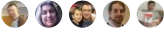


Governance design for a simple workload

02/11/2019 • 6 minutes to read • Contributors 

In this article

[Licensing Azure](#)

[Identity management](#)

[Resource management scope](#)

[Implementing the basic resource access management model](#)

[Next steps](#)

The goal of this guidance is to help you learn the process for designing a resource governance model in Azure to support a single team and a simple workload. You'll look at a set of hypothetical governance requirements, then go through several example implementations that satisfy those requirements.

In the foundational adoption stage, our goal is to deploy a simple workload to Azure. This results in the following requirements:

- Identity management for a single **workload owner** who is responsible for deploying and maintaining the simple workload. The workload owner requires permission to create, read, update, and delete resources as well as permission to delegate these rights to other users in the identity management system.
- Manage all resources for the simple workload as a single management unit.

Licensing Azure

Before you begin designing our governance model, it's important to understand how Azure is licensed. This is because the administrative accounts associated with your Azure license have the highest level of access to your Azure resources. These administrative accounts form the basis of your governance model.

Note

If your organization has an existing [Microsoft Enterprise Agreement](#) that does not include Azure, Azure can be added by making an upfront monetary commitment. See [licensing Azure for the enterprise](#) for more information.

When Azure was added to your organization's Enterprise Agreement, your organization was prompted to create an **Azure account**. During the account creation process, an **Azure account owner** was created, as well as an Azure Active Directory (Azure AD) tenant with a **global administrator** account. An Azure AD tenant is a logical construct that represents a secure, dedicated instance of Azure AD.

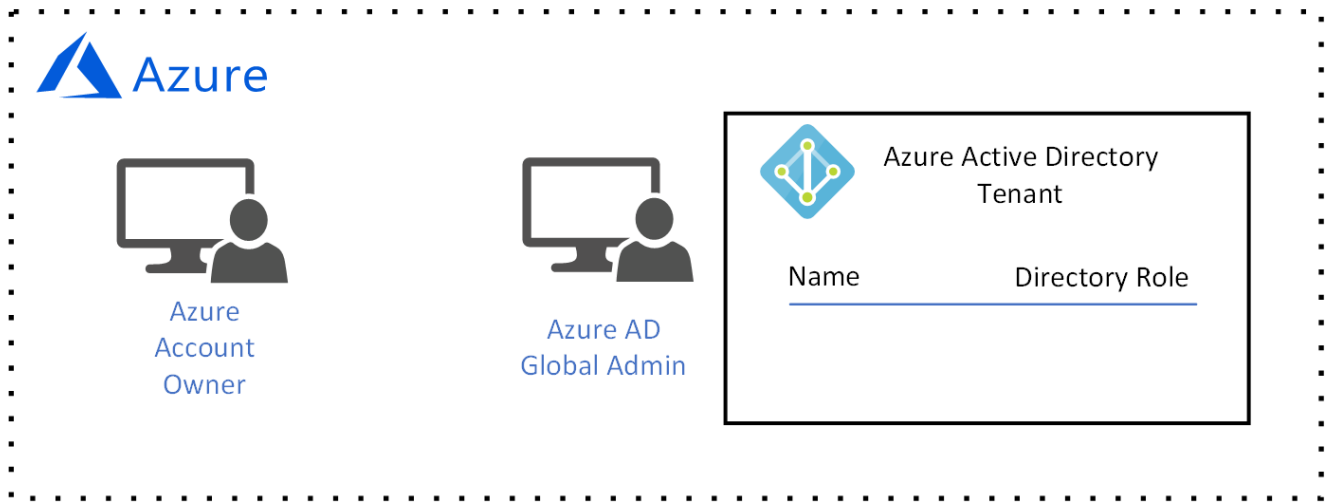


Figure 1. An Azure account with an Account Manager and Azure AD Global Administrator.

Identity management

Azure only trusts [Azure AD](#) to authenticate users and authorize user access to resources, so Azure AD is our identity management system. The Azure AD global administrator has the highest level of permissions and can perform all actions related to identity, including creating users and assigning permissions.

Our requirement is identity management for a single **workload owner** who is responsible for deploying and maintaining the simple workload. The workload owner requires permission to create, read, update, and delete resources as well as permission to delegate these rights to other users in the identity management system.

Our Azure AD global administrator will create the **workload owner** account for the workload owner:

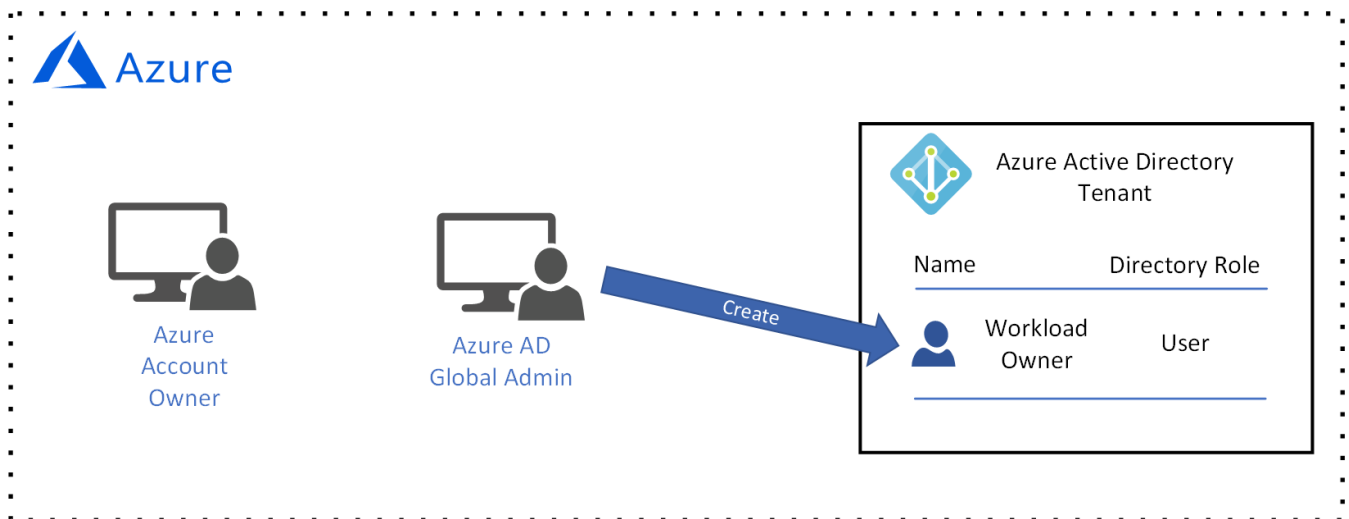


Figure 2. The Azure AD global administrator creates the workload owner user account.

You aren't able to assign resource access permission until this user is added to a **subscription**, so you'll do that in the next two sections.

Resource management scope

As the number of resources deployed by your organization grows, the complexity of governing those resources grows as well. Azure implements a logical container hierarchy to enable your organization to manage your resources in groups at various levels of granularity, also known as **scope**.

The top level of resource management scope is the **subscription** level. A subscription is created by the Azure **account owner**, who establishes the financial commitment and is responsible for paying for all Azure resources associated with the subscription:

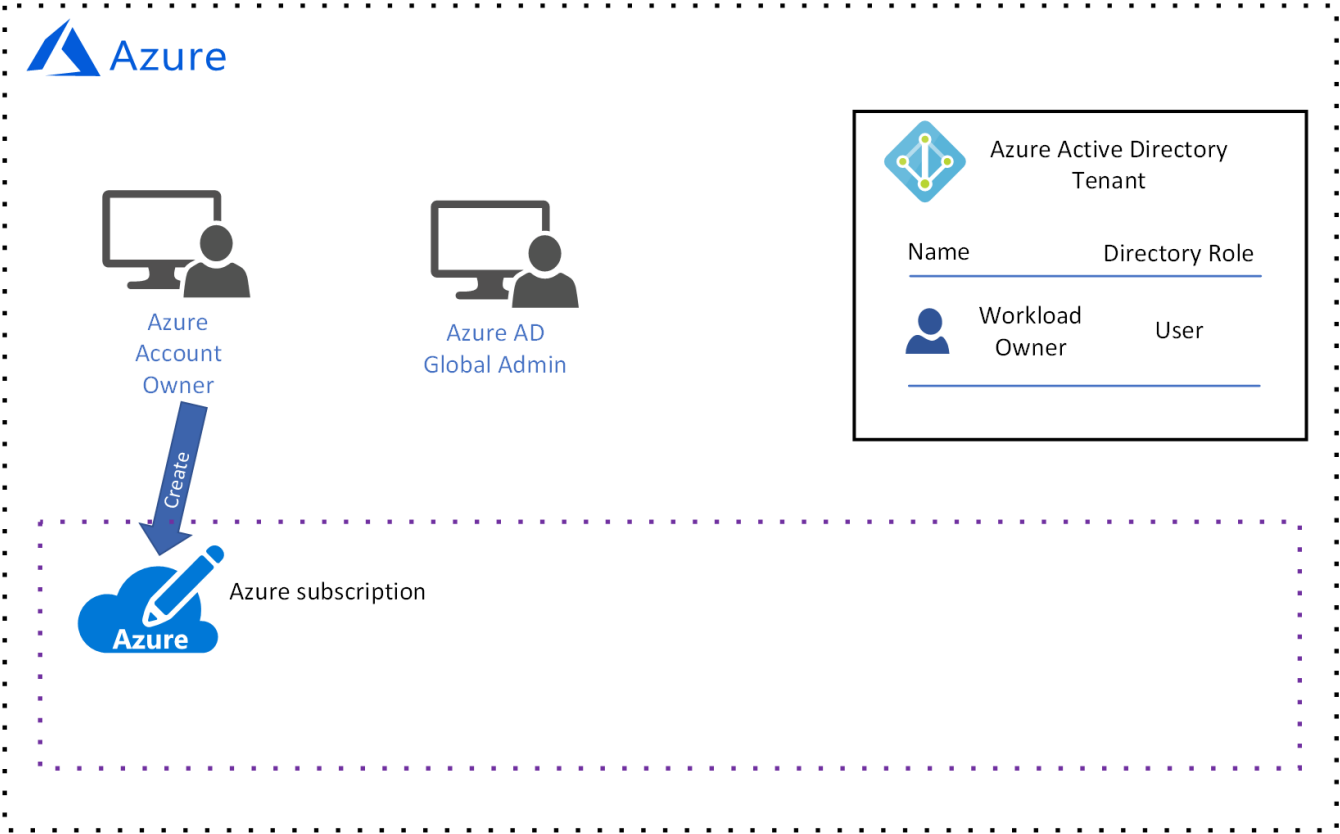


Figure 3. The Azure account owner creates a subscription.

When the subscription is created, the Azure **account owner** associates an Azure AD tenant with the subscription, and this Azure AD tenant is used for authenticating and authorizing users:

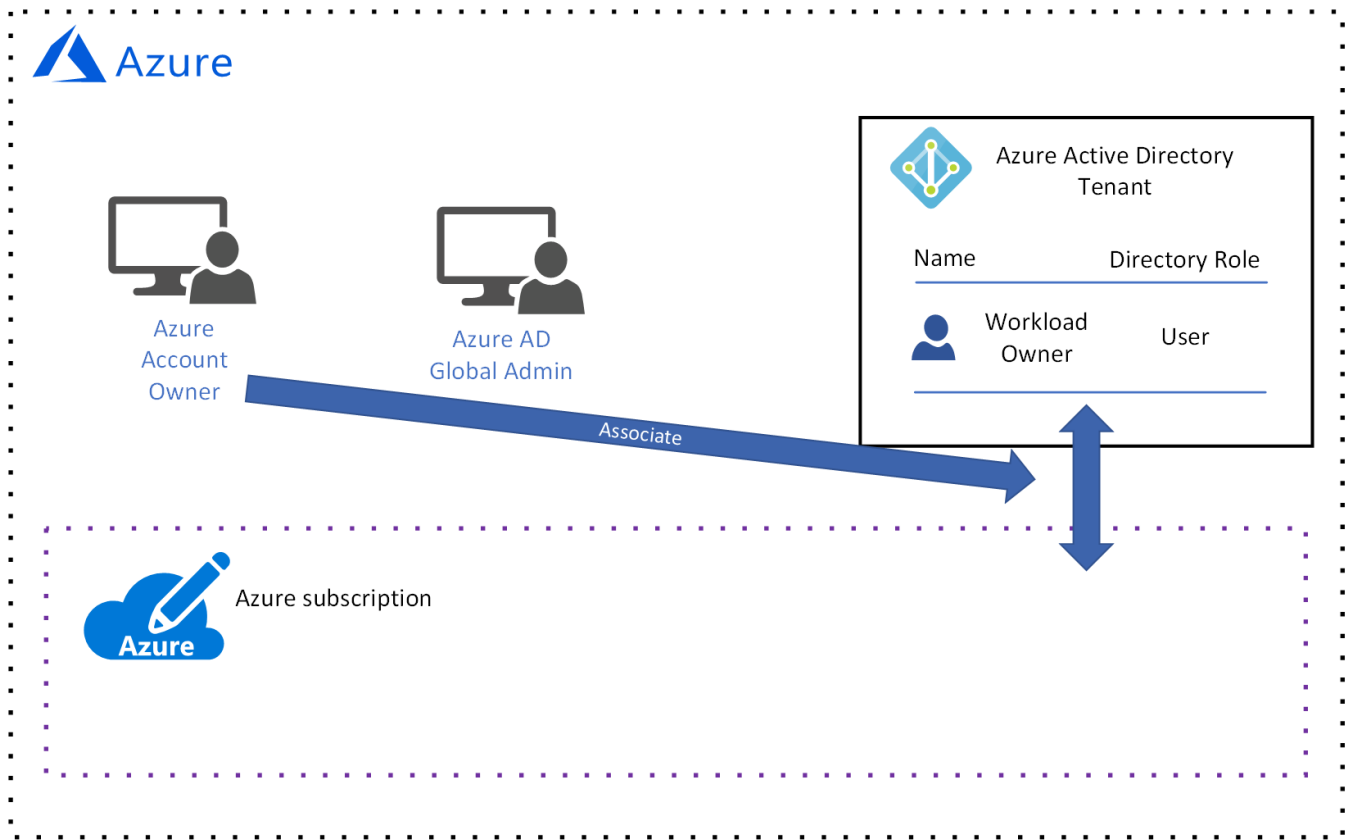


Figure 4. The Azure account owner associates the Azure AD tenant with the subscription.

You may have noticed that there is currently no user associated with the subscription, which means that no one has permission to manage resources. In reality, the **account owner** is the owner of the subscription and has permission to take any action on a resource in the subscription. However, in practical terms the **account owner** is more than likely a finance person in your organization and is not responsible for creating, reading, updating, and deleting resources - those tasks will be performed by the **workload owner**. Therefore, you need to add the **workload owner** to the subscription and assign permissions.

Since the **account owner** is currently the only user with permission to add the **workload owner** to the subscription, they add the **workload owner** to the subscription:

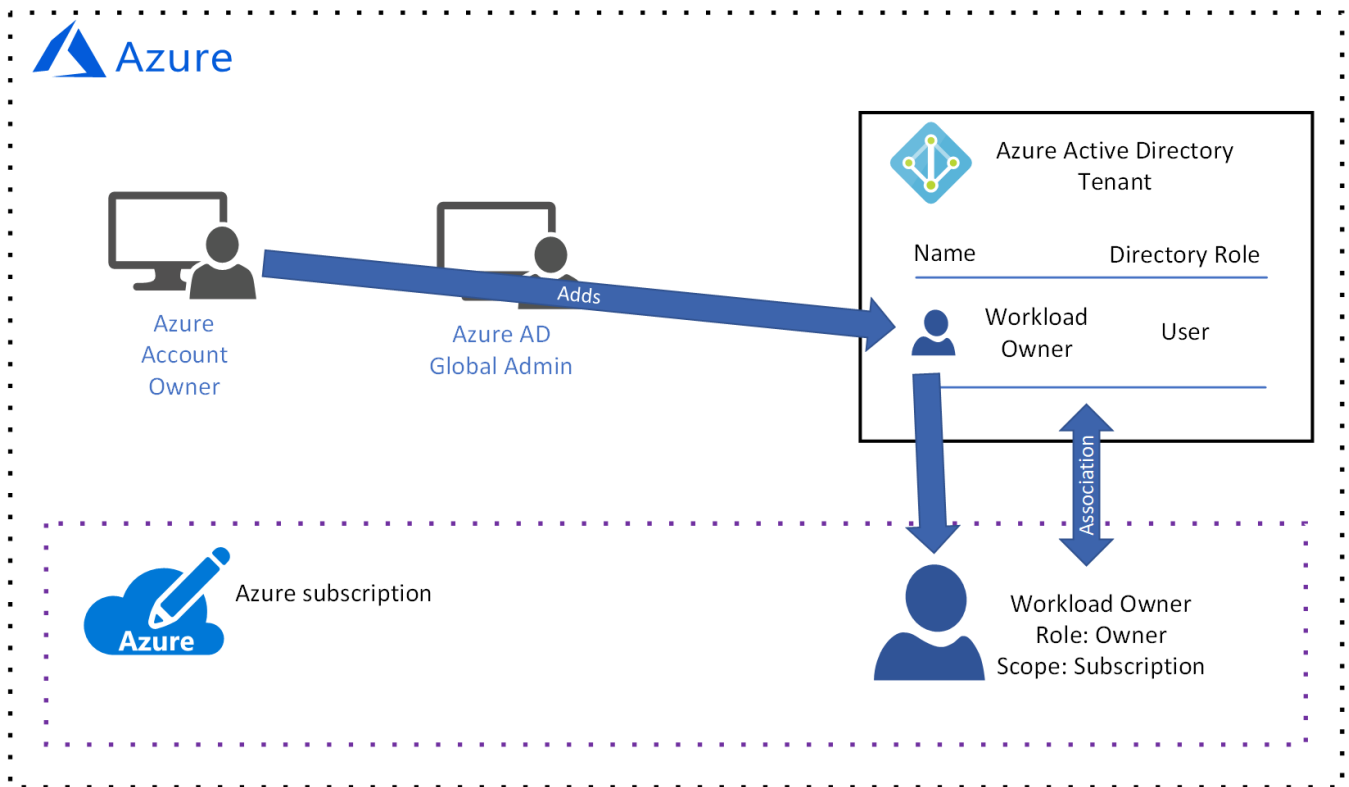


Figure 5. The Azure account owner adds the workload owner to the subscription.

The Azure **account owner** grants permissions to the **workload owner** by assigning a [role-based access control \(RBAC\)](#) role. The RBAC role specifies a set of permissions that the **workload owner** has for an individual resource type or a set of resource types.

Notice that in this example, the **account owner** has assigned the [built-in owner role](#):

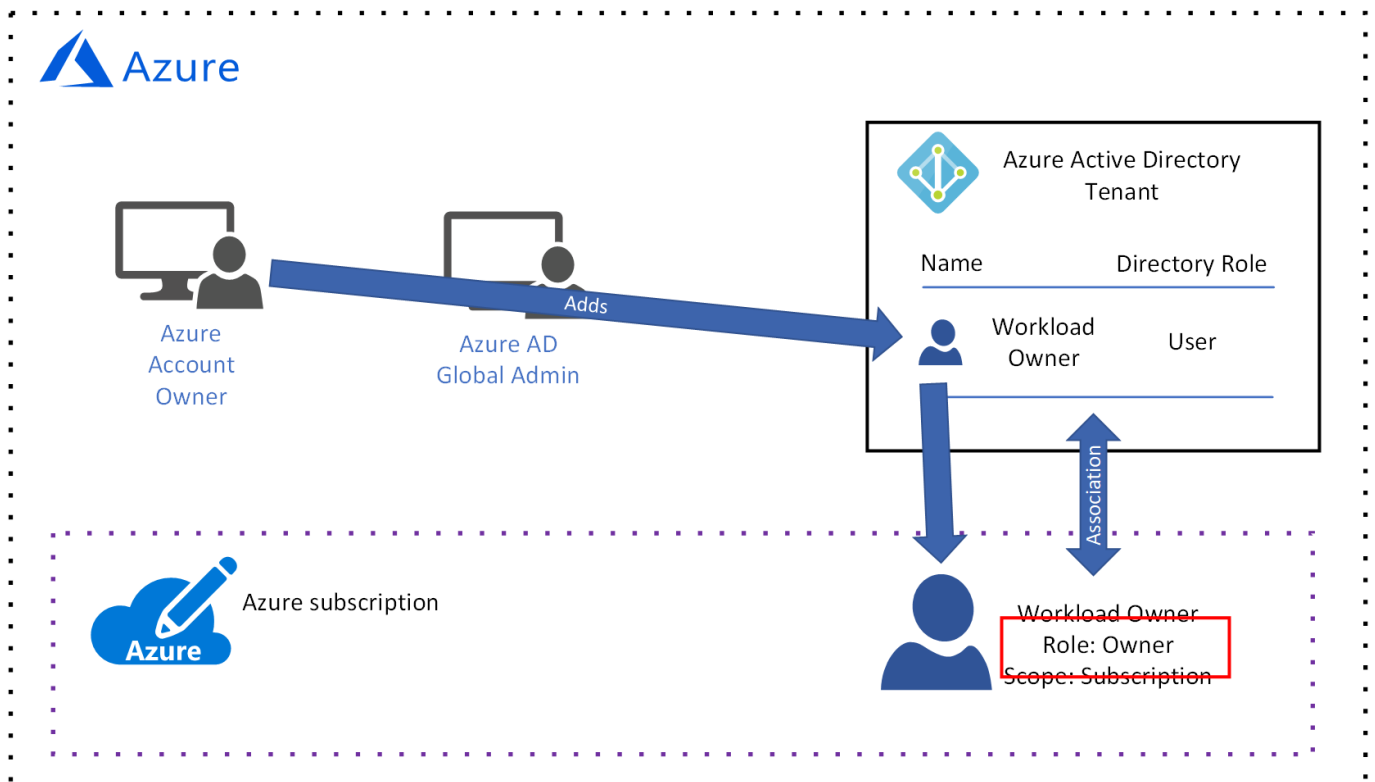


Figure 6. The workload owner was assigned the built-in owner role.

The built-in **owner** role grants all permissions to the **workload owner** at the subscription scope.

❗ Important

The Azure **account owner** is responsible for the financial commitment associated with the subscription, but the **workload owner** has the same permissions. The **account owner** must trust the **workload owner** to deploy resources that are within the subscription budget.

The next level of management scope is the **resource group** level. A resource group is a logical container for resources. Operations applied at the resource group level apply to all resources in a group. Also, it's important to note that permissions for each user are inherited from the next level up unless they are explicitly changed at that scope.

To illustrate this, let's look at what happens when the **workload owner** creates a resource group:

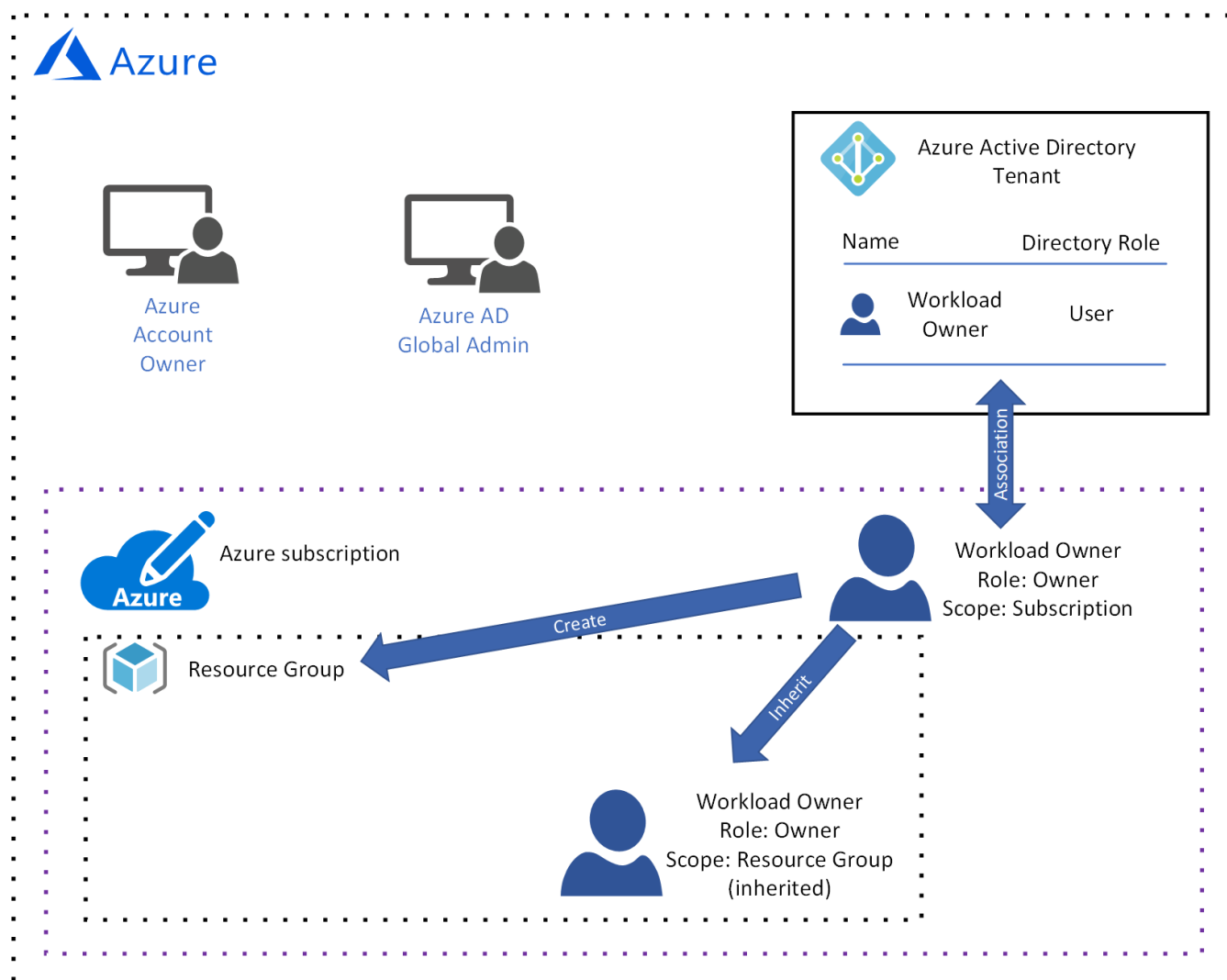


Figure 7. The workload owner creates a resource group and inherits the built-in owner role at the resource group scope.

Again, the built-in **owner** role grants all permissions to the **workload owner** at the resource group scope. As discussed earlier, this role is inherited from the subscription level. If a different role is assigned to this user at this scope, it applies to this scope only.

The lowest level of management scope is at the **resource** level. Operations applied at the resource level apply only to the resource itself. And once again, permissions at the resource level are inherited from resource group scope. For example, let's look at what happens if the **workload owner** deploys a [virtual network](#) into the resource group:

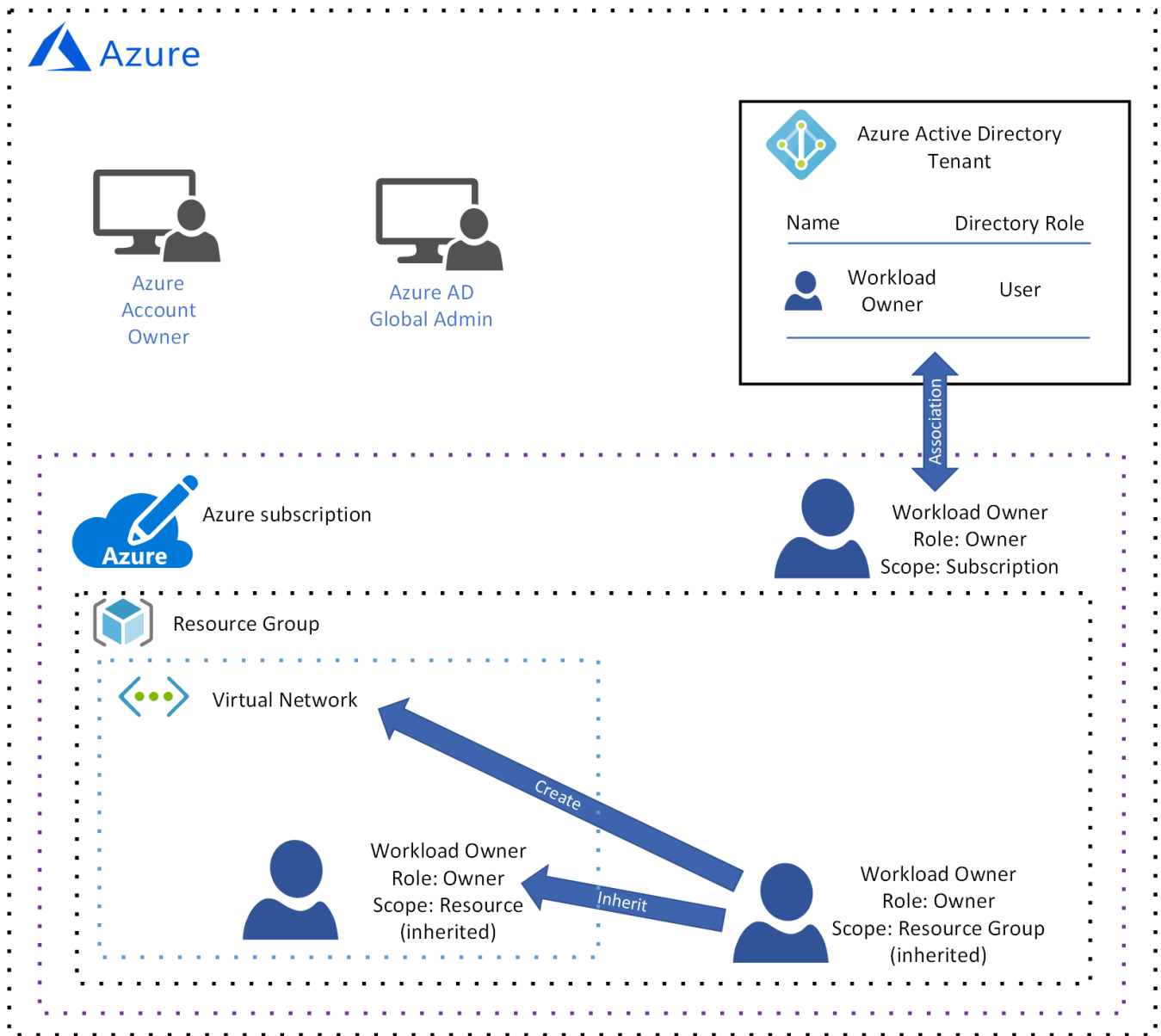


Figure 8. The workload owner creates a resource and inherits the built-in owner role at the resource scope.

The **workload owner** inherits the owner role at the resource scope, which means the workload owner has all permissions for the virtual network.

Implementing the basic resource access management model

Let's move on to learn how to implement the governance model designed earlier.

To begin, your organization requires an Azure account. If your organization has an existing [Microsoft Enterprise Agreement](#) that does not include Azure, Azure can be added by making an upfront monetary commitment. See [licensing Azure for the enterprise](#) for more information.

When your Azure account is created, you specify a person in your organization to be the Azure **account owner**. An Azure Active Directory (Azure AD) tenant is then created by default. Your Azure **account owner** must [create the user account](#) for the person in your organization who is the **workload owner**.

Next, your Azure **account owner** must [create a subscription](#) and [associate the Azure AD tenant](#) with it.

Finally, now that the subscription is created and your Azure AD tenant is associated with it, you can [add the workload owner to the subscription with the built-in owner role](#).

Next steps

[Deploy a basic workload to Azure](#)

[Learn about resource access for multiple teams](#)