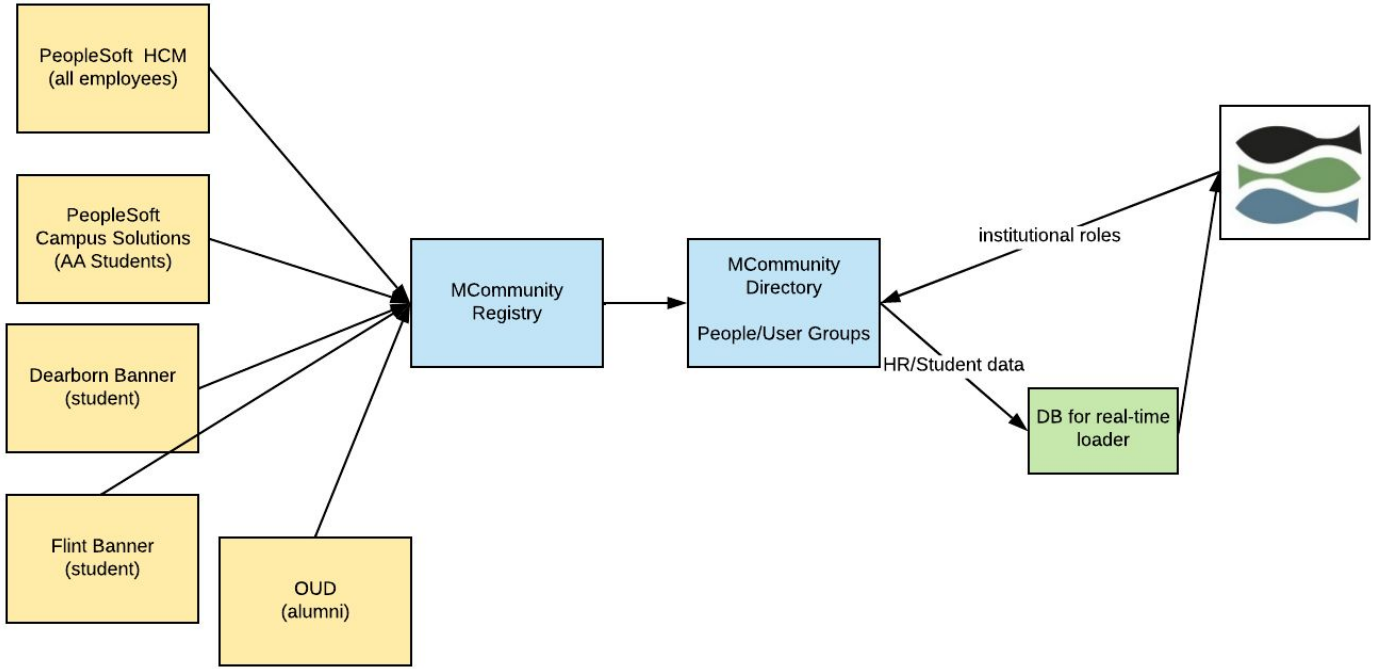

Wolverine vs Grouper 2: ABAC

# Current state of Grouper ABAC at Michigan

- v5.4 of Grouper that is "Production-ready" was released Oct 5th
- We are testing ABAC functionality in our non-Production environments
  - We need to learn how much help departments will need from us
  - We also need to learn how well this will scale
- We are working closely with the Grouper developers on ABAC functionality
  - You should too!

# U-M environment



# Michigan HR appointment data

- department with hierarchy
- faculty/regular staff/temp staff
- jobcode and jobfamily (job classification)
- active / on leave / retired
- primary / secondary job
- supervisor
- tenure for faculty

# LDAP -- row of data in a string

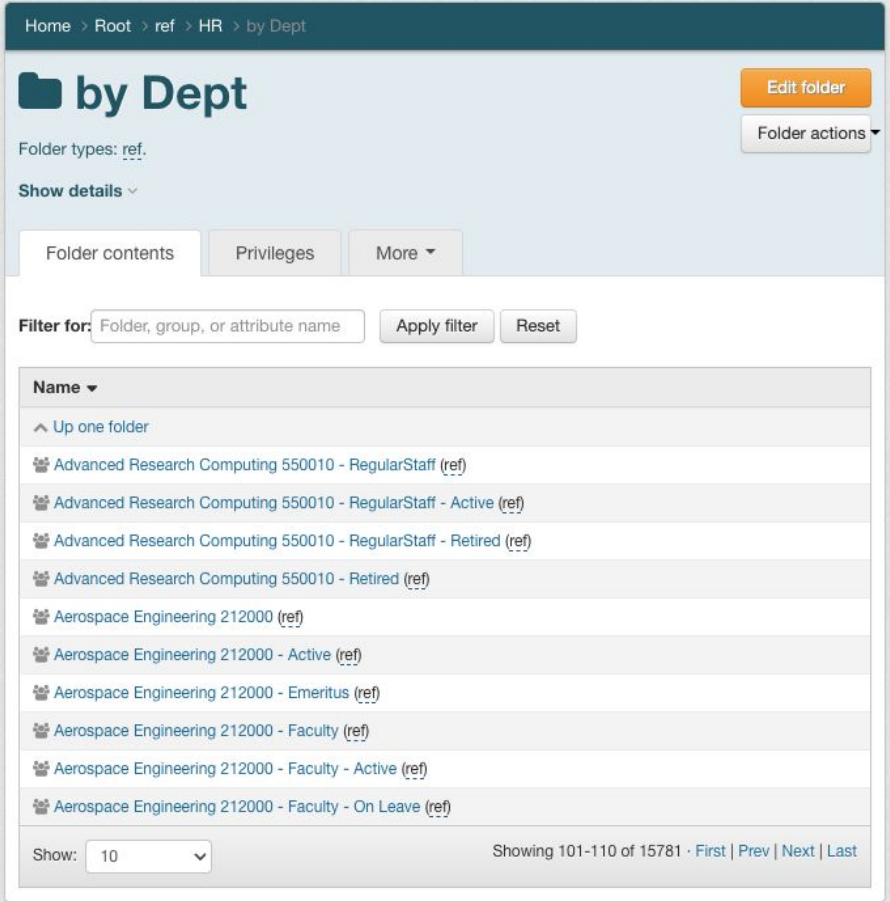
| subject_id | Category | deptId | deptGroup | primary/ secondary | job Family | Status | reg/ Temp | . . . |
|---|---|---|---|---|---|---|---|---|
| John | Faculty | 185500 | LSA | P | 10 | A | R | . . . |
| John | Staff | 185500 | LSA | S | 28 | A | R | . . . |
| . . . | | | | | | | | |


```
{jobCategory=Faculty}:{campus=UM_ANN-ARBOR}:{deptId=185500}:{deptGroup=COLLEGE_OF_LSA}:{deptDescription=LSA
Psychology}:{deptGroupDescription=College of Lit, Science & Arts}:{deptVPArea=PRVST_EXC_VP_ACA_AFF}:{jobcode=201000}:
{jobFamily=10}:{emplStatus=A}:{regTemp=R}:{supervisorId=}:{tenureStatus=TEN}:{jobIndicator=P}

{jobCategory=Staff}:{campus=UM_ANN-ARBOR}:{deptId=185500}:{deptGroup=COLLEGE_OF_LSA}:{deptDescription=LSA
Psychology}:{deptGroupDescription=College of Lit, Science & Arts}: {deptVPArea=PRVST_EXC_VP_ACA_AFF}:{jobcode=107000}:
{jobFamily=28}:{emplStatus=A}:{regTemp=R}:{supervisorId=}:{tenureStatus=NA}:{jobIndicator=S}
```

# The Dream

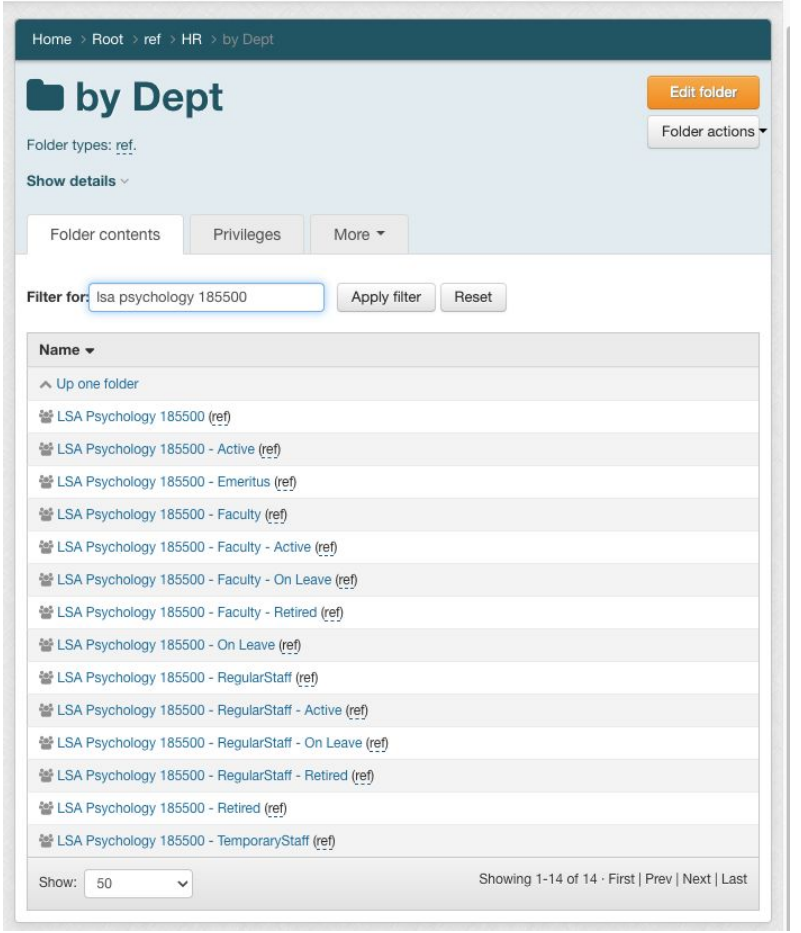
**Grouper**: We made these reference groups for units to use. These 17K groups cover the typical departmental needs.



# The Reality

**Grouper**: Here are the ones for Psychology.

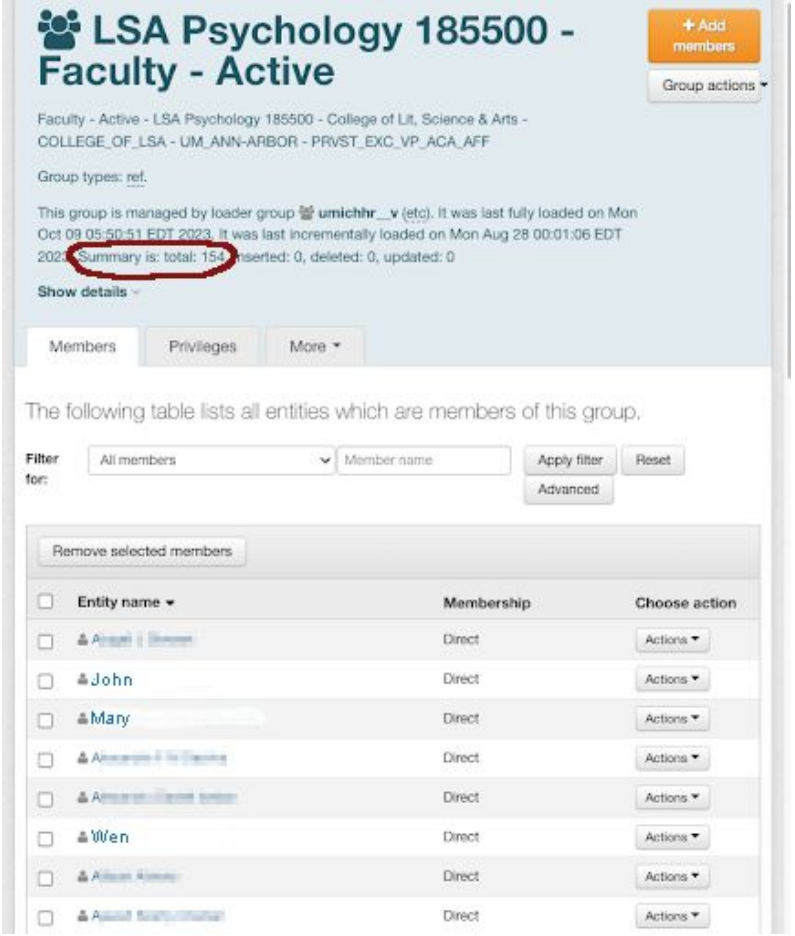
**Psych Wolverine**: Great! I need the group of my Faculty.



**Psych Wolverine**:

WHAT! This group has 154 members. I have 95 Faculty.

Who is Mary? Oh, she's mainly in SPH. I only want MY faculty.



**Grouper**: We just talked to Dentistry. They include secondary appointments in "Faculty"

**Psych**: This is Psychology, not Dentistry. I need MY Faculty.

**Grouper**: Do you want separate groups for Associate, Assistant, and Full? Mechanical Engineering wants that. Tenured?

**Psych**: No. I don’t see why anyone would want that.

**Grouper**: So, we thoughtfully and carefully built 17,000 groups that aren't actually useful.

# Grouper has relied on reference groups

- predefined "standard" combinations of affiliation data
- mechanism to build / maintain groups is efficient and scalable
- batch and near real-time (incremental) updates
- works best when many departments have the same needs

# Custom Loaders for "odd" situations

- can get non-central data
- get data combinations not common enough for global reference groups
- batch loaded, but near real-time updates could become resource hogs
- not efficient or scalable in most service architectures

# At U-M reference groups aren't working

- We have 17K reference groups just for HR data
- For faculty in a department, we have groups for all, active, on leave, retired
- Add primary/secondary, tenure/not/na, full/asst/assoc, we have 55 groups!
- We have 436 departments with faculty
- We have 2500 depts with staff -- similar explosion there
- And then there's student data

# What to do?

- build reference for all possible combinations 😵
- build reference groups as each department asks 😫
- let each department build their own 😱
- or . . .

# Enter Grouper ABAC

- Grouper admins bring curated institutional data INTO Grouper
- user chooses what data elements to combine for their groups
- can be friendlier than raw institutional data
- currently batch loading, near real-time coming
- works when reference groups do not
- more manageable than a horde of custom loaders

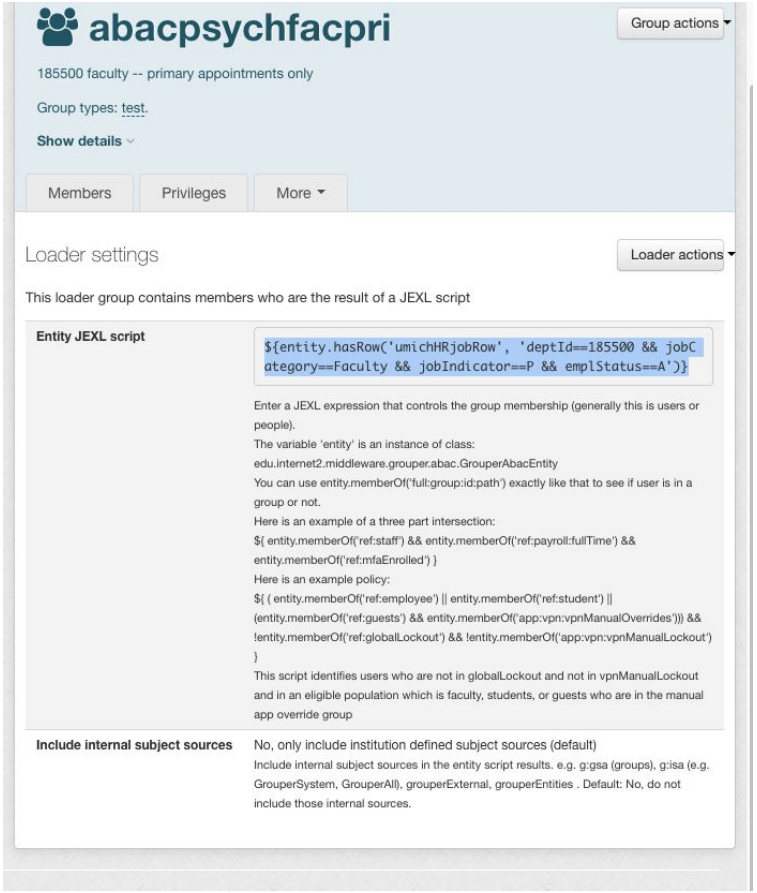
# Now units can get what they actually need!

**Psych Wolverine**: I need active Faculty whose primary department is Psychology.

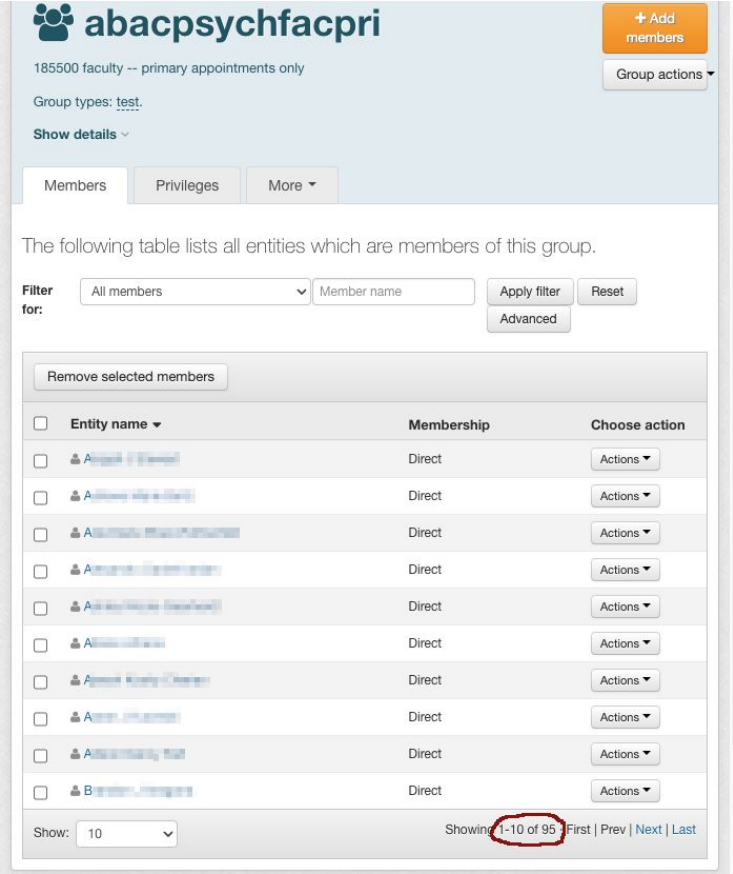
**Grouper**: So, you need all the job rows that have:
jobCategory=Faculty and deptid=185500 and jobIndicator=P and emplStatus=A

Create your group, go to Group actions / Loader, and add this JEXL script:

```
${entity.hasRow('umichHRjobRow', 'deptId==185500 && jobCategory==Faculty &&
jobIndicator==P && emplStatus==A')}
```


abacpsychfacpri
Group actions
185500 faculty -- primary appointments only
Group types: test.
Show details
Members
Privileges
More
Loader settings
Loader actions
This loader group contains members who are the result of a JEXL script
Entity JEXL script
${entity.hasRow('umichHRjobRow', 'deptId==185500 && jobCategory==Faculty && jobIndicator==P && emplStatus==A')}
Enter a JEXL expression that controls the group membership (generally this is users or people).
The variable 'entity' is an instance of class:
edu.internet2.middleware.grouper.abac.GrouperAbacEntity
You can use entity.memberOf('full:group:id:path') exactly like that to see if user is in a group or not.
Here is an example of a three part intersection:
${ entity.memberOf('ref:staff') && entity.memberOf('ref:payroll:fullTime') && entity.memberOf('ref:mfaEnrolled') }
Here is an example policy:
${ ( entity.memberOf('ref:employee') || entity.memberOf('ref:student') || (entity.memberOf('ref:guests') && entity.memberOf('app:vpn:vpnManualOverrides'))) && !entity.memberOf('ref:globalLockout') && !entity.memberOf('app:vpn:vpnManualLockout') }
This script identifies users who are not in globalLockout and not in vpnManualLockout and in an eligible population which is faculty, students, or guests who are in the manual app override group
Include internal subject sources
No, only include institution defined subject sources (default)
Include internal subject sources in the entity script results. e.g. g:gsa (groups), g:isa (e.g. GrouperSystem, GrouperAll), grouperExternal, grouperEntities . Default: No, do not include those internal sources.

# Psych: My faculty!



# Psychology faculty data

| subject_id | Category | deptId | deptGroup | primary/ secondary | job Family | Status | reg/ Temp |
|---|---|---|---|---|---|---|---|
| Mary | Faculty | 583000 | ISR | S | 20 | W | R |
| Mary | Faculty | 185500 | LSA | S | 11 | A | R |
| Mary | Faculty | 458300 | SPH | P | 10 | A | R |
| Mary | Staff | 458300 | SPH | S | 28 | A | R |
| John | Faculty | 185500 | LSA | P | 10 | A | R |
| John | Staff | 185500 | LSA | S | 28 | A | R |
| Wen | Faculty | 185000 | LSA | P | 10 | A | R |

# Functionality in, more features coming

- design decision -- get the functionality working, then tackle UI.
- can control access to the data itself by type (HR, Student . . .)
- restricting who can create groups in specific folders is coming soon
- currently uses pseudo-jexl:

```
${entity.hasRow('umichHRjobRow', 'deptId==185500 &&
jobCategory==Faculty && jobIndicator==P && emplStatus==A')}
```

# How to start using ABAC

- upgrade to Grouper v5
- decide what affiliation data detail will be needed for access decisions
- define those data elements in Grouper
- build query to fetch affiliation data, and map the results into the data elements you defined
- load the affiliation data INTO Grouper, which allows for more flexibility in building groups

# Getting affiliation data into Grouper

- fields - simple values
- rows - multiple field values that must be considered together
- you define what your institution needs
  - names and aliases
  - data types
  - sql or ldap to get values from source data

# Fields

Unitary values:

| subject_id | umichInstRoles |
|---|---|
| Mary | FacultyAA |
| Mary | RegularStaffAA |
| Aimee | StudentAA |

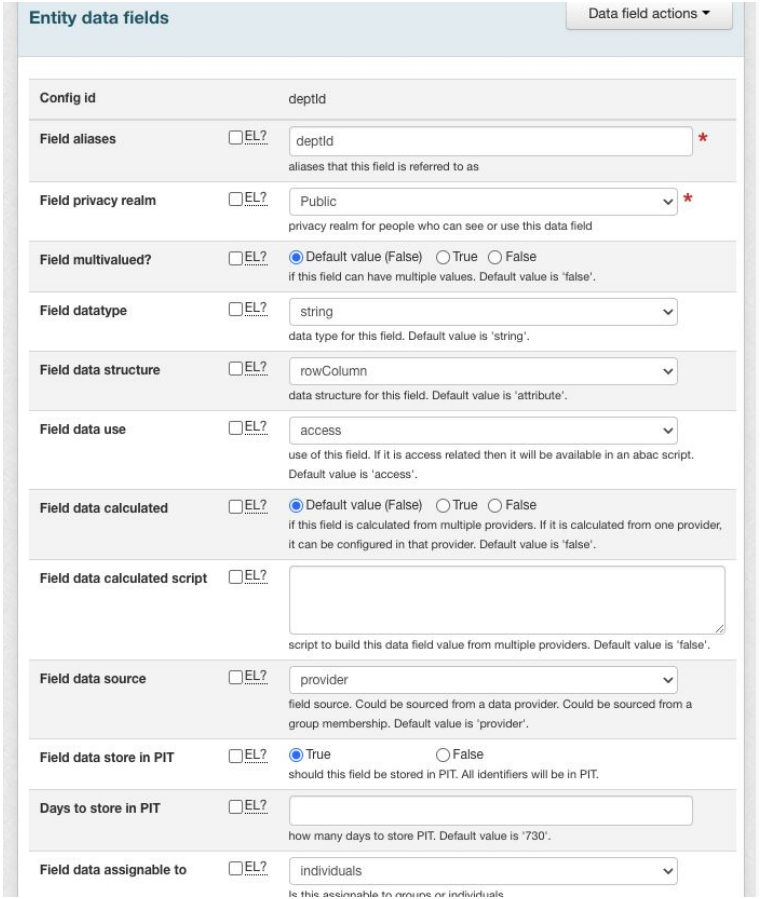
# Rows

One "unit" of data requires several fields

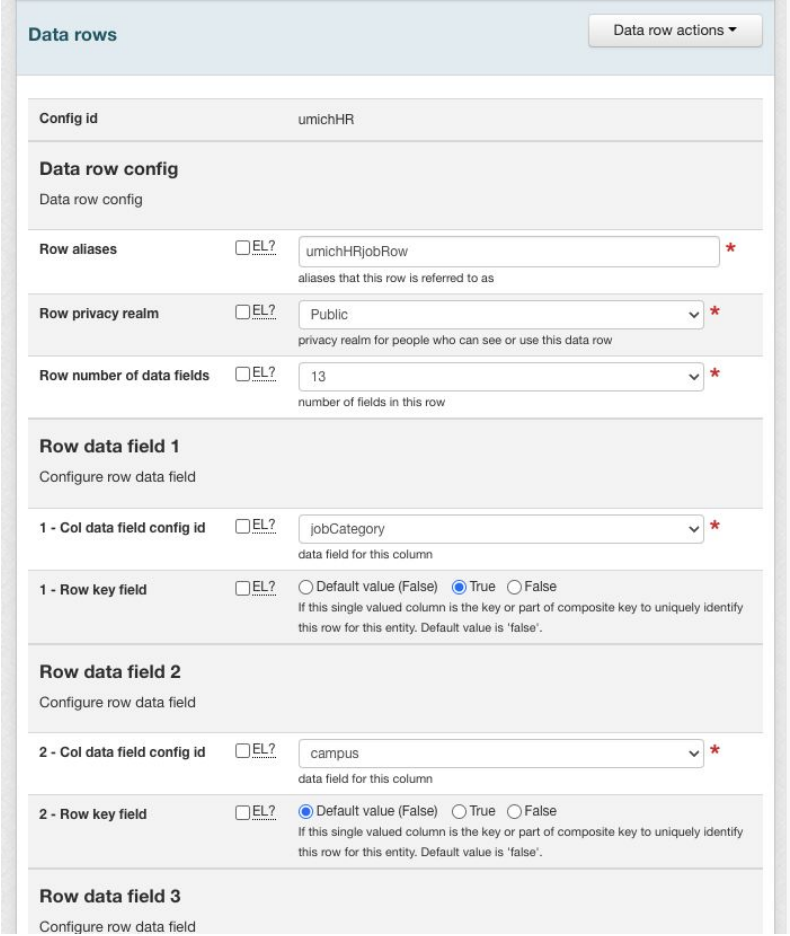
| subject_id | Category | deptId | deptGroup | primary/ secondary | job Family | Status | reg/ Temp |
|---|---|---|---|---|---|---|---|
| Mary | Faculty | 583000 | SPH | P | 20 | A | R |
| Mary | Faculty | 185500 | LSA | S | 11 | A | R |
| John | Staff | 185500 | LSA | S | 28 | A | R |
| Wen | Faculty | 185000 | LSA | P | 10 | A | R |

# Explaining your data to Grouper

Using the Grouper UI to define a field

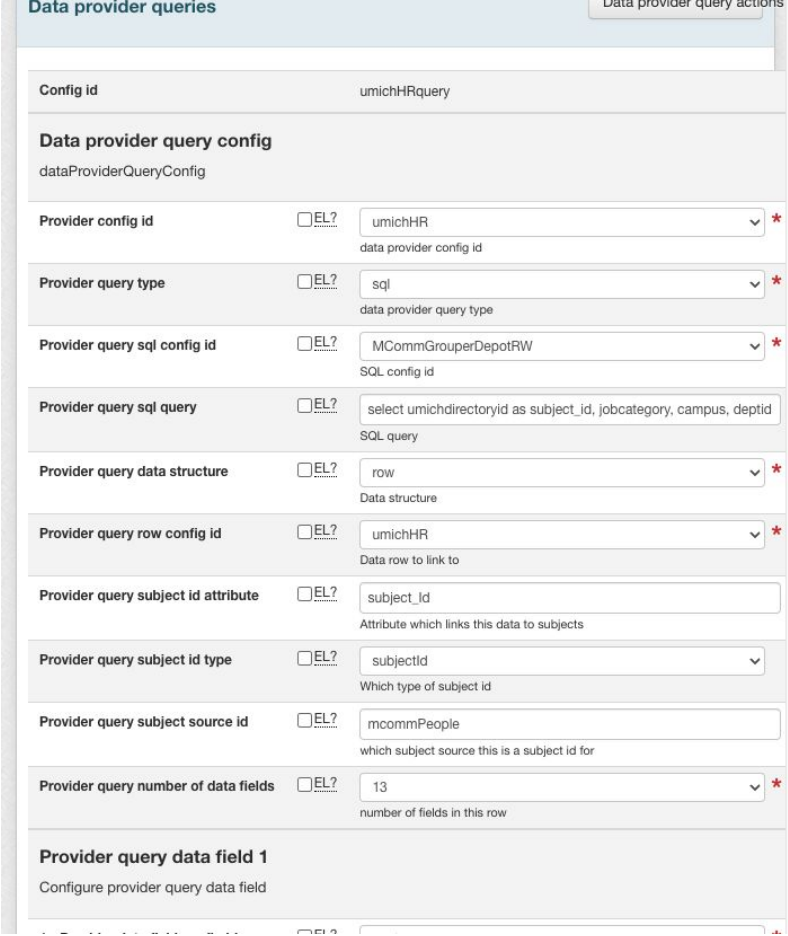


# Using the Grouper UI to define a row



specify query

map results



# The UI just builds Grouper properties files

```
grouperDataField.umichInstRoles.fieldAliases = umichInstRoles
grouperDataField.umichInstRoles.fieldMultiValued = true
grouperDataField.umichInstRoles.fieldDataType = string
grouperDataRow.umichHR.rowDataField.0.colDataFieldConfigId = jobCategory
grouperDataRow.umichHR.rowDataField.2.colDataFieldConfigId = deptId
grouperDataProviderQuery.umichHRquery.providerQueryDataStructure = row
grouperDataProviderQuery.umichHRquery.providerQueryNumberOfDataFields = 13
grouperDataProviderQuery.umichHRquery.providerQueryRowConfigId = umichHRjobRow
grouperDataProviderQuery.umichHRquery.providerQuerySqlConfigId = MCommGrouperDepotRW
grouperDataProviderQuery.umichHRquery.providerQuerySubjectIdAttribute = subjectId
grouperDataProviderQuery.umichHRquery.providerQuerySqlQuery = SELECT umichdirectoryid AS
subject_id, jobcategory, campus, deptid, deptgroup, deptdescription, deptvparea, jobcode,
jobfamily, emplstatus, regtemp, supervisorid, tenurestatus, jobindicator FROM umichhr
```

# Affiliation data is stored in Grouper!

```
SELECT data_row_assign_internal_id, name, the_text
  FROM grouper_data_row_field_assign, grouper_data_alias, grouper_dictionary
 WHERE data_row_assign_internal_id in (1149206, 1149207)
   AND grouper_data_row_field_assign.value_dictionary_internal_id=grouper_dictionary.internal_id
   AND grouper_data_row_field_assign.data_field_internal_id=grouper_data_alias.data_field_internal_id
 ORDER BY data_row_assign_internal_id ;
```

| | | |
|---|---|---|
| 1149206 | deptDescription | LSA Psychology |
| 1149206 | deptId | 185500 |
| 1149206 | jobcode | 107000 |
| 1149206 | jobFamily | 28 |
| 1149206 | regTemp | R |
| 1149206 | deptVPArea | PRVST_EXC_VP_ACA_AFF |
| 1149206 | tenureStatus | NA |
| 1149206 | emplStatus | A |
| 1149206 | jobCategory | Staff |
| 1149206 | jobIndicator | S |
| 1149206 | campus | UM_ANN-ARBOR |
| 1149206 | deptGroup | COLLEGE_OF_LSA |

| | | |
|---|---|---|
| 1149207 | deptDescription | LSA Psychology |
| 1149207 | deptId | 185500 |
| 1149207 | jobcode | 201000 |
| 1149207 | jobFamily | 10 |
| 1149207 | regTemp | R |
| 1149207 | deptVPArea | PRVST_EXC_VP_ACA_AFF |
| 1149207 | tenureStatus | TEN |
| 1149207 | emplStatus | A |
| 1149207 | jobCategory | Faculty |
| 1149207 | jobIndicator | P |
| 1149207 | campus | UM_ANN-ARBOR |
| 1149207 | deptGroup | COLLEGE_OF_LSA |

# Conclusion

- With ABAC, units can build the groups THEY need
- Some understanding of data is required -- more than for reference groups
- Grouper is the "who" part of your access policy. Overall policy enforcement may include other data/circumstances (time of day, location, etc.)