a particular region?

Design for resource tagging

Plan your resource tagging

Resource tagging can be business-aligned or IT-aligned

- Consider your organization's taxonomy
- Determine the reason for the tagging - functional, classification, accounting, partnership, or purpose
- Start with a few tags (mission-critical resources) and then scale out
- Policies could be used to apply tags and enforce tagging rules and conventions - mimic inheritance



Design for resource tags

✓ 100 XP

4 minutes

[Resource tags](#) are another way to organize resources. Tags provide extra information, or metadata, about your resources.

💡 Tip

Before you start a resource tagging project, ask yourself what you want to accomplish. Will the tags be used for reporting or billing? Can you use the tags to enable more effective searching for Tailwind Traders? Maybe the tags can be used in automated scripts. Be sure to clearly define your goals.

Things to know about resource tags

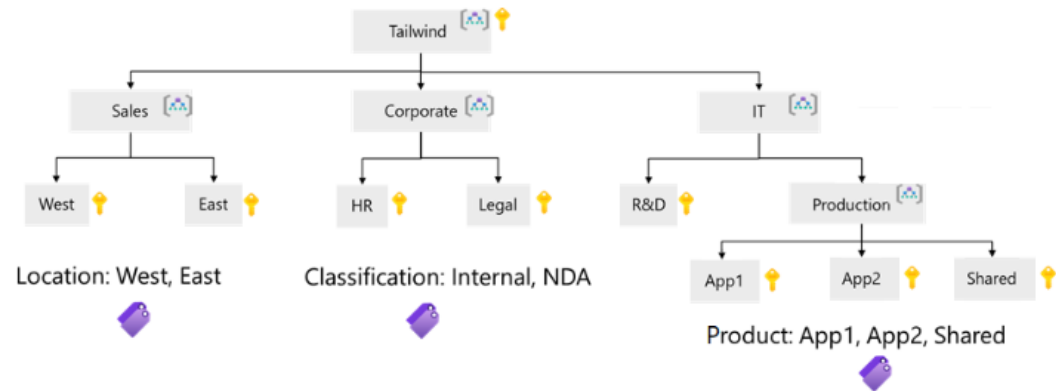
As you plan the governance strategy for Tailwind Traders, consider these characteristics of resource tags:

- A resource tag consists of a name-value pair. For example, `env = production` or `env = dev, test`.
- You can assign one or more tags to each Azure resource, resource group, or subscription.
- Resource tags can be added, modified, and deleted. These actions can be done with PowerShell, the Azure CLI, Azure Resource Manager (ARM) templates, the REST API, or the Azure portal.
- [Tags can be applied](#) to a resource group. However, tags applied to a resource group aren't inherited by the resources in the group.



Things to consider when creating resource tags

You created the organizational hierarchy for Tailwind Traders. Now you need to determine which resource tags to apply.



- Consider your organization's taxonomy. Align your resource tags with accepted department nomenclature to make it easier to understand. Are there recognized terms for compliance or cost reporting for the Tailwind Traders organization? Add tags for office locations, confidentiality levels, or other defined policies.
- Consider whether you need IT-aligned or business-aligned tagging. Implement IT-aligned tagging or business-aligned tagging, or a combination of these approaches to be most effective.




Tip

Many organizations are shifting from IT-aligned to business-aligned tagging strategies.

Alignment	Description	Example scenarios
IT-aligned	The IT-aligned option is useful for tracking workload, application, function, or environment criteria. This option can reduce the complexity of monitoring assets. IT-aligned tagging simplifies making management decisions based on operational requirements.	Tailwind Traders printers are busy 80% of the time. We have five high-speed color printers and should buy more. Use IT-aligned tagging to support printer resource workload and function.
Business-aligned	The Business-aligned option helps to focus on accounting, business ownership, cost responsibility, and business criticality. This option provides improved accounting for costs and value of IT assets to the overall business. You can use Business-aligned tagging to shift the focus from an asset's operational cost to an asset's business value.	The Tailwind Traders Marketing department's promotional literature has increased sales revenue 10%. We should invest in more printing capabilities. Use Business-aligned tagging to support marketing resource ownership, accounting, and cost.



- Consider the type of tagging required. Plan to use different types of resource tags to support the Tailwind Traders organization. Resource tags generally fall into five categories: functional, classification, accounting, partnership, and purpose.

 Expand table

Tag type	Description	Example name-value pairs
Functional	Functional tags categorize resources according to their purpose within a workload. This tag shows the deployed environment for a resource, or other functionality and operational details.	<ul style="list-style-type: none"> - app = catalogsearch1 - tier = web - webserver = apache - env = production, dev, staging
Classification	Classification tags identify a resource by how it's used and what policies apply to it.	<ul style="list-style-type: none"> - confidentiality = private - SLA = 24hours
Accounting	Accounting tags allow a resource to be associated with specific groups within an organization for billing purposes.	<ul style="list-style-type: none"> - department = finance - program = business-initiative - region = northamerica
Partnership	Partnership tags provide information about the people (other than IT members) who are associated with a resource, or otherwise affected by the resource.	<ul style="list-style-type: none"> - owner = jsmith - contactalias = catsearchowners - stakeholders = user1;user2;user3
Purpose	Purpose tags align resources to business functions to better support investment decisions.	<ul style="list-style-type: none"> - businessprocess = support - businessimpact = moderate - revenueimpact = high

- Consider starting with a few tags and then scale out. The resource tagging approach you choose can be simple or complex. Rather than identify all the possible tags required by the Tailwind Traders organization, prototype with just a few important or critical tags. Determine how effective the tagging scheme is before you add more resource tags.
- Consider using Azure policy to apply tags and enforce tagging rules and conventions. Resource tagging is only effective if it's used consistently across an organization. You can use Azure policy to require that certain tags be added to new resources as they're created. You can also define rules that reapply tags that have been removed.
- Consider which resources require tagging. Keep in mind that you don't need to enforce that a specific tag is present on all Tailwind Traders resources. You might decide that only mission-critical resources have the **Impact** tag. All non-tagged resources would then not be considered as mission critical.

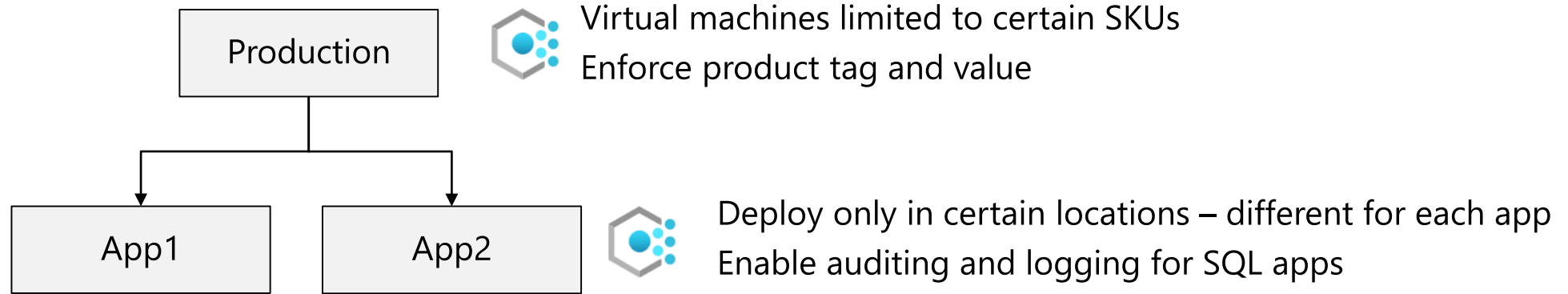
Note

To implement an effective resource tagging structure, be sure to seek input from the different stakeholders in your organization.

Design for Azure Policy and RBAC

When to use Azure Policy

Azure Policy helps to enforce organizational standards and to assess compliance at-scale.



- Large number of built-in policies and you can create custom policies

Examples

- Allow only certain virtual machines sizes for your project
- Ensure all resources are correctly tagged – if not, apply the tag
- Recommend system updates on your servers
- Enable multifactor authentication for all subscription accounts



Things to know about Azure Policy

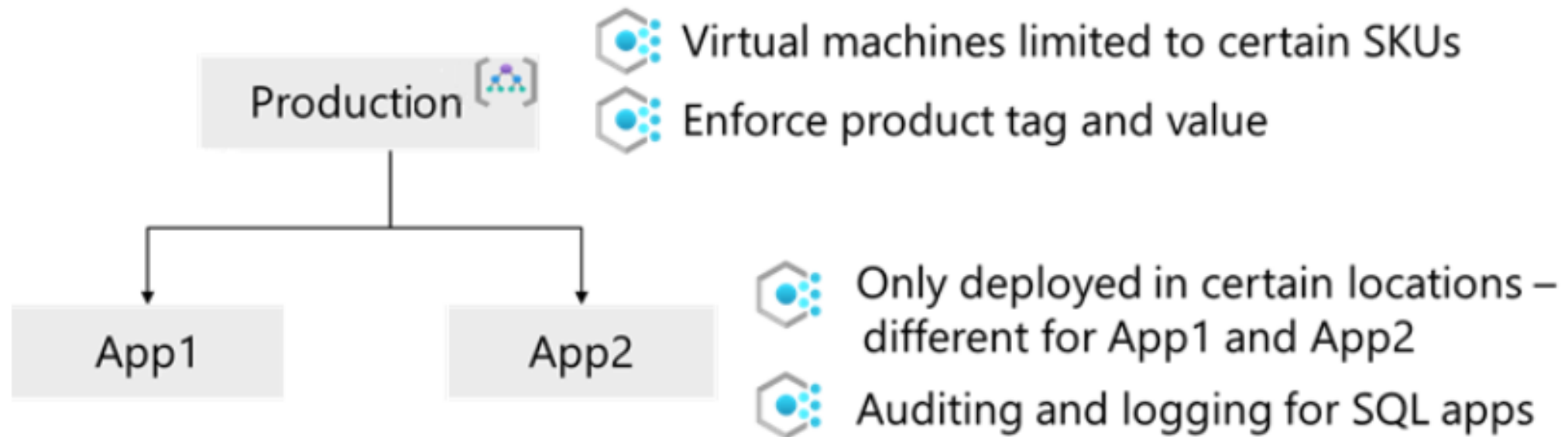
As you plan the governance strategy for Tailwind Traders, consider these characteristics of Azure Policy:

- Azure Policy lets you define both individual policies and groups of related policies, called *initiatives*. Azure Policy comes with many **built-in policy** and **initiative** definitions.
- Azure policies are inherited down the hierarchy.
- You can scope and enforce Azure policies at different levels in the organizational hierarchy.
- Azure Policy evaluates all resources in Azure and Arc-enabled resources (specific resource types that are hosted outside of Azure).
- Azure Policy highlights resources that aren't compliant with the current policies.
- Use Azure Policy to prevent noncompliant resources from being created, and automatically remediate noncompliant resources.
- Azure Policy integrates with Azure DevOps by applying pre-deployment and post-deployment policies.



Things to consider when using Azure Policy

You're ready to consider how to apply Azure Policy settings to your Tailwind Traders applications. You'll probably apply some policies at the Production management group level. Other policies can be assigned at the application level.



- **Consider using the Azure Policy compliance dashboard.** Use the Azure Policy compliance dashboard to analyze the overall state of the environment. The dashboard offers an aggregated view where you can drill down to see Tailwind Traders policies for each resource and level. The tool provides bulk remediation for existing resources and automatic remediation for new resources, to resolve issues rapidly and effectively.
- **Consider when Azure Policy evaluates resources.** Plan for how Azure Policy evaluates your Tailwind Traders resources at specific times. Understand when and how evaluations are triggered. There might be a delay in identifying non-compliant resources. The following events or times trigger an evaluation:
 - A resource is created, deleted, or updated in scope with a policy assignment.
 - A policy or an initiative is newly assigned to a scope.
 - An assigned policy or initiative for a scope is updated.
 - The standard compliance evaluation cycle (occurs once every 24 hours).
- **Consider how to handle a noncompliant resource.** Determine how you're going to handle noncompliant resources for Tailwind Traders. An organization can have a different way of handling noncompliance depending on the resource. Here are some examples:
 - Deny changes to the resource.
 - Log changes to the resource.
 - Alter the resource before or after the change.
 - Deploy related compliant resources.

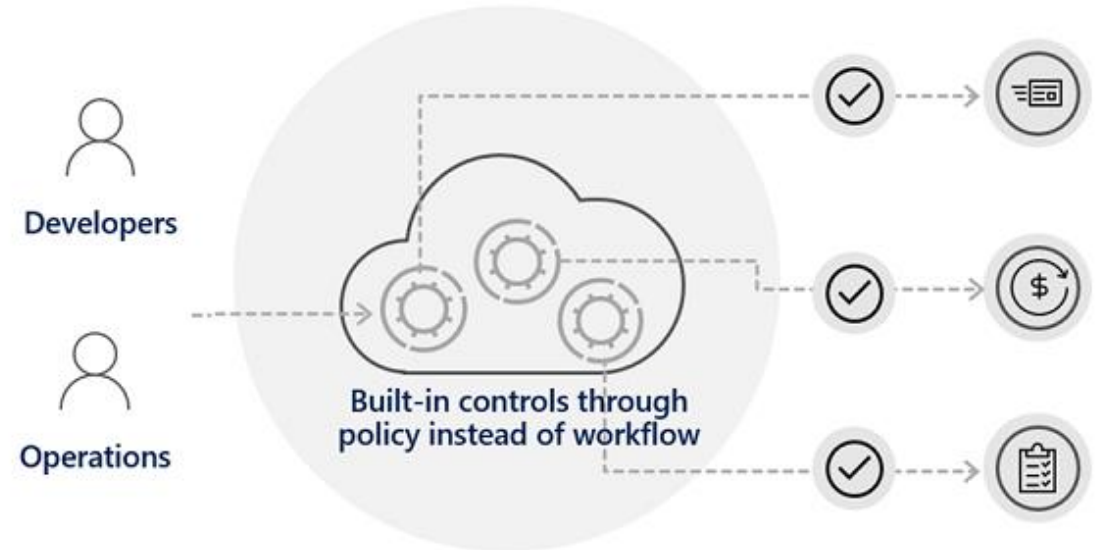


- **Consider when to automatically remediate noncompliant resources.** Decide if you want Azure Policy to do automatic remediation for noncompliant resources. Remediation is especially useful in resource tagging. Azure Policy can tag resources and reapply tags that have been removed. You can use Azure Policy to ensure all resources in a certain resource group are tagged with a specific tag like `Location` to identify the region.
- **Consider how Azure Policy is different from role-based access control (RBAC).** It's important to understand that Azure Policy and Azure RBAC are different. For your Tailwind Traders strategy, Azure RBAC and Azure Policy should be used together to achieve full scope control.
 - You use Azure Policy to ensure the resource state is compliant with the organization's business rules. Compliance doesn't depend on who made the change or who has permission to make changes. Azure Policy evaluates the state of a resource, and acts to ensure the resource stays compliant.
 - You implement Azure RBAC to focus on user actions at different scopes. Azure RBAC manages who can access Azure resources, what they can do with those resources, and what areas they can access. If actions need to be controlled, use Azure RBAC. If a user has access to complete an action, but the result is a noncompliant resource, Azure Policy still blocks the action.



Considerations for Azure Policy





- Apply policy at the highest scope possible
- Know when policies are evaluated
- Decide what to do if a resource is non-compliant
- Consider when to automatically remediate non-compliant resources
- Use the Azure policy compliance dashboard for auditing and review
- Effectively combine Azure policy with RBAC (next slide)



Design for Azure role-based access control (RBAC)

Azure RBAC allows you to grant access to Azure resources that you control.

- Only grant users the access they need
- Assign at the highest scope level that meets the requirements
- Assign roles to groups, not users
- Know when to create a custom role
- Consider what happens if you have overlapping role assignments

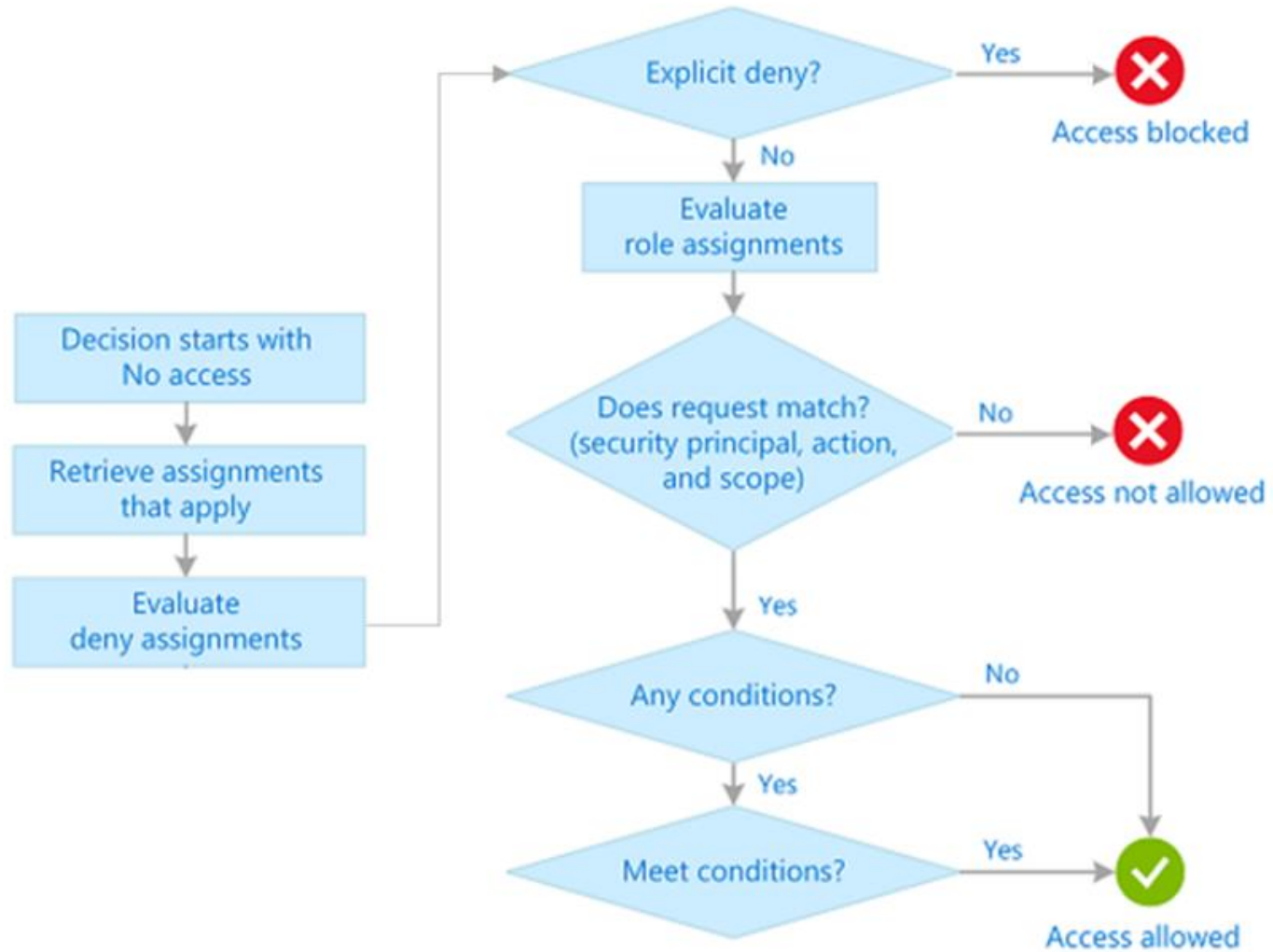
		Role				
		Reader	Resource-specific	Custom	Contributor	Owner
Scope	 Management group	Observers Auditors Reviewers	Helpdesk personnel Developers Users managing resources			Admins
	 Subscription					
	 Resource group					
	 Resource	Automated processes				



Design for role-based access control (RBAC)

4 minutes

Azure RBAC allows you to grant access to Azure resources that you control. Azure RBAC evaluates each request for access and determines if access should be blocked, not allowed, or allowed.



Things to know about Azure RBAC

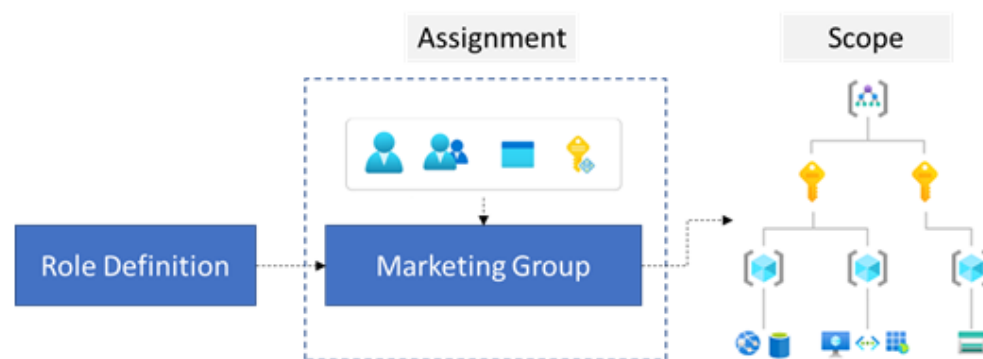
Suppose you need to manage access to resources in Azure for Tailwind Traders Development, Engineering, and Marketing teams. Here are some scenarios you can implement with Azure RBAC:

- Allow one user to manage virtual machines in a subscription, and allow another user to manage virtual networks.
- Allow members of a database administrator group to manage SQL databases in a subscription.
- Allow a user to manage all resources in a resource group, such as virtual machines, websites, and subnets.
- Allow an application to access all resources in a resource group.

Things to consider when using Azure RBAC

You have a plan for how to apply Azure Policy settings to your Tailwind Traders applications. Now consider how to integrate Azure RBAC to control user privileges and resource access.

- **Consider the highest scope level for each requirement.** Your first step is to accurately define each role definition and its permissions. Next, assign the roles to specific users, groups, and service principles. Lastly, scope the roles to management groups, subscriptions, resource groups, and resources. Assign each role at the highest scope level that meets the requirements.




- Consider the access needs for each user. As you plan your access control strategy, it's a best practice to grant users the least privilege they need to get their work done. This method makes it easier to separate team member responsibilities. By limiting roles and scopes, you limit what resources are at risk if a security principle is ever compromised. You can create a diagram like the following example to help plan your Azure RBAC roles for Tailwind Traders.

		Role				
		Reader	Resource-specific	Custom	Contributor	Owner
Scope	Management group	Observers Auditors Reviewers	Helpdesk personnel Developers Users managing resources			Admins
	Subscription					
	Resource group					
	Resource	Automated processes				

- Consider assigning roles to groups, and not users. To make role assignments more manageable, avoid assigning roles directly to users. Instead, assign roles to groups. Assigning roles to groups helps minimize the number of role assignments.



- **Consider when to use Azure policies.** Azure policies are used to focus on resource properties. During deployment, an Azure policy can be used to ensure users can deploy only certain virtual machines in a resource group. By using a combination of Azure policies and Azure RBAC, you can provide effective access control in your Tailwind Traders solution. The following table compares these access models.

 Expand table

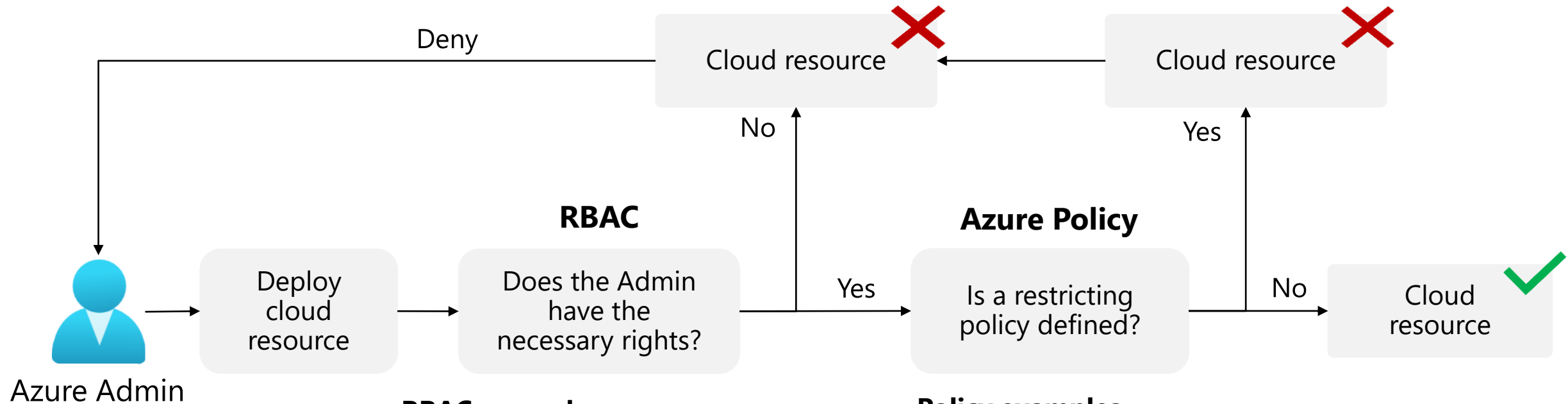
	Azure Policy	Azure RBAC
Description	Defined policies to ensure resources are compliant with a set of rules.	Authorization system that provides fine-grained access controls.
Main focus	Focused on the properties of resources.	Focused on what resources users can access.
Implementation	Specify a set of rules.	Assign roles and scopes.
Default access	By default, policy rules are set to <i>allow</i> .	By default, all access for all users is <i>denied</i> .

- **Consider when to create a custom role.** Sometimes, the built-in roles don't grant the precise level of access you need. Custom roles allow you to define roles that meet the specific needs of your organization. Custom roles can be shared between subscriptions that trust the same Microsoft Entra ID.
- **Consider how to resolve overlapping role assignments.** Azure RBAC is an additive model, so your effective permissions are the sum of your role assignments. Consider a user is granted the **Contributor** role at the subscription scope and the **Reader** role on a resource group. The sum of the Contributor permissions and the Reader permissions is effectively the Contributor role for the subscription. Therefore, in this case, the Reader role assignment has no impact.

<https://learn.microsoft.com/en-us/azure/role-based-access-control/overview#multiple-role-assignments>



When to combine Azure Policy and Azure RBAC



RBAC examples

- Does the Admin have the right to deploy?
- Does the Admin have the right to deploy this resource type?
- Does the Admin have the right to deploy this resource group?

Policy examples

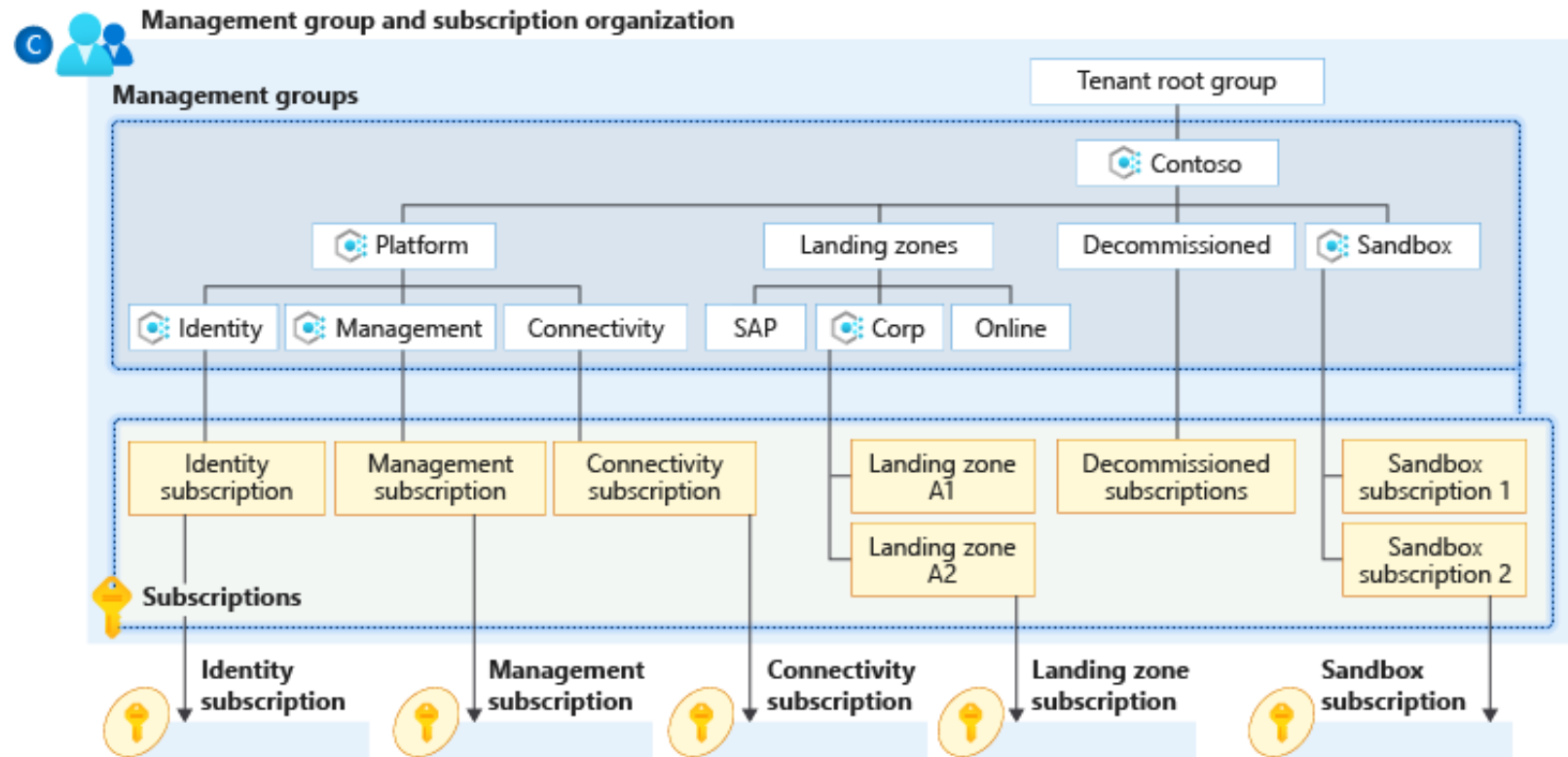
- Is the region restricted?
- Is the resource type restricted?
- Should a tag be applied?

Design for Landing Zones

Implement Landing Zones

A landing zone provides an infrastructure environment for hosting your workloads.

- Implements key foundational principles of governance, security, networking, management, and identity
- Pre-provisions the environment through code
- Good for both migrations and green field situations
- You can transition existing architectures
- Part of the Cloud Adoption Framework Ready phase



Design for Azure landing zones

2 minutes

An [Azure landing zone](#) provides an infrastructure environment for hosting your workloads. Landing zones ensure key foundational principles are put in place before you deploy services.

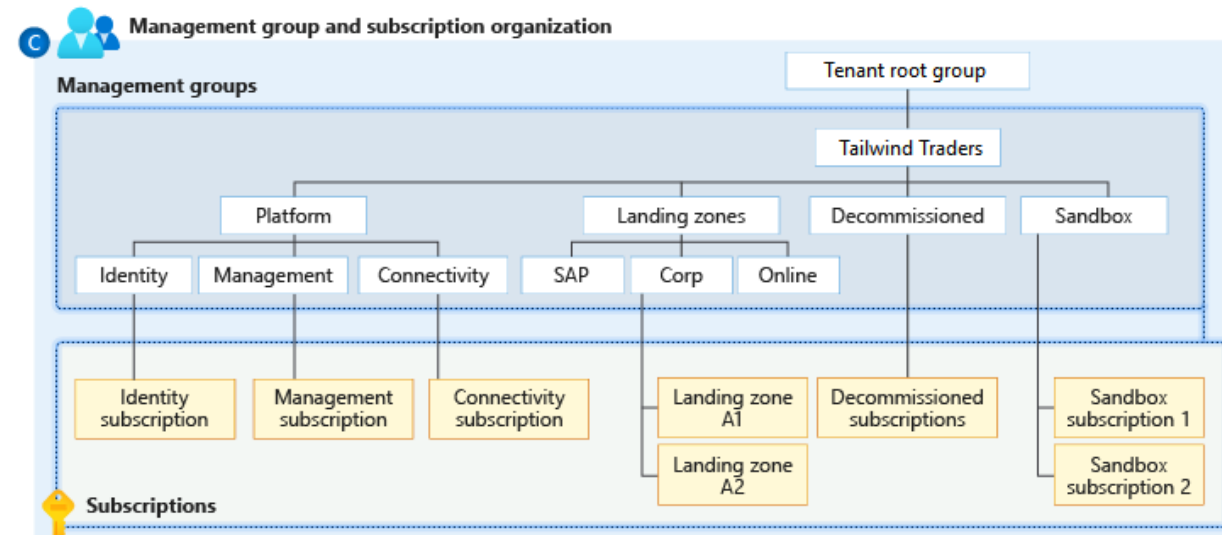
To use an analogy, shared city utilities like water, gas, and electricity are available to new homes before they're built. In the same manner, the network, identity and access management, policies, and monitoring configuration for landing zones must be ready before you try to deploy. These "utilities" for landing zones need to be active and ready to help streamline the application migration process.

Things to know about Azure landing zones

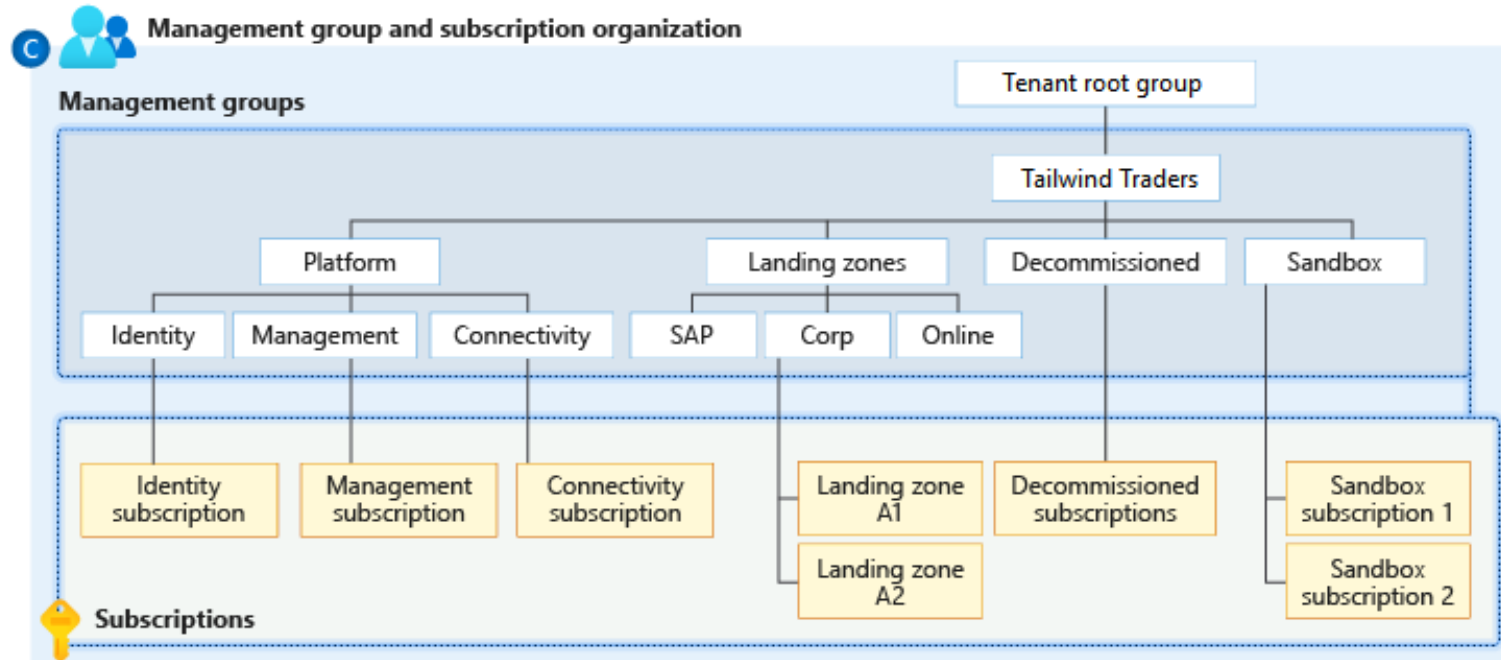
As you plan the governance strategy for Tailwind Traders, consider these characteristics of Azure landing zones:

- Landing zones are defined by management groups and subscriptions that are designed to scale according to business needs and priorities.

The following diagram shows landing zones for SAP, Corporate, and Online applications.



The following diagram shows landing zones for SAP, Corporate, and Online applications.



Azure landing zone accelerators ...
Accelerators are infrastructure-as-code implementations that help you deploy an Azure landing zone correctly.

- Azure policies are associated with landing zones to ensure continued compliance with the organization platform.
- Landing zones are pre-provisioned through code.
- A landing zone can be scoped to support application migrations and development to scale across the organization's full IT portfolio.
- The Azure landing zone accelerator can be deployed into the same Microsoft Entra tenant for an existing Azure architecture. The accelerator is an Azure-portal-based deployment.



Things to consider when using Azure landing zones

You're ready to finalize your governance strategy for Tailwind Traders. Consider how you can use Azure landing zones to scale your design:

- **Consider including landing zones in your design.** Include landing zones in your overall Azure infrastructure design. You can use subscriptions as a unit of management and scale that's aligned with business needs and priorities. Apply Azure Policy to provide guardrails and ensure continued compliance with your organization's platform, along with the applications that are deployed onto it.
- **Consider creating landing zones through code.** Implement landing zones that are pre-provisioned through code. As your situation changes, you should expect to refactor the code. Use an iterative approach that maximizes learning opportunities and minimizes time to business success. You can minimize refactoring by having a central IT team to review both short term and long-term scenarios.
- **Consider using the [Azure landing zone accelerator](#).** Use the accelerator to provide a full implementation of the conceptual architecture, along with opinionated configurations for key components like management groups and policies.
- **Consider focusing on your applications.** Focus on application-centric migrations and development rather than pure infrastructure lift-and-shift migrations, such as moving virtual machines.
- **Consider Azure-native design and aligning with the platform.** Favor using Azure-native platform services and capabilities, when possible. It's crucial to align with the Azure platform roadmap to ensure that new capabilities are made available within customer environments.
- **Consider scoping for both migrations and green field situations.** Scope the landing zone to support application migrations and green field development at scale in Azure. This expansion allows for a design that can scale across your organization's complete IT portfolio, which looks well beyond a short-term cloud-adoption plan.
- **Consider [transitioning existing architectures to Azure landing zones](#).** Take advantage of landing zones for existing Azure architecture. Deploy the Azure landing zone accelerator into the same Microsoft Entra tenant in parallel with the current environment. You can create a new management group structure and ensure that the existing environment isn't affected by these changes.

<https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/ready/landing-zone/#azure-landing-zone-accelerator>

Case Studies and Review

Case study – Cost and accounting

- Tailwind Traders has two main business units that handle Apparel, and Sporting Goods.
 - Each of the business units consist of three departments: Product Development, Marketing, and Sales.
 - Each business unit and subunit will be responsible for tracking their Azure spend.
 - The Enterprise IT team will be responsible for providing company-wide Azure cost reporting.
- What are different ways Tailwind Traders could organize their subscriptions and management groups. Which would be the best to meet their requirements?
 - Design two alternative hierarchies and explain your decision-making process.

Case study – New development project

- The company has a new development project for customer feedback.
 - The CFO wants to ensure all costs associated with the project are captured.
 - For the testing phase workloads should be hosted on lower cost virtual machines.
 - The virtual machines should be named to indicate they are part of the project.
 - Any instances of non-compliance with resource consistency rules should be automatically identified.
- What are the different way Tailwind Traders could track costs for the new development project?
 - How are you ensuring compliance with the requirements for virtual machine sizing and naming?
 - Propose at least two ways of meeting the requirements. Explain your final decision.

End of presentation