

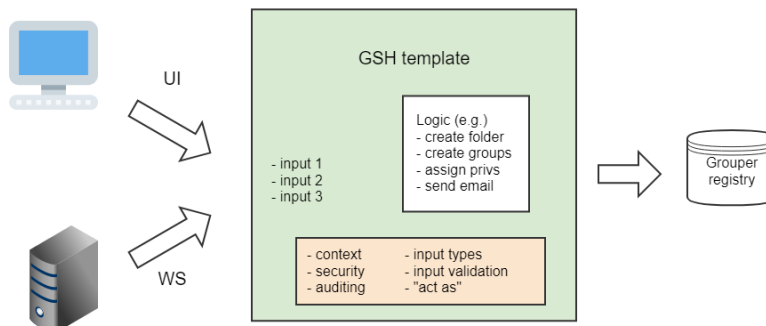
Grouper custom template via GSH

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

The Grouper custom template via GSH is in Grouper 2.5.43+

- [Grouper custom templates via GSH - converting an existing GSH script - insert a row in table](#)
- [Grouper custom template via GSH delegated VPN management](#)
 - [Grouper custom template via GSH delegated VPN management - add](#)
 - [Grouper custom template via GSH delegated VPN management - delete](#)
 - [Grouper custom template via GSH delegated VPN management - manage](#)
- [Grouper custom template via GSH impersonate delete example](#)
- [Grouper custom template via GSH impersonate example](#)
- [Grouper custom template via GSH impersonate testing helper](#)
- [Grouper custom template via GSH inherited privileges WS example](#)
- [Grouper custom template via GSH Internet2 example](#)
 - [Grouper custom template via GSH Internet2 example delete](#)
- [Grouper custom template via GSH invoked by daemon - load group attributes](#)
- [Grouper custom template via GSH manage credential SQL subject source](#)
- [Grouper custom template via GSH onboarding org example](#)
- [Grouper custom template via GSH report compare Banner security envs](#)
- [Grouper custom template via GSH role mapper](#)
- [Grouper custom template via GSH testing the Grouper health](#)
- [Grouper custom template via GSH used by daemon for CSV report](#)
- [Grouper custom template via GSH user membership history](#)
- [Grouper custom template via GSH - V2](#)
 - [Grouper custom template via GSH - V2 - dynamic drop downs](#)
- [Grouper custom template via GSH web service membership counts](#)
 - [Grouper custom template via GSH web service membership counts - V1 - arbitrary inputs map](#)
 - [Grouper custom template via GSH web service membership counts - V1 - basic](#)
 - [Grouper custom template via GSH web service membership counts - V1 - bean inputs](#)
 - [Grouper custom template via GSH web service membership counts - V1 - json inputs / outputs](#)
 - [Grouper custom template via GSH web service membership counts - V2 - arbitrary inputs](#)
 - [Grouper custom template via GSH web service membership counts - V2 - basic](#)
 - [Grouper custom template via GSH web service membership counts - V2 - bean inputs](#)
 - [Grouper custom template via GSH web service membership counts - V2 - bean inputs with test](#)
 - [Grouper custom template via GSH web service membership counts - V2 - json inputs](#)
- [Grouper custom template via GSH zoom deprovisioning](#)
- [Grouper GSH template security](#)
- [Grouper GSH template simplified UI](#)
- [Grouper programmatic JSON](#)

Intro



Custom templates via GSH are a function of Grouper that evolves the idea of an Access Management Platform. You can make a UI screen or a web service call that is completely custom. The inputs are dynamic, flexible, and have built-in validations. The [GSH script](#) to process those inputs has full control of Grouper or Java. There is [security](#) around who can run the template, and the user they run-as. There are easy-to-use APIs to perform a list of actions.

Custom templates via GSH are good for:

1. Performing actions that are not available in the UI
 - check LDAP to see if provisioning happened, for example

- make an ad hoc report and send it to the user
 - add/remove a service principal to a subject source table
2. Automating multiple tasks into one
 3. Creating a custom access management interface for a user to simplify actions
 - e.g. to add a user to a VPN instance, you need to remove an exclude if there, and if not in the automatic population and an employee, add them to the VPN employee group, otherwise guest group. All can be automated
 4. Create a Web Service call that does not exist in Grouper
 5. Reduce time to perform tasks
 6. Reduce Help Desk Tickets
 7. The sky is the limit



Reminder: "Custom UI" and "Custom Template via GSH" are **two different features** in Grouper.

Use a [Custom UI](#) when you want an **end-user facing UI** where the end user can:

- See their status and why (e.g. they might need a training or a different affiliation)
- Can (optionally) enroll or unenroll easily into a service
- Redirect based on certain conditions to other services
- You don't want the end user to see the full Grouper UI

Use Grouper Custom Templates via GSH (the info on this page):

- For people who use the Grouper UI
- Can have custom inputs
- Logic can use the input values and performance queries or tasks
- Automate multiple tasks in one operation
- Can expose via web service

Best practices

- If you get errors with dollar signs in strings use this

```
Script2.groovy: 603: illegal string body character after dollar sign;
  solution: either escape a literal dollar sign "\$5" or bracket the value expression "${5}"
FROM: "$whatever"
TO: '$' + "whatever"
```

- You cannot line wrap unless you put an operator at end of line

```
No signature of method: java.lang.String.positive() is applicable for argument types: () values: []
FROM: String something = ""
      + "hey";
TO: String something = "" +
    "hey";
```

Configuring

There is a wizard in the UI. These are the configs that the wizard controls

grouper.properties

grouperGshTemplate.<configId>.suffix

config id suffix	value example	description	notes
enabled	true false	if this template is enabled	if false, do not show it in menu on folders or allow it to run
templateVersion	V1 V2	V1 is the legacy version, the script is in the gshTemplate V2 is the newer way where you extend a class and can include tests	

showOnGroups	true false	if this template option is available on groups	default false (TODO as of 2.5.36)
groupShowType	certainGroup, groupsInFolder, allGroups	is this supposed to show for one group, groups in a certain folder, or allGroups	required, show if showOnGroups=true (TODO as of 2.5.36)
groupUuidToShow	xyz321	which group to show the template on. could be uuid or name.	required, show if groupShowType=certainGroup
groupShowOnDescendants	oneChildLevel, descendants	oneChildLevel: only show on groups directly in folder descendants: show in all groups under the folder	required, show if groupShowType = groupsInFolder
showOnFolders	true false	if this template option is available on folders	default false
folderShowType	certainFolder, allFolders	where should this template be available	drop down, required, show if showOnFolders = true
folderUuidToShow	abc123	uuid or name of folder to show this template	eventually we can have a folder combobox, currently textfield. show if folderShowType = 'specifiedFolder' or groupShowType = 'groupsInFolder'. Required
folderShowOnDescendants	certainFolder, oneChildLevel, certainFolderAndOneChildLevel, descendants, certainFolderAndDescendants	certainFolder: just show on one folder oneChildLevel, only show on child level under folder certainFolderAndOneChildLevel, show the folder and one level of children descendants: show all folders under the folder certainFolderAndDescendants: show folder and all descendants	if folderShowType = 'specifiedFolder'. Required
runButtonGroupOrFolder	group folder	If you want a run button on the GSH template page, this will allow configuration of a default group or folder	
defaultRunButtonGroupUuidOrName	a:b:c	Default group name or uuid for the run now button	
defaultRunButtonFolderUuidOrName	a:b:c	Default folder name or uuid for the run now button	
securityRunType	wheel specifiedGroup privilegeOnObject everyone	who can run this template. Only GrouperSystem / wheel group, or specify a group. If privilegeOnObject, then check to see if user has certain privileges on the object where the template was invoked from.	drop down, Required
groupUuidCanRun	def456	uuid or name of group that can run this template.	eventually we can have a group combobox, currently textfield. Show if securityRunType = 'specifiedGroup'. Required
requireFolderPrivilege	admin, create, stemAttrRead, stemAttrUpdate	If running this template requires any of these privs on the folder it is run on	show if showOnFolders = true. required if securityRunType = 'privilegeOnObject'
requireGroupPrivilege	admin, read, update, read_and_update, optin, optout, view, groupAttrRead, groupAttrUpdate	the option to run the template will only show if the user has this privilege on the group at least (e.g. if the user has ADMIN they can READ)	show if showOnGroups = true. required if securityRunType = 'privilegeOnObject'
runAsType	currentUser GrouperSystem specifiedSubject	select the type of user to run as. "currentUser" means run as the user using the UI. "GrouperSystem" means run as root user, "specifiedSubject" means you can pick a subject to run as (not common).	drop down. Required
runAsSpecifiedSubjectSourceId	pennperson	select the source ID of the specified subject to run as	drop down (subject source picker). Required
runAsSpecifiedSubjectId	12345678	subject id of the specified subject to run as	eventually we can have a subject combobox, currently textfield. Required
templateNameExternalizedTextKey	grouperGshTemplate_<configId>_templateNameExternalizedTextKey	this is hardcoded for each template	this is readonly, not editable, eventually we can have an externalized text editor to edit that from this screen
templateDescriptionExternalizedTextKey	grouperGshTemplate_<configId>_templateDescriptionExternalizedTextKey	this is hardcoded for each template	this is readonly
simplifiedUi	true false	If the UI should not show the normal Grouper menus, to not confuse users who are not familiar with Grouper	
allowWsFromNoOwner	true false	if WS can call the template without identifying a stem or group owner	
gshTemplate	//	this is the GSH template to run	if we can get a textarea that would be good. Should convert from windows or mac newlines to unix newlines on submit. Required

numberOfInputs	5	number of inputs (form elements on ui or in ws)	drop down from 0-50 repeat group. Default value: 0
input.i.name	gsh_input_folderName	name of the ui form element, ws param, template variable,	validation: must start with gsh_input_, must be alphanumeric/underscore. Required
input.i.labelExternalizedTextKey	grouperGshTemplate_<configId>_input_<inputName>_labelExternalizedTextKey	externalized text key of the label on the UI for this input	readonly
input.i.descriptionExternalizedTextKey	grouperGshTemplate_<configId>_input_<inputName>_descriptionExternalizedTextKey	externalized text key of the label on the UI for this input	readonly
input.i.type	int boolean string	type of the data	drop down with supported types. suggested starting point: int, boolean, string. Default value: string
input.i.formElementType	textfield, dropdown, checkbox	form element type on UI	show if type is not equal to "boolean" (if boolean its a radio). Default value: textfield.
input.i.index	10	form elements will show on the form in order of "index". DefaultValue is 0. If two elements have the same index then they will show in order of configuration.	textfield
input.i.validationType	regex jexl none	type of validation on the input	required (none is not blank, it is an option)
input.i.validationRegex	^[a-zA-Z0-9_]{1,50}\$	regex to check the input and if it doesnt match then fail. For example, this is alphanumeric or underscore length between 1 and 50	show if validationType = 'regex', required
input.i.validationJexl	\${gsh_input_myField.startsWith('whatever')}	run a validation (and include all variables for cross-validations), return true for valid, and false for invalid	show if validationType = 'jexl', required
input.i.validationMessageExternalizedTextKey	grouperGshTemplate_<configId>_input_<inputName>_validationMessageExternalizedTextKey	readonly key in externalized text for validation message (e.g. for jexl or regex)	readonly
input.i.required	true false	if this input is required for template to run	default value: false
input.i.defaultValue	abc	default value for the input if none is provided	show if required is false
input.i.showEl	\${gsh_input_someField = 'something'} for boolean, \${gsh_input_type == "Bulk"} for string value matching.	jexl for if this field should show, note all inputs are available to use as variables	
numberOfTests	3	number of tests to make sure this template functions	drop down from 0-20 repeat group. Default value: 0
test.i.deleteSideEffectsIfExistGsh	gsh script	will be run before and after test (first and last) to clean up what the test and setup and verification do. Will not run after test if test fails	textarea
test.i.setupGsh	gsh script	will be run before test	textarea
test.i.testGsh	gsh script	call the Java API for the template exec with inputs	textarea, required
test.i.verifyGsh	gsh script	verify that the test succeeded	textarea

A grouper admin would configure this in the UI under miscellaneous

Text on screen similar to custom UI

This template could be called via ws REST JSON and pass in the input name/value form elements just like the UI would

Developing GSH templates with java

See [this wiki as an example of developing GSH templates with java](#)

Built in variables and inputs

Before the configured GSH script is executed, it will be prefixed with some built in variables, and the inputs from the user (supplied in the UI or less commonly the WS).

Type	Variable	Value
GshTemplateOutput	gsh_builtin_gshTemplateOutput	Object that helps print output, notify about validation errors, or identify that an error occurred
GshTemplateRuntime	gsh_builtin_gshTemplateRuntime	Internal object that is used to help with other variables
GrouperSession	gsh_builtin_grouperSession	Session based on who the template is running is (this is configured, recommended to be the user who is running the template)
Subject	gsh_builtin_subject	Subject who is running the template
String	gsh_builtin_subjectId	Subject ID of gsh_builtin_subject
String	gsh_builtin_ownerStemName	Folder name of the folder where the script is run (if stem script, otherwise null)
String	gsh_builtin_ownerGroupName	Group name for the group where the script was run (if group script, otherwise null)
String	gsh_input_XXXXXX	String inputs
Integer	gsh_input_YYYYYYY	Integer inputs
Boolean	gsh_input_ZZZZZZ	Boolean inputs

```
import edu.internet2.middleware.grouper.app.gsh.template.*;
import edu.internet2.middleware.grouper.util.*;
GshTemplateOutput gsh_builtin_gshTemplateOutput = GshTemplateOutput.retrieveGshTemplateOutput();
GshTemplateRuntime gsh_builtin_gshTemplateRuntime = GshTemplateRuntime.retrieveGshTemplateRuntime();
GrouperSession gsh_builtin_grouperSession = gsh_builtin_gshTemplateRuntime.getGrouperSession();
Subject gsh_builtin_subject = gsh_builtin_gshTemplateRuntime.getCurrentSubject();
String gsh_builtin_subjectId = \"subjectSubjectId\";
String gsh_builtin_ownerStemName = \"a:b:c\";
String gsh_input_workingGroupExtension = \"test\";
Integer gsh_input_theSize = 55;
Boolean gsh_input_isSympaModerated = false;
... the configured script ...
```

Check SQL for restricted value

Example of getting an input and checking a sql to see if its restricted

```
int existingGroupCount = new GcDbAccess().connectionName("grouper").sql("select count(1) from grouper_groups gg where gg.name = ?").addBindVar(gsh_input_something).select(int.class);

if (existingGroupCount > 0) {
    gsh_builtin_gshTemplateOutput.addValidationLine("gsh_input_something", "Error: group exists");
}

// if anything not valid, stop
if (GrouperUtil.length(gsh_builtin_gshTemplateOutput.getValidationLines()) > 0) {
    gsh_builtin_gshTemplateOutput.assignIsError(true);
    GrouperUtil.gshReturn();
}
```

Check LDAP for restricted value

Example of getting an input and checking a ldap to see if its restricted

```

    List<String> eduPersonAffiliations = edu.internet2.middleware.grouper.ldap.LdapSessionUtils.ldapSession().
list(
    String.class, "personLdap", "ou=People,dc=example,dc=edu",
    edu.internet2.middleware.grouper.ldap.LdapSearchScope.SUBTREE_SCOPE,
    "(&(uid=" + GrouperUtil.ldapFilterEscape(gsh_built_in_subject.getId()) +
    ")(eduPersonAffiliation=" + GrouperUtil.ldapFilterEscape(gsh_input_something) + "))",
    "eduPersonAffiliation");

    if (GrouperUtil.length(eduPersonAffiliations) > 0) {
        gsh_built_in_gshTemplateOutput.addValidationLine("gsh_input_something", "Error: eduPersonAffiliation
exists");
    }

    // if anything not valid, stop
    if (GrouperUtil.length(gsh_built_in_gshTemplateOutput.getValidationLines()) > 0) {
        gsh_built_in_gshTemplateOutput.assignIsError(true);
        GrouperUtil.gshReturn();
    }

```

Zoom add school example

Configuration

```

grouperGshTemplate.zoom.enabled = true
grouperGshTemplate.zoom.showOnFolders = true
grouperGshTemplate.zoom.folderShowType = certainFolder
grouperGshTemplate.zoom.folderShowOnDescendants = certainFolder
grouperGshTemplate.zoom.securityRunType = privilegeOnObject
grouperGshTemplate.zoom.requireFolderPrivilege = admin
grouperGshTemplate.zoom.runAsType = currentUser
grouperGshTemplate.zoom.gshTemplate =

GrouperSession grouperSession = GrouperSession.startRootSession();
String prefix = Character.toUpperCase(gsh_input_orgName.charAt(0)) + gsh_input_orgName.substring(1,
gsh_input_orgName.length());
String prefixLower = gsh_input_orgName;
Group excludeAdHocGroup = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:service:ref:
excludeAdHoc:" + prefixLower + "AdhocExcludeFromZoom").save();
Group excludeLoadedGroup = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:service:ref:
loadedGroupsForExclude:" + prefixLower + "ExcludeLoaded").save();
Group excludeGroup = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:service:ref:
excludeFromZoom:" + prefixLower + "ExcludeFromZoom").save();
Group excludedFromZoom = GroupFinder.findByName(grouperSession, "penn:isc:ait:apps:zoom:service:ref:
usersExcludedFromZoom", true);
excludedFromZoom.addMember(excludeGroup.toSubject(), false);
Group schoolLspGroup = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:security:
schoolCenterAdminsAndLsps:zoom" + prefix + "Lsps").save();
Group lsps = GroupFinder.findByName(grouperSession, "penn:isc:ait:apps:zoom:security:
zoomSchoolCenterLspsPreCheck", true);
lsps.addMember(schoolLspGroup.toSubject(), false);
Group schoolAdminGroup = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:security:
schoolCenterAdminsAndLsps:zoom" + prefix + "Admins").save();
Group admins = GroupFinder.findByName(grouperSession, "penn:isc:ait:apps:zoom:security:
zoomSchoolCenterAdminsPreCheck", true);
admins.addMember(schoolAdminGroup.toSubject(), false);
if (gsh_input_addIncludes) {
    Group loadedGroup= new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:service:ref:loadedGroups:
gsh_input_zoomGroupName).save();
    Group overrideGroup = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:service:ref:
groupsOverride:gsh_input_zoomGroupName).save();
    Group groupToGoToZoom = new GroupSave(grouperSession).assignName("penn:isc:ait:apps:zoom:service:policy:
groupsToGoToZoom:gsh_input_zoomGroupName).save();
    Group zoomCanLogIn = GroupFinder.findByName(grouperSession, "penn:isc:ait:apps:zoom:service:policy:
zoomCanLogIn", true);
    zoomCanLogIn.addMember(groupToGoToZoom.toSubject(), false);
}

grouperGshTemplate.zoom.numberOfInputs = 3
grouperGshTemplate.zoom.input.0.name = gsh_input_orgName
grouperGshTemplate.zoom.input.0.validationType = regex
grouperGshTemplate.zoom.input.0.validationRegex = ^[a-z][a-zA-Z0-9]{1,49}$
grouperGshTemplate.zoom.input.0.required = true
grouperGshTemplate.zoom.input.1.name = gsh_input_addIncludes
grouperGshTemplate.zoom.input.1.type = boolean
grouperGshTemplate.zoom.input.1.defaultValue = false
grouperGshTemplate.zoom.input.2.name = gsh_input_zoomGroupName
grouperGshTemplate.zoom.input.2.showEl = ${gsh_input_addIncludes}
grouperGshTemplate.zoom.input.2.required = true
grouperGshTemplate.zoom.input.2.validationType = regex
grouperGshTemplate.zoom.input.2.validationRegex = ^[a-zA-Z0-9_]{1,50}$

```

Form elements

- Text box: for lower camel case school or center name, e.g. artsAndScience
- Checkbox for "add includes"? (if unchecked only excludes are created)
- Textbox for group name in zoom

grouper.text.en.us.properties

grouperGshTemplate_zoom_templateNameExternalizedTextKey = Add zoom school / center
grouperGshTemplate_zoom_templateDescriptionExternalizedTextKey = Add an organization to zoom. If it is exclude only then do not select 'Add includes'.
grouperGshTemplate_zoom_input_gsh_input_orgName_labelExternalizedTextKey = School or center name
grouperGshTemplate_zoom_input_gsh_input_orgName_descriptionExternalizedTextKey = Enter a camel-cased school or center name, starting with lower case letter, can only contain alpha-numeric characters and must be less than 50 chars
grouperGshTemplate_zoom_input_gsh_input_orgName_validationMessageExternalizedTextKey = Invalid 'School or center name': must be start with lower case letter, can only contain alpha-numeric characters and must be less than 50 chars
grouperGshTemplate_zoom_input_gsh_input_addIncludes_labelExternalizedTextKey = Add 'includes'
grouperGshTemplate_zoom_input_gsh_input_addIncludes_descriptionExternalizedTextKey = If this school or center should have a group in Zoom and in "includes" group to sponsor accounts
grouperGshTemplate_zoom_input_gsh_input_zoomGroupName_labelExternalizedTextKey = Zoom group name
grouperGshTemplate_zoom_input_gsh_input_zoomGroupName_descriptionExternalizedTextKey = Group name in zoom for this population. Must be alphanumeric or underscore and less than 50 chars
grouperGshTemplate_zoom_input_gsh_input_zoomGroupName_validationMessageExternalizedTextKey = Invalid 'Zoom group name' must be alphanumeric or underscore and less than 50 chars

Imagine an HTML form that has some inputs and a submit buttonForm

[Home](#) > [Root](#) > [penn](#) > [isc](#) > [ait](#) > [apps](#) > [zoom](#)

 **zoom**

[Edit folder](#)
[More actions ▾](#)

This folder contains resources for the zoom service.

[More ▾](#)

[Folder contents](#) [Privileges](#) [More ▾](#)

Template type

Add zoom school / center ▾

Pick the template type you want to run

Add an organization to zoom. If it is exclude only then do not select 'Add includes'.

School or center name

*

Enter a camel-cased school or center name, starting with lower case letter, can only contain aloha-numeric characters and must be less than 50 chars

Add 'includes'

☐

If this school or center should have a group in Zoom and in "includes" group to sponsor accounts

[Next](#) [Cancel](#)

Java API

```
TODO FIX THIS
GshTemplateOutput gshTemplateOutput = new GshTemplateExec()
    .assignOwnerType(GshTemplateOwnerType.stem)
    .assignOwnerStemName("a:b:c")
    .addInput(new GshTemplateInput().assignName("gsh_input_something").assignValueString("hello"))
    .addInput(new GshTemplateInput().assignName("gsh_input_something2").assignValueString("hello2"))
    .exec();
System.out.println(gshTemplateOutput.isValid());
System.out.println(gshTemplateOutput.isSuccess());
for (GshOutputLine gshOutputLine : gshTemplateOutput.getOutputLines()) {
    System.out.println(gshOutputLine.getInputName() + ": " + gshOutputLine.getText());
}
```

Web service

Template "current user" is the web service "act as" user if it exists, or the authenticating user is not.

These are a PUT method since the templates should be written as idempotent. If you have something that isn't idempotent that's ok, but aim for idempotency. This means if someone calls the same template with same inputs twice the second call should be a NO-OP (do not do anything). You can also use a POST for any call as well.

HTTP code	Grouper result code	Meaning
200	SUCCESS	Valid and ran successfully
400	INVALID	Invalid didnt run
500	EXCEPTION	Threw exception

There are various ways to call the WS, [see this page for the options](#)

```
#####
##
## HTTP request sample (could be formatted for view by
## indenting or changing dates or other data)
##
#####

POST /grouper-ws/servicesRest/v2_5_000/gshTemplateExec HTTP/1.1
Connection: close
Authorization: Basic xxxxxxxxxxxxxxxxxxxx==
User-Agent: Jakarta Commons-HttpClient/3.1
Host: localhost:8092
Content-Length: 181
Content-Type: application/json; charset=UTF-8

{
  "wsRestGshTemplateExecRequest":{
    "gshTemplateActAsSubjectLookup": {
      "subjectSourceId":"ldap",
      "subjectId":"eisbruch@at.internet2.edu"
    },
    "ownerStemLookup":{
      "stemName":"test2"
    },
    "ownerType":"stem",
    "configId":"testGshTemplateConfig",

    // new in v4.9.4+ and v5.6.1+. arbitrary input based on the template and its needs
    // this is marshaled into a Map which can be converted into a Javabean
    "wsInput": {"gsh_input_prefix":"TEST"}
  }
}
```

```
#####
```

```
##
```

```
## HTTP response sample (could be formatted for view by
```

```
## indenting or changing dates or other data)
```

```
##
```

```
#####
```

```
HTTP/1.1 200
```

```
Set-Cookie: JSESSIONID=1197432C8BE84B39F70438D3EB93E014;path=/grouper-ws/;HttpOnly
```

```
X-Grouper-resultCode: SUCCESS
```

```
X-Grouper-success: T
```

```
X-Grouper-resultCode2: NONE
```

```
Content-Type: application/json;charset=UTF-8
```

```
Content-Length: 4699
```

```
Date: Sun, 28 Feb 2021 02:30:15 GMT
```

```
Connection: close
```

```
Server: Apache TomEE
```

```
{
  "WsGshTemplateExecResult":{
    "gshOutputLines":[
      ]
    ,
    // wsOutput is new in v4.10.0+ and v5.7.0+. arbitrary output based
    // on the template and its needs (assign a Map or Javabean)
    "wsOutput": {
      "totalMembershipCount": 12,
      "immediateMembershipCount": 10
    },
    "gshScriptOutput": "",
    "gshValidationLines":[
      ]
    ,
    "responseMetadata":{
      "millis":"20197",
      "serverVersion":"2.5.0"
    },
    "resultMetadata":{
      "resultCode":"SUCCESS",
      "resultMessage":"Success for: clientVersion: 2.5.0, configId: testGshTemplateConfig, ownerType: stem ,
inputs: Array size: 1: [0]: edu.internet2.middleware.grouper.ws.coresoap.WsGshTemplateInput@52f79ba1\n\n,
actAsSubject: null, paramNames: \n, params: null",
      "success":"T"
    },
    "transaction":true
  }
}
```