v2.1 Release Notes

Release Notes for Grouper v2.1

Grouper v2.1.5 includes 11 fixes and improvements over v2.1.4. See the full list in Jira.

Grouper v2.1.4 includes 19 fixes and improvements over v2.1.3. See the full list in Jira.

Grouper v2.1.3 includes 20 fixes and improvements over v2.1.2. See the full list in Jira.

Grouper v2.1.2 includes 10 fixes and improvements over v2.1.1. See the full list in Jira.

Grouper v2.1.1 includes 31 fixes and improvements over v2.1.0. See the full list in Jira.

Grouper v2.1.0 includes 55 fixes and improvements over v2.0.3. See the full list in Jira.

New Features

PSP	Changes to user access can now happen in real time courtesy of real time and incremental provisioning based on Grouper's changelog. Group and folder moves and renames can be provisioned as well. This is accomplished with the new Provisioning Service Provider (PSP), which replaces LDAPPCNG.
Provision from LDAP	Grouper can now be updated from LDAP via PSP or Grouper Loader.
More Web Services	Improved web services support for applications that outsource their internal access management to Grouper: Operations 1,2,3,4
Local Entities	Improved management of access by service principals to info stored in Grouper using a new "local entity" object type.
Grouper Installer	It's quick and easy to get started using Grouper with the new installer. Really!
Subject Filter and Attribute Decorator	Manage who can see which subject attributes with new subject attribute security support. You can also decorate subjects after retrieval from their source.
Grouper Failover Client	Higher availability of web services using a new failover client library and discovery service library.

For more information about upcoming plans, see the Grouper+Product+Roadmap.

Many other fixes and improvements were also made to all components of the Grouper Toolkit: Grouper API, Administrative & Lite UIs, Grouper Web Services, Grouper Client, Grouper Shell, Grouper Loader, PSP, and the Subject API.

Wiki Home	Download Grouper	Grouper Guides	Community Contributions	Developer Resources	Deployment Guide

> This topic is discussed in the "Grouper Minor Upgrade" training video.

Upgrading from Grouper v1.6+

The following instructions describe how you can upgrade to 2.1 from either 1.6 or 2.0. To give you an idea of how long the database upgrade may take, I performed a test upgrade from 2.0 to 2.1 on an Oracle database with 126,801 groups, 105,916 stems, 1,074,434 memberships, and 132,137 members. The actual database upgrade steps (Step 7, Step 9, Step 10) took the following amount of time.

- Step 6 (generate SQL script): 4 minutes (may have taken up to 3 minutes more if upgrading from 1.6)
- Step 9 (run SQL script): 7 minutes (may have taken up to 13 minutes more if upgrading from 1.6)
- Step 10 (update grouper_members table): Only applicable if upgrading from 1.6. Estimated time: 50 minutes.

1. Now searching for subjects requires a root session. You need to call GrouperSession.startRootSession() before finding subjects, or be in a GrouperSession as another user. If you have GSH scripts, make sure you have a grouper session before finding subjects.
2. If you are upgrading to 2.1.0 (it is fixed in 2.1.1+), then if you have an ldap subject source, and you have capital letters in the subejct ids, and you dont want them toLowerCased, then grab the 2.1.1 subject jar and put it in place of the current subject jar (in all locations), and ignore the log issue if it says the subject.jar is not the right version or size

3. If you are upgrading to 2.1.0 (it is fixed in 2.1.1+), and you use SOAP web services and the wsdl from server, then build with this axis2.xml instead of the one there in webapp/WEB-INF/conf
4. If you are upgrading to 2.1.0 (it is fixed in 2.1.1+), and you will be doing loader jobs from ldap, use this grouper.jar in your loader called from gsh
5. You should get v2.1 versions of the Grouper API, Grouper UI, Grouper WS, Grouper Daemon, etc. You will need to merge configuration files and JARs. See the v2.1 change log for more information. If you are upgrading from a release before v2.0.2 (including any v1.6 release), then see the v2.0 change log before the v2.1 changes. The rest of this document focuses on upgrading the database.
6. First you may want to analyze your tables to help speed up the upgrade. Analyze your tables. At minimum, be sure to analyze grouper_members, grouper_group_set, grouper_memberships, grouper_groups, and grouper_stems.
7. Stop the Grouper Daemon. Once you prevent users from making updates to your Grouper instance, run the changeLogTempToChangeLog daemon to clear out the temp changelog using your existing v1.6 or v2.0 API. Here's an example using GSH.

```
gsh 0% loaderRunOneJob("CHANGE_LOG_changeLogTempToChangeLog")
```

8. If you are not currently using the change log for notifications and you are also not using point in time auditing, then you can instead just clear the temp change log.

```
delete from grouper_change_log_entry_temp; commit;
```

9. Before performing any upgrade steps, export your Grouper registry. Options include performing a database backup or using the XML Export utility in Grouper.
10. Using the 2.1 API, perform a registry check using GSH to create an SQL file that will contain the DDL to update your database. To do this, run: gsh -registry -check For instance..

```
$ ./bin/gsh.sh -registry -check
Using GROUPER_HOME: /srv/grouper
Using GROUPER_CONF: /srv/grouper/conf
Using JAVA: java
using MEMORY: 64m-750m
Grouper starting up: version: 2.1.0, build date: 2012/02/17 11:48:03, env: <no label configured>
grouper.properties read from: /srv/grouper/conf/grouper.properties
Grouper current directory is: /srv/grouper
log4j.properties read from:   /srv/grouper/conf/log4j.properties
Grouper is logging to file:   /srv/grouper/logs/grouper_error.log, at min level WARN for package: edu.
internet2.middleware.grouper, based on log4j.properties
grouper.hibernate.properties: /srv/grouper/conf/grouper.hibernate.properties
grouper.hibernate.properties: ims@jdbc:oracle:thin:@imstst-db.oit.duke.edu:1668:IMSTST
sources.xml read from:        /srv/grouper/conf/sources.xml
sources.xml groupersource id: g:gsa
sources.xml groupersource id: grouperEntities
sources.xml jdbc source id:   jdbc: GrouperJdbcConnectionProvider
(note, might need to type in your response multiple times (Java stdin is flaky))
(note, you can allow or deny db urls and users in the grouper.properties)
Are you sure you want to schemaexport all tables (dropThenCreate=F,writeAndRunScript=F) in db user
'ims', db url 'jdbc:oracle:thin:@imstst-db.oit.duke.edu:1668:IMSTST'? (y|n):
y
Continuing...
Grouper ddl object type 'Grouper' has dbVersion: 23 and java version: 26
Grouper database schema DDL requires updates
(should run script manually and carefully, in sections, verify data before drop statements, backup
/export important data before starting, follow change log on confluence, dont run exact same script in
multiple envs - generate a new one for each env),
script file is:
/srv/grouper/ddlScripts/grouperDdl_20120217_11_49_34_909.sql
Note: this script was not executed due to option passed in
To run script via gsh, carefully review it, then run this:
gsh -registry -runsqlfile /srv/grouper/ddlScripts/grouperDdl_20120217_11_49_34_909.sql
```

11. In this example above, an SQL script called /srv/grouper/ddlScripts/grouperDdl_20120217_11_49_34_909.sql was created.
12. Review the script to make sure it looks okay. The script will be dropping and recreating the table GROUPER_PIT_ATTR_ASSN_VALUE and it will be dropping the GROUPER_FLAT_* tables if you are upgrading from v1.6. It will also drop and recreate views, constraints, and some indexes. The index COMPOSITE_COMPOSITE_IDX should be dropped and recreated as a unique index.
    a. Some tables will have new columns added as well.
        i. If upgrading from v1.6, new columns are added to: GROUPER_ATTRIBUTE_ASSIGN, GROUPER_GROUPS, GROUPER_MEMBERS, and GROUPER_STEMS.
        ii. If upgrading from v2.0, new columns are added to: GROUPER_STEMS and all of the GROUPER_PIT_* tables.

b. If using postgres, you should see foreign keys being dropped at the top of the script. If not, try setting the ddlutils.schema grouper.properties setting and run again. If you still don't see foreign keys being dropped at the top of the script, manually drop all foreign keys before running the script.
c. If using postgres or hsql, you should backup any non grouper views that depend on Grouper views, run the grouper script (which deletes those views due to drop view cascade), and then you should recreate those non grouper views.

13. If you are okay with the SQL script, execute using GSH again. To do this, run: gsh -registry -runsqlfile /path/to/sql/file.sql For instance..

```
$ ./bin/gsh.sh -registry -runsqlfile ddlScripts/grouperDdl_20120217_11_49_34_909.sql
Using GROUPER_HOME: /srv/grouper
Using GROUPER_CONF: /srv/grouper/conf
Using JAVA: java
using MEMORY: 64m-750m
(note, might need to type in your response multiple times (Java stdin is flaky))
(note, you can allow or deny db urls and users in the grouper.properties)
Are you sure you want to run the sql file in db user 'ims', db url 'jdbc:oracle:thin:@imstst-db.oit.duke.
edu:1668:IMSTST'? (y|n):
y
Continuing...
Script was executed successfully

Grouper starting up: version: 2.1.0, build date: 2012/02/17 11:48:03, env: <no label configured>
grouper.properties read from: /srv/grouper/conf/grouper.properties
Grouper current directory is: /srv/grouper
log4j.properties read from:   /srv/grouper/conf/log4j.properties
Grouper is logging to file:   /srv/grouper/logs/grouper_error.log, at min level WARN for package: edu.
internet2.middleware.grouper, based on log4j.properties
grouper.hibernate.properties: /srv/grouper/conf/grouper.hibernate.properties
grouper.hibernate.properties: ims@jdbc:oracle:thin:@imstst-db.oit.duke.edu:1668:IMSTST
sources.xml read from:        /srv/grouper/conf/sources.xml
sources.xml groupersource id: g:gsa
sources.xml groupersource id: grouperEntities
sources.xml jdbc source id:   jdbc: GrouperJdbcConnectionProvider
```

14. Starting with v2.0, Grouper now stores member attributes that you can configure and use to sort and search a list of members. These attributes are populated in the member objects when the subjects are resolved in Grouper. Here is how you can resolve the subjects. Note that if you have a lot of groups or members, you may have to increase your JVM heap size before starting GSH. **You can skip this step if you are upgrading from v2.0.**

```
$ ./bin/gsh.sh
Using GROUPER_HOME: /srv/grouper
Using GROUPER_CONF: /srv/grouper/conf
Using JAVA: java
using MEMORY: 64m-750m
Grouper starting up: version: 2.1.0, build date: 2012/02/17 11:48:03, env: <no label configured>
grouper.properties read from: /srv/grouper/conf/grouper.properties
Grouper current directory is: /srv/grouper
log4j.properties read from:   /srv/grouper/conf/log4j.properties
Grouper is logging to file:   /srv/grouper/logs/grouper_error.log, at min level WARN for package: edu.
internet2.middleware.grouper, based on log4j.properties
grouper.hibernate.properties: /srv/grouper/conf/grouper.hibernate.properties
grouper.hibernate.properties: ims@jdbc:oracle:thin:@imstst-db.oit.duke.edu:1668:IMSTST
sources.xml read from:        /srv/grouper/conf/sources.xml
sources.xml groupersource id: g:gsa
sources.xml groupersource id: grouperEntities
sources.xml jdbc source id:   jdbc: GrouperJdbcConnectionProvider
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInvite
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInviteExpireDate
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInviteDate
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectEmailAddress
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInviteGroupUuids
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInviteMemberId
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
```

```
externalSubjectInviteUuid
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInviteEmailWhenRegistered
Grouper note: auto-created attributeDefName: etc:attribute:attrExternalSubjectInvite:
externalSubjectInviteEmail
Grouper note: auto-created attributeDefName: etc:attribute:rules:rule
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleActAsSubjectId
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleActAsSubjectIdentifier
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleActAsSubjectSourceId
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleCheckType
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleCheckOwnerId
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleCheckOwnerName
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleCheckStemScope
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleCheckArg0
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleCheckArg1
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfOwnerId
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfOwnerName
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfConditionEl
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfConditionEnum
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfConditionEnumArg0
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfConditionEnumArg1
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleIfStemScope
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleThenEl
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleThenEnum
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleThenEnumArg0
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleThenEnumArg1
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleThenEnumArg2
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleValid
Grouper note: auto-created attributeDefName: etc:attribute:rules:ruleRunDaemon
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitExpression
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitIpOnNetworks
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitIpOnNetworkRealm
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitLabelsContain
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitAmountLessThan
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitAmountLessThanOrEqual
Grouper note: auto-created attributeDefName: etc:attribute:permissionLimits:limitWeekday9to5
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoader
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderType
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderDbName
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderScheduleType
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderQuartzCron
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderIntervalSeconds
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderPriority
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderAttrsLike
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderAttrQuery
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderAttrSetQuery
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderActionQuery
Grouper note: auto-created attributeDefName: etc:attribute:attrLoader:attributeLoaderActionSetQuery
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdap
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapType
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapServerId
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapFilter
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapQuartzCron
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapSearchDn
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapSubjectAttribute
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapSourceId
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapSubjectIdType
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapAndGroups
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapSearchScope
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapPriority
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapGroupsLike
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapGroupAttribute
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapExtraAttributes
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapErrorUnresolvable
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:
grouperLoaderLdapGroupNameExpression
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:
grouperLoaderLdapGroupDisplayNameExpression
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:
grouperLoaderLdapGroupDescriptionExpression
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapSubjectExpression
```

```
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapGroupTypes
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapReaders
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapViewers
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapAdmins
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapUpdaters
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapOptins
Grouper note: auto-created attributeDefName: etc:attribute:loaderLdap:grouperLoaderLdapOptouts
Grouper note: auto-created attributeDefName: etc:attribute:entities:entitySubjectIdentifier
Type help() for instructions
gsh 0% GrouperSession.startRootSession()
edu.internet2.middleware.grouper.GrouperSession:
6f94c99d5b0948a3be96f94f00ab4d87,'GrouperSystem','application'
gsh 1% // run USDU to resolve all the subjects with type=person
gsh 1% usdu()
usdu completed successfully
gsh 2% // resolve the groups
gsh 2% GrouperSession.startRootSession();
edu.internet2.middleware.grouper.GrouperSession:
4163fb08b3b24922b55a14010d48e121,'GrouperSystem','application'
gsh 3% for (String g : HibernateSession.byHqlStatic().createQuery("select uuid from Group").listSet
(String.class)) { subj = SubjectFinder.findByIdAndSource(g, "g:gsa", true); GrouperDAOFactory.
getFactory().getMember().findBySubject(subj).updateMemberAttributes(subj, true); }
```

15. Analyze your tables. At minimum, be sure to analyze grouper_members, grouper_group_set, grouper_memberships, grouper_groups, grouper_stems, grouper_pit_members, grouper_pit_group_set, grouper_pit_memberships, grouper_pit_groups, and grouper_pit_stems.
16. See if you are using flattened permissions rules (you probably aren't). Run this SQL, if it returns any rows, you need to remove those rules:

```
SELECT * FROM grouper_rules_v WHERE rule_check_type LIKE 'flattenedPermission%'

Get the IDs like this:

SELECT attribute_assign_id FROM grouper_rules_v WHERE rule_check_type LIKE 'flattenedPermission%'

Then delete them:

GrouperSession.startRootSession();
AttributeDefFinder.findByName("stem:permissionDef", true).getAttributeDelegate().
removeAttributeByAssignId("whateverIdReturnedFromPreviousQuery");

You should coordinate with the owners of the objects about removing this functionality
```

17. Start the Grouper Daemon.

See Also

Grouper Downloads