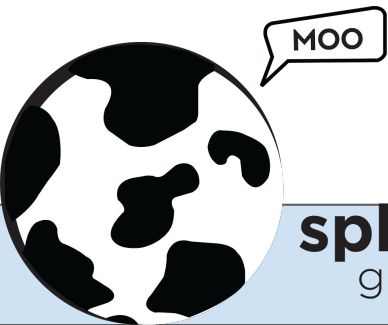


# COmanage Developer Workshop

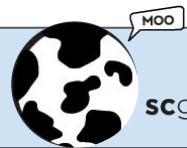
Ann Arbor • March 2020



**spherical cow**  
group

# Tour Itinerary

- About the Products
- About the Code
- About the Project
- Releases and Versioning
- Terminology
- The Data Model
- The REST API
- Multitenancy
- Authentication
- Authorization
- Plugins



# About the Products

- Registry
  - A Person Enrollment and Lifecycle Management system
- Match
  - Identity Deduplication
- (Directory)

# About the Code

- Why PHP? Why not switch to Java?
- Why CakePHP? Why not switch to Laravel?
- Why Postgres? (For Match)
- Domain Expertise + Historical Baggage

# About the Project: Organization & Roles

- Relationship between Internet2, the COmanage Project, and the Commercial Solutions Providers
- Community Working Group
- Developer {Working} Group

# About the Project: Contributing

- CCLA
- Discuss before you write
- JIRA
- Pull Requests
- Developer guidelines
- Contributors vs Committers

# About the Project: Roadmap

- Roadmap (high level)
  - <https://spaces.at.internet2.edu/display/COmanage/COmanage+Product+Roadmap>
  - Roadmap Planning
  - The 80/20 rule
    - Finding the right level of abstraction

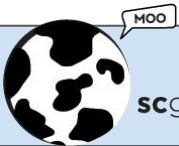
# About the Project: JIRA

- JIRA (high level)
  - <https://todos.internet2.edu/browse/CO>
  - Tracking funding
  - Issue prioritization
    - Bugs
      - Negatively affecting operational deployments?
    - Features
      - Funding?
      - Community feedback?



# Releases and Versioning

- About Semantic Versioning
  - <https://semver.org/>
  - What exactly is a breaking change in our context?
- Current and Next Releases
  - Bug Fix: Registry v3.2.4
  - Feature: Registry v3.3.0, Match v1.0.0
    - Feature freeze in place for both
  - New Feature: Registry v3.4.0, Match v1.1.0



# Releases and Versioning

- About Framework Migration
  - Registry v3 = Cake v2
  - Match v1 = Cake v3 → v4
  - Registry v5 = Cake v4
  - Balancing Framework Migration with Bug Fixes and New Feature Development

# Terminology

- COs
- COUs, CO Groups, Departments
- CO People, CO Person Roles
- Organizational Identities
- Administrators: Platform, CO, COU

# The Data Model

*Mostly about Registry from here*

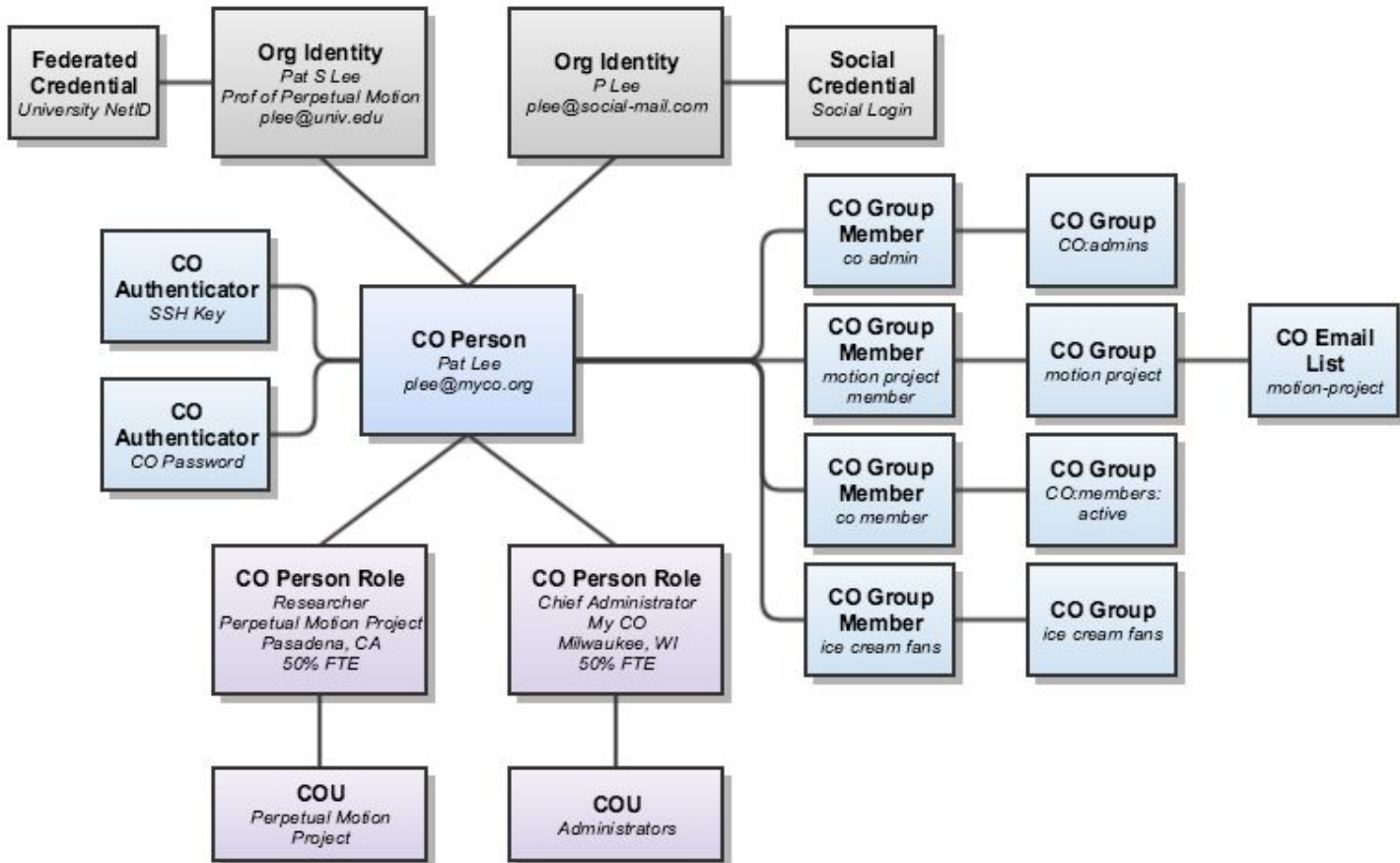
- <https://spaces.at.internet2.edu/display/COmanage/Registry+Data+Model>
- All new features start with the data model
- High level object overview
  - CO Person vs Org Identity, etc
  - Configuration objects vs Operational (Registry) objects
    - No XML files and mostly no JSON files

**COmanage  
Collaboration  
Management  
Platform**

**CO 1  
(COmanage CO)**

**CO 2**

**CO *n***



# The Data Model

- ERD
- Copy On Write
  - Compare against History Records
- Other record metadata
- ADOdb vs DBAL
  - axmls vs custom json format
- Transmogrification to v5

# The REST API

- v1 vs v2
- OpenAPI (swagger) for v2?

# Multitenancy

- Tenants are fully opaque\*
  - Except pooled organizational identities
  - Using Provisioners and OIS to pull records across tenants
- Special characteristics of the COmanage CO
- Determining the current CO



# Authnz

- Authentication
  - \$REMOTE\_USER
  - REST API Users
- Authorization
  - is\_authorized
  - RegistryAuthComponent

# Plugins

- Provisioners
- Organizational Identity Sources
- Enrollers
- (etc)
- Active plugins vs Available plugins