Federating Google Cloud with Active Directory

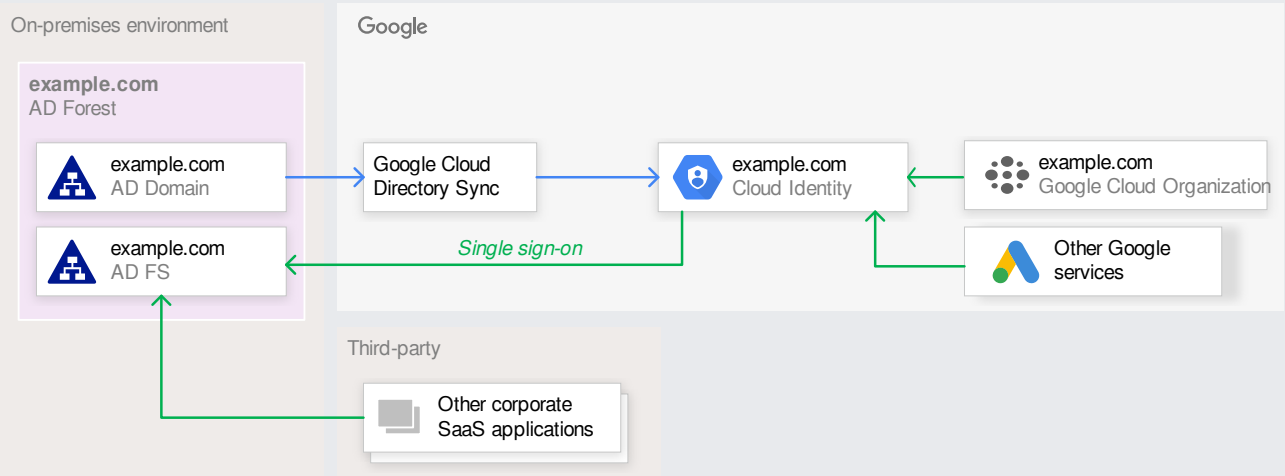
This article describes how you can configure Cloud Identity or G Suite to use Active Directory as IdP and authoritative source (/architecture/identity/reference-architectures#active_directory_as_idp_and_authoritative_source).

The article compares the logical structure of Active Directory with the structure used by Cloud Identity and G Suite and describes how you can map Active Directory forests, domains, users, and groups. The article also provides a flowchart (#choosing_the_right_mapping) that helps you determine the best mapping approach for your scenario.

This article assumes that you're familiar with Active Directory.

Implementing federation

Google Cloud uses Google identities (/architecture/identity/overview-google-authentication#google_identities) for authentication and access management. Manually maintaining Google identities for each employee can add unnecessary management overhead when all employees already have an account in Active Directory. By federating user identities between Google Cloud and your existing identity management system, you can automate the maintenance of Google identities and tie their lifecycle to existing users in Active Directory.



Setting up federation between Active Directory and Cloud Identity or G Suite entails two pieces:

- **Provisioning users**: Relevant users and groups are synchronized periodically from Active Directory to Cloud Identity or G Suite. This process ensures that when you create a new user in Active Directory, it can be referenced in Google Cloud even before the associated user has logged in for the first time. This process also ensures that user deletions are being propagated.

  Provisioning works one way, which means changes in Active Directory are replicated to Google Cloud but not vice versa. Also, provisioning does not include passwords. In a federated setup, Active Directory remains the only system that manages these credentials.

- **Single sign-on**: Whenever a user needs to authenticate, Google Cloud delegates the authentication to Active Directory by using the Security Assertion Markup Language (SAML) protocol. This delegation ensures that only Active Directory manages user credentials and that any applicable policies or multi-factor authentication (MFA) mechanisms are being enforced. For a sign-on to succeed, however, the respective user must have been provisioned before.

To implement federation, you can use the following tools:

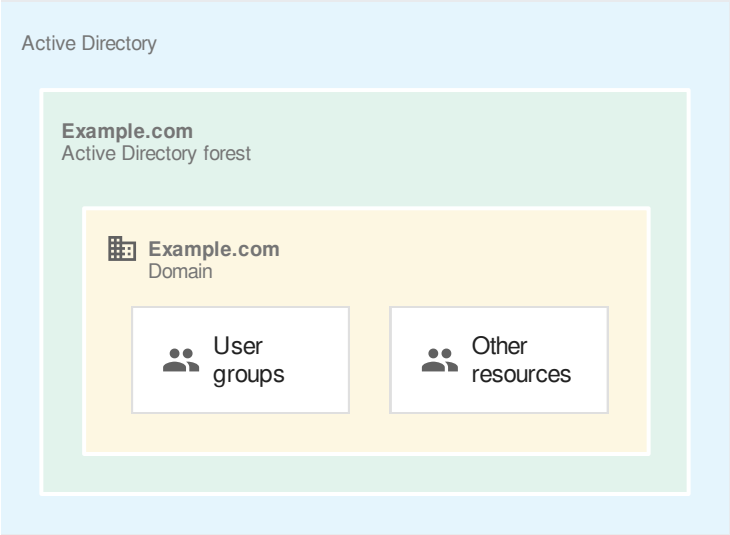
- Google Cloud Directory Sync (https://tools.google.com/dlpage/dirsync/) is a free Google-provided tool that implements the synchronization process. Cloud Directory Sync communicates with Google Cloud over Secure Sockets Layer (SSL) and usually runs in the existing computing environment.
- Active Directory Federation Services (AD FS) is provided by Microsoft as part of Windows Server. With AD FS, you can use Active Directory for federated authentication. AD FS usually runs within the existing computing environment.

Because APIs for Google Cloud are publicly available (https://developers.google.com/admin-sdk/directory/) and SAML is an open standard, many tools are available to implement federation. This article focuses on using Cloud Directory Sync and AD FS.

Logical structure of Active Directory

In an Active Directory infrastructure, the top-level component is the *forest*. The forest serves as a container for one or more domains and derives its name from the forest root domain. Domains in an Active Directory forest trust each other, allowing users who are authenticated in

one domain to access resources that are in another domain. Unless forests are connected by using cross-forest trusts, separate forests don't trust each other by default, and users who are authenticated in one forest cannot access resources that are in a different forest.



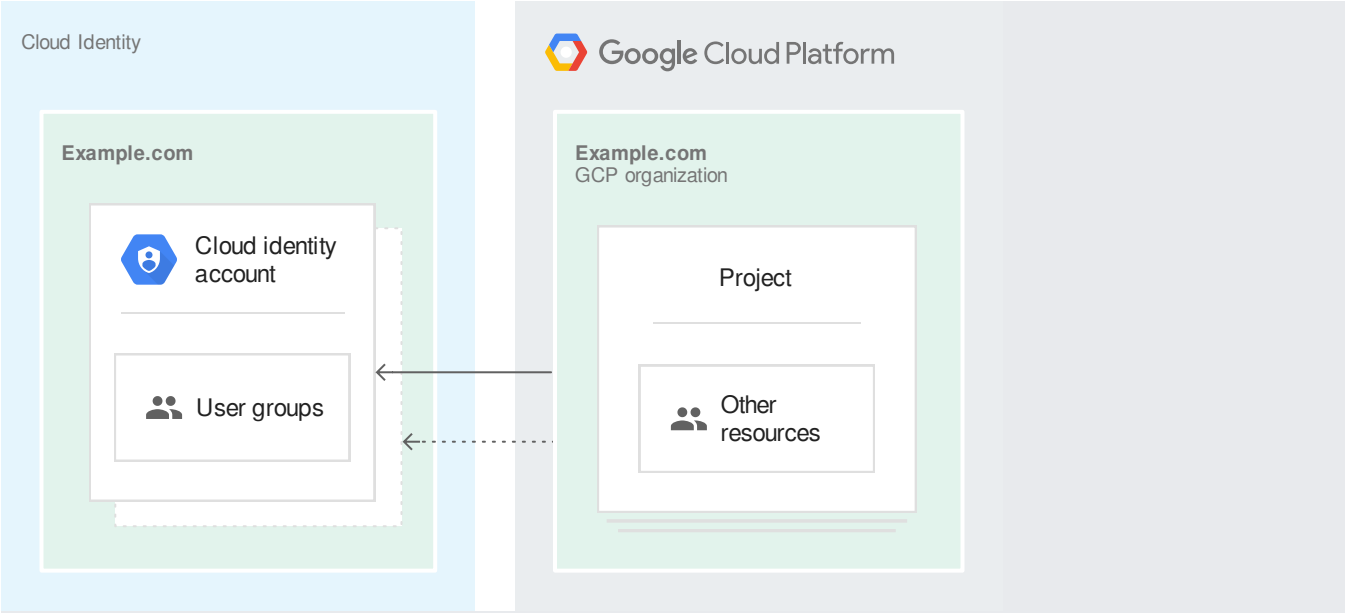
Active Directory domains are containers for managing resources and are considered administrative boundaries. Having multiple domains in a forest is one way to simplify administration or enforce additional structure, but domains in a forest don't represent security boundaries.

Logical structure of Google Cloud

In Google Cloud, organizations (/architecture/identity/overview-google-authentication#organization_node) serve as containers for all resources, and they can be further segmented by using folders and projects (/resource-manager/docs/cloud-platform-resource-hierarchy). Organizations, folders, and projects therefore serve a purpose similar to Active Directory domains.

Active Directory treats users as resources, so user management and authentication are tied to domains. In contrast, Google Cloud doesn't manage users in an organization, except for service accounts (/compute/docs/access/service-accounts). Instead, Google Cloud relies on Cloud Identity or G Suite to manage users.

A Cloud Identity or G Suite account (/architecture/identity/overview-google-authentication#cloud_identity_or_g_suite_account) serves as a private directory for users and groups. As an administrator of the account, you can control the lifecycle and the configuration of users and groups and define how authentication can be performed.



When you create a Cloud Identity or G Suite account, a Google Cloud organization (/resource-manager/docs/creating-managing-organization) is created automatically for you. The Cloud Identity or G Suite account and the Google Cloud organization that's associated with it share the same name and are tied to each other. However, a Google Cloud organization is allowed to reference users and groups from other Cloud Identity or G Suite accounts.

Integrating Active Directory and Google Cloud

Despite certain similarities between the logical structure of Active Directory and Google Cloud, no single mapping between the two structures works equally well in all scenarios. Instead, the right approach to integrating the two systems and mapping the structure depends on multiple factors:

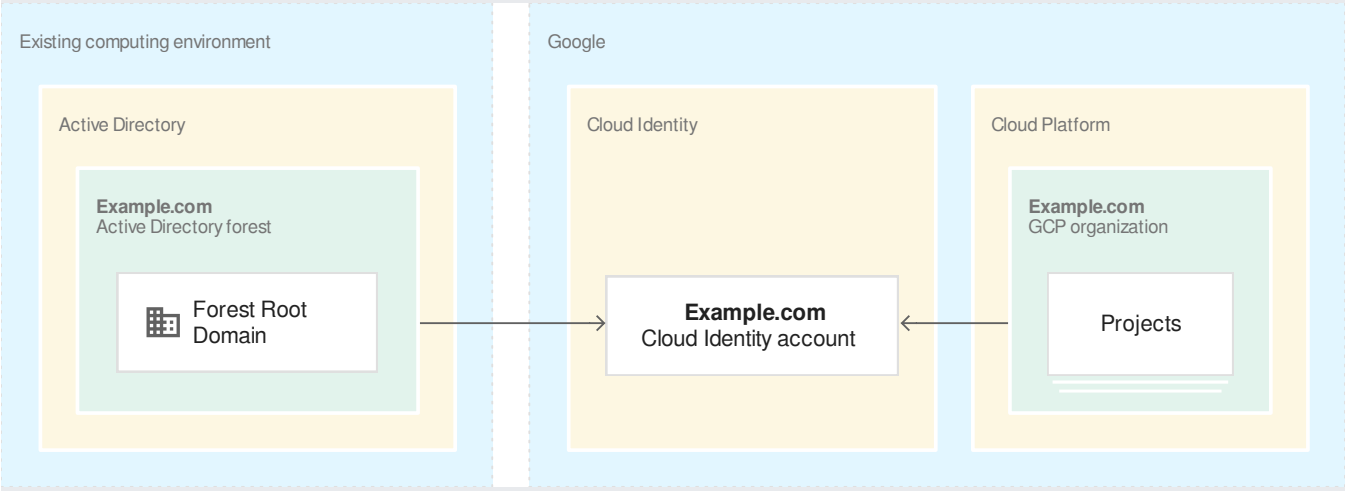
- How to map domains and forests to Cloud Identity or G Suite accounts
- How to map DNS domains
- How to map users
- How to map groups

The following sections look at each of these factors.

Mapping forests

Especially in larger organizations, you often use more than one Active Directory domain to manage identities and access across the enterprise. When you are planning to federate Active Directory and Google Cloud, the first factor to look at is the topology of your Active Directory infrastructure.

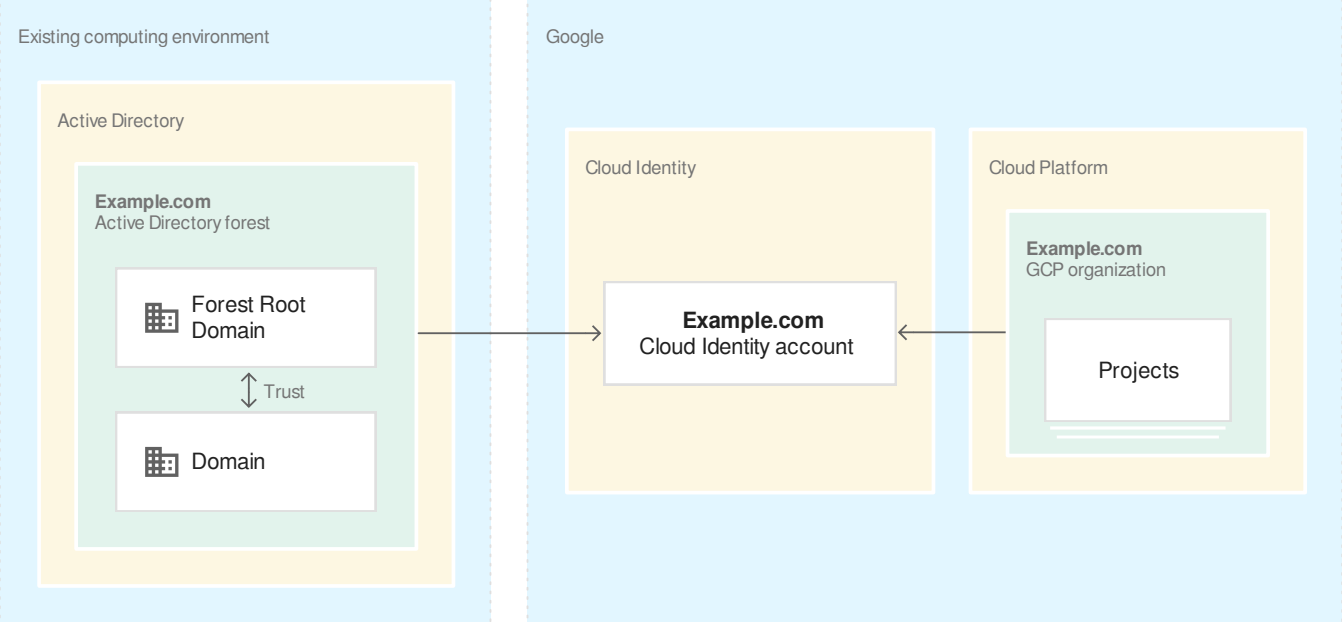
Single forest, single domain



When a forest includes just one domain, you can map the entire Active Directory forest to a single Cloud Identity or G Suite account. This account then provides the basis for a single Google Cloud organization that you can use to manage your Google Cloud resources.

In a single-domain environment, domain controllers and global catalog servers both provide access to all objects that are managed in Active Directory. In most cases, you can run a single instance of Cloud Directory Sync to synchronize user accounts and groups to Google Cloud, and to maintain a single AD FS instance or fleet to handle single sign-on.

Single forest, multiple domains

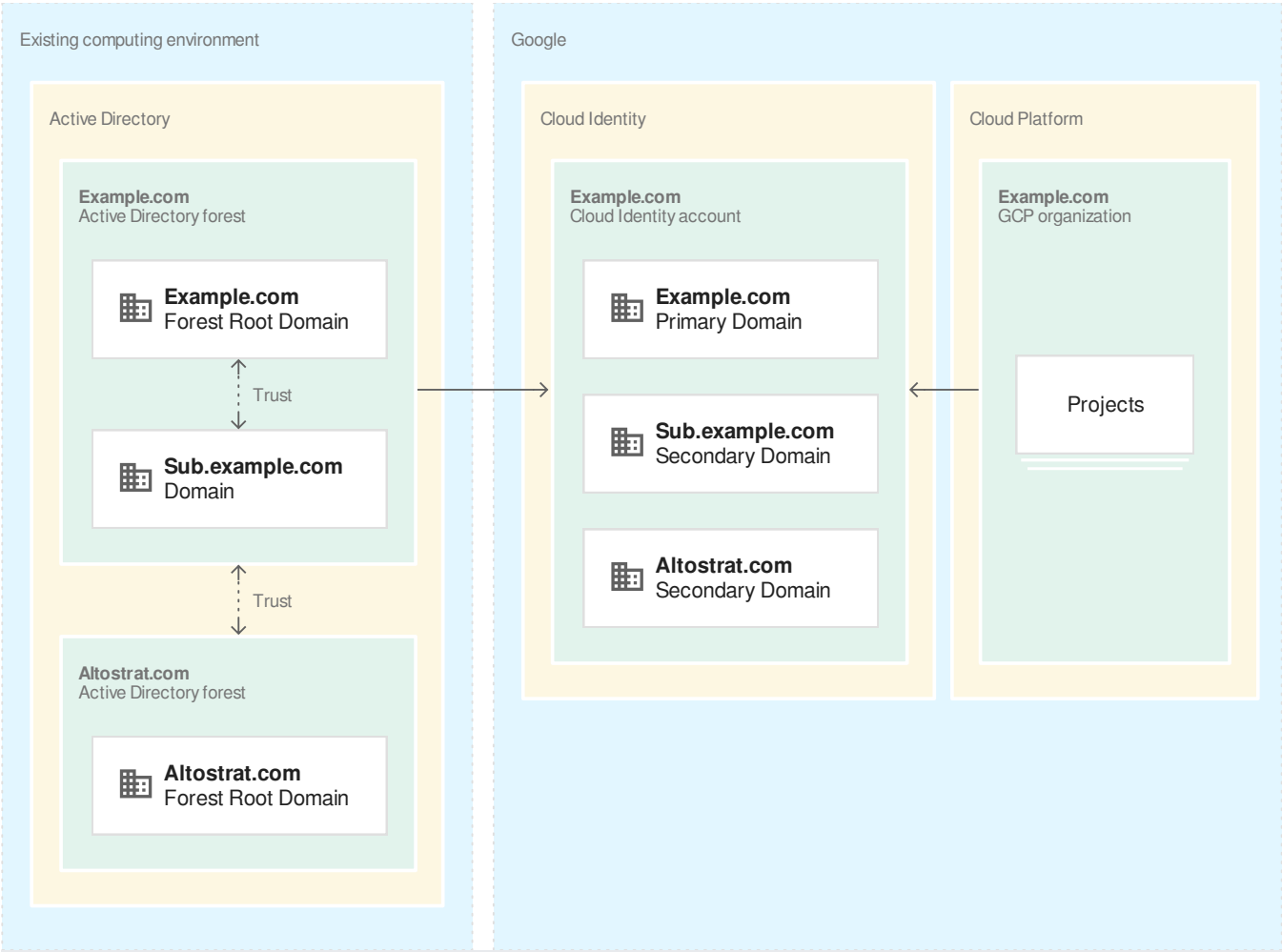


When a forest contains multiple Active Directory domains, you can organize them in one or more domain trees. In both cases, you can map the entire forest to a single Cloud Identity or G Suite account. This account then provides the basis for a single Google Cloud organization that you can use to manage your Google Cloud resources.

In a multi-domain environment, there is a difference between what information can be retrieved from a domain controller and what can be queried from a global catalog server. While domain controllers serve data only from their local domain, global catalog servers provide access to information from all domains in the forest. Crucially, the data that is served by global catalog servers is partial and lacks certain LDAP attributes. This limitation can affect how you configure Cloud Directory Sync to synchronize groups (#mapping_groups).

Depending on how you plan to map groups, federating a multi-domain forest with Google Cloud requires one or more Cloud Directory Sync instances but only a single AD FS instance or fleet to handle single sign-on.

Multiple forests with cross-forest trust



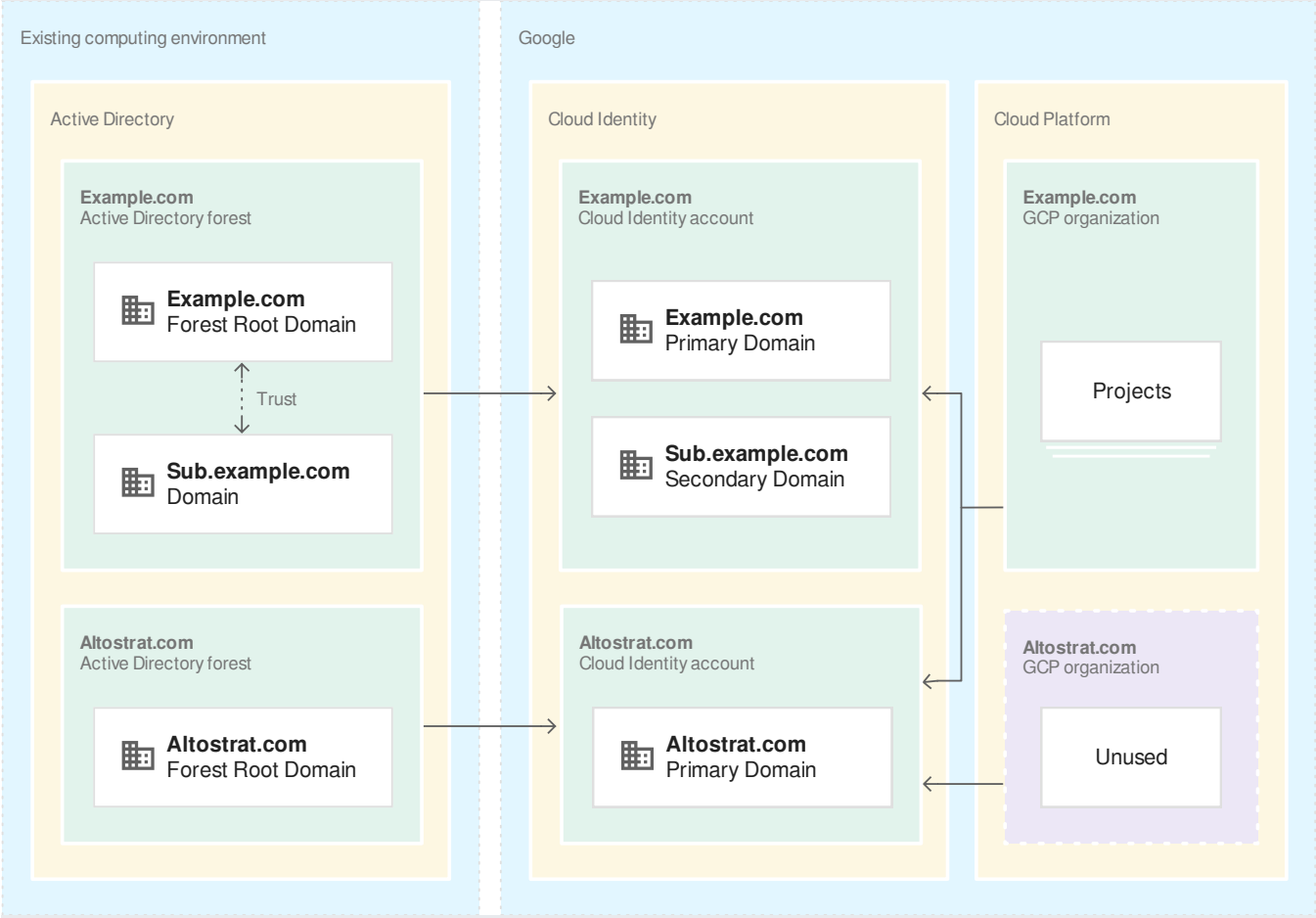
In larger organizations, it's not uncommon to have more than one Active Directory forest, often as a result of a merger or acquisition. You can combine these forests by using a two-way, cross-forest trust so that users can share and access resources across the boundaries of a single forest.

If all forests have a bidirectional trust relationship with another forest, you can map the entire environment to a single Cloud Identity or G Suite account. This account provides the basis for a single Google Cloud organization that you can use to manage your Google Cloud resources.

Although global catalog servers provide access to data from multiple domains, their scope is limited to a single forest. So in a multi-forest environment, you must query multiple domain controllers or global catalog servers to obtain, for example, a complete list of users. As a result of this limitation, federating a multi-forest environment with Google Cloud requires at least one Cloud Directory Sync instance per forest. Cross-forest trusts enable user authentication to work across forest boundaries, so a single AD FS instance or fleet is sufficient to handle single sign-on.

If your environment spans multiple forests without cross-forest trust, but all Active Directory domains that are relevant for federation with Google Cloud are connected through external trusts, then the same considerations apply.

Multiple forests without cross-forest trust



In the environment illustrated here, it's not possible to authenticate or access resources across the forest boundaries. It's also not possible for a single AD FS instance or fleet to handle single sign-on requests for users from all forests.

Therefore, it's not possible to map multiple forests that lack cross-forest trusts to a single Cloud Identity or G Suite account. Instead, each forest must be mapped to a separate Cloud Identity or G Suite account, which involves running at least one Cloud Directory Sync instance and one AD FS server or fleet per forest.

In Google Cloud, a separate organization is created for each Cloud Identity or G Suite account. In most cases, you don't need to maintain multiple, separate organizations. You can select one of the organizations and associate it

(/resource-manager/docs/managing-multiple-orgs#using_multiple_organization_nodes) with the other Cloud Identity or G Suite accounts, effectively creating an organization that is federated with multiple Active Directory forests. The other organizations remain unused.

Mapping DNS domains

DNS plays a crucial role both in Active Directory and for Cloud Identity and G Suite. The second factor to look at when you're planning to federate Active Directory and Google Cloud is how to share or map DNS domains between Active Directory and Google Cloud.

Usage of DNS domains in Active Directory

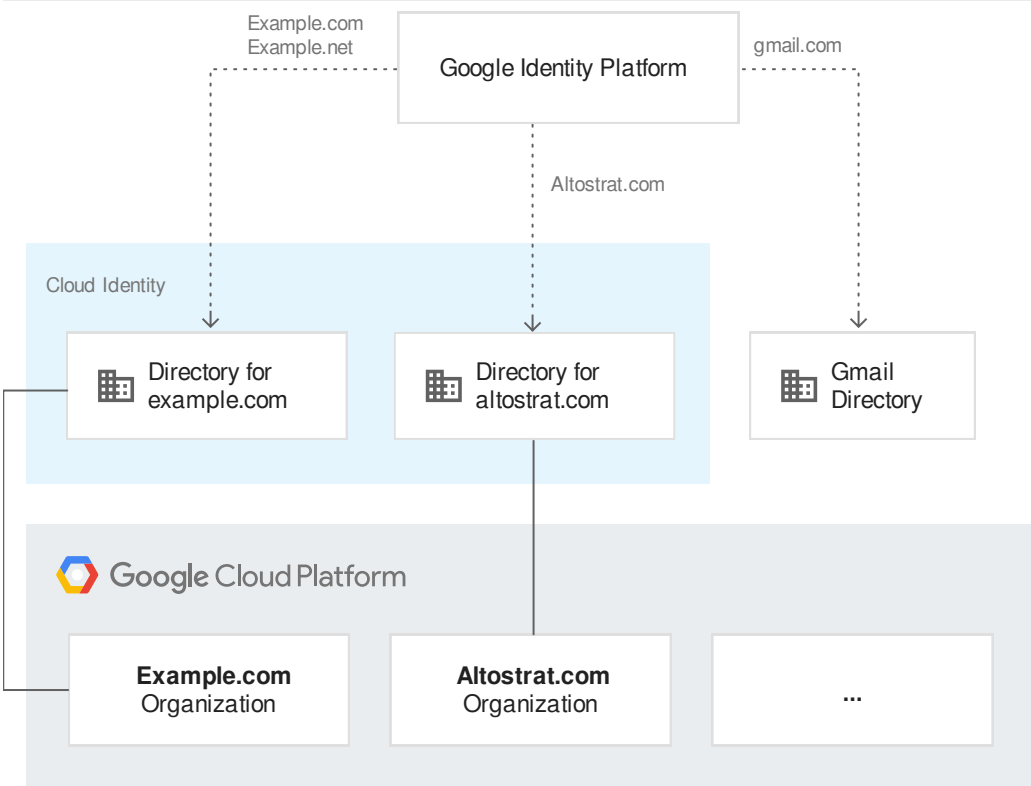
In an Active Directory forest, DNS domains are used in multiple places:

- **Active Directory DNS domains**: Each Active Directory domain corresponds to a DNS domain. This domain might be global, like `corp.example.com`, or can be a local domain name like `corp.local` or `corp.internal`.
- **Mail exchange (MX) domains**: Email addresses use a DNS domain. In some cases, this domain is the same as the Active Directory DNS domain, but in many cases, a different, often shorter, domain such as `example.com` is used. Ideally, users in Active Directory have the email address that is associated with the optional `mail` attribute.
- **UPN suffix domains**: These domains are used for *User Principal Names* (UPN). By default, the Active Directory DNS domain of the user's domain is used to build a UPN. For a user *john* in the domain `corp.example.com`, the default UPN therefore reads `john@corp.example.com`. However, you can configure a forest to use additional DNS domains as UPN suffixes that correspond to neither Active Directory DNS domains nor MX domains. UPNs are optional and are stored in the `userPrincipalName` field of the user.
- **Endpoint domains**: Public-facing servers such as AD FS servers are usually assigned a DNS name, such as `login.external.example.com`. The domain that is used for these purposes can overlap with the MX, UPN suffix, or Active Directory DNS domain, or it can be an entirely different domain.

Usage of DNS domains in Google Cloud

Google Sign-In (https://developers.google.com/identity/), which Google Cloud relies on for authentication, uses email addresses to identify users. Using email addresses not only guarantees that they are globally unique, but also enables Google Cloud to send notification messages to the addresses.

Google Sign-In determines the directory or identity provider to use for authenticating a user based on the domain part of the email addresses, which follows the `@`. For an email address that uses the `gmail.com` domain, for example, Google Sign-In uses the directory of Gmail users for authentication.



When you sign up for a G Suite (https://gsuite.google.com/) or Cloud Identity (/identity) account, you're creating a private directory that Sign-In can use for authentication. In the same way that the Gmail directory is associated with the `gmail.com` domain, G Suite and Cloud Identity accounts need to be associated with a custom domain. Three different kinds of domains are used:

- **Primary domain**: This domain identifies the Cloud Identity or G Suite account and is used as the name for the organization in Google Cloud. When signing up for Cloud Identity or G Suite, you must specify this domain name.
- **Secondary domain**: Along with the primary domain, you can associate other, secondary domains with a Cloud Identity or G Suite account. Each user in the directory is associated

with either the primary domain or one of the secondary domains. Two users, `johndoe@example.com` and `johndoe@secondary.example.com`, are considered separate users if `example.com` is the primary domain and `secondary.example.com` is a secondary domain.

- **Alias domain**: An alias domain is an alternative domain for the primary domain. That is, `johndoe@example.com` and `johndoe@alias.example.com` refer to the same user if `alias.example.com` is set up as an alias domain. An alias domain can only provide an alternative name for the primary domain; it's not possible to add alias domains for secondary domains.

All domains must satisfy the following requirements:

- They must be valid, global DNS domain names. During setup, you might need administrative access to the respective DNS zones in order to verify domain ownership.
- A domain, such as `example.com`, can refer only to a single directory. However, you can use different subdomains, such as `subdomain.example.com`, to refer to different directories.
- Primary and secondary domains should have a valid MX record so that messages sent to email addresses that are formed by using this domain name can be delivered properly.

In order to enable synchronizing between the directories, some mapping is required between the Active Directory domains and the domains that Cloud Identity or G Suite uses. Determining the right mapping depends on how you use Active Directory and requires a closer look at how users are identified in an Active Directory forest and how they can be mapped to Cloud Identity or G Suite.

Mapping users

The third factor to look at when planning to federate Active Directory and Google Cloud is how to map users between Active Directory and Cloud Identity or G Suite.

Identifying users in Active Directory

Internally, Active Directory uses two identifiers to uniquely identify users:

- `objectGUID`: This globally unique ID is generated when a user is created, and never changes.

- `objectSID`: The SID (https://docs.microsoft.com/en-us/windows/desktop/secauthz/security-identifiers), or security identifier, is used for all access checks. While this ID is unique and stable within a domain, it's possible that when moved to a different domain in the forest, a user might be assigned a new SID.

Neither of these IDs is meaningful to users, so Active Directory offers two human-friendly ways to identify users:

- UPN (`userPrincipalName`): The preferred way to identify a user is by UPN. UPNs follow the RFC 822 format of email addresses and are created by combining the username with a UPN suffix domain, as in `johndoe@corp.example.com`. Despite being the preferred way to identify users, UPNs are optional, so some users in your Active Directory forest might lack a UPN.

  Although it's considered a best practice that UPNs be valid email addresses, Active Directory does not enforce this practice.

- Pre–Windows 2000 logon name (`sAMAccountName`): This name combines the NetBIOS domain name and username by using the format ***`domain\user`***, as in `corp\johndoe`. Although these names are considered legacy, they are still commonly used and are the only mandatory identifier of a user.

Notably, Active Directory does not use the user's email address (`mail`) to identify users. Consequently, this field is neither mandatory nor required to be unique in a forest.

All of these identifiers can be changed at any time.

Mapping user identities

Mapping Active Directory users to Cloud Identity or G Suite users requires two pieces of information for each user:

- A stable, unique ID that you can use during synchronization to track which Active Directory user corresponds to which user inCloud Identity or G Suite. On the AD side, the `objectGUID` is perfectly suited for this purpose.
- An email address for which the domain part corresponds to a primary, secondary, or alias domain of your Cloud Identity or G Suite account. Because this email address will be used

throughout Google Cloud, make sure the address is meaningful. Deriving an address from the `objectGUID` is impractical, as are other automatically generated email addresses.

For an Active Directory user, two fields are candidates for providing a Cloud Identity or G Suite email address: `userPrincipalName` and `mail`.

Mapping by User Principal Name

Using the `userPrincipalName` field requires that two criteria be met for all users that are subject to synchronization:

- UPNs must be valid email addresses. All domains that are used as UPN suffix domains also must be MX domains and must be set up so that an email that is sent to a user's UPN is delivered to their email inbox.
- UPN assignments must be complete. All users that are subject to synchronization must have a UPN assigned. The following PowerShell command can help you find users that lack a UPN:

```
Get-ADUser -LDAPFilter "(!userPrincipalName=*)"
```

If these two criteria are met, you can safely map UPNs to Cloud Identity or G Suite email addresses. You can use one of the UPN suffix domains as the primary domain in Cloud Identity or G Suite and add any other UPN suffix domains as secondary domains.

If one of the criteria is not met, you can still map UPNs to Cloud Identity or G Suite email addresses, but the following caveats apply:

- If UPNs are not valid email addresses, users might not receive notification emails that are sent by Google Cloud, which might cause users to miss important information.
- Users without UPNs are ignored during synchronization.
- You can configure the synchronization to replace the UPN suffix domain with a different domain. When you're using multiple UPN suffix domains in a forest, this approach can create duplicates, however, because all UPN suffix domains will be replaced by a single domain. In case of duplicates, only a single user can be synchronized.

A major advantage of using UPNs to map users is that UPNs are guaranteed to be unique across a forest, and they use a curated set of domains, which helps avoid potential synchronization problems.

Mapping by email address

Using the `mail` field requires meeting the following criteria for all users that are subject to synchronization:

- Email assignments must be complete. All users that are subject to synchronization must have the `mail` field populated. The following PowerShell command can help you find users for which this field is not populated:

  ```
  Get-ADUser -LDAPFilter "(!mail=*)"
  ```

- Email addresses must use a curated set of domains, all of which are owned by you. If some of your users use email addresses that refer to partner companies or consumer email providers, those email addresses cannot be used.
- All email addresses must be unique across the forest. Because Active Directory does not enforce uniqueness, you might have to implement custom checks or policies.

If all relevant users meet these criteria, you can identify all domains that are used by these email addresses and use them as primary and secondary domains in Cloud Identity or G Suite.

If one of the criteria is not met, you can still map email addresses to Cloud Identity or G Suite email addresses, with the following caveats:

- During synchronization, users without email addresses will be ignored, as will users with email addresses that use domains that are not associated with the Cloud Identity or G Suite account.
- When two users share the same email address, only one user will be synchronized.
- You can configure the synchronization to replace the domain of email addresses with a different domain. This process can create duplicates, in which case only one user will be synchronized.

Mapping groups

The fourth factor to look at when you're planning to federate Active Directory and Google Cloud is whether to synchronize groups between Active Directory and Google Cloud and how to map them.

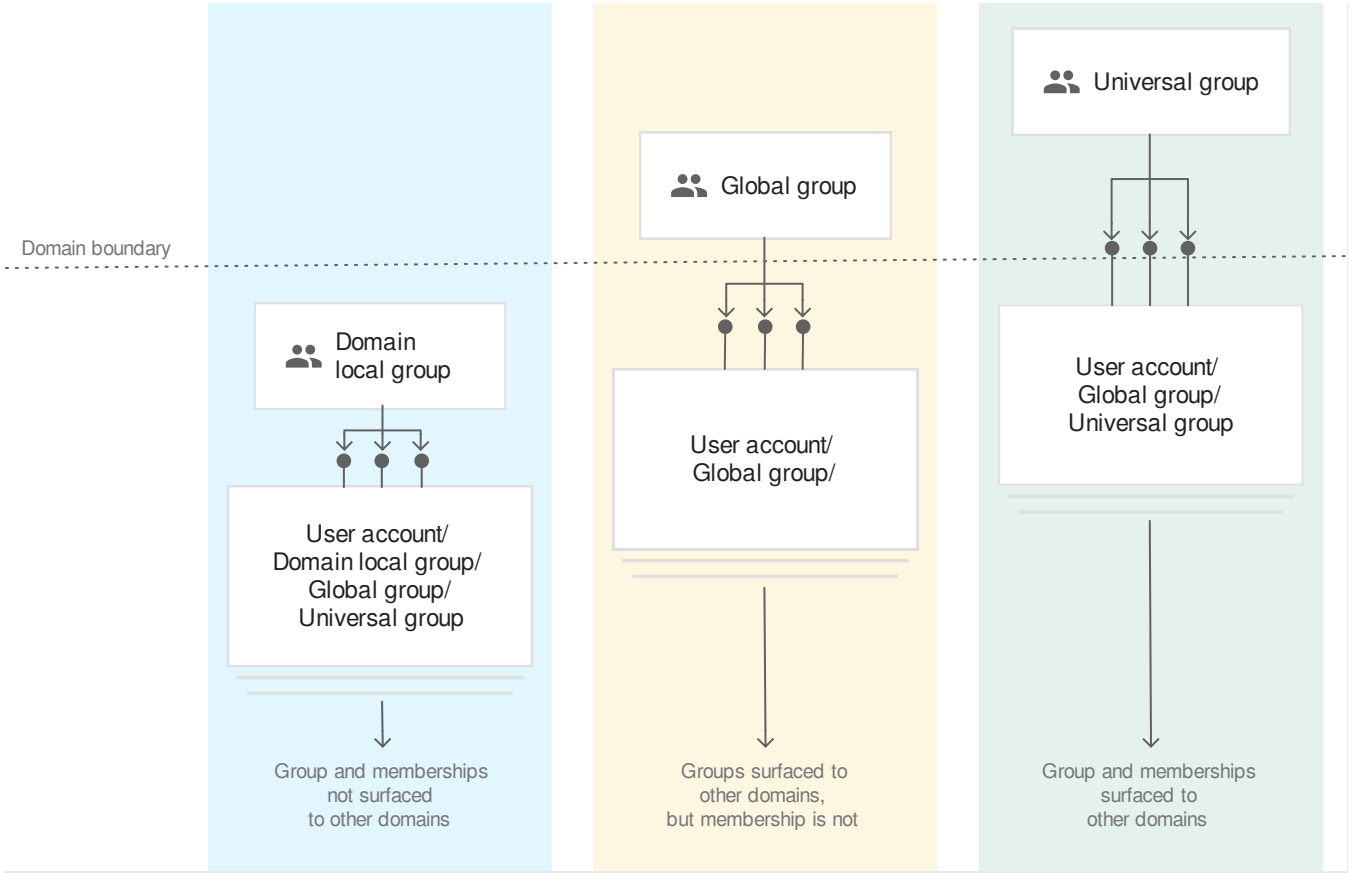
In Google Cloud, groups are commonly used as a way to manage access efficiently (/docs/enterprise/best-practices-for-enterprise-organizations#groups-and-service-accounts) across projects. Rather than assigning individual users to IAM roles in each project, you define a set of groups that model common roles in your organization, and then assign those groups to a set of IAM roles. By modifying the membership of the groups, you can control users' access to an entire set of resources.

Active Directory distinguishes between two kinds of groups: distribution groups and security groups. Distribution groups are used to manage email lists. Synchronizing distribution groups is relevant when you're migrating from Microsoft Exchange to G Suite, so Cloud Directory Sync can handle both regular and dynamic distribution groups. Distribution groups aren't a concern in identity and access management for Google Cloud, however. So this discussion focuses exclusively on security groups.

Mapping groups between Azure AD and Google Cloud is optional. Once you've set up user provisioning, you can create and manage groups directly in Cloud Identity or G Suite, which means that Active Directory remains the central system for identity management but not for access management. To maintain Active Directory as the central system for identity management and access management, we recommend that you synchronize security groups from Active Directory instead of managing them in Cloud Identity or G Suite. With this approach, you can set up IAM so that you can use group memberships in Active Directory to control who has access to certain resources in Google Cloud.

Security groups in Active Directory

Security groups play a foundational role in Windows security and Active Directory access management. This role is facilitated by three different types of Active Directory groups: *domain local groups*, *global groups*, and *universal groups*.



Domain local groups

These groups are relevant only within the scope of a single domain and cannot be referenced in other domains. Because their list of members does not need to be replicated across the forest, domain local groups are the most flexible with respect to the types of members that they can include.

Global groups

These groups are surfaced to and can be referenced in other domains. Their member list is not replicated, however. This limitation restricts the types of members that these groups can include. These groups can only include users and other global groups from the same domain.

Universal groups

These groups, along with their member lists, are replicated across the forest. They can therefore be referenced in other domains and can include not only users and other universal groups but also global groups from all domains.

If your Active Directory forest contains only a single domain, you can synchronize all three types of security groups by using Cloud Directory Sync. If your Active Directory forest uses more than one domain, the type of a group determines whether and how it can be synchronized to Cloud Identity or G Suite.

Because domain local and global groups aren't fully replicated across a forest, global catalog servers contain incomplete information about them. Although this limitation is deliberate and helps to speed up directory replication, it's an obstacle when you want to synchronize such groups to Cloud Identity or G Suite. Specifically, if you configure Cloud Directory Sync to use a global catalog server as a source, then the tool will be able to find groups from all domains across the forest. But only groups that are in the same domain as the global catalog server will contain a membership list and be suitable for replication. To synchronize domain local or global groups in a multi-domain forest, you must run a separate Cloud Directory Sync instance per domain.

Because universal groups are fully replicated across the forest, they don't have this restriction. A single Cloud Directory Sync instance can synchronize universal groups from multiple domains.

Before concluding that you need multiple Cloud Directory Sync instances to synchronize multiple Active Directory domains to Cloud Identity or G Suite, keep in mind that not all groups might need to be synchronized. For this reason, it's worthwhile to look at how different types of security groups are typically used across your Active Directory forest.

Usage of security groups in Active Directory

Resource groups

Windows uses an access model based on access control lists (ACLs). Each resource like a file or LDAP object has an associated ACL that controls which users have access to it. Resources and ACLs are very fine grained, so there are many of them. To simplify the maintenance of ACLs, it's common to create *resource groups* to bundle resources that are frequently used and accessed together. You add the resource group to all affected ACLs once, and manage further access by altering membership of the resource group, not by altering the ACLs.

The resources that are bundled this way typically reside in a single domain. Consequently, a resource group also tends to be referenced only in a single domain, either in ACLs or by other resource groups. Because most resource groups are local, they are usually implemented by using domain local groups in Active Directory.

Role and organization groups

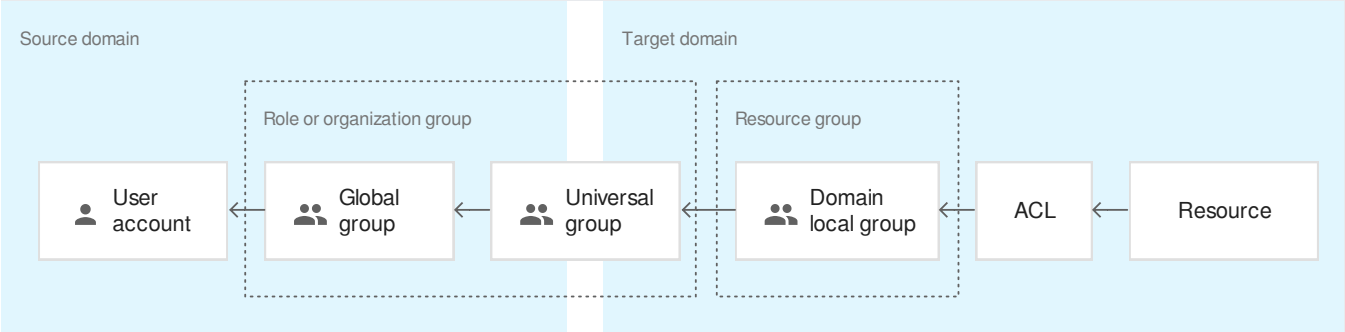
Resource groups help simplify access management, but in a large environment, you might need to add a new user to a large number of resource groups. For this reason, resource groups are commonly complemented by *role groups* or *organization groups*.

Role groups aggregate the permissions that a specific role requires in the organization. A role group that is named Engineering Documentation Viewer, for example, might give members read-only access to all engineering documentation. Practically, you would implement this by creating a role group and making it a member of all resource groups that are used to control access to various kinds of documentation.

In a similar way, organization groups aggregate the permissions that are required by departments within an organization. For example, an organization group that is named Engineering might grant access to all resources that members of the Engineering department commonly require.

Technically, there is no difference between role groups and resource groups, and the two are commonly used in concert. Unlike resource groups, however, role and organization groups can have relevance beyond the boundaries of a domain. To allow such groups to be referenced by resource groups in other domains, role and organization groups are usually implemented by using global groups, which are constrained to members of a single domain, and sometimes complemented by universal groups, which allow members from different domains.

The following diagram shows a nesting pattern that is commonly used in multi-domain Active Directory environments.



Groups in Cloud Identity and G Suite

In Cloud Identity and G Suite, there is only a single type of group. Groups in Cloud Identity and G Suite aren't confined to the scope of the Cloud Identity or G Suite account where they were

defined. Instead, they can include users from different Cloud Identity or G Suite accounts, support being referenced and nested in other accounts, and be used across any Google Cloud organization.

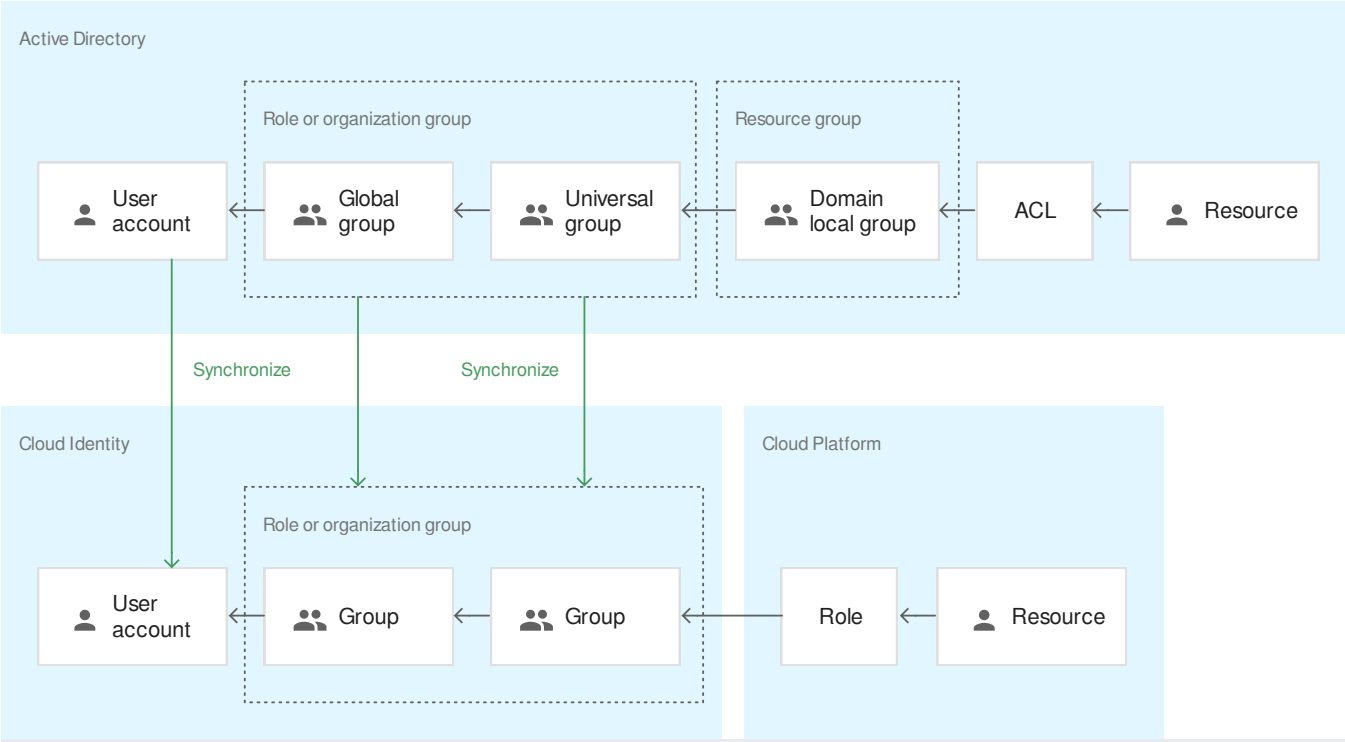
As it does with users, Cloud Identity and G Suite identifies groups by email address. The email address doesn't have to correspond to an actual mailbox, but it must use one of the domains registered for the respective Cloud Identity account.

Unlike Active Directory groups, members of a Cloud Identity or G Suite group are not implicitly granted permission to list other members of the same group. Instead, querying group membership generally requires explicit authorization (https://developers.google.com/admin-sdk/directory/v1/reference/members/list).

Usage of groups in Google Cloud

Google Cloud uses a role-based access model instead of an ACL-based access model. Roles apply to all resources of a certain type that fall within a certain scope. For example, the Kubernetes Engine Developer role has full access to Kubernetes API objects inside all clusters in a given project. Due to the nature of roles, there is little need to maintain resource groups on Google Cloud, and rarely a need to synchronize resource groups to Google Cloud.

Role groups and organization groups are just as relevant in Google Cloud as they are in Active Directory, because they make it easier to manage access for large numbers of users. Synchronizing role and organization groups helps maintain Active Directory as the primary place for managing access.



If you consistently implement resource groups as domain local groups, and role and organization groups as global or universal groups, you can use the group type to ensure that only role and organization groups are synchronized.

The question of whether it's sufficient to run a single Cloud Directory Sync instance per multi-domain forest or whether you need multiple Cloud Directory Sync instances then becomes the question of whether you use global groups. If you implement all your role and organization groups by using universal groups, a single Cloud Directory Sync instance is sufficient; otherwise, you'll need a Cloud Directory Sync instance per domain.

Mapping group identities

Mapping Active Directory security groups to Cloud Identity or G Suite groups requires a common identifier. In Cloud Identity and G Suite, this identifier must be an email address for which the domain part corresponds to a the primary, secondary, or alias domain of the Cloud Identity or G Suite account. Because this email address will be used throughout Google Cloud, the address must be human-readable. The email address doesn't need to correspond to a mailbox.

In Active Directory, groups are identified either by their common name (`cn`) or by a pre–Windows 2000 logon name (`sAMAccountName`). Similar to user accounts, groups can also have an email

address (`mail`), but email addresses are an optional attribute for groups, and Active Directory does not verify uniqueness.

You have two options for mapping group identities between Active Directory and Cloud Identity or G Suite.

Mapping by common name

The advantage of using the common name (`cn`) is that it's guaranteed to be available and, at least within an organizational unit, unique. However, the common name is not an email address, so it needs a suffix `@[DOMAIN]` appended to turn into an email address.

You can configure Cloud Directory Sync to automatically take care of appending a suffix to the group name. Because Active Directory ensures that group names and user names don't conflict, an email address that is derived this way is also unlikely to cause any conflicts.

If your Active Directory forest contains more than a single domain, the following caveats apply:

- If two groups in different domains share a common name, the derived email address will conflict, causing one group to be ignored during synchronization.
- You can only synchronize groups from domains of a single forest. If you synchronize groups from multiple forests by using separate Cloud Directory Sync instances, the email addresses that are derived from the common name don't reflect which forest they correspond to. This ambiguity will cause a Cloud Directory Sync instance to delete any group that has previously been created from a different forest by another Cloud Directory Sync instance.
- You cannot map groups by common name if you use domain substitution for mapping users.

Mapping by email address

Using the email address (`mail`) to map group identities means you must satisfy the same criteria as when using the email address to map users:

- Email assignments must be complete. Although it's common for distribution groups to have an email address, security groups often lack this attribute. To use the email address for mapping identities, security groups that are subject to synchronization must have the

`mail` field populated. The following PowerShell command can help you find accounts for which this field is not populated:

```
Get-ADGroup -LDAPFilter "(!mail=*)"
```

- Email addresses must use a curated set of domains, all of which you own. If some of your users use email addresses that refer to partner companies or consumer email providers, you cannot use those addresses.
- All email addresses must be unique across the forest. Because Active Directory does not enforce uniqueness, you might have to implement custom checks or policies.

If all relevant groups meet these criteria, you can identify all domains that are used by these email addresses and ensure that the list of DNS domains registered in Cloud Identity or G Suite covers these domains.

If one of the criteria is not met, you can still map UPNs to Cloud Identity or G Suite email addresses, with the following caveats:

- Groups without email addresses will be ignored during synchronization, as will email addresses that use domains that aren't associated with the Cloud Identity or G Suite account.
- When two groups share the same email address, only one of them will be synchronized.

Mapping groups by email address is not supported if your Active Directory forest contains more than a single domain and you use domain substitution for mapping users.

Mapping organizational units

Most Active Directory domains make extensive use of organizational units to cluster and organize resources hierarchically, control access, and enforce policies.

In Google Cloud, folders and projects (/resource-manager/docs/creating-managing-folders) serve a similar purpose, although the kinds of resources that are managed within a Google Cloud organization are very different from the resources that are managed in Active Directory. As a result, an appropriate Google Cloud folder hierarchy for an enterprise tends to differ significantly from the structure of organizational units in Active Directory. Automatically

mapping organizational units to folders and projects is therefore rarely practical and not supported by Cloud Directory Sync.

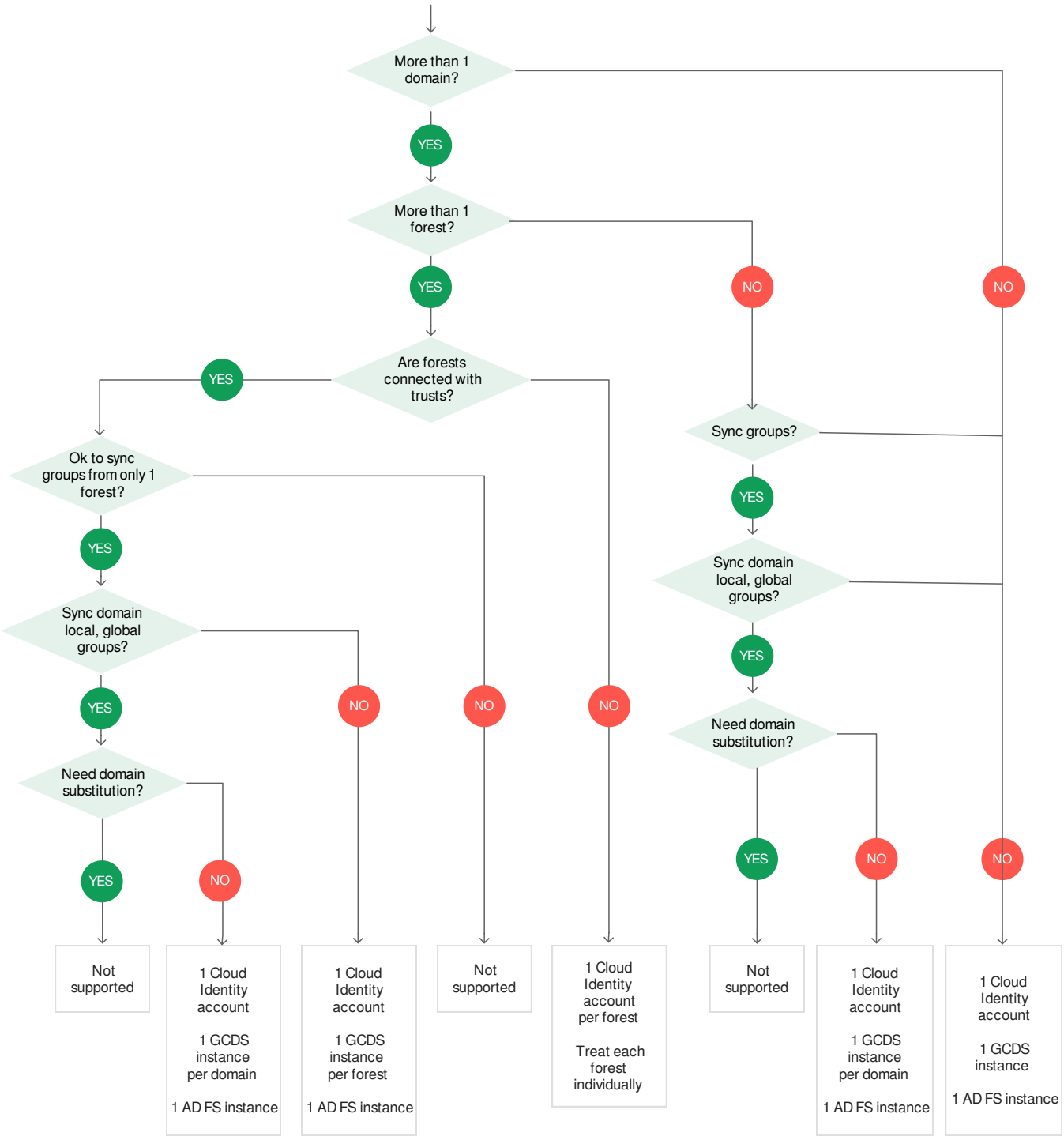
Unrelated to folders, Cloud Identity and G Suite support the concept of organizational units (https://support.google.com/cloudidentity/answer/4352075?hl=en). Organizational units are created to cluster and organize users, similar to Active Directory. But unlike in Active Directory, they apply only to users, not to groups.

Cloud Directory Sync offers the option of synchronizing Active Directory organizational units to Cloud Identity or G Suite. In a setup where Cloud Identity is merely used to extend Active Directory identity management to Google Cloud, mapping organizational units is usually not necessary.

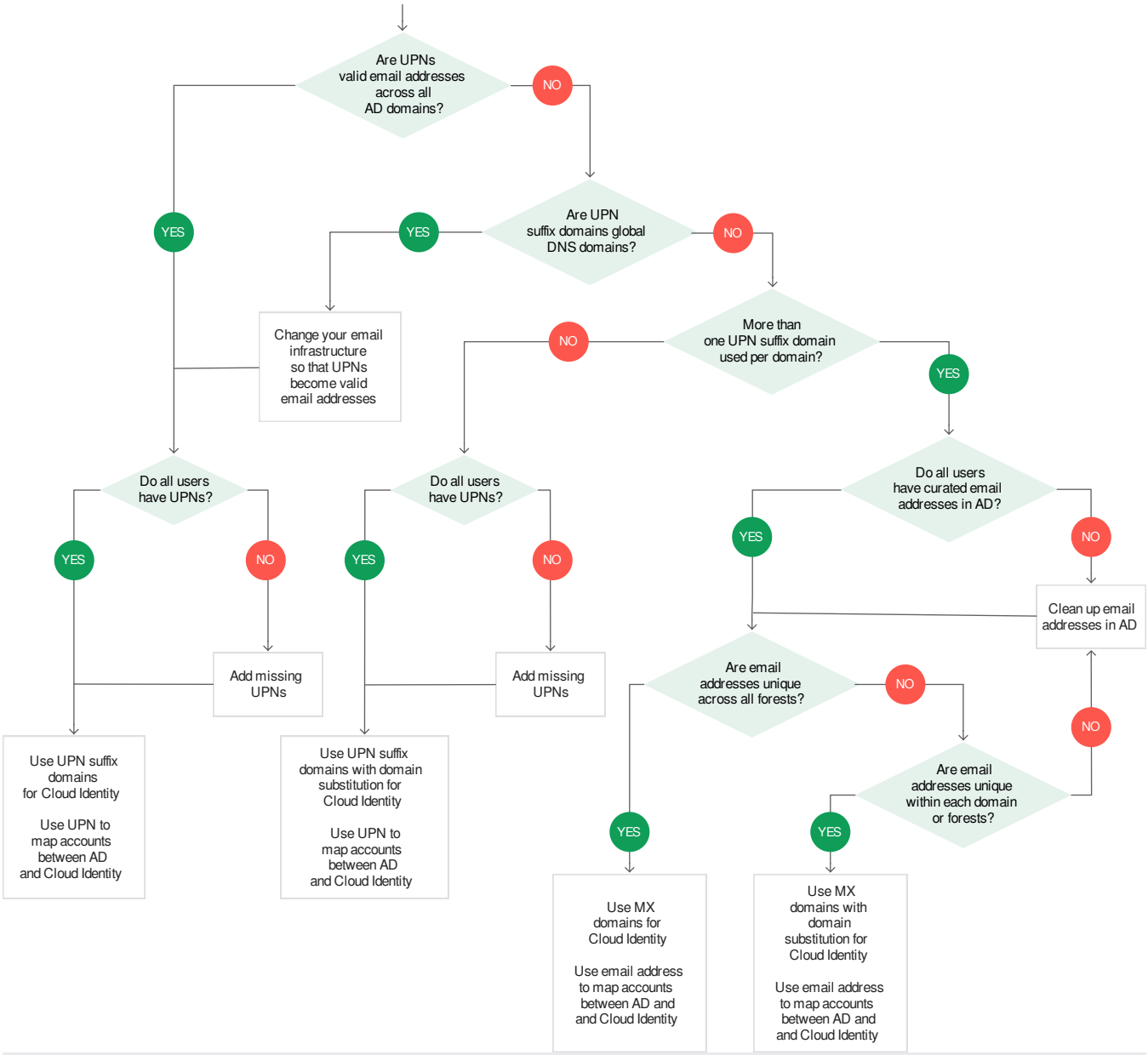
Choosing the right mapping

As noted at the beginning of this article, there is no single best way to map the structures of Active Directory and Google Cloud. To help you choose the right mapping for your scenario, the following decision graphs summarize the criteria that were discussed in the previous sections.

First, refer to the following chart to identify how many Cloud Identity or G Suite accounts, Cloud Directory Sync instances, and AD FS instances or fleets you will need.



Then refer to the second chart to identify the domains to configure in your Cloud Identity or G Suite account.



What's next

- Read about best practices for planning accounts and organizations (/architecture/identity/best-practices-for-planning) and best practices for federating Google Cloud with an external identity provider (/architecture/identity/best-practices-for-federating).
- Configure Cloud Directory Sync to synchronize Active Directory users and groups to Cloud Identity (/architecture/identity/federating-gcp-with-active-directory-synchronizing-user-accounts).

- Configure single sign-on between Active Directory and Google Cloud (/architecture/identity/federating-gcp-with-active-directory-configuring-single-sign-on).
- Lean about best practices for managing super administrator accounts (/resource-manager/docs/super-admin-best-practices)

Last updated 2020-07-21 UTC.