Gathering Empirical Evidence and Building a Business Case for Software Reference Architectures in Industry

Silverio Martínez-Fernández

Software Engineering

Engineering discipline concerned with all aspects of software production.

Many systematic processes, methods, techniques...
Software Architecture: the set of structures needed to reason about the system

Software Reference Architecture (SRA):
a software architecture where the structures and respective elements and relations provide templates for concrete architectures in a particular domain or in a family of software systems
Examples of Software Reference Architectures

Technological domain

For many organizations

Business domain

For a single organization

100+ Software reference architectures in scientific literature (e.g., service-oriented systems, automotive software)
75+ Software reference architecting practices in scientific literature (e.g., design methods, conformance checking)
Empirical Software Engineering

How to **evaluate software engineering methods or techniques** against each other?

**Experimentation**: research using empirical evidence, i.e., gaining knowledge by means of direct and indirect observation or experience.
Should I adopt a software reference architecture in my company?

To make informed decisions, we need empirical evidence!

An industry-academia collaboration

What? Studying SRAs from everis | How? Empirical studies drived by GESSI

RQ 1: How can an organization get corporate evidence that is useful for the SRA engineering?

RQ 2: Is it worth for an organization to invest on the adoption of an SRA?
The Context at everis
An everis’ Client Software Reference Architecture

Source: http://canigo.ctti.gencat.cat/canigo/framework/
Difficulties working with software reference architectures:

- Shortage of experience reports
- How are they being designed and used?

At everis, capturing the architectural knowledge of years of work in a congruent vision → helping everis’ employees

RQ 1: How can an organization get corporate evidence that is useful for the SRA engineering?
Difficulties justifying the investment of software reference architectures to upper management:

- **Shortage of economic models** to precisely evaluate the costs and benefits

At everis, preparing materials for clients to show that adopting a software reference architecture is a good idea and it makes financial sense

RQ 2: Is it worth for an organization to invest on the adoption of an SRA?
Empirical Studies and Action Research Cycles to Answer Research Questions

1. Diagnosis of a problem
2. Options to solve the problem
3. Selection of options and execution
4. Analysis of the results
RQ1: How can an organization get corporate evidence that is useful for the SRA engineering?

“Architectural thinking puts the 'engineering' in software engineering”

David Garlan
RQ1 Cycles

Formative Stage

- Shaping Guidelines
- Cycles 1 and 2
- Mainly at everis
- Packaging of the guidelines

Summative Stage:

- Validating Guidelines
- Cycles 3 and 4
- Mainly at other companies
Action Research about RQ1: 1st Cycle

Study: Informal meetings

Population: SRA stakeholders

Respondents: 3 everis’ managers and 4 GESSI researchers

Instruments: archival data, experience, scientific literature
### Description of the SRA Aspect

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Description of the SRA Aspect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Overview and classification of an SRA</td>
</tr>
<tr>
<td>2</td>
<td>Requirements and quality attributes analysis</td>
</tr>
<tr>
<td>3</td>
<td>Architectural knowledge and decisions</td>
</tr>
<tr>
<td>4</td>
<td>Supportive technologies</td>
</tr>
<tr>
<td>5</td>
<td>Business qualities and architecture competence</td>
</tr>
</tbody>
</table>
Action Research about RQ1: 2nd Cycle

Study: Exploratory case study

Population: Stakeholders from real SRA projects at everis

Respondents: 28 stakeholders (software architecture, architecture developers, and application builders) from 9 projects

Instruments: Face-to-face interviews, online questionnaires, archival documentation
RQ1: 2nd Cycle Results

Some results in *everis*:

**Ben-X → benefits of using SRAs for clients**

**Ven-X → benefits of designing SRAs for vendor**

**Dra-X → drawbacks of using SRAs**

- Reduced development costs (Ben-A) 7 8 8 82%
- Reduced maintenance costs (Ben-B) 5 7 10 78%
- Easier development and increased productivity (Ben-C) 7 3 5 68%
- Incorporation of latest technologies (Ben-D) 3 5 7 53%
- Applications more aligned with business needs (Ben-E) 4 4 5 43%
- Homogenization or standardization (Ben-F) 6 2 1 33%
- Increased reliability of RAs software elements (Ben-G) 5 3 1 33%
- Others benefits (Ben-H) 2 2 1 25%

*Harvest experience for prospective RA projects (Ven-A) 7 5 67%*

*Reuse of architectural knowledge (Ven-B) 4 2 8 33%*

*Gain of reputation (Ven-C) 8 17%*

*Risk reduction (Ven-D) 8 17%*

*Shared architectural mindset (Ven-E) 8 11%*

*It turns tacit knowledge into explicit knowledge (Ven-F) 8 6%*

*Additional learning curve for using the RA (Dra-A) 5 4 9 63%*

*Limited innovation by giving guidelines (Dra-B) 5 2 1 28%*

*Applications’ dependency over the RA (Dra-C) 4 2 1 22%*

*Complexity (Dra-D) 4 2 1 14%*

*None (Dra-E) 8 14%*

*Wrong decisions about the technologies (Dra-F) 2 7%*

*Other drawbacks (Dra-G) 2 2 2 21%*
Can we generalize the results from the previous empirical studies?

Can these materials be articulated into guidelines for many companies?

Corporate Reference Model

- **IT consulting firms**, e.g., Accenture and Capgemini

Platform-specific SRAs

- **Public administrations**, e.g., Germany, The Netherlands, Spain

Industry-specific SRAs

- **Space**, e.g., NASA, ESA
- **Banking**, e.g., Credit Suisse, Cajamar
- **Embedded systems**, e.g., Océ, Volvo, BMW, AUTOSAR
Action Research about RQ1: Guidelines (1/3)

**Inputs**
- Interviews key stakeholders
- Archival data (e.g., documentation, metrics)
- Observation

**Guidelines**
- Context of SRAs in Industry
- Set of Relevant Aspects of SRAs
- Template Survey

**Outputs**
- Evidence of SRAs
Action Research about RQ1: Guidelines (2/3)

Materials? Set of Relevant Aspects of SRAs

List of relevant aspects
Explanations why they may be important

How? Focus group

Who? Software architects
Materials? Template Questions

To get the context (personal data, project, and experience) → generic questions

For each relevant aspect → a set of questions personalized to each stakeholder as well as previous answers

How? A survey or a case study

Who? All stakeholders
Action Research about RQ1: 3rd Cycle

Study: a survey of a world-wide accepted SRA

Population: AUTOSAR practitioners

Respondents: 51 practitioners from world-wide and several partners

Instruments: online questionnaire based on the template questions of the guidelines
**Action Research about RQ1: 3rd Cycle Results**

### Benefits of using AUTOSAR?

- ** Complexity: 33 (65%) **
- ** Initial investment: 30 (59%) **
- ** Learning curve: 26 (51%) **
- ** Term confusion: 21 (41%) **
- ** Too abstract: 18 (35%) **
- ** Dependency in AUTOSAR: 15 (29%) **
- ** Inefficient instantiation: 11 (22%) **
- ** Bad documentation: 10 (20%) **
- ** Too specific or limiting: 8 (16%) **
- ** Bad quality: 5 (10%) **

### Drawbacks and risks of using AUTOSAR?

- ** Complexity: 33 (65%) **
- ** Initial investment: 30 (59%) **
- ** Learning curve: 26 (51%) **
- ** Term confusion: 21 (41%) **
- ** Too abstract: 18 (35%) **
- ** Dependency in AUTOSAR: 15 (29%) **
- ** Inefficient instantiation: 11 (22%) **
- ** Bad documentation: 10 (20%) **
- ** Too specific or limiting: 8 (16%) **
- ** Bad quality: 5 (10%) **
- ** Other: 2 (2%) **
- ** None: 2 (2%) **

---

**Guidelines applied successfully!**

Highly accessed! Statistics in one year (published in May 2015)

1100+ reads in Research Gate

150+ downloads in ACM DL
Action Research about RQ1: 4th Cycle

Study: a meta-analysis

Population: empirical studies about SRAs benefits and drawbacks

Respondents/Sampling: 5 primary studies

Instruments: systematic review for selection, structured synthesis method for aggregation, and data analysis for studying the context
SRA effects present in different contexts:

• improved interoperability
• reduced development costs
• better communication
• higher learning curve

Effects without significant changes in their belief value or only present under certain contexts → looking for explanations, hypothesis
Aggregating evidence helps to:

**Increase the confidence** of the results of single empirical studies

Possibility to formulate **new theories**

Reduce the effort of people interested in a topic.
RQ 2: Is it worth for an organization to invest on the adoption of an SRA?

“Architects regularly make architectural design decisions but are usually unable to evaluate the economic impact of those decisions. [...] They need tools and techniques to make better decisions and to better justify those decision to their stakeholders.”

Rick Kazman
**RQ2 Cycles**

**Formative Stage:**
- Shaping Guidelines
- Cycles 1 and 2
- Mainly at everis
- Packaging of the guidelines

**Summative stage:**
- Validating Guidelines
- Cycle 3
- Mainly at other venues
Study: a survey

Participants: software architects and application builders

Respondents: 5 software architects, and 5 application builders

Instrument: online questionnaire
To calculate projects’ costs: effort metrics

To calculate benefits (e.g., reuse): software metrics

Software development is a low-validity environment → Need to move to a high-validity environment.

• Time tracking
• Continuous feedback
• Test-driven development
Action Research about RQ2: 2nd Cycle

Study: a case study

Participants: SRA projects at everis

Respondents/Sampling: a public administration in Spain

Instruments: archival documentation, effort metrics from JIRA, software quality metrics from Sonar
How Many Instantiations are necessary before savings pay off for the up-front investment?

Payback: 16.5 months
7 applications

The bigger applications, the greater the benefits

Better time-to-market: 589 hours development costs avoidance

ROI within 2 years

Three-years ROI of 42%
Action Research about RQ2: Guidelines (1/3)

Inputs

- Interviews key stakeholders
- Archival data (e.g., documentation, metrics)
- Observation

Guidelines

- Context of SRAs in Industry
- Checklist of value-driven data
- Economic Model (REARM)

Outputs

- Business Case of SRAs
Action Research about RQ2: Guidelines (2/3)

Materials? Checklist of value-driven data

Sources and metrics to extract the costs

Sources and metrics to extract the benefits

How? A survey

Who? Software architects for SRA projects, application builders for concrete architecture projects.
Action Research about RQ2: Guidelines (3/3)

Materials: REARM economic model

10 basic parameters

6 cost-benefit factors

Business case parameters

Scenarios to calculate the ROI

How? A case study

Who? The case under evaluation

ROI = \frac{Benefits - Costs}{Costs} \quad (D.1)

Benefits = \sum_{i=1}^{n} (DCA_i + SCA_i) \quad (D.2)

Costs = CSWD + CSWS + \sum_{i=1}^{n} LIDC_i + \sum_{j=1}^{m} CSWE_j \quad (D.3)
Action Research about RQ2: 3rd Cycle

*Study*: Presentations and Workshops

*Population*: Practitioners and researchers

*Respondents*: an everis’ client organization, audience of ICSR, and a research group

*Instrument*: Results communication
Action Research about RQ2: 3rd Cycle Results

Management of a client organization
- Prefer outputs in terms of effort
- Prefer scenario SRA vs. no SRA

Workshop at the ESE group of UFRJ
- Suggest to include qualitative terms
- Remark that cost-benefit factors of REARM represent effects caused by SRAs

Audience of ICSR (future work)
- Remark that different levels of reuse in applications may change the outputs of REARM → sensitivity analysis

10 citations (excluding self-citations) of REARM (Google Scholar). Some authors:
Closure and Contributions – What is new in this PhD?

Source: [http://matt.might.net/articles/phd-school-in-pictures/](http://matt.might.net/articles/phd-school-in-pictures/)
Contributions (1/2)

Relationships between empirical theory and software architecture theory
Falessi, Ali Babar, Cantone and Kruchten 2010 [27]
Empirical Research Methodology of this PhD Thesis

- **Software Reference Architecture (SRA) Theory**: build
  - Thesis’ Results:
    - SLR on SRA engineering
    - Review criteria for SRA
    - The benefits and drawbacks of SRAs (in everis, in AUTOSAR, and research synthesis)
    - The artifacts of SRAs in everis
    - Common data in SRA projects
    - REARM, a reuse-based model to build the business case for SRAs

- **Thesis’ Challenges**:
  - RQ 1: Improving SRA engineering from evidence
  - RQ 2: Calculating the ROI of SRA adoption

- **Empirical Software Engineering (ESE) Theory**
  - includes
  - improves
  - allows

- **Empirical assessment**:
  - Design and conduction of diverse empirical studies

- **Lessons Learned on our industry-academia collaboration**
  - produces
  - produces
Journals (1 Elsevier, 1 Springer)


Conferences (3 Full & 1 Short CORE-A)


5) S. Martínez-Fernández, J. Bisbal, and X. Franch, “QuPreSS: A Service-Oriented Framework for Predictive Services Quality Assessment,” in *KMO 2012*, 525-536
**Workshops**


3) S. Martínez-Fernández, X. Franch, and J. Bisbal, “Verifying predictive services’ quality with Mercury,” in *WASDeTT@ECSA 2013*.


**Doctoral Symposium**


**Ongoing Journals**

1) Extended version of the Case Study about the Benefits and Drawbacks of SRAs → ACM TOSEM

2) Systematic Mapping about SRAs → IST
Thank you!!

Silverio Martínez-Fernández
Advisors: Claudia Ayala and Xavier Franch