How to Manage a Network of Software Architects within your Company

Frances Paulisch and Rüdiger Kreuter
Corporate Technology, Munich, Germany
Siemens Qualification Program (Curriculum) for Architects

Goals:

- Key positions in critical software-related projects are filled with highly qualified persons
- Help ensure that they are empowered, help remove blind spots (avoid the technical focus only)
- Improve the ability to meet targets in software-related projects and master uncertainty
- Improve the failure culture
- Foster networking so that beneficial to the individual as well as to their organization
- Increase the common vision of responsibility and collaboration of key roles across company
- Motivate employees and high-potentials
- Acquisition of external talents, keep high-potentials

- Siemens has established since 2007 an architecture-oriented qualification program
- Original focus on software architects
- Meanwhile we have also extended to system architects and test architects
- Over 400 persons have successfully completed one of the programs.
(Core) Learning Programs driven by the Siemens Software Initiative (SWI)

R&D Manager Workshop, 2012
Transfer into eLearning

Senior Software Architect (SSWA), 2007
Software Architect (SWA), 2009
System Architect, 2012
Test Architect, 2016 (NEW)
Requirements Engineering for PM/PLM, 2010

System Architect
Software Architect
Test Architect

Senior Certified
Certified
Functional Trainings
Senior Software Architect
Software Architect Spider Diagram – Five major topic areas

- Business Understanding
  - Business models
  - Standards, Regulations, Legal issues
  - Lifecycle issues
  - Domain knowledge
  - Customer & System requirements
  - Software requirements
  - Quality requirements

- Architecture and Development
  - Product Line Engineering
  - Architecture & design
  - Coding & Technology
  - Lean & Agile
  - Governance & Guidance
  - Basic
  - Advanced
  - Expert

- Leadership
  - Social skills
  - Project management

- Testing and Quality
  - Test methods
  - Test strategy

- Social Skills and Leadership

SWA: Software Architect
SSWA: Senior Software Architect

Basic: Knows about
Advanced: Drives, coaches and is able to improve in own domain & scope
Expert: Unrestricted

Unrestricted © Siemens AG 2016
### Guiding Principles

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Architecture as well as the continuous governance of it is the key throughout the whole lifecycle as well as across releases.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Build on existing basis where feasible (from technical and business perspective) and be able to recognize when such reuse is not suitable.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Avoid unnecessary technological platform development and use technical standards and products available on the market.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>The product (lifecycle) manager in product / system business and the project manager in project / solution business are and must act as the owner of the main requirements and quality characteristics.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>Pay particular attention to specifying, testing, and realizing non-functional requirements (NFRs), often overlooked but are extremely important.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>Be prepared and able to handle changing requirements, but be aware about the risk of late changes.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>Synchronize and balance well across the technical disciplines (software, mechanics, electronics, etc.) and have a close synchronization with business aspects.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>Work together truly as a team, avoid &quot;silo&quot; thinking, be willing and able to speak and understand the other roles and disciplines.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.</td>
<td>Work iteratively and test-driven, foster defect prevention from the beginning, and strive to identify and resolve technical and business risks early. Getting real and early feedback, both from customer and from realization team, is essential.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td>Structure the system to avoid unnecessary complexity, and to actively enable and support multi-site development.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.</td>
<td>Strive for transparency and base decisions on clear business / technical (not political) reasons.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.</td>
<td>Do not underestimate the importance of soft skills, these can be particularly important for convincing and motivating.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Key parts of the Qualification Program

- Setting standards for software development
- Assuring practical experience for critical projects
- Fostering best practice sharing and experience exchange

Qualification (courses and on-the-job training)

Certification

Software architects’ network
Networking to foster best practice sharing and experience exchange

- Provide a forum that is of real benefit to the community of architects (must want to attend and contribute)
- Make known who has what specific expertise
- Exchange best practices but also speak openly about failures
- Balanced combination of “in person” meetings and online collaboration (intranet, online meetings, webcasts,....)
- Both formal (annual workshop) and informal (brown-bag) in person meetings
- Continuously encourage use of the architects wiki as the central online information source e.g. doing joint “homeworks” of the training there. (Closed and Open Wikis)
- Actively involve architects in cross-organizational activities e.g. “hackathons”
Dimensions of Networking

- Within a class and across classes e.g. within software architect program
- Across the various programs e.g. software architect / system architect
- With other non-architecture roles e.g. architect / product manager / project manager
- Across business / process / technology view
- With “key experts”
- Within a geographic area e.g. networking workshops in Germany (all architects)
- Within a business unit e.g. establishing a dedicated architecture group
- “mentoring” e.g. of SWAs by SSWAs
- Across company (across domains) e.g. take advantage that software is a cross-cutting topic
Feedback from Architects Regarding the Qualification Program

- Due to level of trust and personal connection, persons are more willing to ask colleagues for advice.
- The programs help when discussing tradeoffs because there is a common understanding established in the training.
- Strength in numbers: the common understanding among the architect community helps strengthen the voice of the individual architects.
- The SWA and SSWA foster good networking amongst architect community and delivered value to my organization.
- There are several major benefits of the program - a common Siemens approach to architecture, the networking aspect, applying the learnings directly in the project, learning from colleagues in other business units.
- Software Architecture is a part of craftsmanship which must be learnt and practiced in a systematic manner. These learning programs provide a structured and a holistic view, accelerating the whole learning process.
- Besides all topics which have been covered and discussed during the workshops and which improved my daily work in many ways, the network of SWAs proved to be very helpful during the past months.
Summary / Lessons Learned

- Have architecture-based foundation that is broadly applicable and enables a consistent understanding e.g. between architect / product manager

- Right balance of centrally-driven and grass roots is key

- Don’t “overcontrol!” – carefully motivate, support, encourage – this may seem slower, but is only way to become self-sustaining

- Motivate through positive visibility

- Establish an open and constructive working environment and explicitly focus also on social skills – very important for successful networking

- Enable working in atmosphere of trust (start with closed wikis among architects from one class)

- Take explicit advantage of the alumni network e.g. when architects move to other positions in the company

- Being able to immediately benefit from the program – and similarly to immediately benefit from the networking – is a key advantage to get buy-in both from architects and their organizations

- “Bonding” is most effective when one jointly experiences significant collaboration – actively seek out and arrange such situations (e.g. joint homeworks, architectural reviews, hackathons,...)