Access Management Use Case from Penn

Overview

At Penn, we have several payroll systems

These payroll systems have **shared permissions** as far as which
business administrators can view/edit which centers of orgs, and/or orgs of employees.

Characteristics:

- The centers (groups of orgs) are not just names of centers, but could be ranges of centers.
- The orgs (groups of employees) are not just names of orgs, but could be ranges of orgs.

Access Management Requirements:

To be able to associate privileges (read or write) to a user, or to a group of users.

For example:

- **John S.** can view centers: 3, 5, 7; and orgs: 512, 611; and edit centers: 3, 7; and orgs: 611
- **medicalGroup** can view centers: 2, 4-6; and orgs 712, 934, 975-1034, 1115-1120; and edit centers: 2, 5; and orgs: 934, 982-1005

Additional Details:

- Each privilege should have an optional **startTimestamp** and **endTimestamp**. These are for permissions that are not enabled until the person starts employment, or to expire when their term ends. These aren't for time of day.
- There should also be a **hook** so that we can veto a business administrator being assigned to their own org.
- It would also be nice if we had a **hook to centralize the decision making code**, so it is not replicated in all our systems that need to make this decision. So it would be possible to make a call via web service, asking "Is user 123 allowed to view org 345?"
- There would be another hook to see which orgs (including looking at ranges) user 123 can view, and see if 345 is in there. And if not, do an external query to see which center org 345 is in, and then see if the user can read that center. Also it would honor the startTimestamp / endTimestamp. It would return T or F.