

Registry Groups

- [Group Attributes](#)
 - [Open vs Closed](#)
 - [Automatic Groups](#)
 - [Identifiers](#)
- [Group Permissions](#)
 - [Group Owners](#)
- [Special Groups](#)
 - [Admin Groups](#)
 - [Members Groups](#)
 - [Owners Groups](#)
- [Group Memberships](#)
 - [Group Memberships and Enrollment](#)
 - [Group Nestings](#)
 - [Group Nesting Boolean Logic](#)
 - [Group Membership Validity](#)
 - [Group Membership Reconciliation](#)
- [Changes From Earlier Versions](#)
 - [As of Registry v5.0.0](#)

COmanage Groups are primary Registry objects, and are defined at the CO level. Group Memberships attach at the [Person](#) level, although they may be created from [Person Roles](#) and other sources via various mechanisms.

Broadly, there are two categories of Groups: *Standard* Groups and *System* Groups. System Groups have various special characteristics, and include *Automatic Groups*, *Admin Groups*, and *Owners Groups* (described below).



Standard Groups may not be named beginning with the prefix CO: , which is reserved for System Groups ([AR-Group-9](#)).

Group Attributes

Open vs Closed

An *open* group is one that allows anyone to join. Participants can self-join, no administrator action is required. Memberships in a *closed* group can only be set by the group owner.

In addition, CO Administrators can manage any Group within their CO.

Automatic Groups

Automatic Groups are those which Registry automatically manages the memberships of. Registry manages Automatic Groups for all members and all active members of each CO and each COU. See *Special Groups*, below, for more information.

Identifiers

[Identifiers](#) can be attached to Groups, either manually or via Identifier Assignment.



Identifiers cannot be assigned to Automatic CO Groups. ([CO-1829](#))

Group Permissions

Group Owners

A *Group Owner* has permission to add and remove Group Members to and from the Group, including closed Groups. A Person need not be a Member of a Group to be an Owner (though they do need to be a Member of the *Owners Group*).

Special Groups

Admin Groups

Admin Groups are used to determine [Registry Administrators](#). Admin Groups are automatically created when a CO or COU is created. The Platform Administrator typically sets the initial CO Administrator.

- The Admin Group for the CO is indicated by the group type `GroupEnum::Admins` and a null `cou_id`. The default name for the group is `CO:admins`.
- The Admin Group for a COU is indicated by the group type `GroupEnum::Admins` and a non-null `cou_id`. The default name for COU admin groups is `CO:COU:COU_Name:admins`.

Members Groups

Members Groups are automatic groups that are updated with all members of the CO or COU. Members Groups are automatically created and updated, and cannot be manually manipulated.

- Members of the CO in [Active or Grace Period](#) status are available in the group identified by the group type `GroupEnum::ActiveMembers` and a null `cou_id`. The default name for the group is `CO:members:active`.
- All members of the CO (except those in Deleted status) are available in the group identified by the group type `GroupEnum::AllMembers` and a null `cou_id`. The default name for the group is `CO:members:all`.
- Members of a given COU with an [Active or Grace Period](#) status role are available in the group identified by the group type `GroupEnum::ActiveMembers` and a non-null `cou_id`. The default name for the group is `CO:COU:COU_Name:members:active`.
- All members of a given COU (except those with only roles in Deleted status) are available in the group identified by the group type `GroupEnum::AllMembers` and a non-null `cou_id`. The default name for the group is `CO:COU:COU_Name:members:all`.

Owners Groups

When a new Group is created, a corresponding *Owners Group* is also created. The Group Owners (as described above) for the new Group are managed by adding the designated People to the Owners Group. (If the Person who created the new Group is *not* an administrator, that Person will automatically be added to the Owners Group; [AR-Group-7](#)) While Owners Groups are implemented using the same Group infrastructure as Standard Groups, there are several restrictions on Owners Groups:

1. The *name*, *description*, and *status* of a Group of type *Owners* cannot be manually changed ([AR-Group-4](#)). They will automatically be kept in sync with the primary Standard Group.
2. Additional attributes (such as Identifiers) cannot be attached to a Group of type *Owners* ([AR-Group-5](#)).
3. Groups of type *Owners* cannot be provisioned ([AR-Group-6](#)).
4. When a Group is deleted, its corresponding Owners Group is also deleted ([AR-Group-8](#)).
5. Membership in Owners Groups must be managed by an administrator.

In general, Owners Groups should be treated as "black box" functionality within Registry, and should not be provisioned to or otherwise integrated with external systems. Note that Group Nestings *are* supported for Owners Groups, so that a single Group can be used as a Standard Group (and therefore be provisioned) and also to populate an Owners Group.

Note that non-administrator access to Group Management functionality requires the use of the Group Dashboard Widget ([CFM-316](#)).

Group Memberships

Group Memberships and Enrollment

Section not yet updated.

Group Memberships can be added as part of an Enrollment Flow by adding an attribute of the appropriate type. For more details, see [Registry Enrollment Flow Configuration](#).

Group Memberships can also be added via Organizational Identity Sources when connected to Pipelines.

Group Nestings

Group Nestings allow the Members of one Group (the "nested" or source Group) to be automatically included as Members of another Group (the "target" Group). Group Nestings only confer membership, they cannot be used to manage ownership.

Currently, only CO administrators can create or remove nestings.

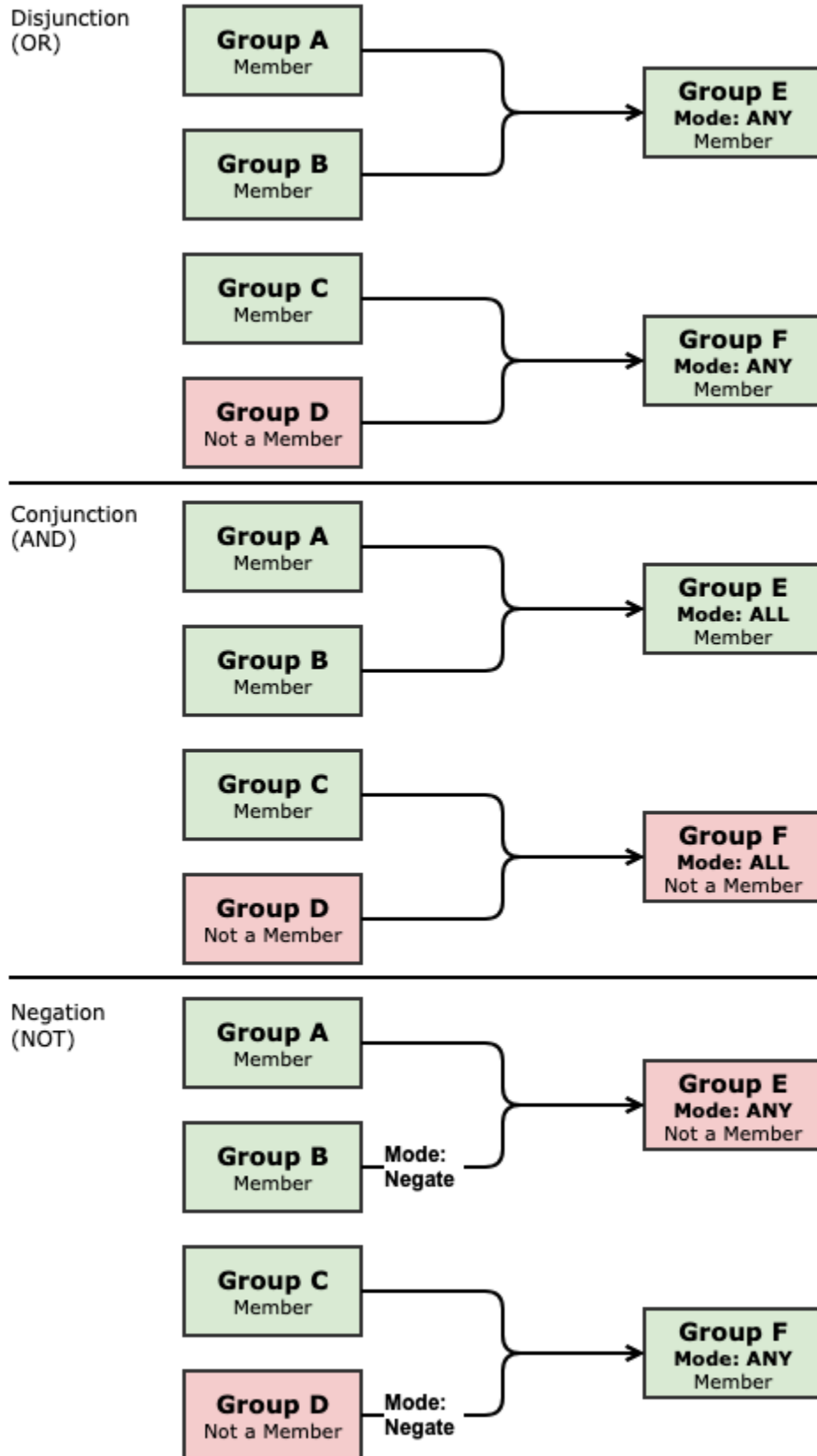
Nested Groups do not imply any sort of hierarchy ([CO-1223](#)).

Group Nesting Boolean Logic

It is possible to apply boolean operators to Group Nestings. The configuration takes place in two places:

1. **AND/OR**: Conjunction and disjunction is configured on the *Target* Group, using the setting *Require All For Nested Memberships*.
 - When enabled, the Person must be a member of **all** nested Groups (and not be a member of **any** exclusion Group) in order to become a member of the target Group.

- Otherwise, the Person may be a member of **any** nested Group (and not be a member of **any** exclusion Group) in order to be a member of the target Group.
 - Changing this setting on the Target Group does not automatically recalculate memberships in the Group.
2. **NOT:** Negation is configured on the Group Nesting itself, via the setting *Negate Nesting*. This setting can be used to create an "exclusion" group, or "negative memberships".
- When enabled, members of the nested Group **can not** become members of the target Group via nesting. (The Person can still be added directly to the target Group manually.)
 - This setting does not impact People who are *not* members of the source Group.
 - Changing this setting does not automatically recalculate memberships in the Group.



Group Memberships can have *Valid From* and *Valid Through* dates attached. See [Registry Validity Dates](#) for more information.

Group Nestings honor validity dates. If a Membership is not yet valid or has expired, it will not propagate to the Target Group. When a Group Member validity date takes effect, Group Nesting changes will not automatically happen. The [ValidateGroupMember](#) job must be used to process these changes.

Group Membership Reconciliation

In general, nested Group Memberships and Memberships of automatic Groups are updated in real time as needed. However, If an automatic Group or a Group with Group Nestings appears to have incorrect Group Memberships, the Group may be manually reconciled to fix incorrect memberships. To reconcile a Group, find the desired Group and click **Reconcile**.

Manually reconciling a Group will *not* automatically reconcile related Groups. For example, if Group A has nested Group B which in turn has nested Group C, and Group C is manually reconciled, it will probably also be necessary manually reconcile Group B.

Changes From Earlier Versions

As of Registry v5.0.0

- Group Ownership is managed via *Owners Groups*.