Using Social Network Analysis to Investigate the Collaboration between Architects and Agile Teams

Ömer Uludağ, 22nd of May, 2019, XP Conference, Montreal, Canada
Outline

Motivation

Foundations

Case Study

Social Network Analysis

Lessons Learned
Outline

Motivation

Foundations

Case Study

Social Network Analysis

Lessons Learned
Agile Software Development

- Small, co-located self-organizing teams
- Closely working with business customers on single-project context
- Frequent feedback loops
- Maximization of quality for software products
- Rapid iterations
- Maximization of customer value

- Agile methods are originally tailored for small teams
- Large enterprises intend to benefit from agile methods and, thus, are interested in scaling them to include larger teams and inter-team coordination and communication

Agile methods share some common characteristics

- Existing Agile Methods have common characteristics:
  - Iterative and incremental life cycles
  - Focus on small releases
  - Co-located teams
- They more or less agree to the values of the Agile Manifesto

### Problem with emergent architecture:

- In large-scale agile development, this design might not work anymore
- It requires excessive redesign efforts, architectural divergence, and functional redundancy

### Problem with intentional architecture:

- Fear of “BDUF”
- It may delay starting point of implementation and the architecture might not be contemporary

---

**Principles behind the Agile Manifesto**

The right balance between emergent and intentional design is required.

**Abrahamson et al. (2010): Agility and architecture: can they coexist?**

**Nord et al. (2012): Making architecture visible to improve flow management in lean software development.**

**Mocker (2009): What is complex about 273 applications? untangling application architecture complexity in a case of European investment banking.**
Outline

Motivation

Foundations

Case Study

Social Network Analysis

Lessons Learned
Using Social Network Analysis to Investigate the Collaboration between Architects and Agile Teams

Using Social Network Analysis to Investigate the Collaboration between Architects and Agile Teams

Kniberg and Ivarsson (2012): Scaling Agile @ Spotify with Tribes, Squads, Chapters & Guilds.
Communication Network Patterns

Outline

Motivation

Foundations

Case Study

Social Network Analysis

Lessons Learned
**Case Study Design**

**Goal:**
Investigating the collaboration between architects and agile teams in a large-scale agile development endeavor

**Semi-structured Interviews**

<table>
<thead>
<tr>
<th>No</th>
<th>Role</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Chief Solution Architect</td>
<td>34 minutes</td>
</tr>
<tr>
<td>2</td>
<td>Solution Architect</td>
<td>42 minutes</td>
</tr>
<tr>
<td>3</td>
<td>Solution Architect</td>
<td>53 minutes</td>
</tr>
<tr>
<td>4</td>
<td>Release Train Engineer</td>
<td>22 minutes</td>
</tr>
<tr>
<td>5</td>
<td>Agile Coach</td>
<td>1 hour and 13 minutes</td>
</tr>
<tr>
<td>6</td>
<td>Developer</td>
<td>31 minutes</td>
</tr>
<tr>
<td>7</td>
<td>Developer</td>
<td>18 minutes</td>
</tr>
</tbody>
</table>

**Social Network Analysis**

- Online survey through Questback
  - How frequently do you exchange **architectural information** with this person?
  - How frequently do you see this person?
- Total amount of teams: 5
- Total amount of program members: 62
- Response rate: 47%
- Visualization of survey data through Gephi
- 758 architecture sharing connections
- Total amount of created graphs: 116
In 2016, the German Consumer Electronics company decided to relaunch a CRM project. Due to the complexity of the project, SAFe was chosen. The adoption of SAFe was initiated with the help of a pilot project. After a few Program Increments, the large-scale agile development endeavor showed a quick learning curve. SAFe has been adopted on the program level. According to the case, SAFe did not provide detailed guidance on the coordination of their agile teams. The case organization decided to organize the Agile Release Train with the help of the Spotify Model (PI 10).
Adopting the Spotify Model for organizing the large-scale agile development program

- Program members were divided into Tribes, Chapters, and Squads
- Due to the ongoing transformation, Guilds (CoPs) were paused
- The Tribe consisted of a Scaled Team and four Squads
- The large-scale agile development program also included both a chapter for solution architects as well as a chapter for business process architects

<table>
<thead>
<tr>
<th>Program Members</th>
<th>Tribe</th>
<th>Chapter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program Owner</td>
<td>PO</td>
<td></td>
</tr>
<tr>
<td>Delivery Manager</td>
<td>DM</td>
<td></td>
</tr>
<tr>
<td>IT Project Manager</td>
<td>IT-PM</td>
<td></td>
</tr>
<tr>
<td>Product Owner</td>
<td>PO</td>
<td></td>
</tr>
<tr>
<td>Business Process Architect</td>
<td>BPA</td>
<td></td>
</tr>
<tr>
<td>Data Analyst</td>
<td>DA</td>
<td></td>
</tr>
<tr>
<td>Development Team</td>
<td>Dev Team</td>
<td></td>
</tr>
<tr>
<td>Test Analyst</td>
<td>TA</td>
<td></td>
</tr>
<tr>
<td>Scrum Master</td>
<td>SM</td>
<td></td>
</tr>
<tr>
<td>Solution Architect</td>
<td>SA</td>
<td></td>
</tr>
</tbody>
</table>

© sebis 190522 Uludağ - Using Social Network Analysis to Investigate the Collaboration between Architects and Agile Teams
Outline

Motivation

Foundations

Case Study

Social Network Analysis

Lessons Learned
Social network analysis provides suitable means for visualizing architecture sharing in a large-scale agile development program.

Program Overview

Program Overview with Salient Roles
Solution architects facilitate constant exchange of architecture-related information between agile teams

Inter-Team Architecture Sharing with Focus on Architects and Frequency
Solution architects are supporting team architects and play an important role in sharing architecture-related information within the team.

Inter-Team Architecture Sharing with Focus on Architects and Frequency
Constant intra-team architecture sharing is mainly performed by architects.
Measure 1 – Degree Centrality

Definition of Degree Centrality
Degree centrality assigns an importance score based purely on the number of links held by each node.

Formula for Normalized Degree Centrality (NDC)

\[ NDC = \frac{\sum A}{n-1} \]

NDC for Intra-Team Architecture Sharing with Focus on Architects

<table>
<thead>
<tr>
<th>Team A</th>
<th>Team B</th>
<th>Team C</th>
<th>Team D</th>
<th>Team E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution Architect (SA)</td>
<td>-</td>
<td>0,91</td>
<td>1,0</td>
<td>1,0</td>
</tr>
<tr>
<td>Business Process Architect (BPA)</td>
<td>-</td>
<td>0,09</td>
<td>0,64</td>
<td>0,2</td>
</tr>
</tbody>
</table>

NDC for Inter-Team Architecture Sharing (Top 10)

<table>
<thead>
<tr>
<th>Rank</th>
<th>Role</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CPO_A</td>
<td>1,0</td>
</tr>
<tr>
<td>2</td>
<td>SM_D</td>
<td>0,95</td>
</tr>
<tr>
<td>3</td>
<td>PO_C</td>
<td>0,93</td>
</tr>
<tr>
<td>4</td>
<td>SA_E</td>
<td>0,92</td>
</tr>
<tr>
<td>5</td>
<td>SA_D</td>
<td>0,90</td>
</tr>
<tr>
<td>6</td>
<td>TM_A</td>
<td>0,90</td>
</tr>
<tr>
<td>7</td>
<td>BPA_E</td>
<td>0,89</td>
</tr>
<tr>
<td>8</td>
<td>Dev_F</td>
<td>0,87</td>
</tr>
<tr>
<td>9</td>
<td>PRE_E</td>
<td>0,84</td>
</tr>
<tr>
<td>10</td>
<td>PO_D</td>
<td>0,82</td>
</tr>
</tbody>
</table>

NDC for Inter-Team Architecture Sharing with Focus on Architects

<table>
<thead>
<tr>
<th>Role</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA_B(intern)</td>
<td>0,41</td>
</tr>
<tr>
<td>SA_B(extern)</td>
<td>0,28</td>
</tr>
<tr>
<td>SA_C_B(extern)</td>
<td>0,36</td>
</tr>
<tr>
<td>SA_B(extern)</td>
<td>0,31</td>
</tr>
<tr>
<td>SA_C</td>
<td>0,72</td>
</tr>
<tr>
<td>SA_D</td>
<td>0,90</td>
</tr>
<tr>
<td>SA_E</td>
<td>0,92</td>
</tr>
</tbody>
</table>

System architecture is a role collaboration
**Measure 2 – Network Density**

**Definition of Network Density**

The proportion of direct ties in a network relative to the total number possible.

**Formula for Network Density (ND)**

\[
ND = \frac{\sum A}{n(n-1)/2}
\]

**ND for Intra-Team Architecture Sharing**

<table>
<thead>
<tr>
<th>Team A</th>
<th>Team B</th>
<th>Team C</th>
<th>Team D</th>
<th>Team E</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.44</td>
<td>0.36</td>
<td>0.80</td>
<td>0.35</td>
<td>0.96</td>
</tr>
</tbody>
</table>

**ND for Inter-Team Architecture Sharing**

<table>
<thead>
<tr>
<th></th>
<th>Team A</th>
<th>Team B</th>
<th>Team C</th>
<th>Team D</th>
<th>Team E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team A</td>
<td>0.24</td>
<td></td>
<td>0.31</td>
<td>0.33</td>
<td>0.60</td>
</tr>
<tr>
<td>Team B</td>
<td>0.24</td>
<td>0.22</td>
<td>0.19</td>
<td>0.66</td>
<td></td>
</tr>
<tr>
<td>Team C</td>
<td>0.31</td>
<td>0.22</td>
<td>0.24</td>
<td>0.39</td>
<td></td>
</tr>
<tr>
<td>Team D</td>
<td>0.33</td>
<td>0.19</td>
<td>0.24</td>
<td>0.69</td>
<td></td>
</tr>
<tr>
<td>Team E</td>
<td>0.60</td>
<td>0.66</td>
<td>0.39</td>
<td>0.69</td>
<td></td>
</tr>
</tbody>
</table>

System architecture is also a cross-team collaboration
Agile teams and architects prefer decentralized communication for the exchange of architecture-related information.
Outline

Motivation

Foundations

Case Study

Social Network Analysis

Lessons Learned
Lessons Learned

1. System architecture is a role collaboration

2. Agile architects

3. The exchange of architecture-related information within the squads is decentralized

4. Internal architects play a double role in the team

5. Solution architects play the most central role in inter- and intra-team architecture sharing
QUESTIONS?

THANK YOU!

Please take part in our questionnaire on scaling agile frameworks:
https://tinyurl.com/agilescalingframeworks
Ömer Uludağ (M.Sc.)
Research Assistant

Technical University of Munich
Departments of Informatics
Chair for Software Engineering for Business Information Systems

Boltzmannstraße 3
85748 Garching bei München

Tel +49.89.289.17141
Fax +49.89.289.17136

oemer.uludag@tum.de
www.matthes.in.tum.de