Penn LMS Grouper groups

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

Current state - three loader jobs

We have a loader job to load who has ever taken a course. We only load the courses that are used in Grouper since there are a LOT of courses and its not really needed

We have a loader job that attaches dates to courses. Currently we dont use this much but for instance the result is people who have taken the FERPA course in the last year

We have a loader job that consists of people whose training will expire in the next 30 days. The rules for the number of days are in the SQL where clause. We join that with a "team" group so that the BA can see who in the team needs training and they can bug them. Though the LMS sends out emails too

Everywhere where we use training we have "reprieve groups" which automatically set a 1 week expire date. People who need access or whose training expired get placed in the reprieve group by the BA or security analyst and they have 5 days to take the training (assuming it takes time for data to get from LMS to data warehouse, and data warehouse to Grouper).

Dynamic training policies

After a discussion on slack, here is a design for dynamic training policies

Create custom attributes on empty groups in grouper

Group a:b:c   (represents people who have taken a certain training with certain specifications (e.g. this fiscal year)

- LMS group attribute marker
    - BasedOnTrainingId: 123.4.5.6
    - MembershipType: expiresInDays
    - NumberOfDays: 365

Loader job

Assume the course completion date is in the grouper database in a table or view (e.g. by ETL or grouper SQL sync)

- Looks at groups in grouper with LMS attribute
- Looks at the type of membership (expires in days, for fiscal, etc)
- Joins on course completion date and sql logic to populate all groups at once

Note: if there are performance problems with all those joins, you might want to use the SQL sync to take parts of it and sync to tables so the performance is better.