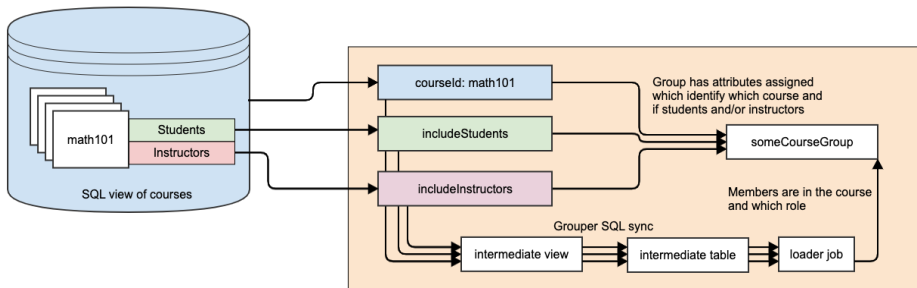


Grouper selective loader from attributes example

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

The design here is to not load all groups from a loader, but be able to selectively load groups as needed, with certain options, and located anywhere in the registry.

An example of this use case is someone can mark an arbitrary group as "load people from a course", while also selecting if that means students, instructors, or both.



Sample data

Add the groups, some memberships, and the attributes.

```
import edu.internet2.middleware.grouper.*;
import edu.internet2.middleware.grouper.attr.*;
import edu.internet2.middleware.grouper.attr.assign.*;
import edu.internet2.middleware.grouper.attr.finder.*;

GrouperSession grouperSession = GrouperSession.startRootSession();

AttributeDef attributeDef = new AttributeDefSave(grouperSession).assignName("attr:courseDef").
assignCreateParentStemsIfNotExist(true).assignCreateParentStemsIfNotExist(true).assignToGroup(true).
assignAttributeDefType(AttributeDefType.attr).assignMultiAssignable(false).assignMultiValued(false).
assignValueType(AttributeDefValueType.marker).save();
AttributeDef attributeValueDef = new AttributeDefSave(grouperSession).assignName("attr:courseValueDef").
assignCreateParentStemsIfNotExist(true).assignToGroupAssn(true).assignAttributeDefType(AttributeDefType.attr).
assignMultiAssignable(false).assignMultiValued(false).assignValueType(AttributeDefValueType.string).save();

AttributeDef attributeValueMarkerDef = new AttributeDefSave(grouperSession).assignName("attr:
courseValueMarkerDef").assignCreateParentStemsIfNotExist(true).assignToGroupAssn(true).assignAttributeDefType
(AttributeDefType.attr).assignMultiAssignable(false).assignMultiValued(false).assignValueType
(AttributeDefValueType.marker).save();

attributeDef.getAttributeDefActionDelegate().configureActionList("assign");
attributeValueDef.getAttributeDefActionDelegate().configureActionList("assign");
attributeValueMarkerDef.getAttributeDefActionDelegate().configureActionList("assign");

course = new AttributeDefNameSave(grouperSession, attributeDef).assignName("attr:course").
assignCreateParentStemsIfNotExist(true).save();

attributeValueDef.getAttributeDefScopeDelegate().assignOwnerNameEquals("attr:course");
attributeValueMarkerDef.getAttributeDefScopeDelegate().assignOwnerNameEquals("attr:course");

courseId = new AttributeDefNameSave(grouperSession, attributeValueDef).assignName("attr:courseId").
assignCreateParentStemsIfNotExist(true).save();
includeStudents = new AttributeDefNameSave(grouperSession, attributeValueMarkerDef).assignName("attr:
includeStudents").assignCreateParentStemsIfNotExist(true).save();
includeInstructors = new AttributeDefNameSave(grouperSession, attributeValueMarkerDef).assignName("attr:
includeInstructors").assignCreateParentStemsIfNotExist(true).save();

Group math101 = new GroupSave(grouperSession).assignName("org:artsAndSciences:courses:math101").
```

```

assignCreateParentStemsIfNotExist(true).save();

attributeAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType(AttributeAssignType.
group).assignAttributeDefName(course).assignOwnerGroup(math101);

attributeAssignOnAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType
(AttributeAssignType.group_asgn).assignAttributeDefName(courseId).addValue("math101");

attributeAssignSave.addAttributeAssignOnThisAssignment(attributeAssignOnAssignSave);

attributeAssignOnAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType
(AttributeAssignType.group_asgn).assignAttributeDefName(includeStudents);

attributeAssignSave.addAttributeAssignOnThisAssignment(attributeAssignOnAssignSave);

attributeAssignSave.save();

Group clinical101 = new GroupSave(grouperSession).assignName("org:nursing:classes:clinical101").
assignCreateParentStemsIfNotExist(true).save();

attributeAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType(AttributeAssignType.
group).assignAttributeDefName(course).assignOwnerGroup(clinical101);

attributeAssignOnAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType
(AttributeAssignType.group_asgn).assignAttributeDefName(courseId).addValue("clinical101");

attributeAssignSave.addAttributeAssignOnThisAssignment(attributeAssignOnAssignSave);

attributeAssignOnAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType
(AttributeAssignType.group_asgn).assignAttributeDefName(includeStudents);

attributeAssignSave.addAttributeAssignOnThisAssignment(attributeAssignOnAssignSave);

attributeAssignOnAssignSave = new AttributeAssignSave(grouperSession).assignAttributeAssignType
(AttributeAssignType.group_asgn).assignAttributeDefName(includeInstructors);

attributeAssignSave.addAttributeAssignOnThisAssignment(attributeAssignOnAssignSave);

attributeAssignSave.save();

```

Attribute assignments on groups

Note, if you delegate who has access to the course attribute def names, then others can control their own groups

Two groups with attributes:

Home > Root > org > artsAndSciences > courses > math101


math101

More actions ▾

More ▾

MembersPrivilegesMore ▾

Attribute assignments

+ Assign attribute

The following table lists all attributes assigned to this group

Assignment type	Attribute name	Enabled?	Assignment values	Attribute definition	Choose action
Direct assignment	 course	enabled		 courseDef	Actions ▾
Metadata on assignment	 courseId	enabled	math101 ▾	 courseValueDef	Actions ▾
Metadata on assignment	 includeStudents	enabled		 courseValueMarkerDef	Actions ▾

Home > Root > org > nursing > classes > clinical101


clinical101

More actions ▾

More ▾

MembersPrivilegesMore ▾

Attribute assignments

+ Assign attribute

The following table lists all attributes assigned to this group

Assignment type	Attribute name	Enabled?	Assignment values	Attribute definition	Choose action
Direct assignment	 course	enabled		 courseDef	Actions ▾
Metadata on assignment	 courseId	enabled	clinical101 ▾	 courseValueDef	Actions ▾
Metadata on assignment	 includeInstructors	enabled		 courseValueMarkerDef	Actions ▾
Metadata on assignment	 includeStudents	enabled		 courseValueMarkerDef	Actions ▾

Control views

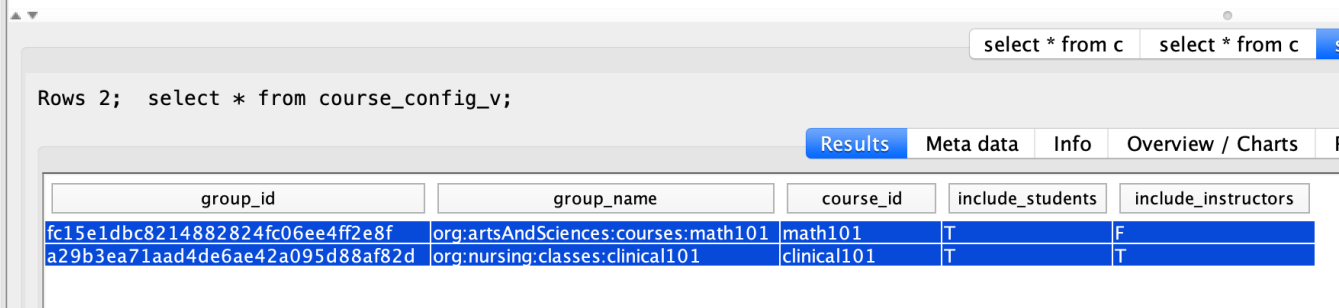
The attribute assignment queries are complicated and have performance implications. So to simplify things, we will make a view, and sync that into a table, so when we do the loader query, it is simpler and performs better.

View for which groups are courses

Note, this is for mysql but will either work or will work with minor edits in other databases

```
create view course_config_v as
select distinct gaagv.group_id, gaagv.group_name,
(select gaaagv.value_string from grouper_attr_asn_asn_group_v gaaagv where gaaagv.attribute_def_name_name2 =
'attr:courseId' and gaaagv.group_name = gaagv.group_name and gaaagv.enabled2 = 'T' limit 1)
as course_id,
case when exists
(select 1 from grouper_attr_asn_asn_group_v gaaagv where gaaagv.attribute_def_name_name2 = 'attr:
includeStudents' and gaaagv.group_name = gaagv.group_name and gaaagv.enabled2 = 'T')
then 'T'
else 'F'
end as include_students,
case when exists
(select 1 from grouper_attr_asn_asn_group_v gaaagv where gaaagv.attribute_def_name_name2 = 'attr:
includeInstructors' and gaaagv.group_name = gaagv.group_name and gaaagv.enabled2 = 'T')
then 'T'
else 'F'
end as include_instructors
from grouper_attr_asn_group_v gaagv
where gaagv.attribute_def_name_name = 'attr:course'
and gaagv.enabled = 'T';
```

```
select * from course_config_v;
```



Rows 2; select * from course_config_v;

group_id	group_name	course_id	include_students	include_instructors
fc15e1dbc8214882824fc06ee4ff2e8f	org:artsAndSciences:courses:math101	math101	T	F
a29b3ea71aad4de6ae42a095d88af82d	org:nursing:classes:clinical101	clinical101	T	T

Control table

To simplify things and improve performance, we will sync this view into a table

Convert to table:

```
create table course_config as select * from course_config_v;
ALTER TABLE course_config MODIFY group_id varchar(40);
ALTER TABLE course_config ADD PRIMARY KEY(group_id);
CREATE INDEX course_config_name_idx ON course_config (group_name);
```

Sync the view to table.

grouper.client.properties

```
grouperClient.syncTable.courseConfig.databaseFrom = grouper
grouperClient.syncTable.courseConfig.tableFrom = course_config_v
grouperClient.syncTable.courseConfig.databaseTo = grouper
grouperClient.syncTable.courseConfig.tableTo = course_config
grouperClient.syncTable.courseConfig.columns = *
grouperClient.syncTable.courseConfig.primaryKeyColumns = group_id
```

Lets schedule this hourly at 50 minutes passed the hour

grouper-loader.properties

```
otherJob.courseConfigFull.class = edu.internet2.middleware.grouper.app.tableSync.TableSyncOtherJob
otherJob.courseConfigFull.quartzCron = 0 50 * * * ?
otherJob.courseConfigFull.grouperClientTableSyncConfigKey = courseConfig
otherJob.courseConfigFull.syncType = fullSyncFull
```

Run the job and make sure it is ok

Home > Miscellaneous > All daemon jobs

All daemon jobs

More actions ▾

Filter for:

Common filters ▾

☐ Show extended results?

☐ Show only errors?

Apply filter

Reset

Next refresh: 25

Job name	State	Overall status	Last run status	Actions	Schedule
OTHER_JOB_courseConfigFull	ENABLED	SUCCESS	SUCCESS	Job actions ▾	CRON: 0 50 *** ? At 50 minutes past the hour

Show:

Showing 1-1 of 1 · [First](#) | [Prev](#) | [Next](#) | [Last](#)

Course loader data

Normally this would be a view against the SIS or an ETL to copy that data to a warehouse or something. Here we will just make some sample data

```
CREATE TABLE courses (course_id VARCHAR(20), role VARCHAR(20), subject_id VARCHAR(20));

ALTER TABLE courses ADD PRIMARY KEY(course_id, role, subject_id);

insert into courses (course_id, role, subject_id) values ('english101', 'student', 'test.subject.0');
insert into courses (course_id, role, subject_id) values ('english101', 'instructor', 'test.subject.1');
insert into courses (course_id, role, subject_id) values ('clinical101', 'student', 'test.subject.2');
insert into courses (course_id, role, subject_id) values ('clinical101', 'student', 'test.subject.3');
insert into courses (course_id, role, subject_id) values ('clinical101', 'instructor', 'test.subject.4');
insert into courses (course_id, role, subject_id) values ('math101', 'student', 'test.subject.5');
insert into courses (course_id, role, subject_id) values ('math101', 'student', 'test.subject.6');
insert into courses (course_id, role, subject_id) values ('math101', 'instructor', 'test.subject.7');
commit;
```

Membership view

This view has the loader results to load the groups via attributes.

```

create view courses_memberships_v as
select courses.subject_id,
       'jdbc' as subject_source_id,
       course_config.group_name
from courses,
     course_config
where
  course_config.course_id = courses.course_id and course_config.course_id is not null
  and (course_config.include_students = 'T' or course_config.include_instructors = 'T')
  and ((course_config.include_students = 'T' and courses.role = 'student')
       or (course_config.include_instructors = 'T' and courses.role = 'instructor'));

```

```
select subject_id, subject_source_id, group_name from courses_memberships_v;
```

Rows 5; select subject_id, subject_source_id, group_name from courses_memberships_v;

Results Meta data

subject_id	subject_source_id	group_name
test.subject.4	jdbc	org:nursing:classes:clinical101
test.subject.2	jdbc	org:nursing:classes:clinical101
test.subject.3	jdbc	org:nursing:classes:clinical101
test.subject.5	jdbc	org:artsAndSciences:courses:math101
test.subject.6	jdbc	org:artsAndSciences:courses:math101

Loader job

courseLoader

More actions ▾

More ▾

Members

Privileges

More ▾

Loader settings

Loader actions ▾

This group has loader configuration

State	ENABLED
Source type	SQL pull the members from a SQL database. Can be SQL or LDAP
Loader type	SQL_GROUP_LIST the SQL query loads the members of multiple groups. Can be SQL_SIMPLE or SQL_GROUP_LIST
Database name	grouper jdbc:mysql://localhost:3306/grouper_v2_5?useSSL=false server ID that is configured in the grouper-loader.properties that identifies the connection information to the database server. Note: "grouper" means use the Grouper registry database connection.
SQL query	select subject_id, subject_source_id, group_name from courses_memberships_v query for memberships. Since this is SQL_GROUP_LIST, the required GROUP_NAME column holds the group system name of the membership. the SUBJECT_ID or SUBJECT_IDENTIFIER or SUBJECT_ID_OR_IDENTIFIER column is required, and the SUBJECT_SOURCE_ID column is optional (but recommended for better performance). SUBJECT_ID has the best performance, and SUBJECT_IDENTIFIER and SUBJECT_ID_OR_IDENTIFIER are slower since they require subject API lookups. If the data has group names as members, it must be in a SUBJECT_IDENTIFIER column.
Schedule type	CRON Cron setting runs on a certain schedule. Can be CRON (recommended) or START_TO_START_INTERVAL
Schedule	0 0 * * * ? Every hour
Priority	this job has the default and middle priority of 5 (higher numbers have a higher priority)
Require members in other group(s)	
Group query (metadata on groups)	query (optional) for SQL_GROUP_LIST which should return cols: group_name, group_display_name (optional), group_description (optional). This should return all groups managed by this job. The name and display name are the full folder path. If there is a column named any of the following: readers, viewers, admins, updaters, optins, optouts, group_attr_readers, group_attr_updaters, then the data in the column (comma separated subjectId's or subjectIdentifiers which can include group names) will be assigned to that group's privilege list. Note you can use inherited privileges on a folder instead.

Home > Root > ref > student > courseLoader

 **courseLoader**

More actions ▾

More ▾

MembersPrivilegesMore ▾

Loader logs

Loader actions ▾

Filter for:

Start time between:

yyyy-mm-dd hh:mi:ss

yyyy-mm-dd hh:mi:ss

End time between:

yyyy-mm-dd hh:mi:ss

yyyy-mm-dd hh:mi:ss

Last updated between:

yyyy-mm-dd hh:mi:ss

yyyy-mm-dd hh:mi:ss

Subjobs:

☐ Show subjobs

Status:

☐ Success ☐ Error ☐ Started ☐ Running ☐ Warning ☐ Config error ☐ Subject problems

Number of rows:

400



Apply filter

Reset

1 logs found for job name: SQL_GROUP_LIST__ref:student:courseLoader__369201e4b008407a89ee30cbe20ba5c9

Status	Loaded group	Job type	Start time	End time	Millis	Millis get data	Millis load data	Total count	Add count	Update count	Delete count	Unresolvable count	Log ID
Success	multiple	overall	2020-06-17 14:51:53.0	2020-06-17 14:51:53.0	153	19	134	5	0	0	0	5	8ae421f872

See memberships on any groups with course attributes

math101

[+ Add members](#)

[More actions ▾](#)

math101 auto-created by grouperLoader

This group is managed by loader group  **courseLoader**. It was last fully loaded on Wed Jun 17 14:55:08 EDT 2020. Summary is: total: 2 inserted: 2 deleted: 0 updated: 0

More ▾

Members

Privileges

More ▾

The following table lists all entities which are members of this group.

Filter for: All members ▾ Member name Apply filter Reset Advanced

[Remove selected members](#)

☐	Entity name ▾	Membership	Choose action
☐	 my name is test.subject.5	Direct	Actions ▾
☐	 my name is test.subject.6	Direct	Actions ▾

Show: 50 ▾

Showing 1-2 of 2 · [First](#) | [Prev](#) | [Next](#) | [Last](#)

[+ Add members](#)[More actions ▾](#)

clinical101 auto-created by grouperLoader

This group is managed by loader group **courseLoader**. It was last fully loaded on Wed Jun 17 14:55:08 EDT 2020. Summary is: total: 3 inserted: 3 deleted: 0 updated: 0

[More ▾](#)[Members](#)[Privileges](#)[More ▾](#)

The following table lists all entities which are members of this group.

Filter for:

[All members ▾](#)[Apply filter](#)[Reset](#)[Advanced](#)[Remove selected members](#)

☐ Entity name ▾	Membership	Choose action
☐ my name is test.subject.2	Direct	Actions ▾
☐ my name is test.subject.3	Direct	Actions ▾
☐ my name is test.subject.4	Direct	Actions ▾

Show:

[50 ▾](#)Showing 1-3 of 3 · [First](#) | [Prev](#) | [Next](#) | [Last](#)

Real time updates

This is not implemented yet but you could easily have a change log consumer that looks for attribute changes for the applicable attributes, and runs a full loader job based on events. Ask the Grouper team for an example if this is a requirement. Otherwise the above will run periodically (this example is hourly)