

Grouper Getting Started Quickly

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



This page is outdated as of 2019. To Get Started Quickly, it is suggested to use the [InCommon Trusted Access Platform Docker Containers](#)

See also [this page](#)

These instructions on how to get started with Grouper quickly were provided as an example by University of Pennsylvania.

API:

1. Unzip the grouper binary api ([Grouper download page](#))
2. If you checked it out, or got the source dist, run: ant dist
3. Hsql doesnt work too well (though if you want to use it instructions below), so switch to mysql. Create a schema and user in mysql (e.g. grouperuser@localhost). I recommend the free version of sqlYog to manage mysql.
4. Change these settings in conf/grouper.hibernate.properties
 - a. hibernate.dialect = org.hibernate.dialect.MySQL5Dialect
 - b. hibernate.connection.driver_class = com.mysql.jdbc.Driver
 - c. hibernate.connection.url = jdbc:mysql://localhost:3306/grouper
 - d. hibernate.connection.username = grouper
 - e. hibernate.connection.password = whateverYouSet
5. Add tables: bin\gsh -registry -runscript
 - a. If there is an error, check the logs carefully, check the db, to be sure it failed. If a lot of views are there, things might be ok. If grouper_fields is empty, run: bin\gsh -registry -reset
 - b. You should now be able to browse the DB with sqlYog or whatever tool you use to admin the db
6. Check tables: bin\gsh -registry -check
 - a. Should output: NOTE: database table/object structure (ddl) is up to date
7. Start gsh and add a subject: bin\gsh
 - a. gsh 0% addSubject("test.subject.0", "person", "test.subject.0")
 - gsh 1% exit

Note: insert rows in the subject attribute table to correspond with your subject, something like:

```
insert into subjectattribute (subjectId, name, value, searchValue) values('test.subject.0','description','description.test.subject.0','description.test.subject.0');
insert into subjectattribute (subjectId, name, value, searchValue) values('test.subject.0','email','test.subject.0@somewhere.someSchool.edu','test.subject.0@somewhere.someschool.edu');
insert into subjectattribute (subjectId, name, value, searchValue) values('test.subject.0','loginid','id.test.subject.0','id.test.subject.0');
insert into subjectattribute (subjectId, name, value, searchValue) values('test.subject.0','name','name.test.subject.0','name.test.subject.0');
commit;
```

1. In grouper.properties, I will change/add these settings:
 - a. groups.wheel.use = true
 - groups.wheel.group = etc:sysadmingroup
 - configuration.autocreate.group.name.0 = etc:sysadmingroup
 - configuration.autocreate.group.description.0 = super users
 - configuration.autocreate.group.subjects.0 = mchyzer
 - configuration.autocreate.group.name.1 = etc:webServiceUsers
 - configuration.autocreate.group.description.1 = users allowed to log in to the WS
 - configuration.autocreate.group.subjects.1 = mchyzer
 - configuration.autocreate.group.name.2 = etc:webServiceActAsUsers
 - configuration.autocreate.group.description.2 = users allowed to act as other in the WS
 - configuration.autocreate.group.subjects.2 = mchyzer
2. Start gsh again: bin\gsh see if the user is in the groups
 - a. gsh 0% **grouperSession = GrouperSession.startRootSession();**
edu.internet2.middleware.grouper.GrouperSession: f802d876-b876-4315-b76e-0586bcc561b1,'GrouperSystem','application'
 - gsh 1% **subject = findSubject("mchyzer");**
subject: id='mchyzer' type='person' source='jdbc' name='Chris Hyzer'
 - gsh 2% **member = MemberFinder.findBySubject(grouperSession, subject);**
member: id='mchyzer' type='person' source='jdbc' uuid='1324c75e-9435-4c45-97e9-af40f2b71046'
 - gsh 3% **member.getGroups();**
group: name='etc:sysadmingroup' displayName='etc:sysadmingroup' uuid='38990f70-3d93-4a80-933c-6358c524024c'
 - group: name='etc:wsActAsUsers' displayName='etc:webServiceActAsUsers' uuid='e87171ed-69e4-4cf8-91ea-c463770b71b1'

```
group: name='etc:wsUsers' displayName='etc:webServiceUsers' uuid='6b2928d8-08f7-4cab-91f0-cb42dc794456'
group: name='etc:uiUsers' displayName='etc:userInterfaceUsers' uuid='4103ede1-fdd1-419b-a5a7-0d2a6b3220eb'
gsh 4%
```

3. If you want quickstart data, [download this file](#) into quickstart.xml, and [this file](#) into subjects.sql. (note, you need to act as admin, or adjust permissions to see this data, but you can verify in the DB that there a lot of groups/stems/etc)
 - a. Import with: bin\gsh -registry -runsqlfile subjects.sql
 - b. Import with: bin\gsh -xmlimportold GrouperSystem quickstart.xml

Grouper loader

You should run the grouper loader all the time for any Grouper deployment. It also runs background tasks even if you arent automatically loading groups.

```
gsh - loader
```

Grouper UI

1. Download or unzip grouper-ui ([Grouper download page](#))
 - a. e.g. for 2.0.0 [download the UI here](#)
2. Run ant - exit
3. Edit the build.properties,
 - a. set the grouper.folder where the API is, if not ../grouper
 - b. set dist.home, e.g. dist.home=dist
 - c. set no metainf context: should.copy.context.xml.to.metainf=false
4. Run ant - dist
5. Edit your tomcat_home/conf/server.xml, add a context for the UI
 - a. <Engine defaultHost="localhost" name="Catalina">
 <Realm className="org.apache.catalina.realm.UserDatabaseRealm" resourceName="UserDatabase"/>
 <Host appBase="webapps" autoDeploy="true" name="localhost" unpackWARs="true" xmlNamespaceAware="false" xmlValidation="false">
 <Context docBase="C:\dev_instleclipse\workspace_v33\grouper-ui\dist\grouper" path="/grouper" reloadable="false"/>
 </Host>
 </Engine>
6. Add the user/pass to your tomcat_home/conf/tomcat-users.xml
 - a. <tomcat-users>
 <role rolename="grouper_user"/>
<user username="mchzyer" password="whateveryouwant" roles="grouper_user"/>
 </tomcat-users>
7. Start tomcat
8. Go to URL: <http://localhost:8080/grouper> (or wherever you mapped this to tomcat)
9. You should be able to login with the credentials in the tomcat-users.xml, and you should see the act as admin dropdown in the upper right

Grouper Web Services

1. Checkout or download grouper web services ([Grouper download page](#))
 - a. e.g. for the 2.0.0, [download the WS here](#)
2. Stop tomcat
3. Do a build in ws: ant quick
4. Change these settings in the build.properties:
 - a. grouper.lib.dir=../grouper/lib/grouper
 grouper.jar.name=../grouper/dist/lib/grouper.jar
 grouper.conf.dir=../grouper/conf
 generated.client.project.dir=../grouper-ws-java-generated-client_HEAD
5. Edit these lines in the grouper-ws.properties
 - a. ws.act.as.group = etc:webServiceActAsUsers
 ws.client.user.group.name = etc:webServiceUsers
6. Do another build: ant quick
7. Edit your tomcat_home/conf/server.xml, add a context for the WS
 - a. <Engine defaultHost="localhost" name="Catalina">
 <Realm className="org.apache.catalina.realm.UserDatabaseRealm" resourceName="UserDatabase"/>
 <Host appBase="webapps" autoDeploy="true" name="localhost" unpackWARs="true" xmlNamespaceAware="false" xmlValidation="false">
 <Context docBase="C:\dev_instleclipse\workspace_v33\grouper-ui\dist\grouper" path="/grouper" reloadable="false"/>
 <Context docBase="C:\dev_instleclipse\workspace_v33\grouper-ws_HEAD\build\dist\grouper-ws" path="/grouperWs" reloadable="false"/>
 </Host>
 </Engine>
8. Start tomcat

Grouper Client

1. Download or checkout grouper client ([Grouper download page](#))
 - a. e.g. for the 2.0.0 [download here](#)
2. Run from grouper_client_home: ant
3. Customize these in the conf/grouper.client.properties

- a. grouperClient.webService.url = <http://localhost:8090/grouperWs/servicesRest>
grouperClient.webService.login = mchyzer
grouperClient.webService.password = whateveryouwant
4. Do another build: ant
5. Cd to the institution dir, and try to run the grouper client:
 - a. F:\temp\grouperClientTemp\dist\institution\grouperClient.institution-1.4.0-rc2> java -jar grouperClient.jar --operation=getGroupsWs --subjectIds=mchyzer
SubjectIndex 0: success: T: code: SUCCESS: subject: mchyzer: groupIndex: 0: etc:sysadmingroup
SubjectIndex 0: success: T: code: SUCCESS: subject: mchyzer: groupIndex: 1: etc:webServiceActAsUsers
SubjectIndex 0: success: T: code: SUCCESS: subject: mchyzer: groupIndex: 2: etc:webServiceUsers
SubjectIndex 0: success: T: code: SUCCESS: subject: mchyzer: groupIndex: 3: etc:userInterfaceUsers

F:\temp\grouperClientTemp\dist\institution\grouperClient.institution-1.4.0-rc2>

1. asdf

Hsql

To use hsql, you should go in server mode. To do this,

- start the server:

```
F:\temp\grouper1.4\grouper-api-1.4.1\bin>java -cp ..\lib\jdbcSamples\hsqldb.jar org.hsqldb.Server -database.0
file:grouper -dbname.0 grouper
```

* Change your grouper.hibernate.properties:

```
hibernate.dialect                = org.hibernate.dialect.HSQLDialect
hibernate.connection.driver_class = org.hsqldb.jdbcDriver
hibernate.connection.url = jdbc:hsqldb:hsql://localhost/grouper
hibernate.connection.username    = sa
hibernate.connection.password    =
```

* Start admin tool if you like

```
F:\temp\grouper1.4\grouper-api-1.4.1\bin>java -cp ..\lib\jdbcSamples\hsqldb.jar org.hsqldb.util.DatabaseManager
-url jdbc:hsqldb:hsql://localhost/grouper
```

* To delete the database, delete the files from the dir where you started hsqldb: grouper.log, grouper.properties, grouper.script