Successful Distributed Agile Team Working Patterns

Scrum Gathering Shanghai
April 19-20, 2010
By Monica Yap
Monica Yap

• Started program in 1992
• Started doing Agile in 2000
• Became a Scrum and XP coach
• Still love to pair with developers
• Loves to travel around the world
Outline

• Distributed Development Myths
• Can Distributed Development Be Successful?
  – Case Study 1 – WDS Global
  – Case Study 2 & 3 – SirsiDynix & Xebia
• Challenges You Must Overcome
• Distributed Scrum Models
• Boot Camp
• Rotating Guru
Outline – Con’t

• High Communication Modes
• Remote Pairing
• Ambassador
• Shared Community
• Integrated Global Code Base & Single CI Server
• Technology Alignment
• Other Distributed Team Best Practices
• Q&A
Distributed Development Myths

• There is no additional overhead
• You need more specification defined upfront in distributed development
• You can’t do Scrum/Agile with distributed teams
Can Distributed Development Be Successful?

- Better chance of success with Scrum + XP
- Scrum maintain customer focus
  - All locations are delivering the high priority features
  - Daily standups or hand offs between locations
  - Sprint reviews
- XP solves integration and quality problems
  - Continuous integration
  - TDD
  - Refactoring
  - Pairing
Case Study 1 – WDS Global

- One global XP team in 3 locations
  - US, UK, Singapore
  - 4+ years
  - Integrated global code base
  - Single CI server
  - Daily hand offs
  - Weekly production releases
  - 2 week iterations
## Case Study 2 & 3 – SirsiDynix & Xebia*

<table>
<thead>
<tr>
<th></th>
<th>Colocated Scrum*</th>
<th>Waterfall*</th>
<th>SirsiDynix Distributed Scrum**</th>
<th>Xebia Distributed Scrum</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Person Months</strong></td>
<td>54</td>
<td>540</td>
<td>827</td>
<td>125</td>
</tr>
<tr>
<td><strong>Lines of Java</strong></td>
<td>51000</td>
<td>58000</td>
<td>671688</td>
<td>100000</td>
</tr>
<tr>
<td><strong>Function Points</strong></td>
<td>959</td>
<td>900</td>
<td>12673</td>
<td>1887</td>
</tr>
<tr>
<td><strong>FP per dev. per month</strong></td>
<td>17.8</td>
<td>1.7</td>
<td>15.3</td>
<td>15.1</td>
</tr>
</tbody>
</table>

* See Reference on Jeff Sutherland’s distributed Scrum case study
Challenges You Must Overcome

• Build trust
• Time zone, language and cultural barriers
• Communication
• Technical alignments
  • Architecture design
  • Best engineering practices
  • Standards
  • Tooling
• Project and process management
  • Synchronizing work between distributed locations
Distributed Scrum Models

- Common distributed Scrum models*
  
  **Isolated Scrums** – Teams are isolated across locations
  
  **Distributed Scrum of Scrums*** – Scrum teams are isolated across locations and integrated by a Scrum of Scrums
  
  **Fully distributed Scrums** – Experienced Scrum teams are cross-functional with members distributed across locations

- The right team structure and organization
  
  – Cross functional team at each location
  
  – ScrumMaster at each location

*Best practice recommended by Scrum Alliance – Jeff Sutherland
Boot Camp – Distributed Pattern

• Co-located project kick off
• Whole team (or enough) from each location
• Team go through initial Sprints together
• Co-located key milestones – release planning
• Start with a shared understanding
  – Customer context
  – Align everyone on standards, tooling, definition of done, natural roles in team formed
• Start building trust
Rotating Guru – Distributed Pattern

- Different team members rotate through locations continuously
- Work with remote team members locally
- Visiting member gains local context
- Cross pollination of technical knowledge
- Maintain personal trust
High Communication Modes – Distributed Pattern

• Face-to-face communication is best
• Establish overlapping work hours
• Provide high bandwidth communication channels
  – Must be easy to use, not always high cost
  – Video conferencing (Skype, streaming servers)
  – Desktop sharing (VNC tools, TeamViewer, etc.)
  – Instant messaging (GoogleTalk, MSN Messenger, etc.)
  – Voice always on (Skype phone)
High Communication Modes – Con’t

• Dedicated space setup with conference equipment
• Involve other location at key events
  – Daily Scrum/handoffs (always)
  – Sprint planning (may not be always)
  – Sprint review (may not be always)
  – Retrospectives (may not be always)
  – Use ‘talking stick’
Remote Pairing

• Team members in different locations pair together
  – Story tasks
  – Problem solving
  – Debugging

• Fixed time, fixed schedule

• Continue to build trust and understanding

• Avoid code ownership

• Share knowledge and experience
Ambassador

- Act as continuous local champion for remote location
- With Rotating Guru
  - The rotating guru become ambassador
  - Trust relationship continued
- Frequent communication (High Communication Modes)
- Frequent pairing (Remote Pairing)
- Maintain trust relationship
Shared Community

• A virtual community accessible by all locations online and real time

• Provide tools
  – One Wiki – one source for knowledge
  – Online project management tool – transparency
  – Shared mailing list and folder – everyone is notified

• Publish each team member info
  – Profile, photo, personal interests, etc.

• Build common language & knowledge

• One source for information
Integrated Global Code Base & Single CI Server

- Single code repository
  - Everyone check into same repository
  - Global policy for “when can I check in”

- Single CI server
  - Ensure high bandwidth connections
  - Hot back up in each location (power outages, server down)
  - Global policy when build breaks
Technology Alignment

• Technology alignment
  – Coding standards
  – Framework used
  – How to introduce new technology

• Tool alignments
  – IDE, project configuration
  – Change policy

• Engineering best practices
  – TDD, refactoring, pairing
  – Code coverage policy
Other Distributed Team Best Practices

• Daily meetings of Product Owner team
• Hourly (minimum) automated builds from one central repository
• No distinction between developers at different sites on the same team
Conclusion

• Expectation must be set
  – With additional cost
  – Overhead
  – Burden on team members

• There are many success stories, but they put in a lot of effort to make it successful
References

• Follow the Sun: Distributed Extreme Programming Development – by Monica Yap

• Distributed Scrum case study – by Jeff Sutherland
Q & A