

Using SQL to do things in Grouper

Wiki Home	Grouper Release Announcements	Grouper Guides	Grouper Deployment Guide	Community Contributions	Internal Developer Resources
---------------------------	---	--------------------------------	--	---	--

Overview

For Grouper 2.6 and above, please see [this page for SQL provisioning](#).

In more advanced/complex use case you may find it useful to use SQL to explore the Grouper data and or to achieve very complex tasks. This page is a Catch All page for things that SQL can be used to do. Expand as needed

SQL Loader Job to roll up group(s) using attributes as markers on groups to be rolled up

You might find it useful to use an SQL loader job that could create groups of groups or other kinds of constructs based on the Grouper Attribute Framework. This kind of a construct can let you use attributes to drive how and what the Loader job does and is really only limited by your SQL knowledge and your imagination for how to use the Attribute Framework.

As a simple example: Let's say that you create an attribute (etc:local:DepartmentXGroup) that you allow "normal grouper users" to assign to a group that should be considered "lists of people in Department X". And you want to maintain a reference group "ref:All_Departments:DepartmentX" that should have all groups that are "marked" with the "DepartmentXGroup".

You can use an SQL statement like the following (example MYSQL,)

find groups tagged by an attributeName

```
select group_name From grouper_attr_asn_group_v where enabled = 'T' and attribute_def_name_name = 'etc:local:DepartmentXGroup'
```

This example clearly does not scale well. You would need to create SQL loader jobs per attributeName (AKA: "Department Marker"). However, it could be expanded to looking for a set of markers that could also "group by" the attribute_def_name_name or use other SQL constructs to create the groupName value for the SQL Loader job. too. (Again, only limited by your S:QL knowledge and imagination. 😊