TU Delft Delft University of Technology

Software Improvement Group

Criteria for the Evaluation of Implemented Architectures

Eric Bouwers, Joost Visser, Arie van Deursen

2009/09/22

Arent Janszoon Ernststraat 595-H
NL-1082 LD Amsterdam
info@sig.nl
www.sig.nl

Why evaluate software architectures?

Software Improvement Group

'Architectures allow or preclude nearly all of the system's quality attributes'

-- from `Evaluating Software Architectures`
by P. Clements, R. Kazman and M. Klein

Why evaluate implemented architectures?

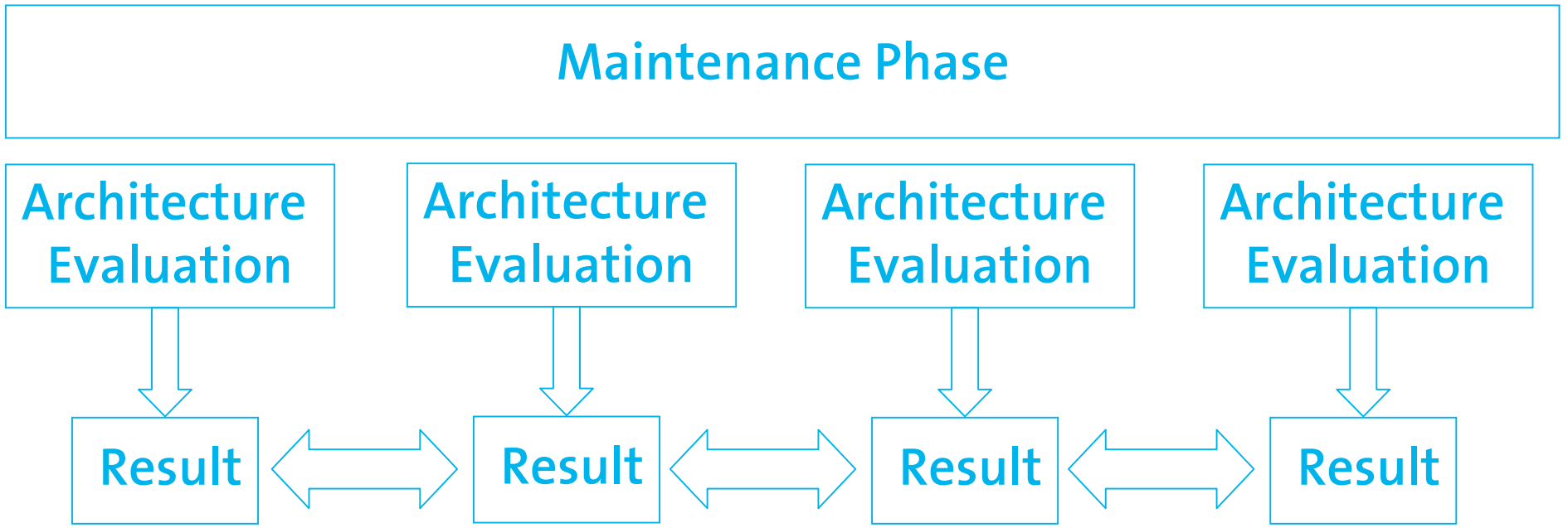
Software Improvement Group

Why evaluate implemented architectures?

Software Improvement Group

Why do we want criteria for the evaluation?

Software Improvement Group



Literature reviews

Software Improvement Group

A survey on software architecture analysis methods.

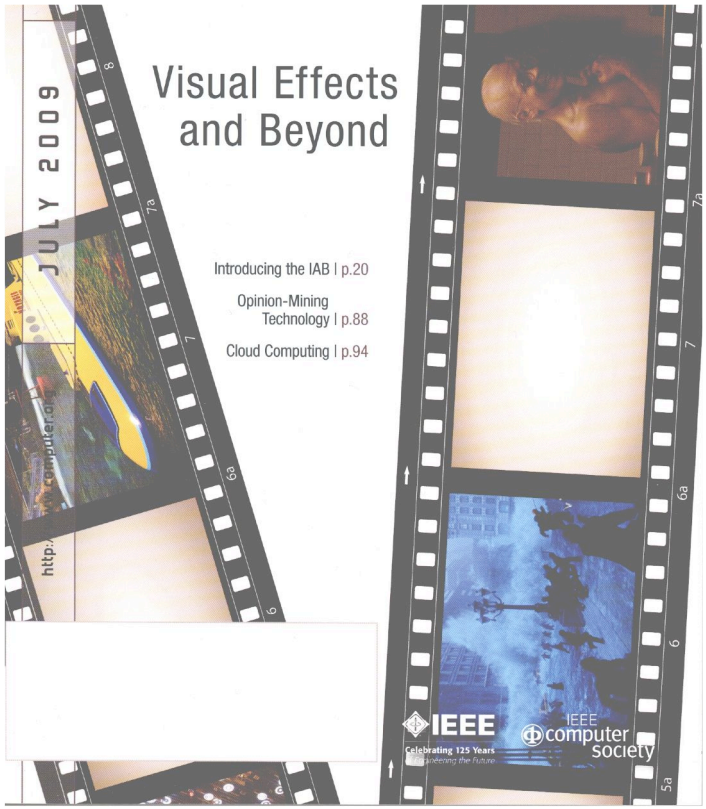
L. Dobrica and E. Niemelä.

A framework for classifying and comparing software architecture evaluation methods.

M. Babar, L. Zhu, and D. R. Jeffery

IEEE Computer, July 2009

Software Improvement Group



'Technology transfer remains a major challenge. There is limited out of the box process and tool support for organizations that want to start reviews.'

'Software Architecture Review: The state of Practice'
by M. Babar, I. Gorton

Software Risk Assessment (SRA)

Software Improvement Group

A one-time investigation into software maintainability and related business risks

Criteria for the Evaluation of Implemented Architectures

Architectural properties

Software Improvement Group

High Level Design:

GUI

Business Logic

Data Access

Modularization:

Saving | Paying | Stocks

Separation of Concerns:

```
if(hasPayed() && !isAdmin()){
   showThankYou();
}
```

Business logic/Security

Research questions

Software Improvement Group

Q1: Which system attributes do experts take into account when evaluating architectural system properties?

Q2: How do these system attributes influence the architectural system properties?

Empirical study resources

Software Improvement Group

A selection of our clients

Software Improvement Group

Financials

Public

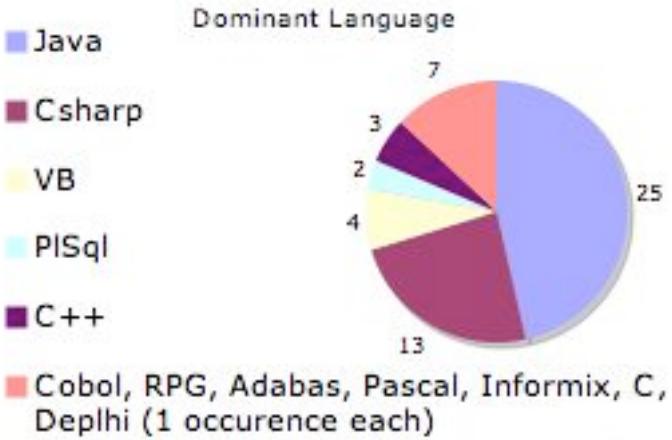
Logistics

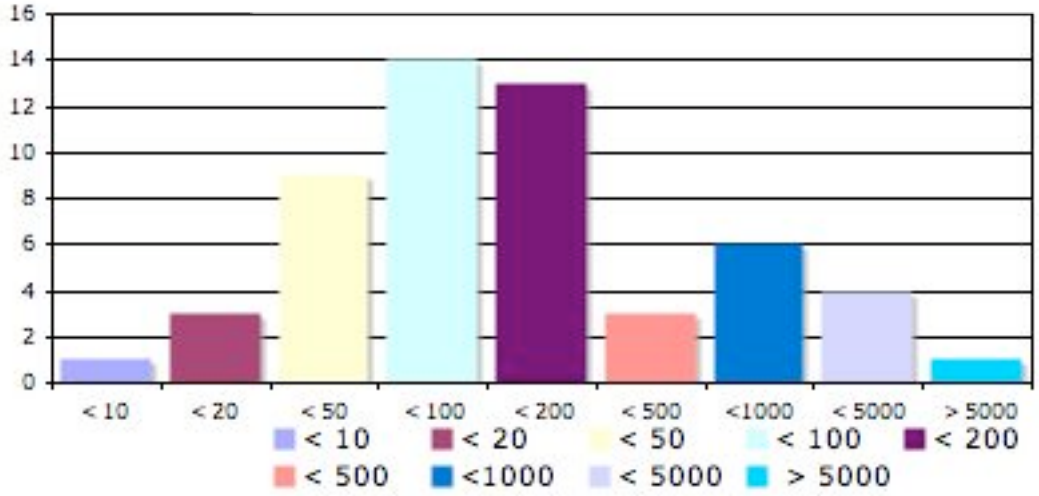
IT

Other

Evaluated systems characteristics

Software Improvement Group





Empirical study design overview

Software Improvement Group

Experimental design (1/3)
Creation of the attribute list

Software Improvement Group

For each report:

1) Check table with property ratings for arguments
if this table does not exists
use specific paragraphs

2) For all arguments found
determine system attribute
if the attribute is not on the list
add system attribute to the list

Experimental design (2/3)
Create mapping from attributes to properties

Software Improvement Group

For each report:

1) Extract arguments per property

2) For all arguments found
determine system attribute
add one to count of attribute for property

Experimental design (3/3)
Validation of Attributes and Mapping

Software Improvement Group

Results of Empirical Study (1/3)

Software Improvement Group

15 System attributes

Software Improvement Group

- Abstraction
- Layering
- Logic in Database
- Module Inconsistency
- Module Size
- Source Grouping
- Technology Combination
- Textual Duplication
- Functional Duplication
- Libraries / Frameworks Usage
- Module Dependencies
- Module Functionality
- Relation Documentation and Implementation
- Technology Age
- Technology Usage

More details available in the paper

Software Improvement Group

Name	Description	Assessment Approach
Libraries / Frameworks	The usage of standard libraries and frameworks	Inspecting the list of imports and structure of the source- and build-files
Module Dependencies	The static dependencies (i.e. calls, includes) between modules	Inspecting the call-graph on module level, matching this against expected dependencies

Results of empirical study (2/3)

Software Improvement Group

Attributes used for each property

Software Improvement Group

	Module Dependencies	Module Functionality	Relation Doc. / Impl.	Functional Duplication	Technology Combination	Technology Age	.....	.....
High Level Design	7	4	2	2	5	13		
Modularization	11	32	3	6	1	0		
Separation of Concerns	6	13	0	18	0	0		

Attributes used for each property

Software Improvement Group

	Module Dependencies	Module Functionality	Relation Doc. / Impl.	Functional Duplication	Technology Combination	Technology Age	.....	.....
High Level Design			X		X	X		
Modularization	X	X	X					
Separation of Concerns				X				

Results of empirical study (3/3)

Software Improvement Group

Interview results

Software Improvement Group

	Module Dependencies	Module Functionality	Relation Doc. / Impl.	Functional Duplication	Technology Combination	Technology Age	.....	.....
High Level Design	E1		E2					
Modularization		E1, E2	E2					
Separation of Concerns				E1				

Threats to validity

Software Improvement Group

Representative data

Reliability of measurements

Generalization of the results

'What is in it for me?'

-- you

Our contributions

Software Improvement Group

- A description of an empirical study using over 40 reports
- The identification of 15 system attributes that have an impact on the maintainability of an architecture
- An analysis of the projection of the found system attributes onto three architectural system properties

Application of the results

Software Improvement Group

Software Improvement Group

Criteria for the Evaluation of Implemented Architectures

Future research opportunities

Software Improvement Group

Software Improvement Group

Criteria for the Evaluation of Implemented Architectures

Software Improvement Group

Criteria for the Evaluation of Implemented Architectures

Questions or comments?

Criteria for the Evaluation of Implemented Architectures

- A description of an empirical study using over 40 SRA reports
- The identification of 15 system attributes that have an impact on the maintainability of an implemented architecture
- An analysis of the projection of the found system attributes onto three architectural system properties

Feel free to contact me:

e.bouwers@sig.eu