Delft University of Technology

Software Improvement Group

Kwaliteitsmetrieken voor software architecturen

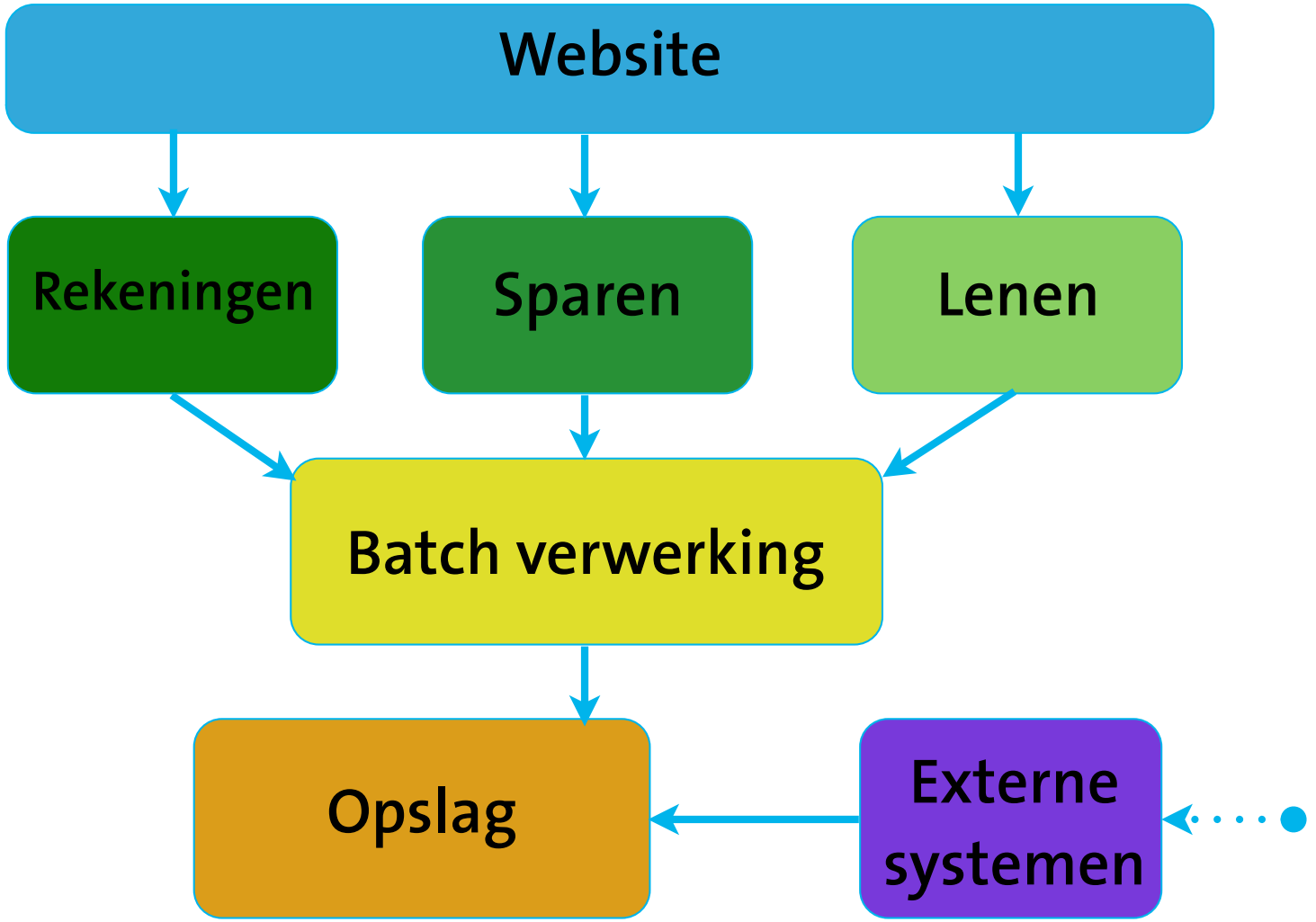
Eric Bouwers

Landelijk Architectuur Congres, 24 november 2010

Amstelplein 1
1096 HA Amsterdam
info@sig.eu
www.sig.eu

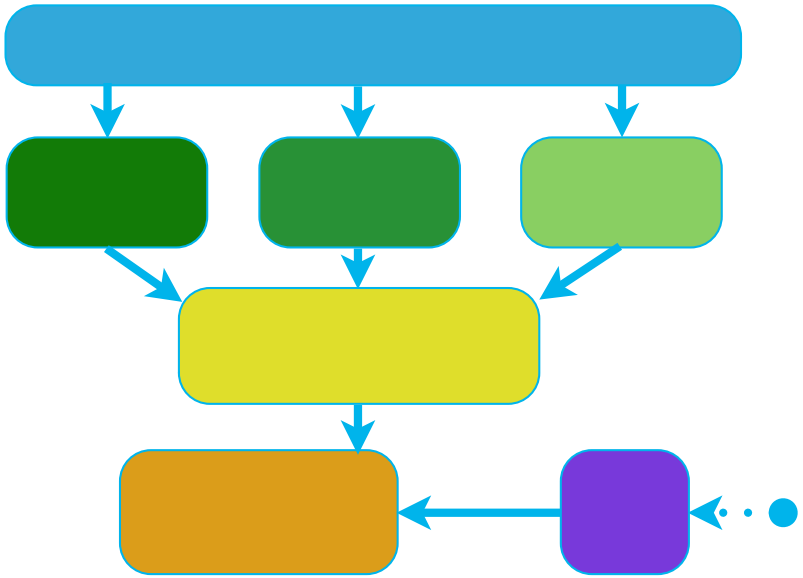
Ontworpen architectuur

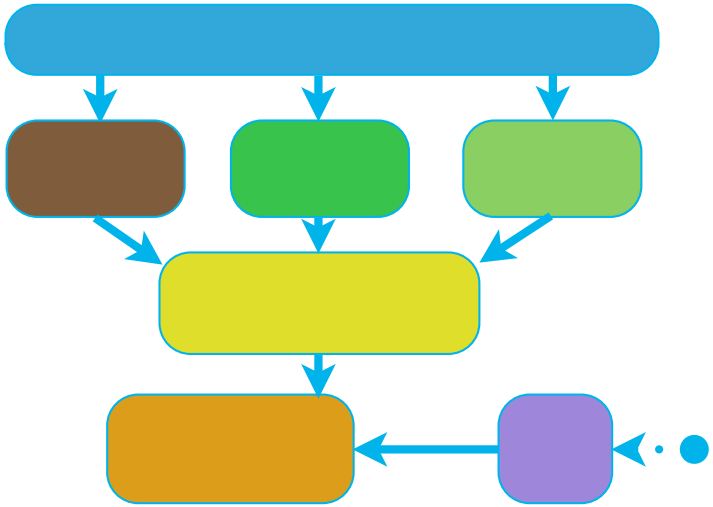
Software Improvement Group

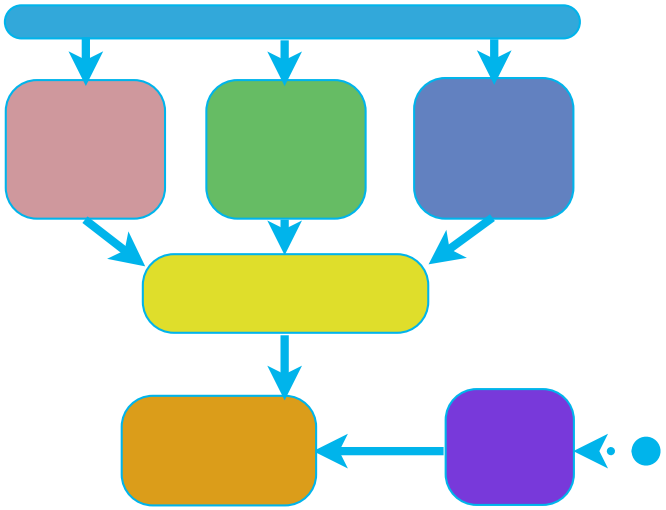


De ideale wereld

Software Improvement Group

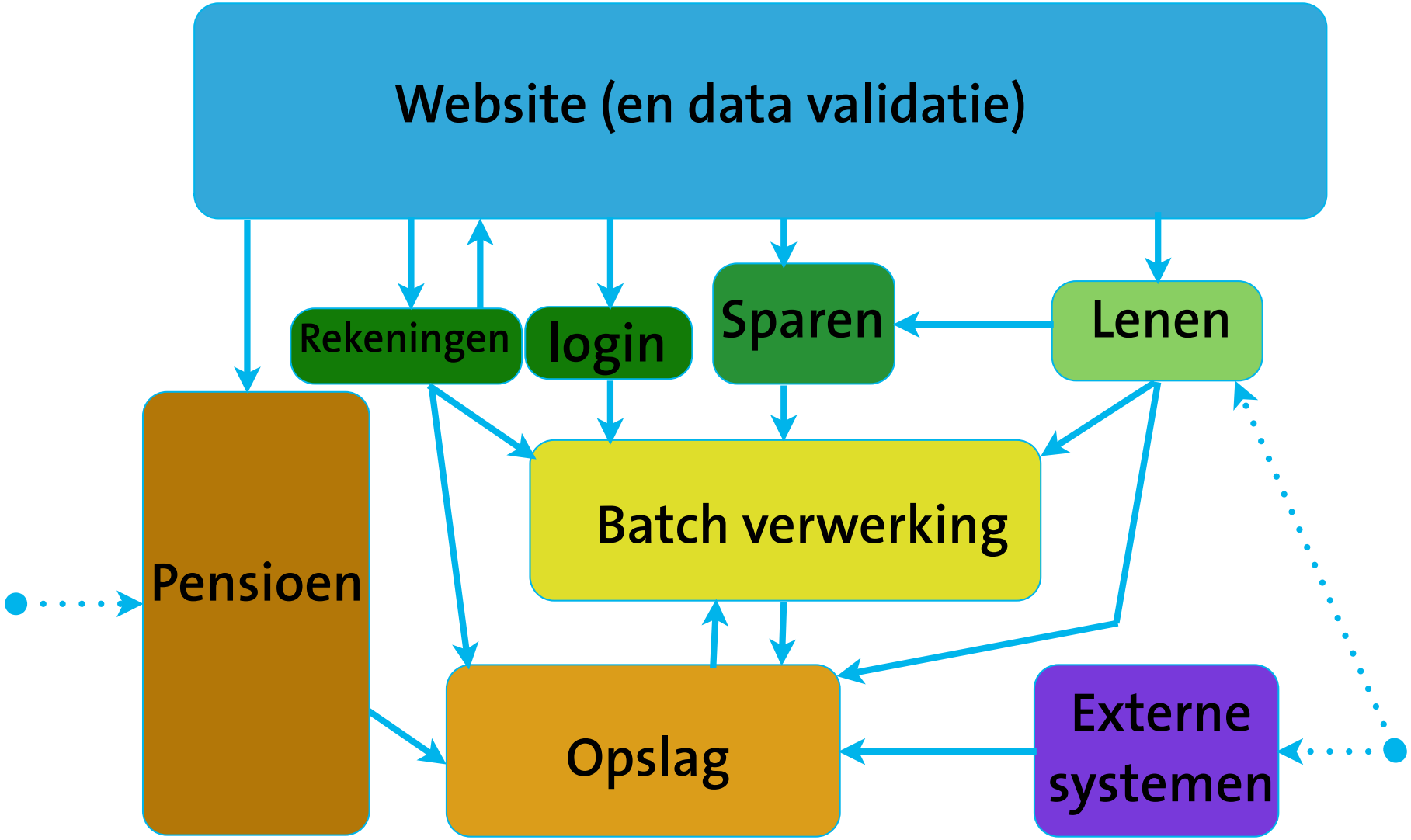






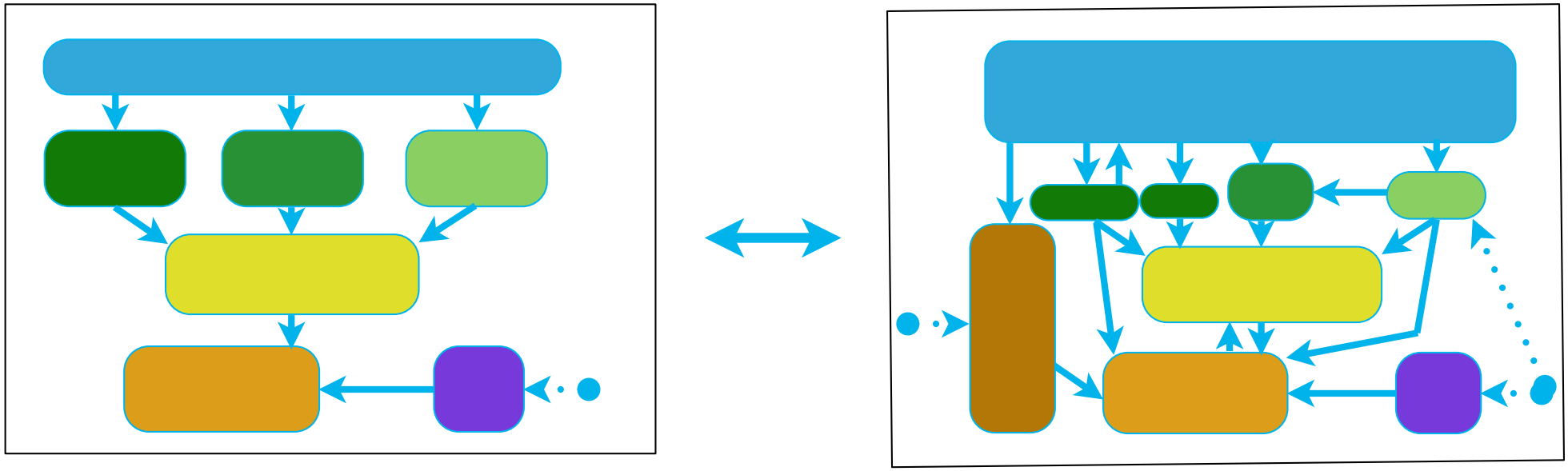
In werkelijkheid ...

Software Improvement Group



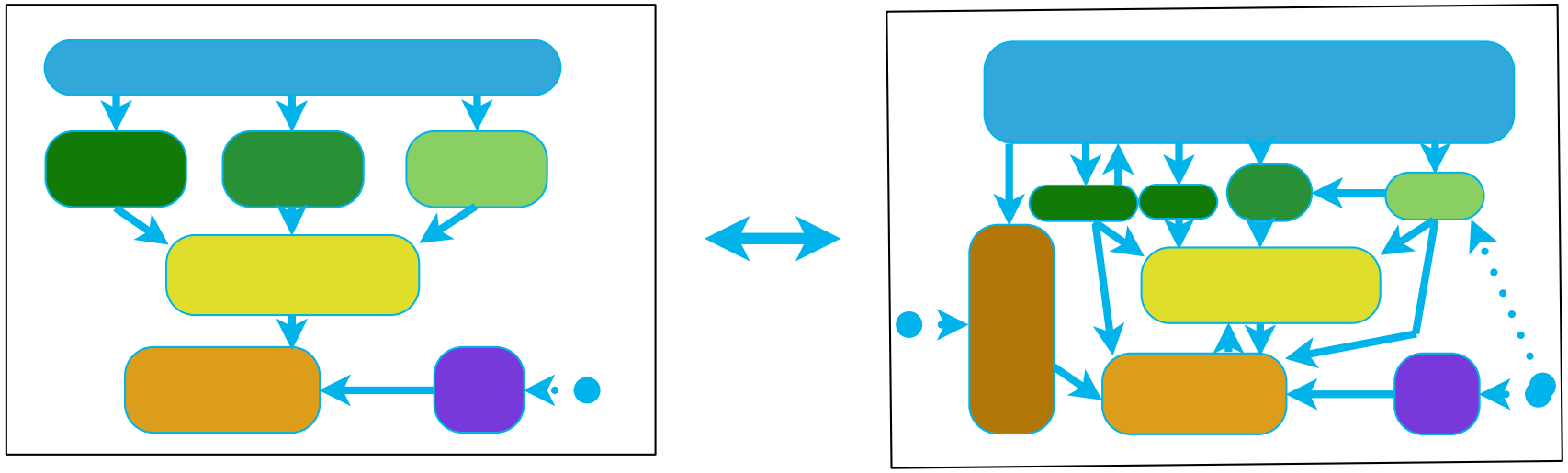
Consequencies

Software Improvement Group



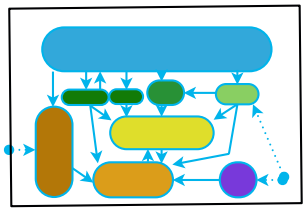
Oorzaken

Software Improvement Group



Preventie

Software Improvement Group



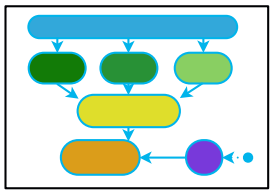
Check | Check | Check | Check | Check | Check | Check

Start project

Continu checken

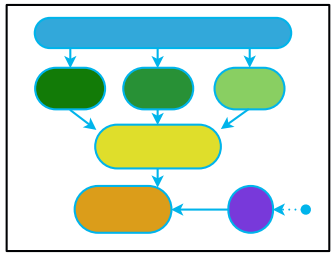
Go Live

Evaluatie | Evaluatie | Evaluatie | Evaluatie



Evaluatie
Ontworpen architectuur

Software Improvement Group



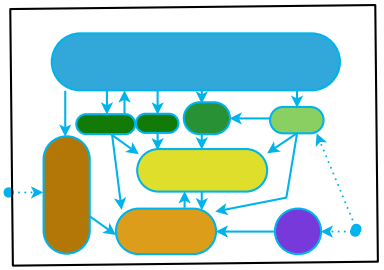
A survey on software architecture analysis methods,
L. Dobrica and E Niemelë, IEEE Transaction 2002

A framework for classifying and comparing software architecture evaluation methods,
M. Babar, L. Zhu and D. R. Jeffery, ASWEC 2004

ATAM, ARID, ALMA, ALPSM,
SAAM, SAAMER, SAAMCS,
SBAR, ESAAMI, SAEM, QASAR, ...

Evaluatie
Geïmplementeerde architectuur

Software Improvement Group



'There is limited out of the box process and tool support for organizations that want to start [architecture] reviews.'

'Software Architecture Review: The state of Practice' by M. Babar, I. Gorton, IEEE Computer July 2009

Attribuut identificatie

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

'Criteria for the Evaluation of Implemented Architectures'
by E. Bouwers, J.Visser and A. van Deursen, ICSM 2009

Operationeel maken

Software Improvement Group

http://www.sig.eu/en/liscia

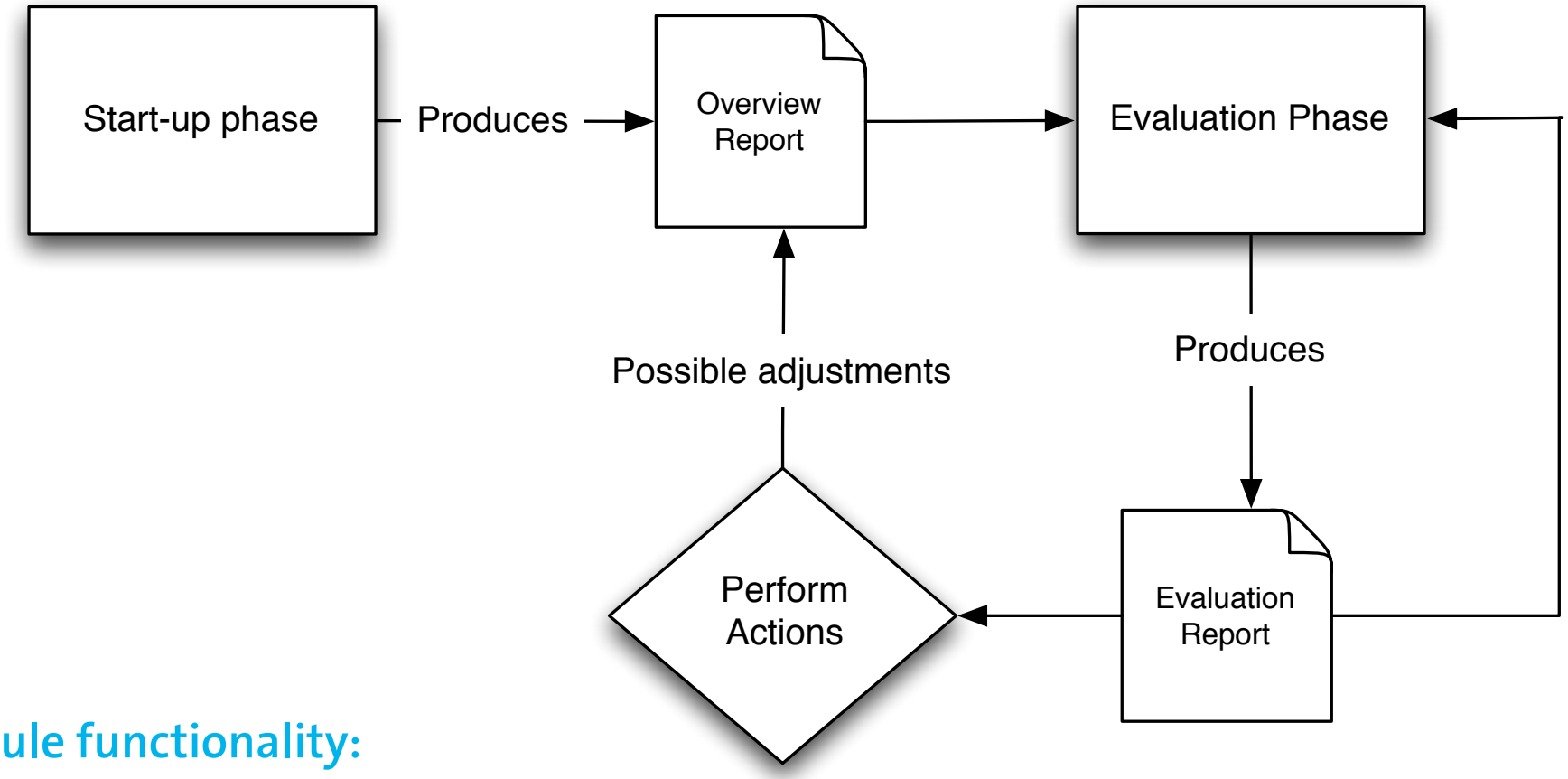
28 vragen

28 acties

5 onderwerpen

'A Lightweight Sanity Check for Implemented Architectures'
by E. Bouwers and A. van Deursen, IEEE Software July 2010

LiSCIA proces



Module functionality:

3.2.3: *Can the functionality of each module be described in a single sentence?*

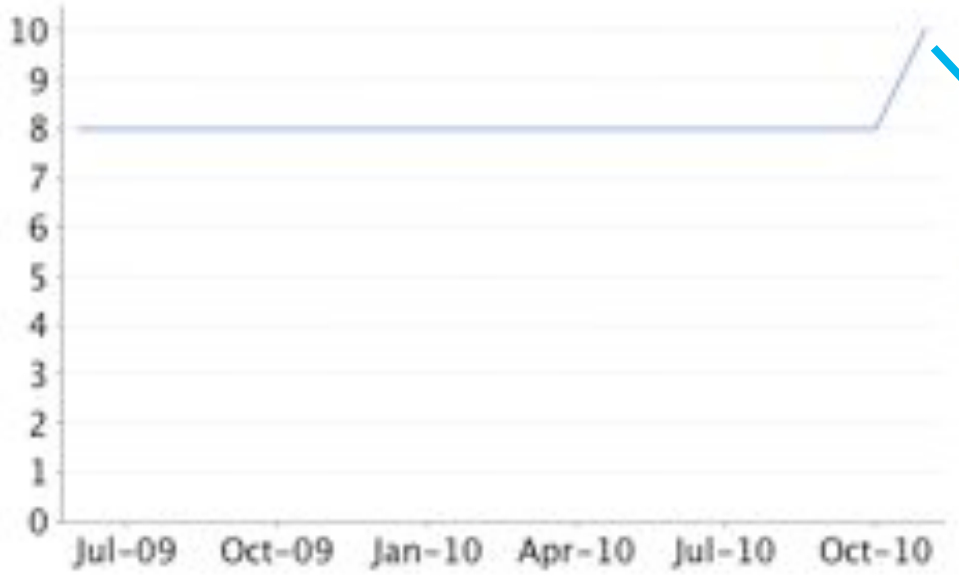
Module dependencies:

3.4.5: *5. Are there any new dependencies? Is this to be expected?*

Architectuur metrieken

Software Improvement Group

Aantal afhankelijkheden over tijd

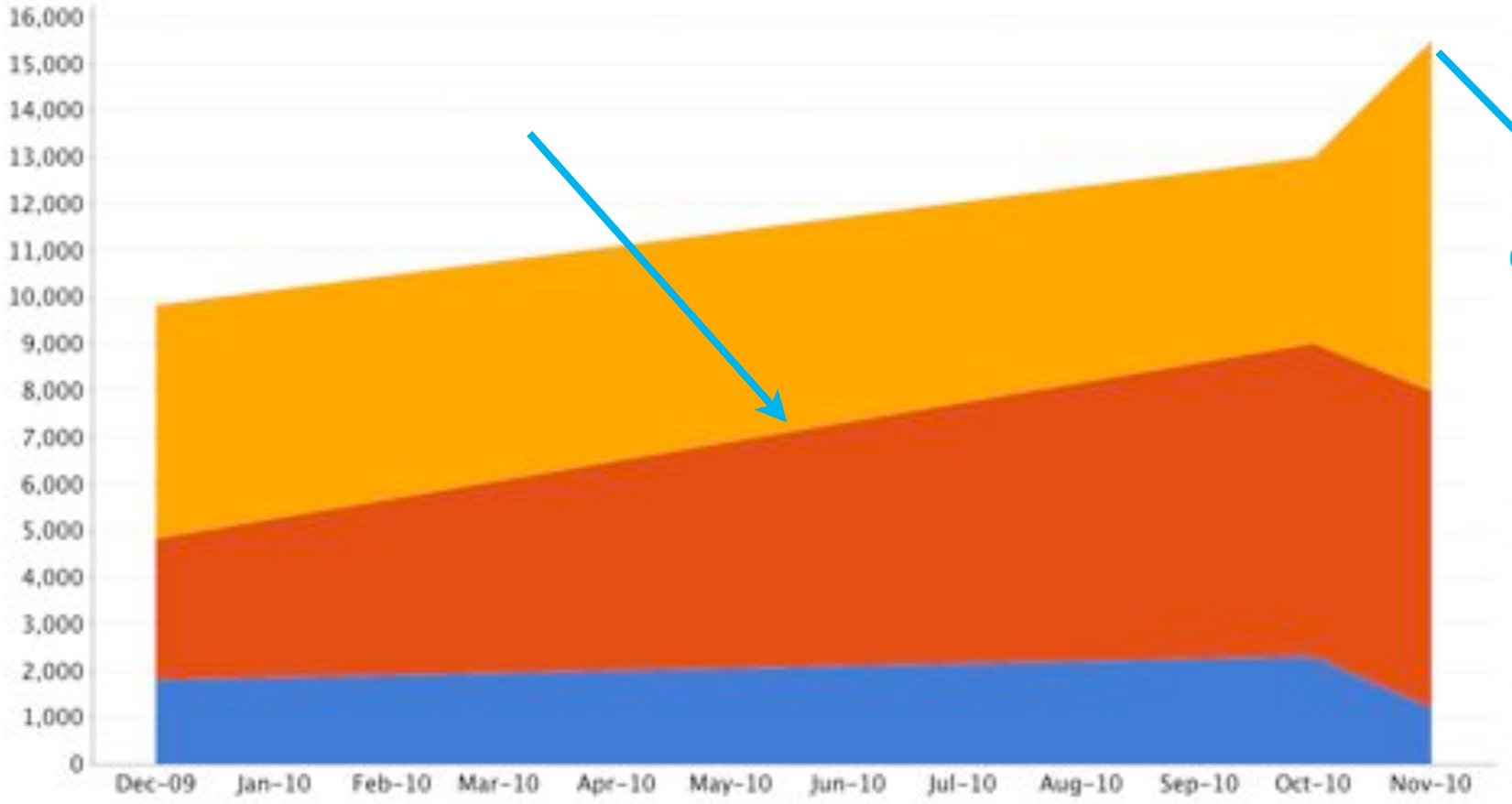


Checklist >> Evaluatie

Architectuur metrieken

Software Improvement Group

Module grootte over tijd



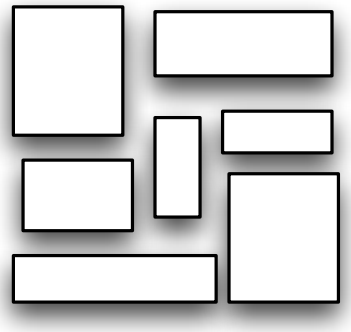
Checklist >>
Evaluatie

Metriek hergebruiken
Gini-coëfficiënt

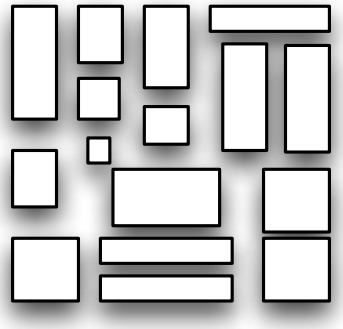
Software Improvement Group

C. Gini. *Measurement of inequality of income.*
Economic Journal, 31:22–43, 1921.

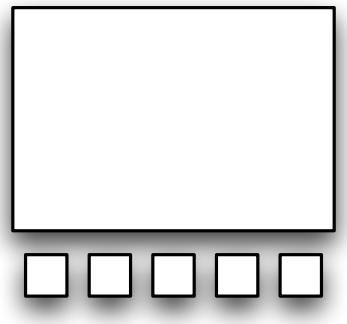
Module Size Uniformity = 1 - Gini



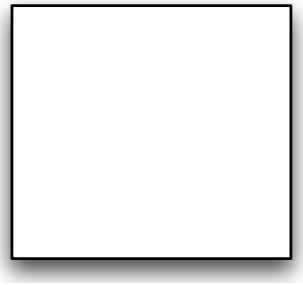
0.8



0.7



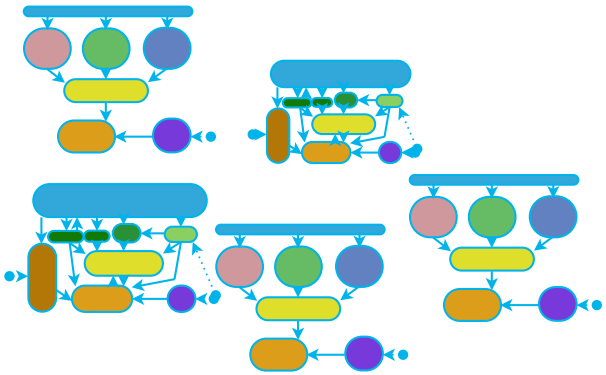
0.2

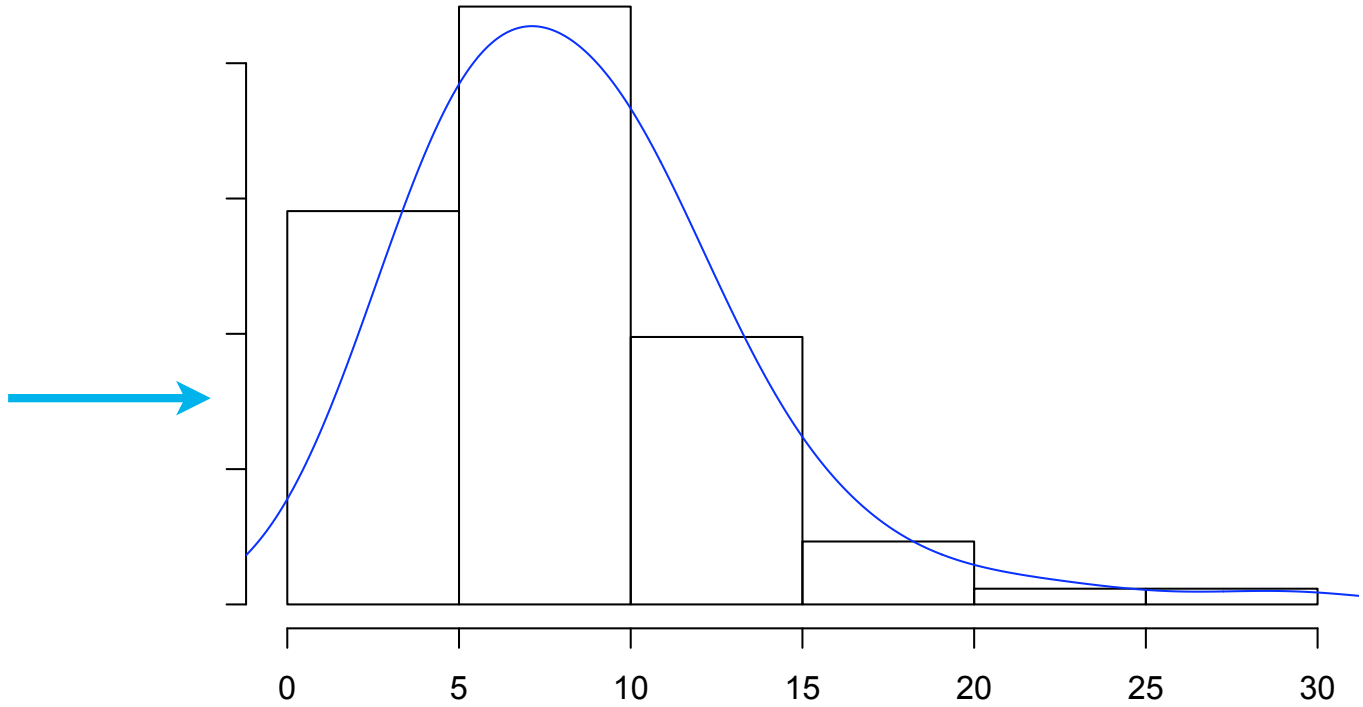


1

Modules inventariseren

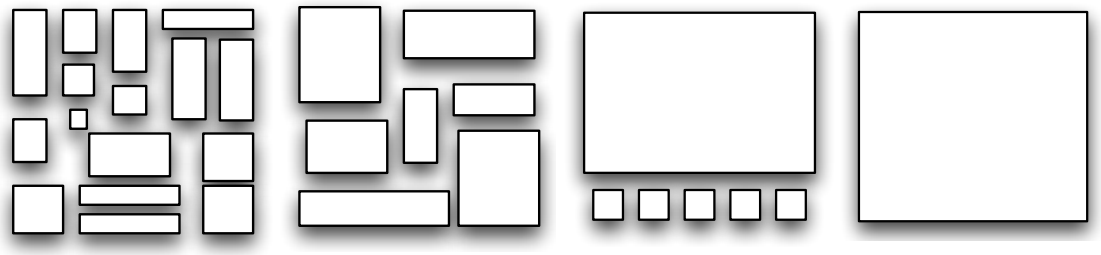
Software Improvement Group





Module Balance

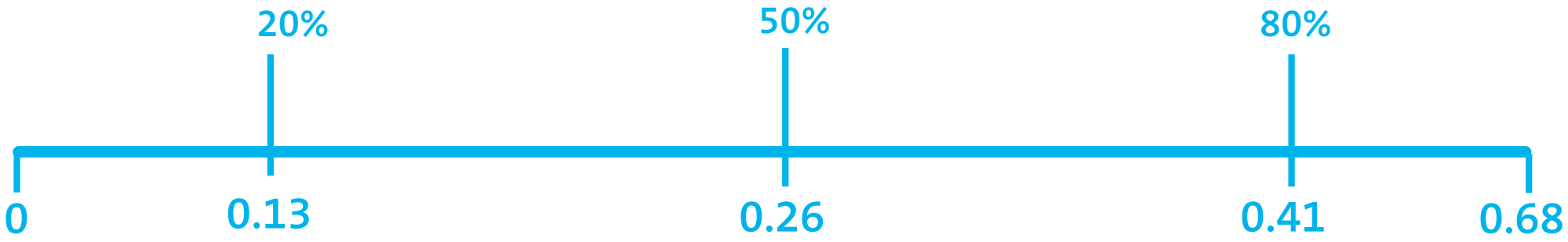
Software Improvement Group



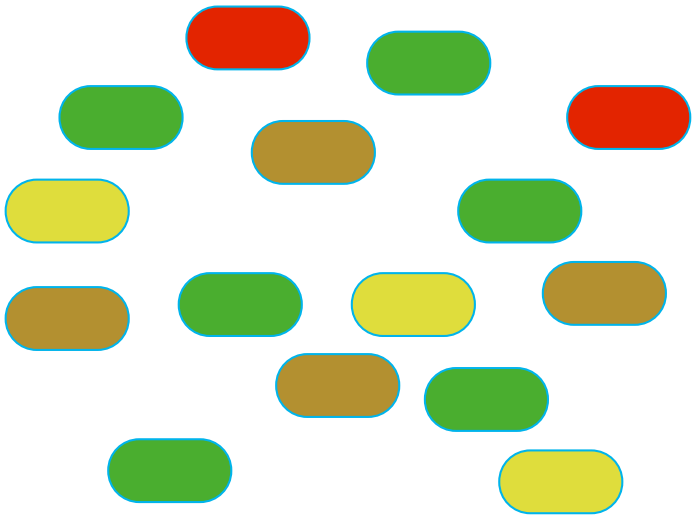
Module Size Uniformity:	0.8	0.8	0,2	1
System Breakdown Optimality:	0.1	0.8	0.8	0
Module Balance:	0.08	0.64	0.16	0

Grenswaarden bepalen
Benchmarking

Software Improvement Group



0.05 omlaag? >> checklist >>evaluatie



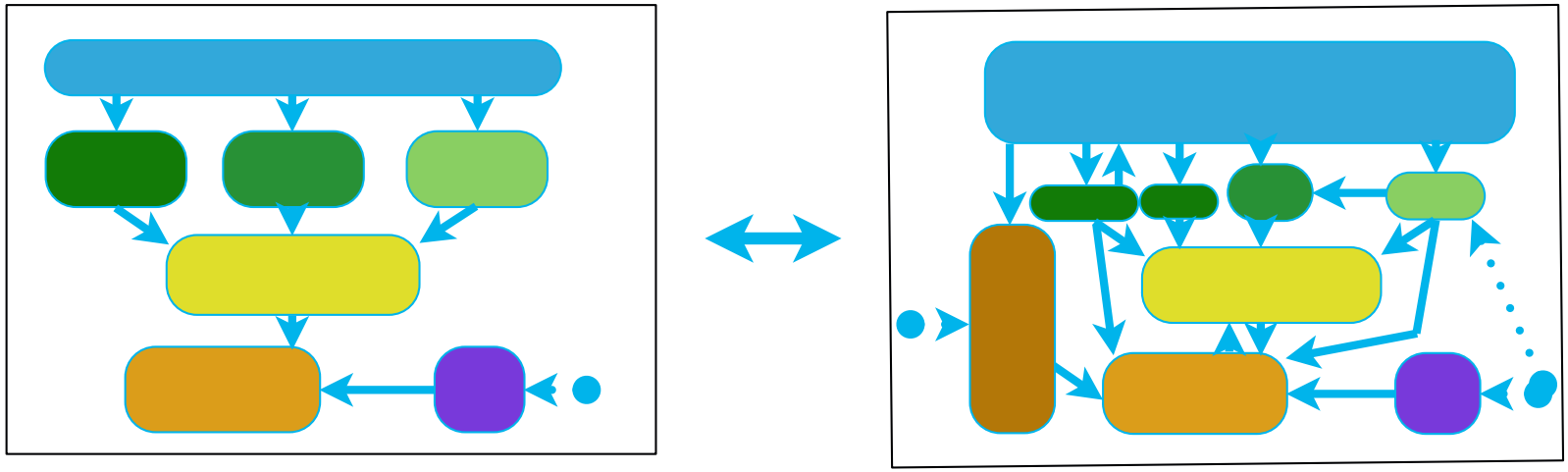
Toekomstplannen

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

Conclusie

Software Improvement Group



http://www.sig.eu/en/liscia

Contact

Eric Bouwers

E.Bouwers@sig.eu