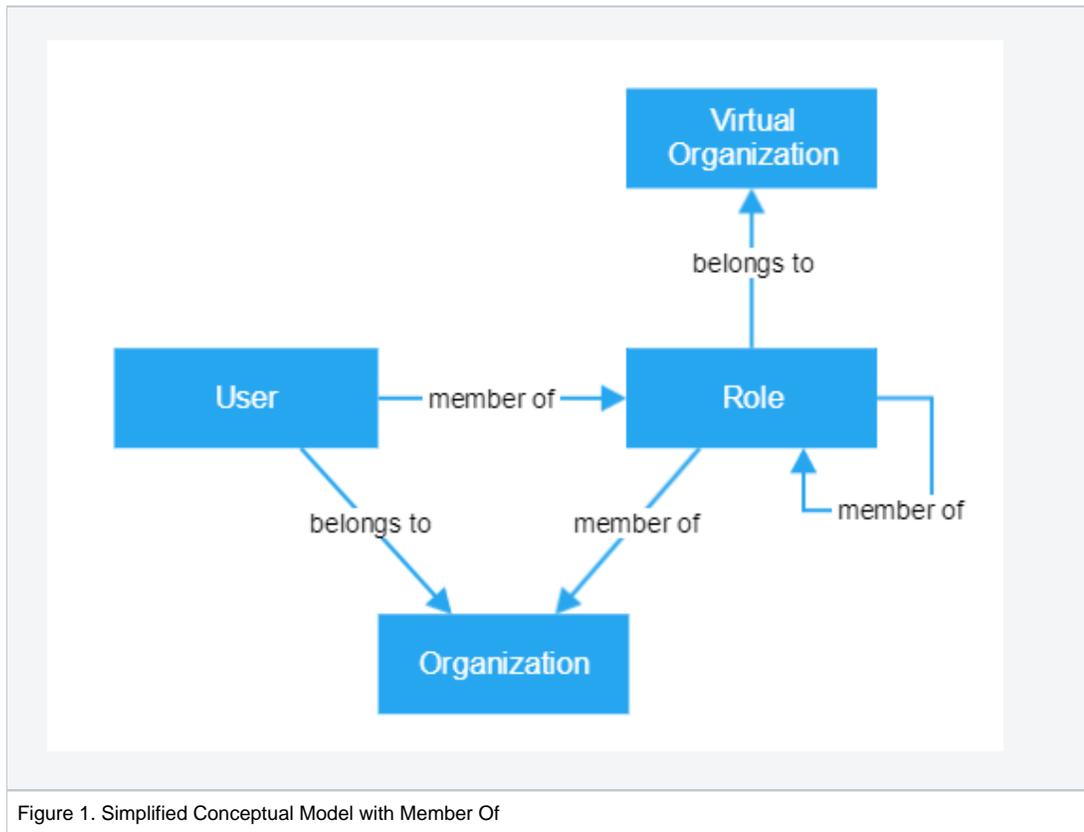# Role hierarchy rules - CustomerID

Figure 1. Simplified Conceptual Model with Member Of

Role hierarchy rules allow you to make automatic role to role mappings. Role hierarchy rules are conditional statements that first match their conditions against all the existing organizations in the system and on finding a match, make the appropriate changes or additions to the organizations specified by the target statements. The conditions that have to be met by the rules are specified using the source and target statements. Only the target role statement is unconditional, and required; all other statements impose successive constraints that have to be met for the rule to be applied.

These configurations are available in the `eidm2.properties` file.

role.hierarchy.N.source.role, role.hierarchy.N.target.role

This property sets the source/target role. It is mandatory to set the source and the target role.
Default is `<not set>`
Example:

```
# All OrganizationMainUsers are also OrganizationUsers
# in the same Organization.
role.hierarchy.1.source.role = OrganizationMainUser
role.hierarchy.1.target.role = OrganizationUser
```

This property sets the source/target organization associated with the role hierarchy rule.
Default is `<not set>`
Example:

```
# All OrganizationUsers of Org1 are also OrganizationUsers of Org2.
role.hierarchy.2.source.role = OrganizationUser
role.hierarchy.2.source.organization = Org1
role.hierarchy.2.target.role = OrganizationUser
role.hierarchy.2.target.organization = Org2
```

role.hierarchy.N.source.organization.type

This property sets the source organization type (was previously called "class"). This part of the rule will be matched if an organization has the organization type that has been set here.
Default is `<not set>`
Example:

```
# OrganizationUsers of 'testType' type organizations are also OrganizationUsers in Org1.
role.hierarchy.3.source.role = OrganizationUser
role.hierarchy.3.source.organization.type = testType
role.hierarchy.3.target.role = OrganizationUser
role.hierarchy.3.target.organization = Org1
```

role.hierarchy.N.target.organization.type

This property sets the organization type (was previously called "class") of the target organization. The matching source roles will have target roles in all organizations that belong to this organization type.
Default is `<not set>`
Example:

```
# Every UserReviewer in Org2 is also a UserReviewer
# in every 'reviewed' type organization.
role.hierarchy.4.source.role = UserReviewer
role.hierarchy.4.source.organization = Org2
role.hierarchy.4.target.role = UserReviewer
role.hierarchy.4.target.organization.type = reviewed
```

role.hierarchy.N.source.organization.virtual

If this property is set to true, only virtual organizations will be matched for the target rules. Otherwise, only physical organizations will be matched for the target rules. If not defined, neither restriction will exist.
Default is `<not set>`
Example:

```
# OrganizationUsers of physical organizations are also OrganizationUsers of Org3.
role.hierarchy.7.source.role = OrganizationUser
role.hierarchy.7.source.organization.virtual = false
role.hierarchy.7.target.role = OrganizationUser
role.hierarchy.7.target.organization = Org3
```

role.hierarchy.N.target.organization.virtual

If this property is set to true, only virtual organizations will be selected from the matching target organizations. Otherwise, only physical organizations will be selected. If not defined, neither restriction will exist.
Default is `<not set>`
Example:

```
# OrganizationMainUsers in any organization are also OrganizationUsers in# the virtual organizations of the
organization type 'type8'.
role.hierarchy.8.source.role = OrganizationMainUser
role.hierarchy.8.target.organization.virtual = true
role.hierarchy.8.target.organization.class = type8
role.hierarchy.8.target.role = OrganizationUser
```

role.hierarchy.N.target.organization.ancestor

If this property is set to true, the target organizations only include organizations that are among the ancestors of a source organization. That is, the target organizations include the parent of the source organization, its parent, and so on, until we reach a top-level organization. If, on the other hand, the key is set to false, we exclude the ancestor organizations of every other organization from the list, to be further limited by other source or target statements.
Default is `<not set>`
Example:

```
# OrganizationMainUsers are also OrganizationUsers in their ancestor organizations.
role.hierarchy.9.source.role = OrganizationMainUser
role.hierarchy.9.target.organization.ancestor = true
role.hierarchy.9.target.role = OrganizationUser
```

role.hierarchy.N.target.organization.descendant

If this property is set to true, the target organizations only include organizations that are among the descendants of a source organization. That is, the target organizations include the sub-organizations (or the children) of the source organization, all the sub-organizations of those sub-organizations, and so on, until there are no further sub-organizations to include. If the key is set to false, we match every organization *except* the descendants of the source organizations.
Default is `<not set>`
Example:

```
# OrganizationMainUsers in any organization are also OrganizationMainUsers in the descendant organizations of
their organizations.
role.hierarchy.10.source.role = OrganizationMainUser
role.hierarchy.10.target.organization.descendant = true
role.hierarchy.10.target.role = OrganizationMainUser
```

role.hierarchy.N.target.organization.level

This property sets the level in which the target organization has to be in the hierarchy. Level 1 corresponds to top-level organizations, level 2 to organizations that have a parent on level 1, and so on.
Default is `<not set>`
Example:

```
# OrganizationMainUser is also UserReviewer of the top level organization
role.hierarchy.12.source.role = OrganizationMainUser
role.hierarchy.12.target.role = UserReviewer
role.hierarchy.12.target.organization.ancestor = true
role.hierarchy.12.target.organization.virtual = false
role.hierarchy.12.target.organization.level = 1
```