Grouper™

Apereo Grouper Seminar
Part 3 – Hands on Grouper

Chris Hyzer
University of Pennsylvania and Internet2

INTERNET®

Agenda

- Grouper Loader LDAP example
- Naming best practice (folders, grouper, roles)
- Setting up reference groups (via loader)
- Composite group setup and management
- Resource/permission inheritance (Penn's unix/tomcat example)

Agenda - continued

- Logical progression from basic to production
- Managing Grouper in multiple environments

INTERNET®

Grouper Loader LDAP example

- Searched internet for public LDAP
- ldap.andrew.cmu.edu
- ou=person
- guid=?
- cn=John Smith

Grouper Loader LDAP example (continued)

- Need a source with the users in there (normally your installation will already have this)
- Get sources.xml from wiki

Grouper Loader LDAP example (continued)

- Create folder/group test:testGroup
- Use new attribute framework to assign ldap loader

Grouper Loader LDAP example (continued)

localhost:8090/grouper/grouperUi/appHtml/grouper.html?operation=SimpleAttributeUpdate.assignInit

Owner type: * Group

Attribute definition: etc:attribute:loaderLdap:grouperLoaderLdapDef

Attribute name: etc:attribute:loaderLdap:Grouper loader LDAP

Owner group: test:testGroup

Enabled / disabled: Enabled only

Filter Assign

Attribute assignments

	Owner group	Attribute name	Enabled?	Assignment values	Attribute definition	Assignment
	testGroup	Grouper loader LDAP	enabled		grouperLoaderLdapDef	94264...
Metadata on assignment		Grouper loader LDAP subject attribute name	enabled	guid	grouperLoaderLdapValueDef	08f68...
Metadata on assignment		Grouper loader LDAP search base DN	enabled	ou=person	grouperLoaderLdapValueDef	66e3c...
Metadata on assignment		Grouper loader LDAP quartz cron	enabled	0 0 8 * * ?	grouperLoaderLdapValueDef	76620...
Metadata on assignment		Grouper loader LDAP filter	enabled	(& (cmuAndrewCommonNamespaceId=*dest*) (objectClass=cmuPerson))	grouperLoaderLdapValueDef	7cb2c...
Metadata on assignment		Grouper loader LDAP server ID	enabled	personLdap	grouperLoaderLdapValueDef	accfc...
Metadata on assignment		Grouper loader LDAP type	enabled	LDAP_SIMPLE	grouperLoaderLdapValueDef	c2eef...

INTERNET®

Grouper Loader LDAP example (continued)

- You can debug the loader
- log4j.properties
- Run GSH: C:\grouper\bin> gsh
  gsh 0% grouperSession = GrouperSession.startRootSession();
  gsh 1% loaderGroup = GroupFinder.findByName(grouperSession, "test:testGroup");
  gsh 2% loaderRunOneJob(loaderGroup);

Naming best practices

- Might want to have a top level folder for your institution, something short
  - E.g. at Penn, it is penn:
  - E.g. at Chicago, it is uc:
- This will make group names generally globally unique
- At Penn we also have a top level folder "test:"
- Our "test" grouper instance is for testing new upgrades to grouper, the "test" folder in prod is for clients' test environments. Not for load testing

Naming best practices (continued)

- Folder structure matches the privilege delgation
- For instance, your top level folders (under the institution folder) might be schools and centers in the institution

Naming best practices (continued)

Grouper, the Internet2 gro

https://medley.isc-seo.upenn.edu/grouper/browseStemsAll.do?currentNode=f24f25a7-2266-4a95-bca5-6975e6e

My tools

- Explore
- Search
- Folder workspace
- Group workspace
- Entity workspace
- Group types
- Lite UI
- Help

Grouper is sponsored by

Browse groups hierarchy

You can look for groups throughout the hierarchy.
(You might not be able to see some groups if you lack appropriate privileges.)

Browse or list groups

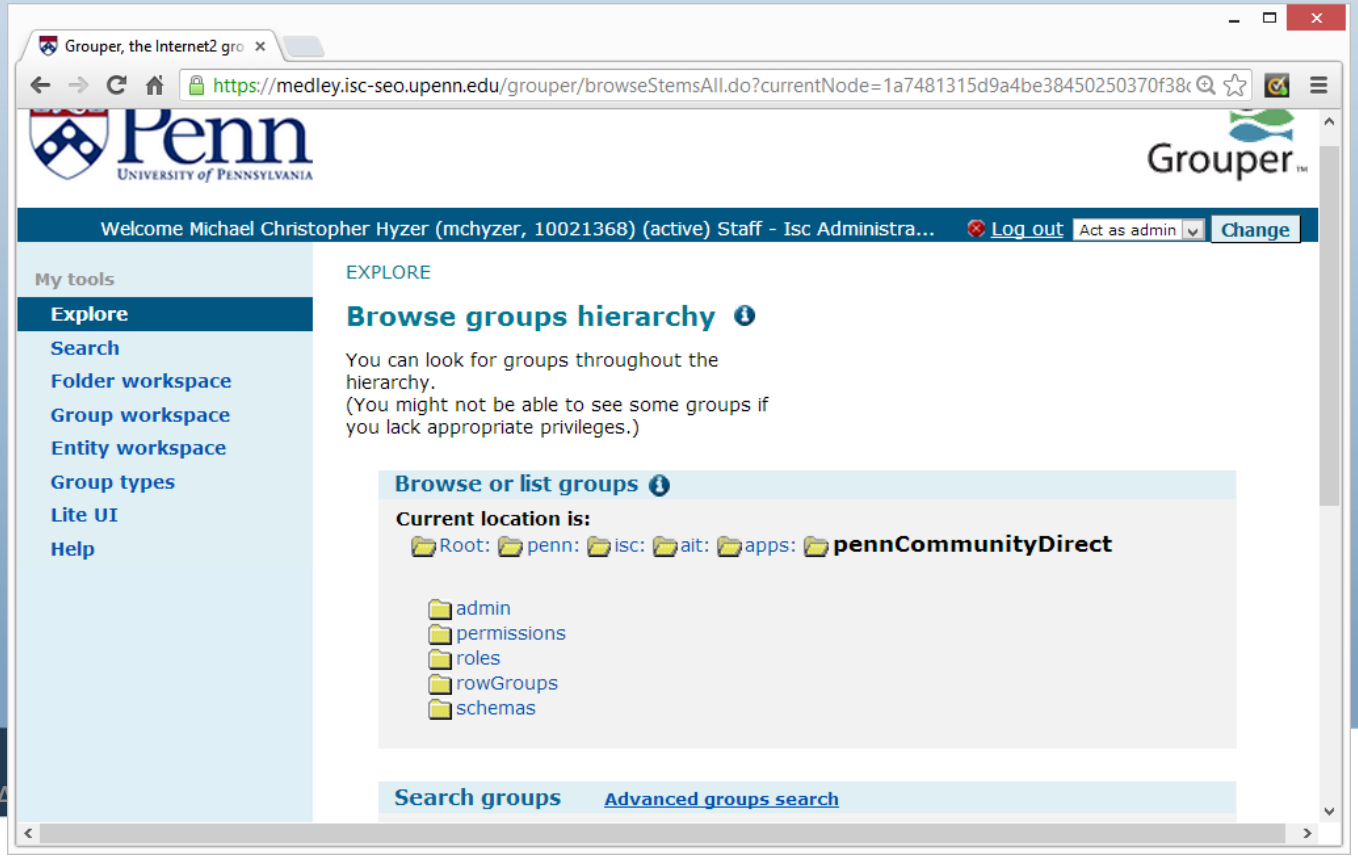
Current location is:

Root: **penn**

- community
- etc
- evp
- gse
- isc
- library
- med
- pennlaw
- presidentsOffice
- sas
- seas
- uphs
- wharton

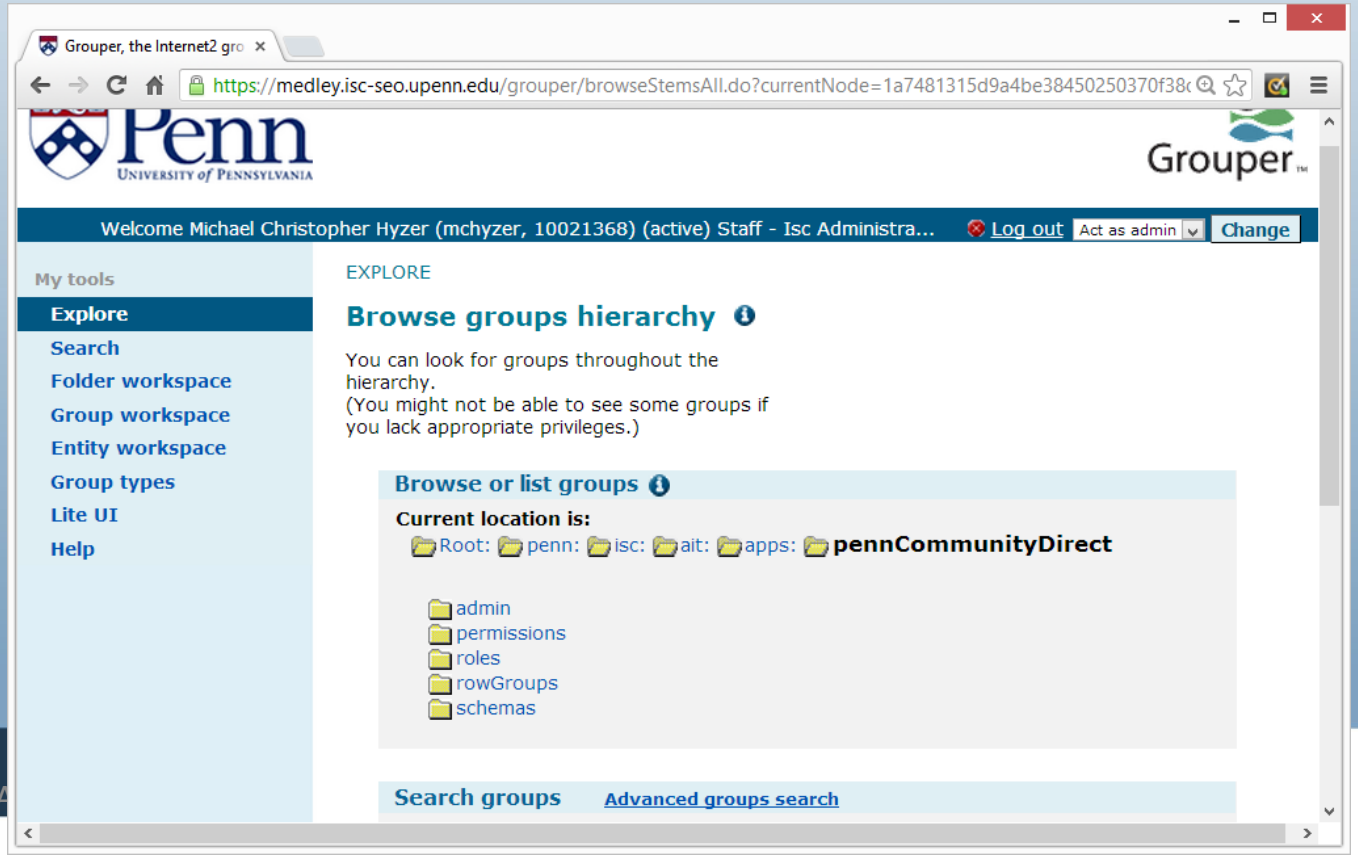
Naming best practices (continued)

- Keep groups / roles / permissions organized in separate folders



Naming best practices (continued)

- Keep groups / roles / permissions organized in separate folders



Naming best practices (continued)

- Enforce a policy on which characters are allowed
- Keep in mind down-stream systems

index.html | PageLinkInclude.java | ByCriteria.java | ByCriteriaStatic.jav | HibUtils.java | fastConfig.xml | fastConfigBase.xml | grouper.properties | 27

```
374
375
376##################################
377## Group attribute validation via regex
378## You can attach a regex to an attribute name (including built ins)
379## If none are registered, the built in hook will not be enabled
380## The built ins are description, displayName, extension, displayExtension, name
381## Configure a group.attribute.validator.attributeName.X for attribute name
382## group.attribute.validator.regex.X for the regex
383## group.attribute.validator.vetoMessage.X for the veto message (can contain the variable $attributeValue$ which will substitute)
384## the X must be a sequential integer which groups the config entries together.
385## do not repeat two config entries
386##################################
387
388#Attach a regex validator by attribute name
389group.attribute.validator.attributeName.0=name
390group.attribute.validator.regex.0=^[a-zA-Z0-9_:.-]+$
391group.attribute.validator.vetoMessage.0=Group ID or ID Path is invalid since it must contain only alpha-numerics, underscore, colon, dot, or das
392
393#group.attribute.validator.attributeName.1=displayExtension
394#group.attribute.validator.regex.1=^[a-zA-Z0-9 ]+$
395#group.attribute.validator.vetoMessage.1=Group name '$attributeValue$' is invalid since it must contain only alpha-numerics or spaces
396
397#####################################
```

Naming best practices (continued)

- Could start with extensions that are the same as display extensions
  - Some people like spaces and title case instead of camel case

Naming best practices (continued)

- Have a high-level apps folder
  - Note: Penn doesn't do this, though some institutions recommend it
- Have a high-level community folder
  - Commonly used groups generally from loader
- Descriptive extensions
  - Some screens only show the extension
  - Instead of "admins", use "ptoAdmins"

Reference groups via loader

- Have a high-level community folder
  - Commonly used groups from loader

Reference groups via loader (continued)

Grouper, the Internet2 gro

https://medley.isc-seo.upenn.edu/grouper/browseStemsAll.do?currentNode=a78654ba-75dd-43c8-a458-d139bc

Lite UI

Help

Grouper is sponsored by

Current location is:

Root: penn: **community**

- alumni
- employee
- student
- **activeNonAlumniWithPennname**
- **affiliate**
- **courtesyAffiliation**
- **employee**
- **employeeActiveFullTimePartTime**
- **employeeIncludingUphs**
- **employeeNonTemp**
- **employeeOrServiceProvider**
- **employeeServiceProvider**
- **faculty**
- **facultyStudentStaff**
- **lockout**
- **overseersTrustees**
- **religiousCommunities**
- **researchers**
- **serviceProviders**
- **staff**
- **student**
- **uphs**
- **visitingFaculty**
- **visitingScholar**

Reference groups via loader (continued)

- Courses
- Could have include/exclude
- Could filter which courses are needed
- Each course should be a folder
- Course list, instructors, guests, etc

Reference groups via loader (continued)

Grouper, the Internet2 gro

https://medley.isc-seo.upenn.edu/grouper/browseStemsAll.do?currentNode=e8e1a884c34b48ef9b769604d6f19e

Group types

Lite UI

Help

Grouper is sponsored by

Browse or list groups

Current location is:

Root: penn: community: student: course: 2013C: EG: CIS: 110: **001**

- all
- assistants
- assistants excludes
- assistants includes
- assistants system of record and includes
- assistants_systemOfRecord
- guests
- instructors
- instructors excludes
- instructors includes
- instructors system of record and includes
- instructors_systemOfRecord
- students
- students excludes
- students includes
- students system of record and includes
- students_systemOfRecord

Search groups Advanced groups search

Search groups

Reference groups via loader (continued)

- Employee orgs similar to courses
- Should organize such that changes in org namespace do not affect group names (been burned)

Reference groups via loader (continued)

Grouper, the Internet2 gro

https://medley.isc-seo.upenn.edu/grouper/browseStemsAll.do?currentNode=4a1c190dea4444a9a806590b86492

Penn
University of Pennsylvania

Grouper

Welcome Michael Christopher Hyzer (mchyzer, 10021368) (active) Staff - Isc Administra... Log out Act as admin Change

My tools
- Explore
- Search
- Folder workspace
- Group workspace
- Entity workspace
- Group types
- Lite UI
- Help

Browse groups hierarchy

You can look for groups throughout the hierarchy.
(You might not be able to see some groups if you lack appropriate privileges.)

Browse or list groups

Current location is:
Root: penn: community: employee: org: **9147 - ASTT and Information Security**

9147 - ASTT and Information Security

Search groups

Advanced groups search

Search groups

Search from Root

Reference groups via loader (continued)

- Employee orgs can have rollups based on descendant orgs

Reference groups via loader (continued)

Grouper, the Internet2 gro

https://medley.isc-seo.upenn.edu/grouper/browseStemsAll.do?currentNode=59d99997afa44a82a0b4ec1f895889

Penn
University of Pennsylvania

Grouper

Welcome Michael Christopher Hyzer (mchyzer, 10021368) (active) Staff - Isc Administra... Log out Act as admin Change

My tools

- Explore
- Search
- Folder workspace
- Group workspace
- Entity workspace
- Group types
- Lite UI
- Help

Browse groups hierarchy

You can look for groups throughout the hierarchy.
(You might not be able to see some groups if you lack appropriate privileges.)

Browse or list groups

Current location is:

Root: penn: community: employee: org: **91XX - Information Systems and Computing Parent**

- 91XX - Information Systems and Computing Parent
- 91XX - Information Systems and Computing Parent excludes
- 91XX - Information Systems and Computing Parent includes
- 91XX - Information Systems and Computing Parent system of record
- 91XX - Information Systems and Computing Parent system of record and includes

Reference groups via loader (continued)

Grouper, the Internet2 gro

https://medley.isc-seo.upenn.edu/grouper/populateGroupMembers.do?groupName=91XX_rolluporg_systemOfR

Folder workspace
Group workspace
Entity workspace
Group types
Lite UI
Help

Root: penn: community: employee: org: 91XX - Information Systems and Computing Parent: **91XX - Information Systems and Computing Parent system of record**

Membership list

- (•) Show DIRECT members of this group
- ( ) Show INDIRECT members of this group
- ( ) Show ALL members of this group (direct and indirect)

Change display

Enter search text to find members in the list:

Search for members

First name | Change sort attribute

- [ ] **91YY - ISC Other Parent** is a direct member
- [ ] **AIS - Administrative Information Parent** is a direct member
- [ ] **ITS - Information Technology Services Parent** is a direct member
- [ ] **NETO - Network Operations Parent** is a direct member

Remove selected members | Remove all members

Add members | Create composite group | Back to group summary

Grouper is sponsored by INTERNET2

Reference groups via loader (continued)

- Loader has 5 categories
  - SQL_SIMPLE
  - SQL_GROUP_LIST
  - LDAP_SIMPLE
  - LDAP_GROUP_LIST
  - LDAP_GROUPS_FROM_ATTRIBUTES
- See grouper loader wiki and intro images

Composite groups

- Three types of composites
  - Union
    - Never use this, just add group as member of another group which is more efficient
  - Intersection
    - Good for requiring members of a group to be members of another group
  - Minus
    - Good for excluding people from a group

Composite groups (continued)

- You can set these up
  - Manually
  - Via loader attributes
  - Via group attributes

Composite groups (continued)

- Composite include/exclude can delegate privileges well
- “System of record” groups is the group used prior to the composite calculation
- Composite groups do not remove the user from the system of record group

Composite groups (continued)

- Rules to the rescue
- Grouper rule can remove user from the system of record group when not employee
- When rehired, user will have to go back through the intake process
- Will not work with loader system of record (should *never* edit that!)

Permissions inheritance

- Penn uses permissions in several apps
- One (which is not quite live yet) is managing unix permissions

Permissions inheritance (continued)

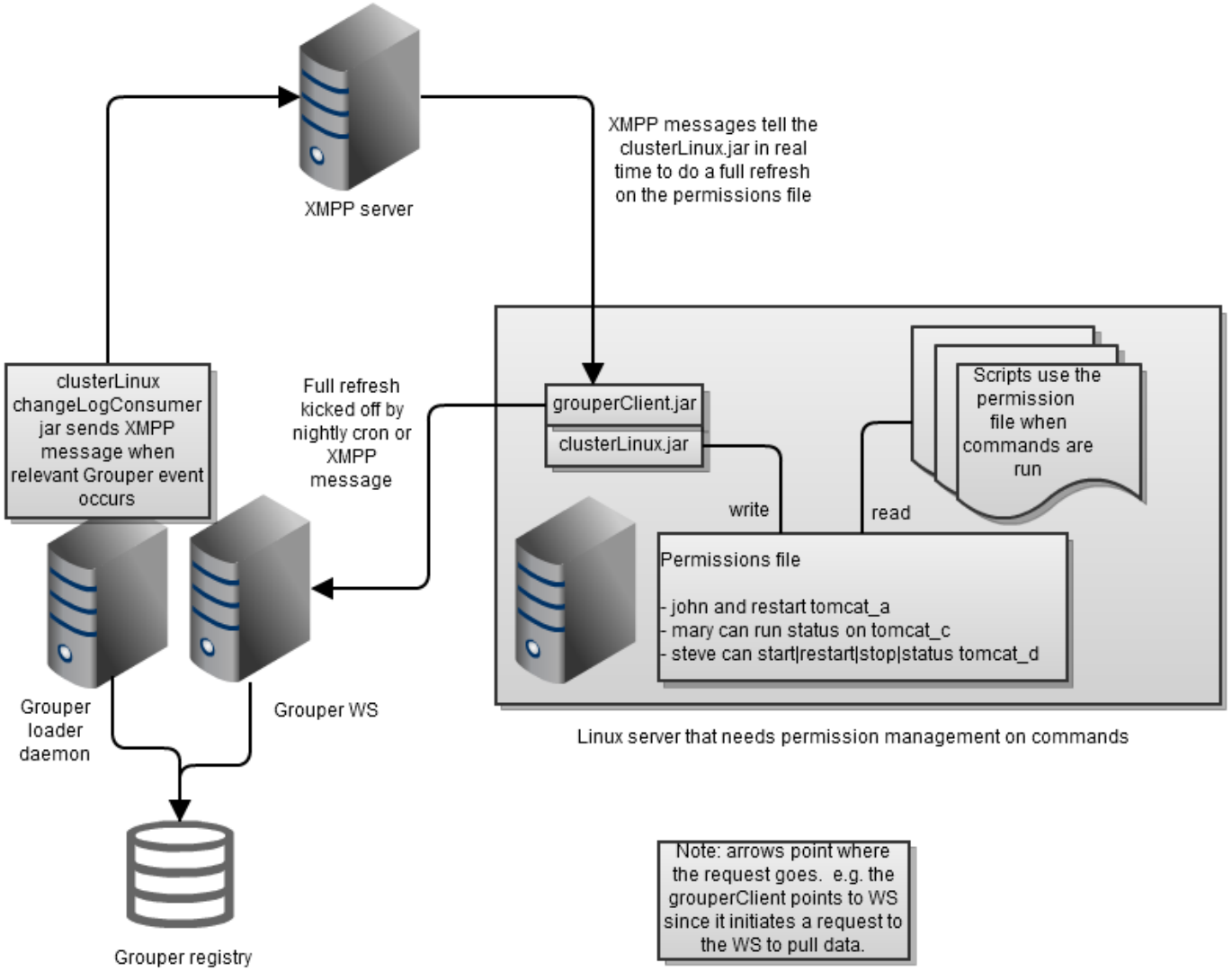
- Support staff for applications have various permissions for various applications
- Restart tomcat
- Stop tomcat
- Start tomcat
- Status tomcat
- Apache configtest
- Apache graceful
- View logs
- Redeploy

Permissions inheritance (continued)

- Users are the unix users
- Role is clusterUser
- Permission is the application
- Action is tomcatRestart / apacheGraceful / etc

- Real time and batch provisioning

Permissions inheritance



Permissions inheritance (continued)

- Group inheritance
  - Could have a group of student-based applications support staff that all share the same permissions

Permissions inheritance (continued)

- Role inheritance
  - There could be a clusterAdminRole role that inherits everything that clusterRole has, and includes all actions on all applications

Permissions inheritance (continued)

- Action inheritance
  - “tomcatAll” action could include: tomcatStatus, tomcatRestart, tomcatStop, tomcatStart
  - “clusterAll” action could include all actions to give someone full control of app

Permissions inheritance (continued)

- Permission inheritance
  - Can make collections of applications so you can assign permissions to multiple related applications with one assignment
  - E.g. researchApplications could include the five permissions for the five research applications

Progression basic to production

- Start with the installer
- Do manual builds based on installer output
- Tweak some config settings, see changes

Progression basic to production (continued)

- Subject source
  - SQL or LDAP
  - Might have more flexibility with JDBC (make a view or data feed with whatever you want)
  - If everything you need is in JNDI, and you have a highly available env, use that

Progression basic to production (continued)

- Subject source
  - Subjects should “always” be resolvable
  - ID generally is an opaque unchanging permanent id
  - Identifier is a netId, eppn, something that needs to resolve to a subject

Progression basic to production (continued)

- Subject source
  - Description is what is generally shown on screen, at Penn:
  - Michael Christopher Hyzer (mchyzer, 10021368) (active) Staff - Isc Administrative Systems Tools And Technologies - Application Architect (also: Alumni)

Progression basic to production (continued)

- Customize the UI
- At least put a logo (media.properties)

INTERNET2®

Progression basic to production (continued)

- Customize the UI authentication
- Easy with shib, CAS, cosign
- Web server plugins will work with REMOTE_USER
- Can do a servlet filter with whatever authentication

Progression basic to production (continued)

- Look in media.properties, grouper.properties, grouper-loader.properties, see which settings you want to change

Progression basic to production (continued)

- Provision to LDAP / AD
  - PSP
  - Batch and real-time

Progression basic to production (continued)

- Document your Grouper deployment for your users
- Delegate privileges for high level folders as needed
- Train admins on using Grouper
- Integrate projects

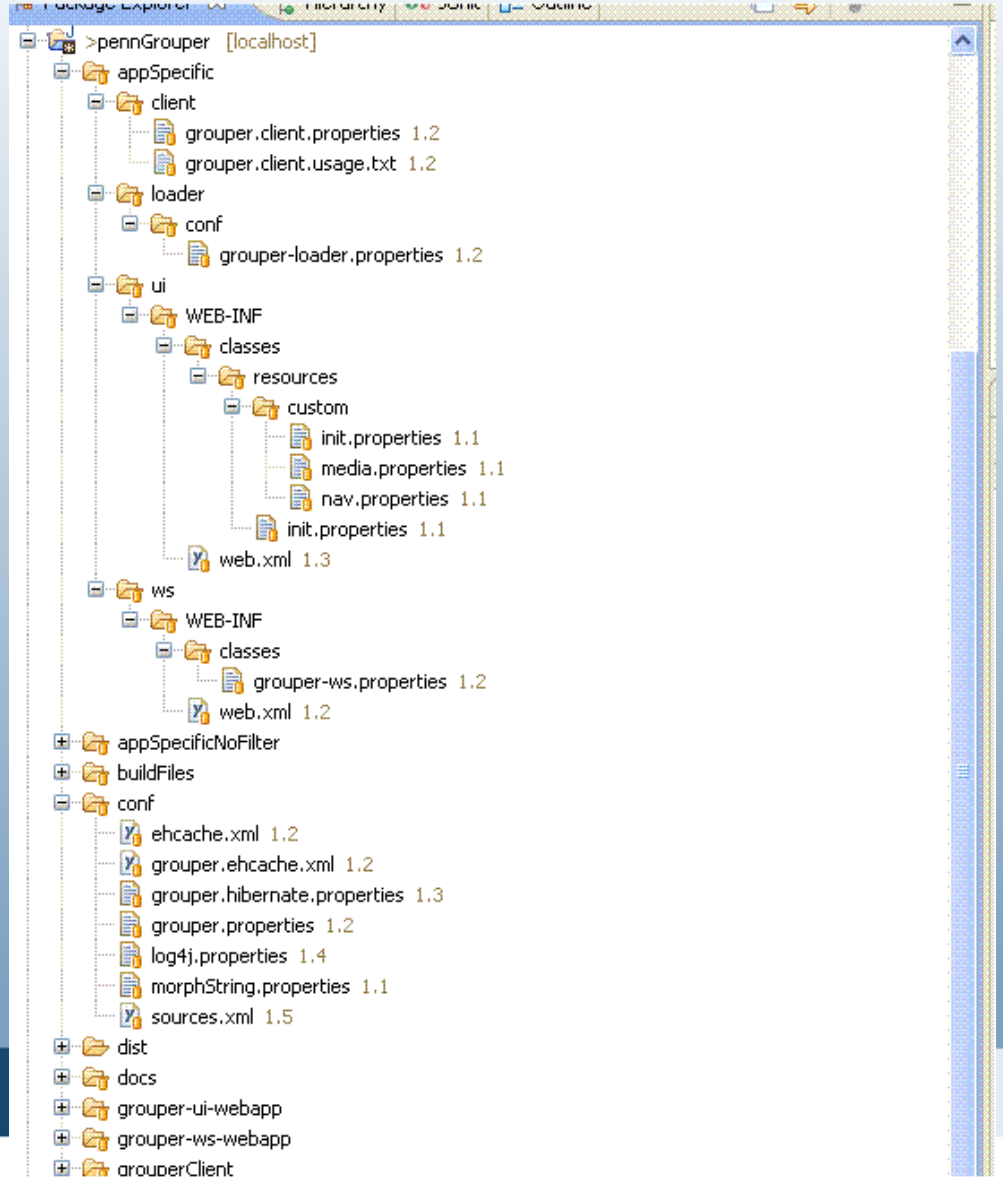
Progression basic to production (continued)

- Decide which environments to have
  - Prod
  - Test
  - Dev?
  - Train?
- See which config settings are different for each environment
- Keep your settings in your revision control
- Have a build script to war up your builds

Manage Grouper in multiple environments

- Penn shared an ant build script
- Out of the box builds a dev / test / prod

Manage Grouper in multiple environments



Manage Grouper in multiple environments

- Config files have variables who’s values are controlled by the build.properties

Manage Grouper in multiple environments

Here are the grouper.hibernate.properties variables

```
hibernate.connection.url = @dbUrl@

hibernate.connection.username          = @dbUser@

hibernate.connection.password          = @dbPass@
```

There are entries per env. Note the passwords are encrypted with the morphString Internet2 library, so the encrypted values are in a file system file (better for storage of config files in CVS nad hiding plaintext passwords)

```
devDbUrl=jdbc:oracle:thin:@devserver:1521:devsid
testDbUrl=jdbc:oracle:thin:@testserver:1521:testsid
prodDbUrl=jdbc:oracle:thin:@prodserver:1521:prodsid

devDbUser=myuser
testDbUser=myuser
prodDbUser=myuser

localdevDbPass=r:/home/appadmin/pass/grouper/grouperMorphDev.pass
devDbPass=/home/appadmin/pass/grouper/grouperMorphDev.pass
testDbPass=/home/appadmin/pass/grouper/grouperMorphTest.pass
prodDbPass=/home/appadmin/pass/grouper/grouperMorphProd.pass
```

Thanks!

Further information:

Infosheets, mail lists, wiki, downloads, etc:
www.internet2.edu/grouper

Grouper demo server:
https://grouperdemo.internet2.edu/