Scalable Web Programming

CS193S - Jan Jannink - 1/28/10

Weekly Syllabus

1. Scalability: *(Jan.)*

2. Agile Practices

3. Ecology/Mashups

4. **Browser/Client**

5. Data/Server: *(Feb.)*

6. Security/Privacy

7. Analytics*

8. Cloud/Map-Reduce

9. Publish APIs: *(Mar.)**

10. Future

* assignment due

Lesson Plan

- Projects
  - Advocates
  - Team kickoff
- Browser Scalability

Projects

- Online IDE

  - Eclipse in the cloud

- Specification Repo

  - claim verifier

- Financial Trends

  - investment finder

- Student Errands

  - Facebook Connect

- Translation API

- Twitter Trends

- Degree Requirements

  - CourseRank add on

Team Repository

- Each team will have a git repository on AFS

- First task (due on Tuesday)

  - add your name to a team repo file

    - advocate creates team file initially
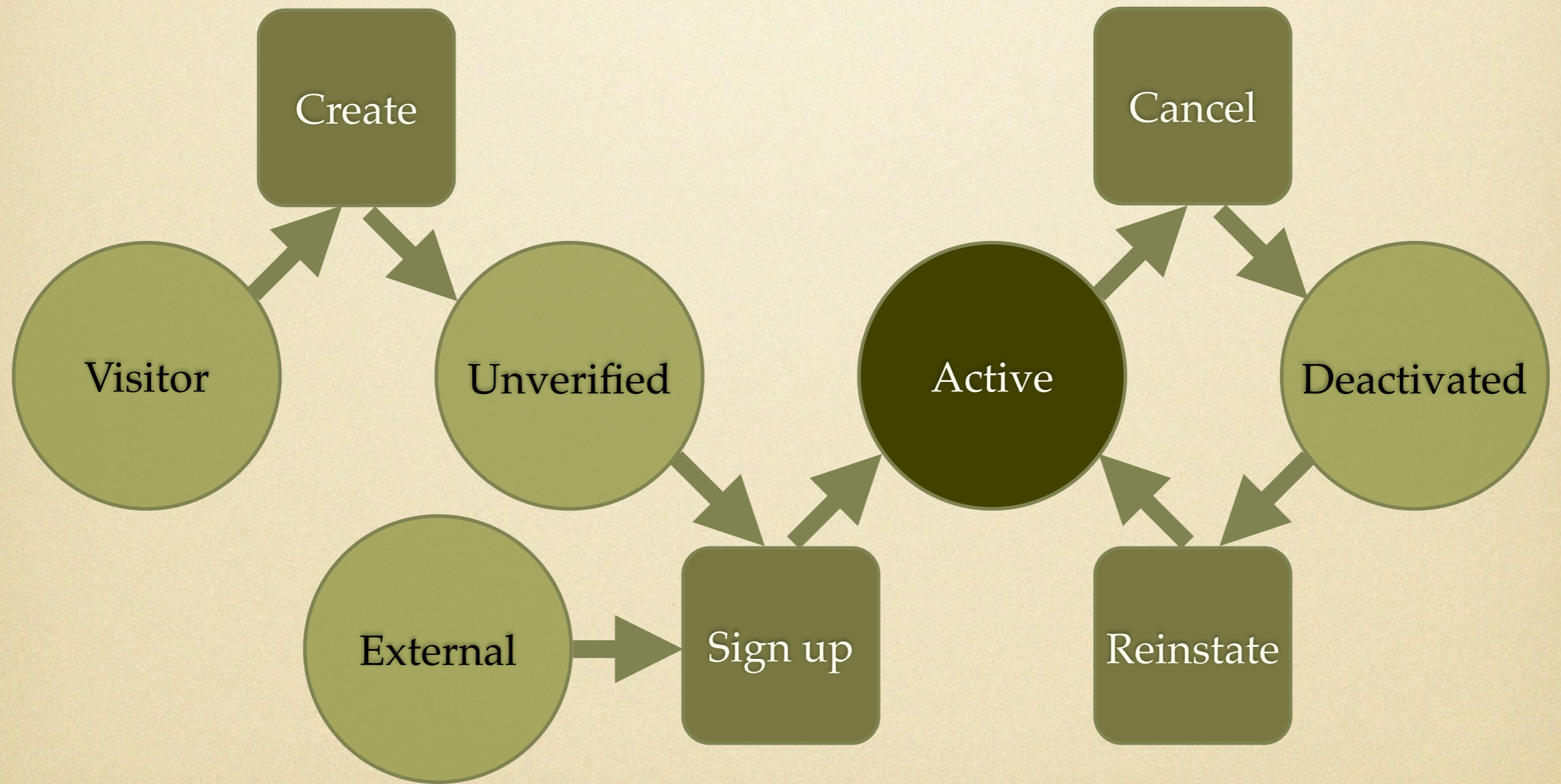
  - add a file to team repo

    - contents: email from advocate to you

Client Scalability

- Simplify data access

- One manager for all data flow

- Add caching at manager level

  - Lock personal profile info in cache

  - Asynchronously write through to servers

Account Manager API

- {x} = CreateAccount(credentials)

- {x, ...} = Login(credentials)

- Update(x)

- Logout(x)

Account States

Account Verification

- Identity proxy

  - Email

  - Sms

- External account

  - Facebook

  - LinkedIn

Data Manager API

- {x} = CreatePage()

- {x, ...} = GetPage(x)

  - have the server side construct needed set

- UpdatePage(x)

  - in general, only one updater per "x"

User Generated Data

DBItem — Basic Presentation

Account Support — User — Content — Features
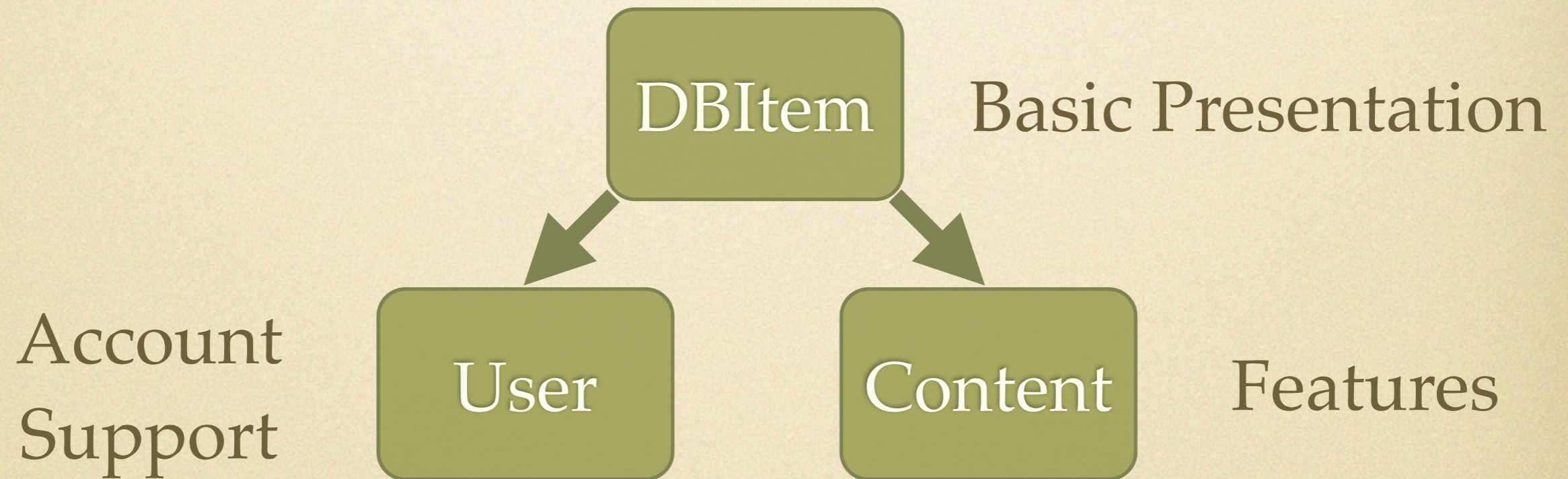
Presentation API

- Display({x, ...})

  - x.Display()

  - x.DisplayShort()

  - x.DisplayLink()

Offline Capabilities

- Google Gears

    - yet another SQLite application

    - very powerful full text index

    - very tiny code footprint

- Simply add interface to Data Manager

Data Consistency

- Don't require transactional consistency

- Eventual consistency is sufficient 99% of time

- "Just in time" consistency

  - clean up before anyone notices

  - strong links should not see inconsistent state

  - changes propagate to strong links first

Edit the Constitution

Multiple Editors

- Problems: simultaneous changes, large deletes

- Maintain edit sequence at server

  - authoritative history

- Allow full access to document history

  - recovery support

Wikipedia Example

- Founded 2001 - 14M entries

  - "Wikipedia" article in over 200 languages

- OED second edition ~ 500k entries

- Cabal

  - core group of privileged editors

- Vandalism vs. inadvertent data clobbering

Worth Checking Out

- Google Gears

  - http://gears.google.com/

- WikiTrust

  - http://wikitrust.soe.ucsc.edu/

- Bringing Silicon Valley to Sacramento

  - http://www.techcrunch.com/2010/01/24/bringing-silicon-valley-to-sacramento-why-entrepreneurs-need-to-help-rebuild-californias-it-systems/

Q & A Topics

- Team activity
  - Advocates
  - Free agents
  - Our managerial role
- Tools for teams
  - git, eclipse, newsgroup, email