Agile Experience Report
Scrum in Training

Jean Barmash, Alfresco Software
jean@alfresco.com
January 15, 2008

About Me / Alfresco

- Director, Technical Services
  - Blog – www.nywebguy.com
- Alfresco Software, www.alfresco.com
- Open Source Enterprise Content Management
  - Document Management
  - Web Content Management
  - Enterprise 2.0 / Social Computing Platform

Agenda

- Program Introduction
- Iteration 1 – Spring Program
- Iteration 2 – Summer Program
- Iteration 3 – Fall Program
- Conclusions

Training Boot Camp

- Top Financial Services Company
- 10 Week Program
- 1 Week Traditional Training
- 6-Week-long Project, different each time
- Students straight out of college

- 3 Times / Year
  - Spring - 22 People
  - Summer – 22 People
  - Fall – 16 People

Goals of the Program

- Enterprise Coding Practices
- Complete project lifecycle
- Teamwork
- Integrated Final Project
- QA Practices
- Final Presentation to Senior Management
  - Everybody must talk
  - Demos

Iteration 1 - Spring Program

- 22 Students
- Project - Operations Dashboard
- Easy technology
- Working with existing team
- Goal – move to production
  - Code Review Process
  - JUnit Tests

Basic Process for All Projects

- Agile-Like
- Teams of 3-5
  - Split up based on Technology components
  - Clearly defined responsibilities
  - Each Team has a Team Lead
- Daily Team Meetings
- Special Team for “Operations”
- Everybody in the same training room

Rough Schedule

- Week 1 – Kickoff, Design
- Week 2 – Initial Implementation
- Week 3 – Final Component Implementation
- Week 4 – Integration / QA
- Week 5 – Presentation / Practice
- Week 6 – Final Presentation

The Room

Spring Program - Conclusions

- Outcome – Pretty Good
- Major problems with integration
- Some QA, but could be better
  - Code Reviews
  - Documentation
  - Integration Tests
  - System Tests

Lessons Learned

- Need more show and tells
- Need better prioritization
- Wanted team to have flip charts of priorities
- Need “Spikes”
- Checklists for how to approach a task
- Focus on Integration from beginning

Iteration 2 - Summer Program

- Complicated research-oriented project
- Challenging Technical Problem
- Emerging Technologies
- Lots of integrations between components
- Rambunctious Class

Challenges

- Hard Technologies
- Emphasis on learning technologies
- No Build Process, almost until end
- Performance Testing – very hard
- Let Quality Lapse
  - Not as important
  - No Code Reviews
- Integration – not until late

Status Dashboard

Integration Dashboard

1 – Have talked about it, very little wor 2 – Have a working proof of concept 3 – working and tested

Component / Integration Method	27-Aug	29-Aug am	30-Aug am	30-Aug pm	31-Aug am	5-Sep	6-Sep
Integration1	1.5	2	2	2.7	2.7	2.7	3
Integration2	1.9	2.01	2.7	2.7	2.7	2.7	3
Integration3	1	2	2.7	2.7	2.7	2.7	3
Integration4	2	2	2	2	1.5	2.7	3
Integration5					1.5	2.7	3
Integration6	1.5	2	2.3	2.3	2.6	2.9	3
Integration7	3	3	3	3	3	3	3
Integration8	1.5	1.7	2.7	2.7	2.8	2.8	3
Integration9			2	2	1.5	2.9	3
Integration10	3	3	3	3	3	3	3
Integration11	2	1.75	2	2	2.7	2.8	3
Integration12	2	1	3	3	2.5	2.7	3
Integration13		2.5	2.5	2.5	2.7	2.9	3
Integration14		1.3	2	2	2	2	3
Integration15					1	2.7	3
Integration16						2	2.7

Release Process							
Team1	1.5	2	2.4	2.4	2.7	2.7	2.7
Team2	2	2	2	2	2.3	2.7	2.7
Team3	2.5	2.5	2.6	2.6	2.7	2.8	2.8
Team4	2	1.5	1.5	1.5	2.5	2.3	2.3
Team5	1	1	1	1	1	1.3	2.4
Team6	2	2	2	2.4	2.5	2.6	2.7

Summer– Lessons Learned

- Need clearer prioritization
- Dashboard was very useful
- Need to renew emphasis on Quality
- Need to prioritize build process
- Introduce Scrum?

Iteration 3 -Fall Program

- Document Generation Component
- To be shared across other teams
- Production-quality
- New Technology, but not too difficult

- Smaller Class - 16
- Most Diverse, both in terms of roles and geographies

Fall Program

- Decided to use Scrum
- Luxury of "decreeing" it
- Challenge – how do you teach and use scrum at the same time?

- Scrum Elements
  - Well Defined Sprints
  - Task Board
  - Impediments List
  - Burn Down Chart
  - Daily meetings (from before)
  - Spikes
  - Team Meetings
  - QA Push – QA Leads
  - Show and Tells
    - External
    - Internal

Prioritization Email Example

Congratulations on (almost) completing the Design Sprint (Spring #1). I think we have good momentum, let's keep it up as we get to the actual implementation of the project.

Priorities for the Week

We have three priorities left over from last week – we need to close those out ASAP, hopefully by EOD Tuesday.

1. Builds – every team should have a working build and have access to SVN. The operations team should start running the master build daily ASAP.
2. Design Documents – those need to be finished by Tuesday the latest. We have a design a review with our sponsors on Wednesday.
3. Sprint Backlog – I want to make sure each team identifies several tasks to tackle this week, estimated them and put them up on task board.

Priorities for this week:

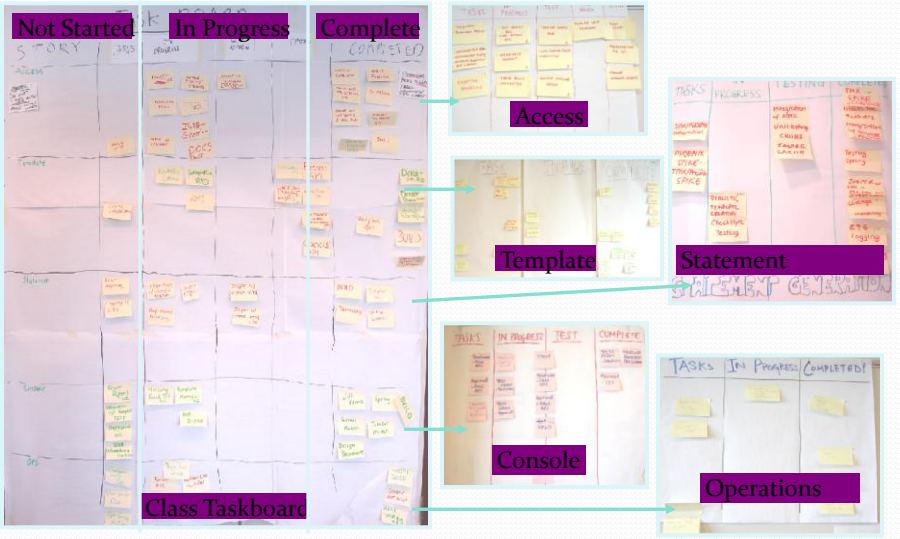
1. **Implementation & Early Integration**. Let's build our core functionality this week if possible so we can get to some interesting additional items. Try to think of what's the most basic functionality you can start with, get that working, and iterate. We want to integrate code between teams as early as possible, so whenever you have dependency with other components, make sure you agree on interfaces and start using the same interfaces.
2. **Code Review Process**. Basically, after you done writing the code (and tests if you are doing TDD), I want to do a code review before you check-in your code. This assures consistency.

Team

Scrum Experience

- Well Defined Sprints- good
- Task Board – good, after changes
- Burn Down Chart- bad
  - No Estimation
- Daily meetings (had all along)
- Spikes
- QA Push – QA Leads
- Show and Tells
  - External
  - Internal

Task Board



Problems

- What's a Task?
  - Requirements Management Tool
  - Issue Tracking
  - Task Boards
  - Dashboard
- Dashboard overload – had too many categories
- Burn down Charts
- Team Meetings

QA Dashboard Example

Test Plan	✓
JUnit Tests	✓
Build Runs Tests	✓
Code Coverage	75%
CheckStyle Style	✓
Stubbing / Faking	✓
Integration Testing	✓
Tests Modeled in Mercury	✓

Scrum Lessons Learned Summary

- Task Board – For Each Team
- Dashboard – Too Much
- Burn Down Chart
- Daily Scrum Meeting – 30 mins minimum
- QA Dashboard
- QA Leads

Social Task Board

Fall Results

- Iteration 3 - Best Project of the Three
- Most Work Accomplished
- Emphasis on Quality and Process

- Reasons
  - We got better – this was my third time
  - Students had good attitude, smaller team
  - Scrum

Questions / Suggestions?