Architectural guidance and governance in industrial software ecosystems

Klaus-Benedikt Schultis
Christoph Elsner
Daniel Lohmann

European Workshop on software ecosystems 2012

Siemens Corporate Technology
Software Architecture Development
Motivation

Observed Trend: Moving towards software ecosystems

- expand technological base outside the organizational boundaries
- involve external business units
- initially not designed as software ecosystem
- architecture, collaboration and business challenges
- software architecture as central pillar

Architectural guidance and governance to counter architectural challenges

- **elicit** SECO specific requirements
- **determine** appropriate tools and methods

Architectural guidance and governance for minimizing architectural erosion to keep a healthy and flexible software ecosystem.
Industrial software ecosystems

A software ecosystem consists of

- a set of co-acting business units, together with the relationships among them
- a shared market for software and services
- a common technological base comprising a reference architecture, core assets and/or standards

and operates through the exchange of

- information, resources and artifacts. (adapted from [1])

An industrial software ecosystem focuses on

- a key company maintaining ultra-large software projects involving mainly internal business units
- where the organizational structure moves
- from hierarchies towards decentralized topologies.

Industrial Software Ecosystems (ISECO) focus on the interdependency of multiple interacting internal business units.
Elicit guidance and governance requirements (1/3)

Model industrial software ecosystems

Model an industrial software ecosystem including

- the participating business units,
- the relationships among them
- and their technical characteristics which are crucial for
  - identifying and refining emerging architectural challenges
  - and for determining architectural guidance and governance approaches to address those challenges.

Case Studies:

We are modeling two industrial software ecosystems from the industry and healthcare sectors at Siemens based on

- interviews with involved, well experienced software architects
- and literature and document reviews.

General observations:

- often comprise only a relative small number of business units
- often posses a relative equable distribution of influence, power and control between some internal business units

Model industrial software ecosystems as enabler to identify and refine architectural challenges and to determine counteractive approaches.
Identify, refine and prioritize abstract architectural challenges based on the modeled industrial software ecosystems.

Birds’ eye-view on the observed architectural challenges

- **Evolution**
  - use of interface as intended
  - compliance to interface access rules
  - facilitate commodifications
  - ensure coevolution between platform and applications

- **Interface stability**
  - enable predictable and transparent interface evolution
  - provide significant time to developers to adjust their app.
  - ensure backward compatibility where negotiated

- **Further challenges**
  - architecture openness, common quality models, cooperative development models, scoping/variability mgmt., …

**Case Studies:**

We are identifying, refining and prioritizing abstract challenges based on the two modeled industrial software ecosystems at Siemens in order to **counter them**.

**General observations:**

- identified architectural challenges are often **similar** to well known software ecosystem specific challenges
- **evolution** and **interface stability** as key challenges
- equable distribution of power demands **consensus rather than normative approaches** to address challenges
Answer architectural guidance and governance key questions to counter architectural challenges – arising in an industrial software ecosystem context.
A method for determining appropriate guidance and governance tools and methods

Elicit guidance and governance requirements

Model industrial software ecosystems
Identify participants and their characteristics

Identify, refine and prioritize architectural challenges
Evolution, Interface stability,…

Answer guidance and governance key questions

How to guide and govern developers?
How to access guidelines?
What should be guided and governed?

Determine appropriate tools and methods

Identify, combine, adapt and extend existing guidance and governance approaches according to the elicited requirements.

- **Handle general issues** like mindset, transparency of ROI, pragmatism, knowledge within a single person
- **Iteratively elicit and refine requirements** on guidance and governance approaches to counter architectural challenges.
- **Identify, combine, adapt and extend existing approaches** to counter **high prioritized** pain points according to the elicited requirements.
- **Case studies**: We are applying the method on two industrial software ecosystems at Siemens with the input of involved, well experienced software architects.

Currently we are **eliciting requirements** with the goal to provide an ISECO-Metamodel and a method to determine customized consensus-based G&G approaches in an ISECO.
Do ISECOs and their observed characteristics appear in other organizations? (e.g. at SAP, Phillips or in further ultra-large organizations)

Are our planned contributions relevant? (Metamodel for ISECOs, a method to provide customized G&G approaches in ISECO, consensus-based G&G)

Does a validation with two to three ISECOs at Siemens delivers satisfying and transferable results?
References


Thank you for your attention!

Klaus-Benedikt Schultis
Ph.D. student

Corporate Technology,
Software Architecture Development

Wladimirstraße 3, 91058 Erlangen, Germany

Fax: +49 (9131) 7-33190
Mobile: +49 (176) 30943982

E-mail: klaus-benedikt.schultis.ext@siemens.com