SYSTEMS ENGINEERING RESEARCH CENTER

SERC TALKS

# WELCOME

***Why Are Ontologies and Languages for Software Quality Increasingly Important?***

**Xavier Franch**, Full Professor, Polytechnic University of Catalonia (BarcelonaTech) – franch@essi.upc.edu

**December 11 | 1:00 PM ET**

- ❑ Today's session will be recorded.
- ❑ An archive of today's talk will be available at: www.sercuarc.org/serc-talks/
- ❑ Use the Q&A box to queue questions, reserving the chat box for comments, and questions will be answered during the last 5-10 minutes of the session.
- ❑ If you are connected via the dial-in information only, please email questions or comments to SERCtalks@stevens.edu.
- ❑ Any issues? Use the chat feature for any technical difficulties or other comments, or email SERCtalks@stevens.edu.

# Why are Ontologies and Languages for Software Quality Increasingly Important?

**Prof. Xavier Franch**
**UPC-BarcelonaTech, Barcelona, Catalonia (Spain)**
**SERC Talk, December 2018**

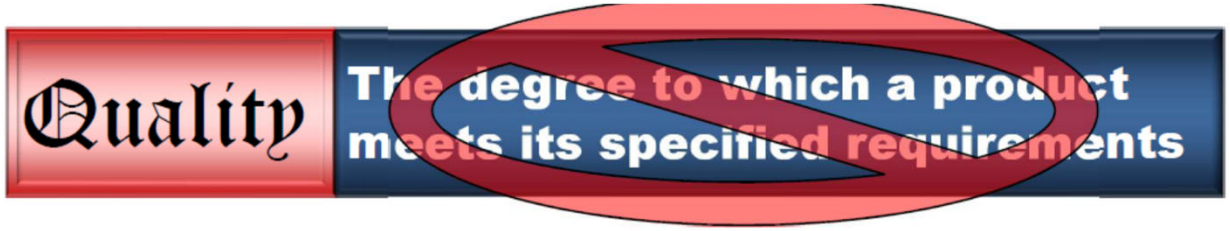
Grup de recerca en Enginyeria del Software
per als Sistemes d´Informació
UNIVERSITAT POLITÈCNICA DE CATALUNYA

# Outline

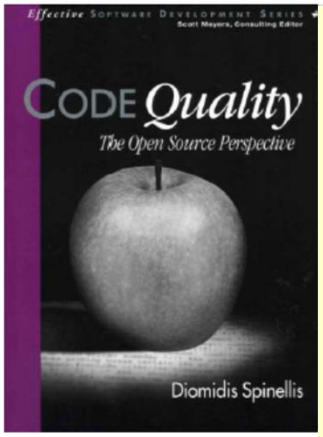
- Motivation
- Ontologies
- Quality models
  - How are they used in practice
- Customization of quality models
  - An example: the Q-Rapids project
  - Another example: requirement patterns
- Expressing ontologies
  - An example: the NFR framework
- Summary

# Motivation

Why ~~are Ontologies and Languages for~~ *is* Software Quality Increasingly Important?




**Problem** — Customers struggle to state functional requirements. They do not understand non-functional requirements.




"…a failure to satisfy a non-functional requirement can be critical, even catastrophic…"

*Bill Curtis, SERC Talk Oct. 2018*

# Quality is elusive…

- Difficult to specify
- Difficult to quantify
- Difficult to verify
- ***Difficult to agree upon***

IEEE Software

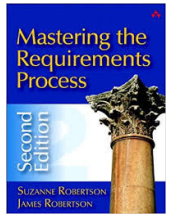
**Software Quality: The Elusive Target**

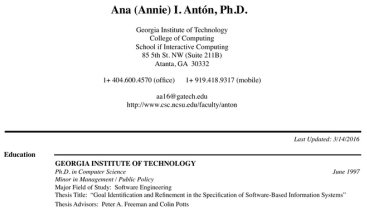
Issue No. 01 - January (1996 vol. 13)
ISSN: 0740-7459
pp: 12-21
DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/52.476281

Barbara Kitchenham
Shari Lawrence Pfleeger

# Example: the NFR concept itself

"A **property**, or **quality**, that the product must have, such as an appearance, or a speed or accuracy"



**Ana (Annie) I. Antón, Ph.D.**

Georgia Institute of Technology
College of Computing
School if Interactive Computing
85 5th St. NW (Suite 211B)
Atanta, GA 30332

1+ 404.600.4570 (office) 1+ 919.418.9317 (mobile)

aa16@gatech.edu
http://www.csc.ncsu.edu/faculty/anton

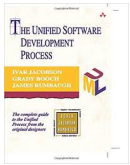
*Last Updated: 3/14/2016*

**Education**

**GEORGIA INSTITUTE OF TECHNOLOGY** *June 1997*
*Ph.D. in Computer Science*
*Minor in Management / Public Policy*
Major Field of Study: Software Engineering
Thesis Title: "Goal Identification and Refinement in the Specification of Software-Based Information Systems"
Thesis Advisors: Peter A. Freeman and Colin Potts

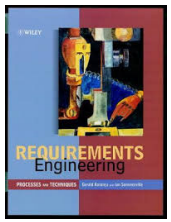
"Describe the **nonbehavioral aspects** of a system, capturing the **properties** and **constraints** under which a system must operate"

"Requirement that specifies **criteria** that can be used to judge the operation of a system, rather than specific behaviors"

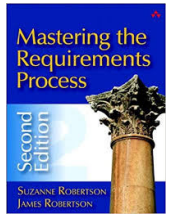

"A requirement that specifies **physical constraints** on a functional requirement"

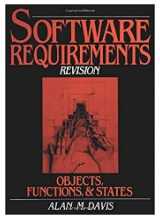
"They place **restrictions** on the product being developed and the development process, and they specify **external constraints** that the product must meet"



# Example: categories of quality

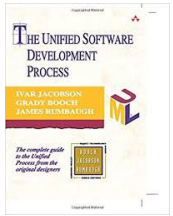
"A property, or quality, that the product must have, such as an **appearance**, or a **speed** or **accuracy**"

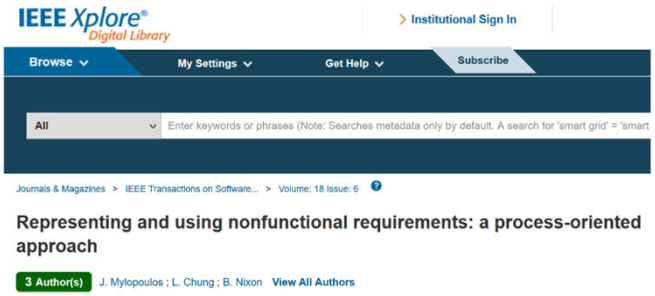



"The required overall attributes of the system, including **portability**, **reliability**, **efficiency**, **human engineering**, **testability**, **understandability**, and **modifiability**"

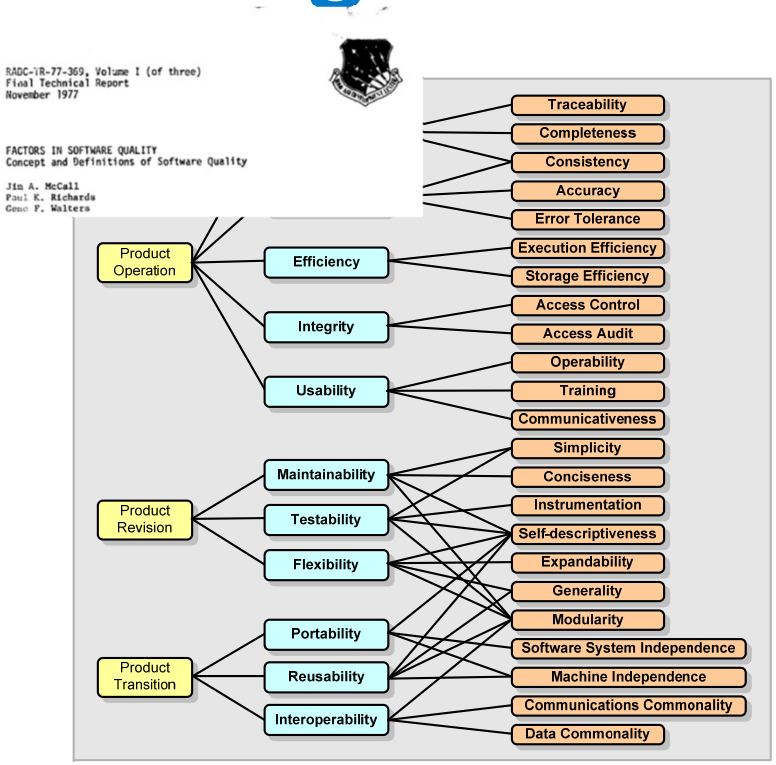
"A requirement that specifies system properties, such as **environmental and implementation constraints**, **performance**, **platform dependencies**, **maintainability**, **extensibility**, and **reliability**"

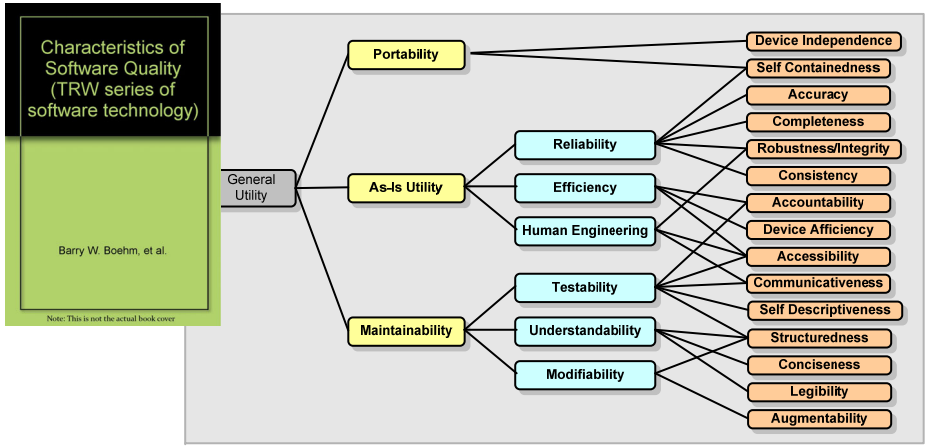


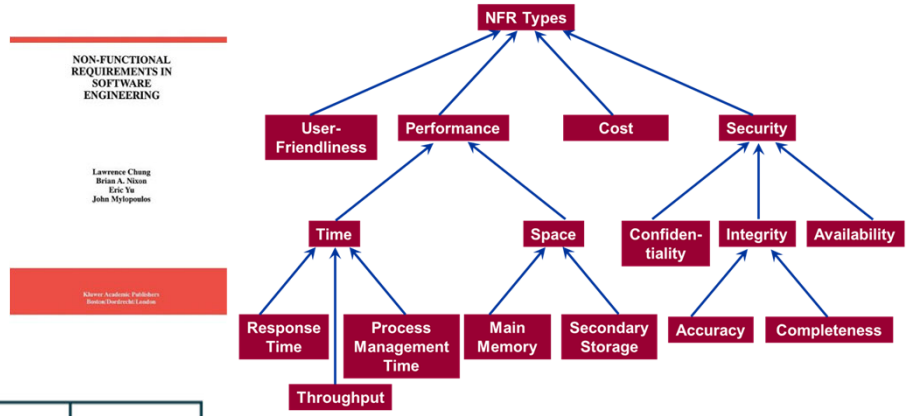



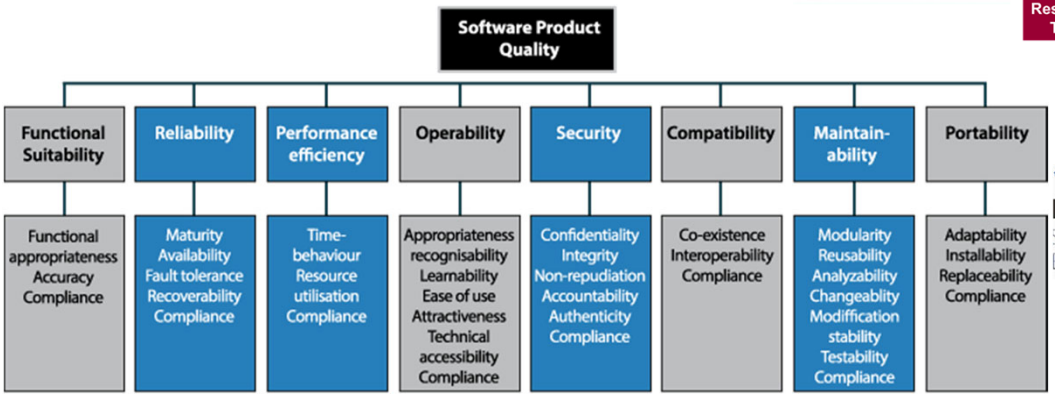

"Global requirements on its **development** or **operational cost**, **performance**, **reliability**, **maintainability**, **portability**, **robustness**, and the like"

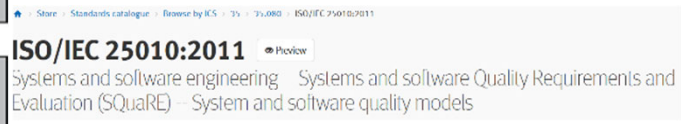
# Categories of quality aspects



# How to cope with this diversity?

Concept of **ontology**

- Branch of philosophy which deals specially with the nature of being
- Explicit specification of a shared conceptualization
- Provide a precise and unambiguous terminology

**A translation approach to portable ontology specifications**

Author: Thomas R. Gruber

1993 Article

Published in:
· Journal
Knowledge Acquisition - Special issue: Current issues in knowledge modeling archive
Volume 5 Issue 2, June 1993
Pages 199 - 220
Academic Press Ltd. London, UK, UK
table of contents doi>10.1006/knac.1993.1008

Bibliometrics

- Citation Count: 1,883
- Downloads (cumulative): n/a
- Downloads (12 Months): n/a
- Downloads (6 Weeks): n/a

# Use of ontologies in RE

Three main categories:

Vol 10 No 6 (2010): Global Journal of Researches in Engineering

Articles

The Use of Ontologies in Requirements Engineering

Vol 10 Issue 6

VerÃ³nica CastaÃ±eda, Luciana Ballejos, Ma. Laura Caliusco, Ma. Rosa Galli

PDF

# Use of ontologies in RE

Three main categories:

- Structure of specification documents


OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"

OTM 2009: On the Move to Meaningful Internet Systems: OTM 2009 pp 966-982 | Cite as

How to Write and Use the Ontology Requirements Specification Document

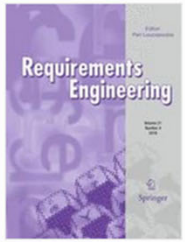
Authors | Authors and affiliations

Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, Boris Villazón-Terrazas

# Use of ontologies in RE

Three main categories:

- Structure of specification documents
- Fundamental ontologies




Requirements Engineering
November 2016, Volume 21, Issue 4, pp 405–437 | Cite as

Applications of ontologies in requirements engineering: a systematic review of the literature

Authors  Authors and affiliations

Diego Dermeval, Jéssyka Vilela, Ig Ibert Bittencourt, Jaelson Castro, Seiji Isotani, Patrick Brito, Alan Silva

# Use of ontologies in RE

Three main categories:

- Structure of specification documents
- Fundamental ontologies
- Application domain ontologies


International Conference on Advanced Information Systems Engineering

CAiSE 2012: Advanced Information Systems Engineering Workshops pp 61-69 | Cite as

Ontologies for Security Requirements: A Literature Survey and Classification

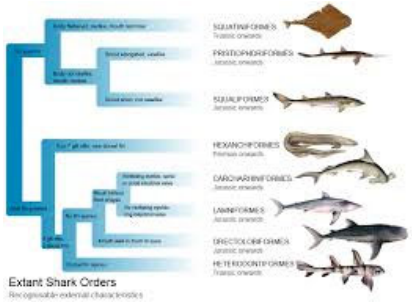
Authors | Authors and affiliations

Amina Souag, Camille Salinesi, Isabelle Comyn-Wattiau

# Expressing an ontology

An ontology needs to represent:

- Lexicon of concepts
- Taxonomical arrangement
- Properties among concepts


$$E = mc^2$$

# Some approaches exist for quality

INCOSE International Symposium

Volume 25, Issue 1
October 2015
Pages 341-356

Regular article

## An Initial Ontology for System Qualities

Barry Boehm, Nupul Kukreja

First published: 29 October 2015 | https://doi.org/10.1002/j.2334-5837.2015.00067.x

PDF TOOLS SHARE

Access Related Inform

The primary need for a system ilities ontology is to enable more effective collaboration in system definition, development, and evolution across system stakeholders from different disciplines.

# Quality Models (QM)

- A lightweight implementation of ontologies for quality:

  "The set of characteristics and the relationships between them which provide the basis for specifying quality requirements and evaluating quality"

  - Characteristics ≅ quality attributes
  - Relationships ≅ (taxonomic) decomposition


INTERNATIONAL STANDARD

ISO/IEC 14598-1

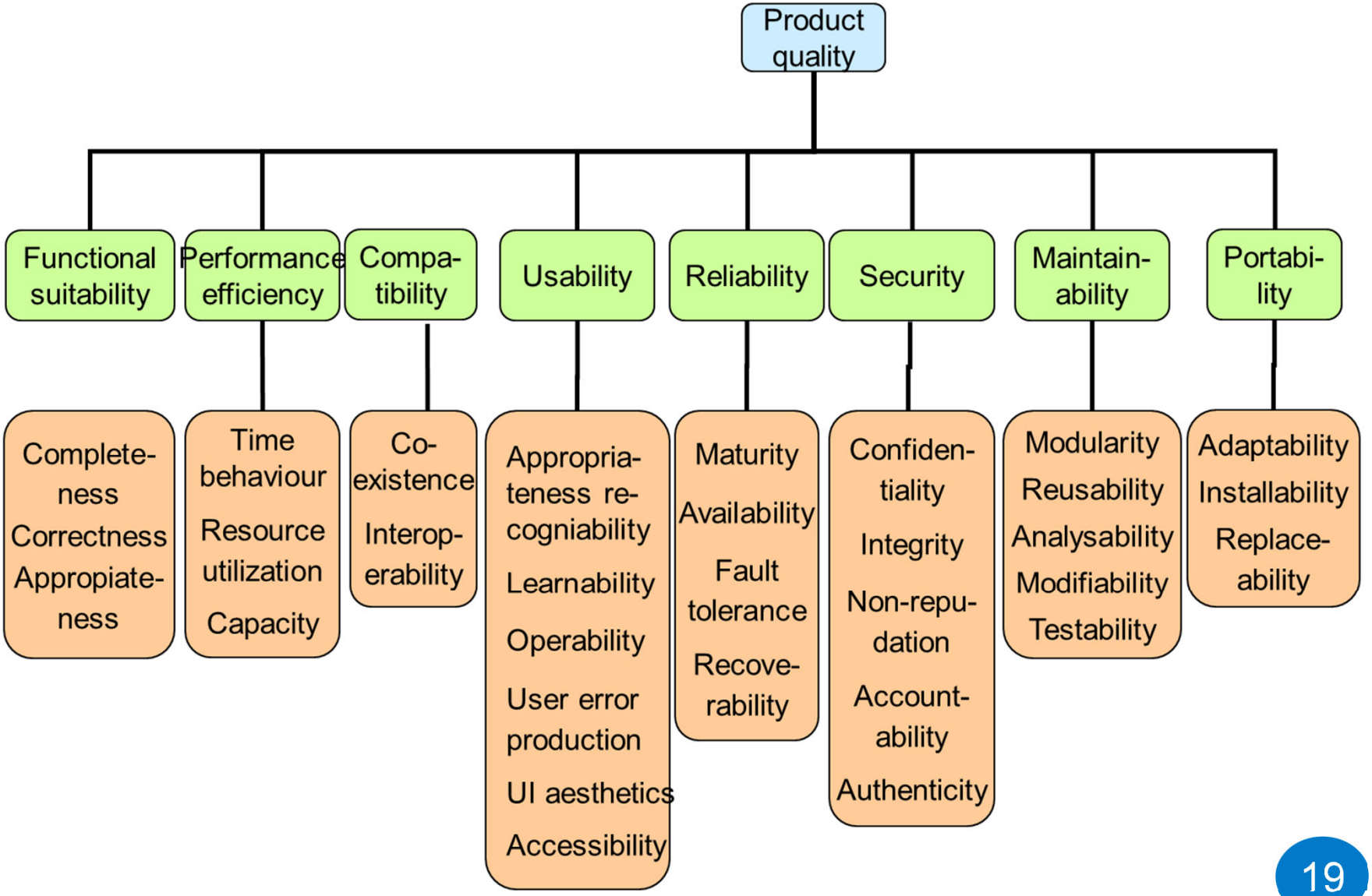
Information technology — Software product evaluation —

Part 1: General overview

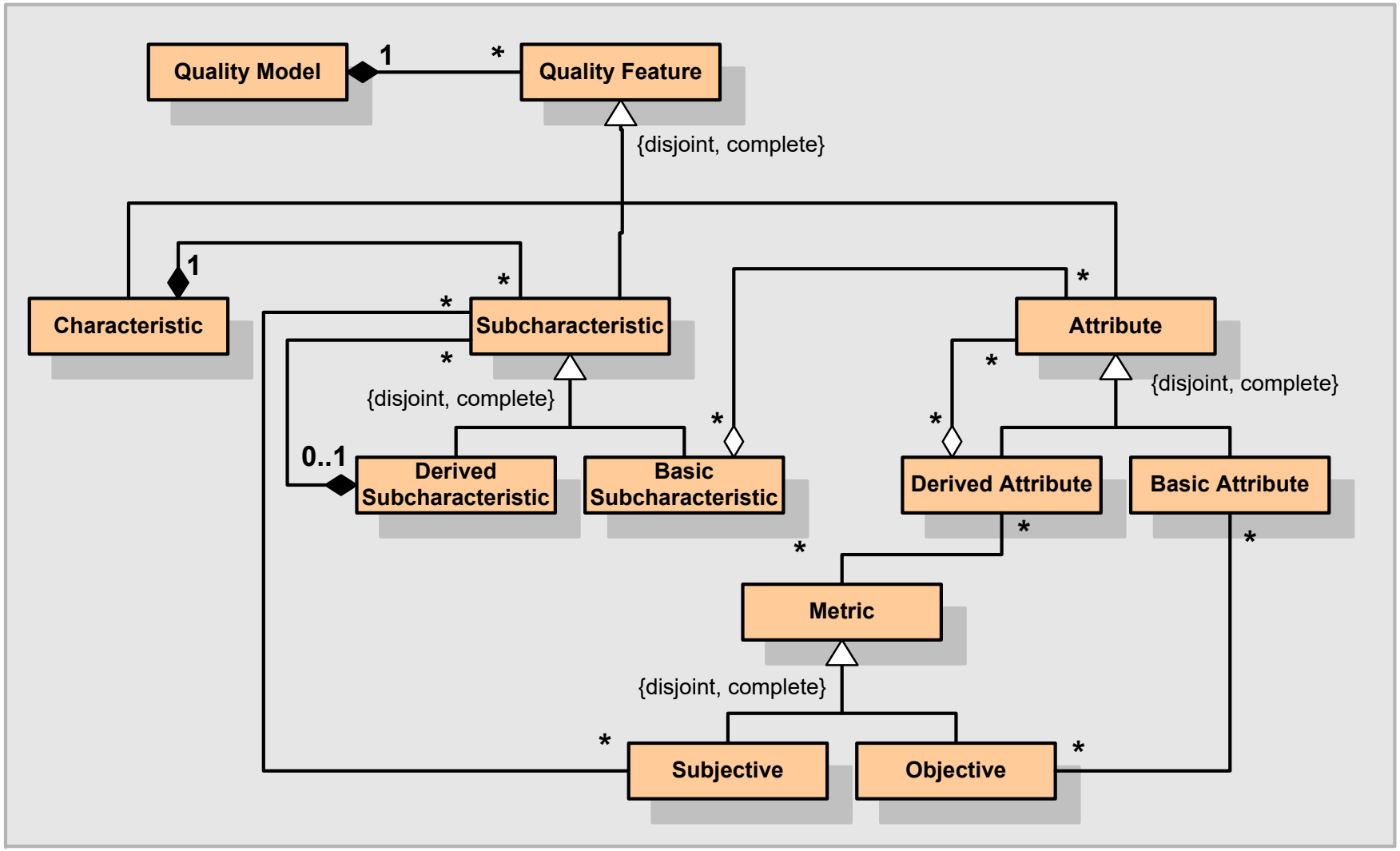
# Example: ISO/IEC 25010 – Terms

| Characteristic | Definition |
|---|---|
| Performance efficiency | performance relative to the amount of resources used under stated conditions |
| Compa-tibility | degree to which a product, system or component can exchange information with other products, systems or components, and/or perform its required functions, while sharing the same hardware or software environment |
| Usability | degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use |
| Reliability | degree to which a system, product or component performs specified functions under specified conditions for a specified period of time |
| Security | degree to which a product or system protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization |
| Mainta-inability | degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers |
| Portability | degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another |

# ISO/IEC 25010 – Taxonomy



# ISO/IEC 25010 - Metamodel



# How does the community work?

ELSEVIER

Information and Software Technology

Volume 56, Issue 10, October 2014, Pages 1167-1182


## Quality models for web services: A systematic mapping

Marc Oriol, Jordi Marco, Xavier Franch

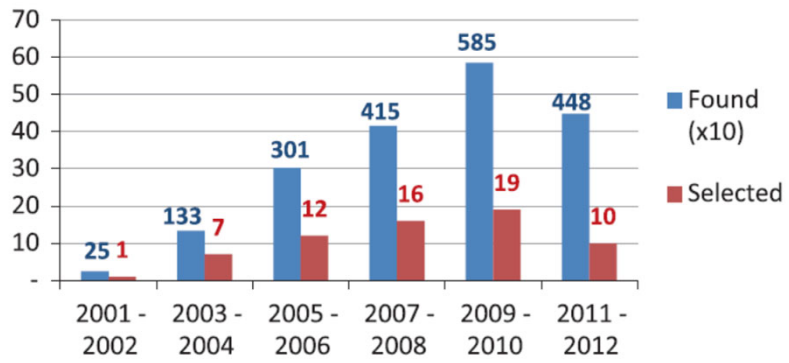
Show more

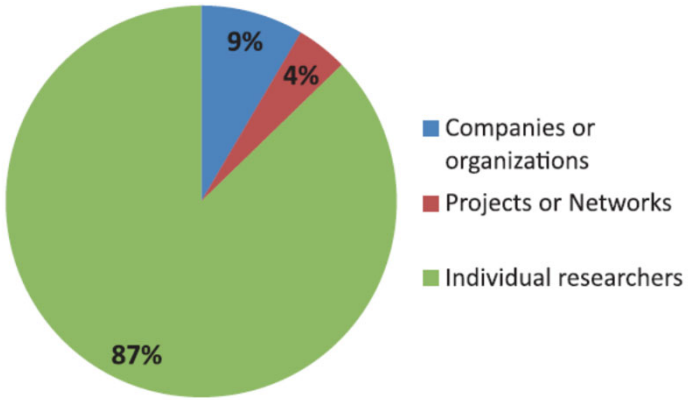
https://doi.org/10.1016/j.infsof.2014.03.012

Get rights and content

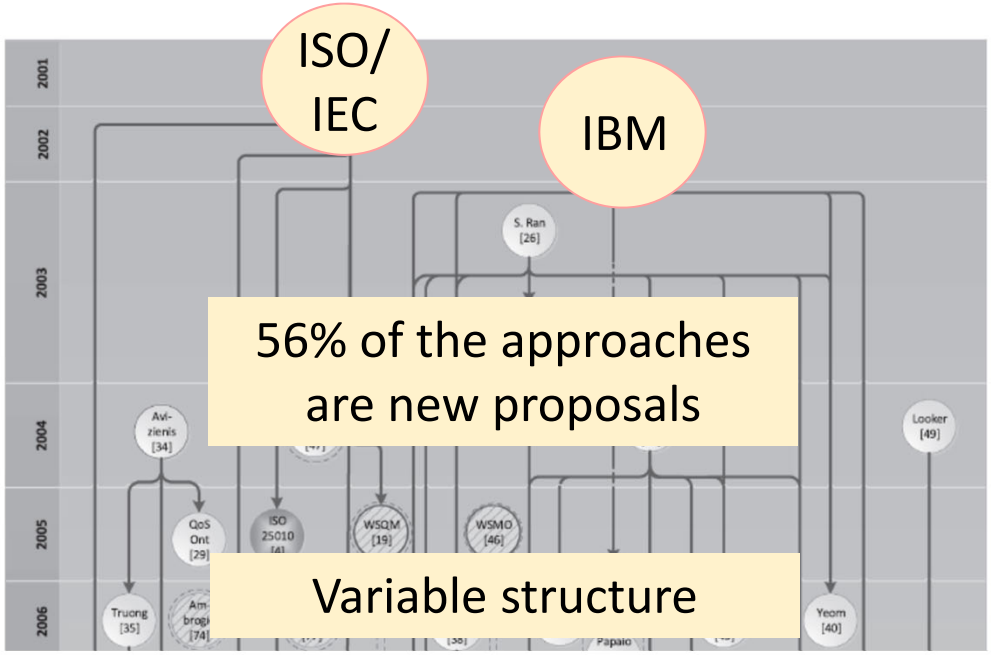
# Based on a literature review

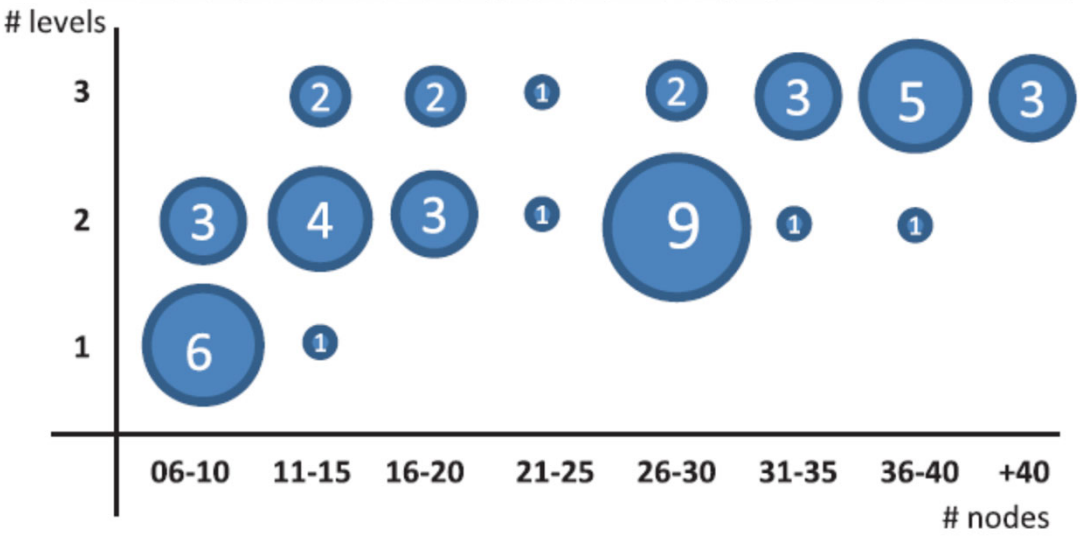
65 works presenting 47 different proposals of quality models in the period 2001-2013



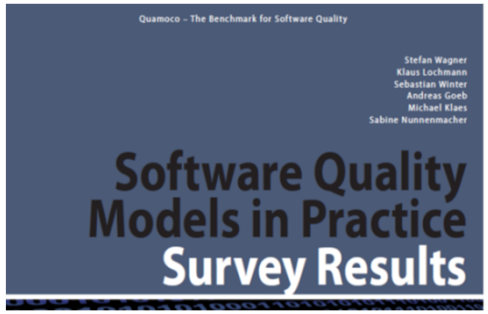



# Results


ISO/
IEC
IBM
56% of the approaches
are new proposals
Variable structure
2001
2002
2003
2004
2005
2006
S. Ran
[26]
Avi-
zienis
[34]
QoS
Ont
[29]
ISO
25010
WSQM
[19]
WSMO
[46]
Looker
[49]
Truong
[35]
Yeom
[40]


# levels
3
2
1
2
2
1
2
3
5
3
3
4
3
1
9
1
1
6
1
06-10
11-15
16-20
21-25
26-30
31-35
36-40
+40
# nodes

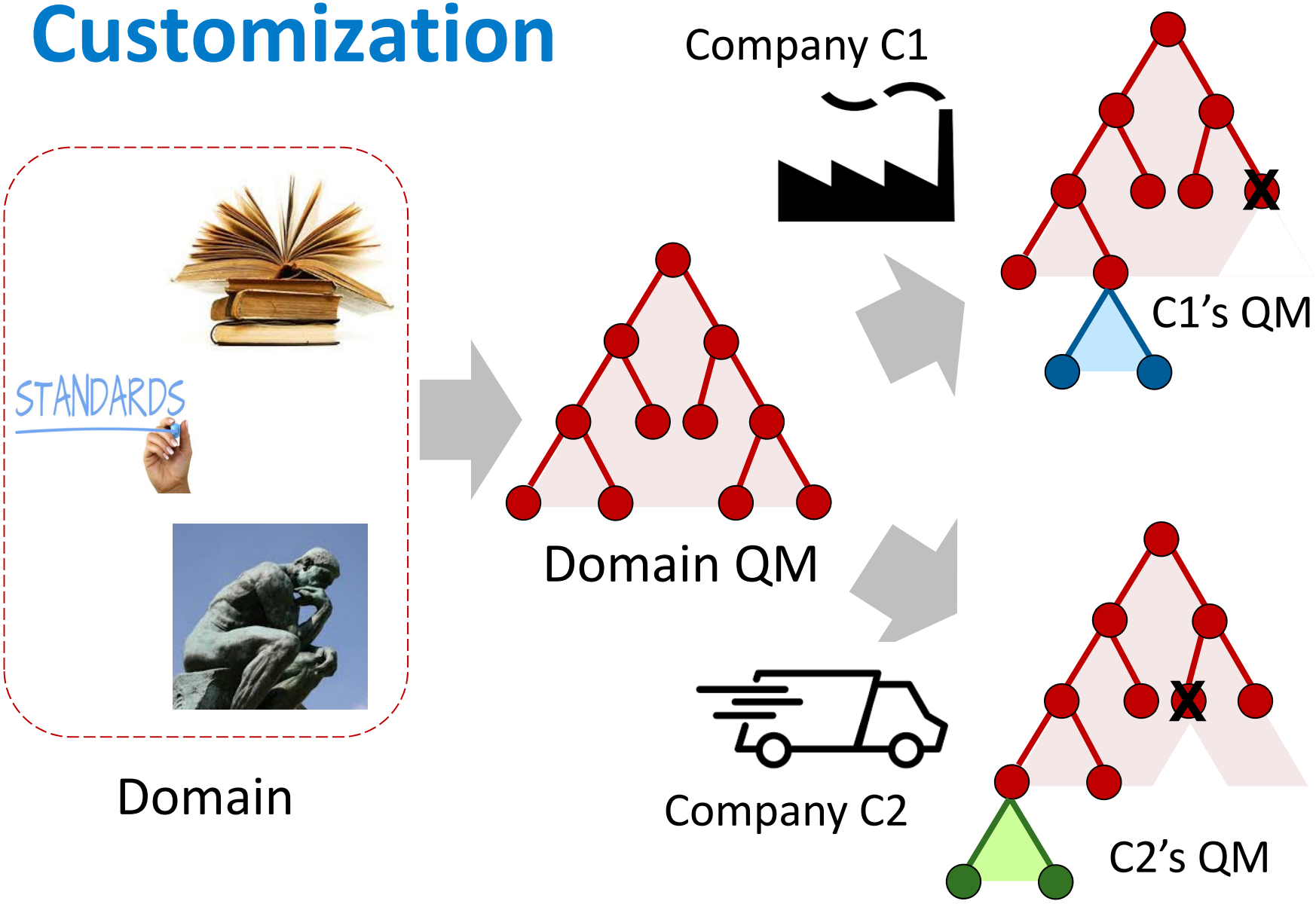
# Use in practice




More than 85% of companies may need customized quality models, adapted to their needs

We need effective ways to build such customized quality models while enforcing ontological understanding as much as possible

# Customization



# Building a domain model

- Locate the appropriate sources
- Organize the existing knowledge
  - See Web Services case
- Analyse the quality (sub)characteristics that are more relevant in the domain
  - E.g., using the ISO/IEC 25010 quality model (or any other)
- Identify quality attributes
  - Effort in those (sub)characteristics that are more relevant

# WS case: ISO/IEC 25010 coverage

| | Functional suitability | Performance efficiency | Compati-bility | Usability | Reliability | Security | Maintaina-bility | Portability |
|---|---|---|---|---|---|---|---|---|
| **Y+** | 2% | 49% | 0% | 6% | 77% | 55% | 11% | 2% |
| **Y** | 2% | 19% | 0% | 4% | 4% | 19% | 6% | 2% |
| **P+** | 6% | 17% | 2% | 4% | 11% | 4% | 4% | 2% |
| **P** | 57% | 6% | 34% | 13% | 8% | 4% | 15% | 40% |
| **ND** | 32% | 9% | 64% | 72% | 0% | 17% | 64% | 53% |

# WS case: ISO/IEC 25010 coverage

| | 5. Reliability | 5.1 Maturity | 5.2 Availability | 5.3 Fault tolerance | 5.4 Recovera-bility |
|---|---|---|---|---|---|
| Y+ | 77% | 0% | 28% | 13% | 23% |
| Y | 4% | 2% | 66% | 26% | 6% |
| P+ | 11% | 0% | 0% | 0% | 2% |
| P | 9% | 11% | 0% | 0% | 13% |
| (ND) | 0% | 87% | 6% | 62% | 55% |

| | 4. Usability | 4.1 Appropriate recognizability | 4.2 Learnability | 4.3 Operability | 4.4 User error protection | 4.5Userinterfa-ce aesthetics | 4.6 Accessi-bility |
|---|---|---|---|---|---|---|---|
| Y+ | 6% | 6% | 0% | 4% | 0% | 0% | 0% |
| Y | 4% | 2% | 4% | 2% | 0% | 0% | 0% |
| P+ | 4% | 0% | 0% | 0% | 0% | 0% | 0% |
| P | 13% | 6% | 0% | 2% | 0% | 0% | 0% |
| (ND) | 72% | 85% | 95% | 91% | 100% | 100% | 100% |

# Defining quality attributes

| ISO/IEC 25010 ⇨ subcharacteristics | Quality attribute | Definition | Definition usage % | Name usage % |
|---|---|---|---|---|
| | | Functional suitability | | |
| Functional ⇨ correctness | **Accuracy** | Error rate produced by the service [26][28]. | 62% | 42% |
| | | Performance efficiency | | |
| Capacity ⇨ | **Accessibility** | Degree the service is capable of serving a web service request [53][16]. | 47% | 100% |
| Capacity ⇨ | **Capacity**[1] | Limit of concurrent requests for guaranteed performance [26][55]. | 51% | 65% |
| Time behaviour ⇨ | **Response Time** | Time to complete a WS request (from a client perspective) [26][28]. | 83% | 73% |
| Time behaviour ⇨ | **Throughput** | Number of web service requests served at a given time period [28][16]. | 60% | 100% |
| | | Compatibility | | |
| *(subcharacteristic)* | **Interoperability** | Ease with which a consumer application or agent interoperates with a service [27]. | 36% | 100% |
| | | Reliability | | |
| Availability ⇨ | **Availability**[1] | Probability that the service can respond to consumer requests [51][27]. | 94% | 98% |
| Recoverability ⇨ | **MTTR** | Mean Time to Repair [30][27]. | 30% | 86% |
| Fault tolerance ⇨ | **Robustness** | Degree to which a service can function correctly in the presence of invalid, incomplete or conflicting inputs [26][28]. | 38% | 94% |
| | | Security | | |
| *(subcharacteristic)* | **Authentication** | Measure of how the service authenticates principals (users or other services) who can access service and data [26]. | 45% | 95% |
| *(subcharacteristic)* | **Confidentiality** | Measure of how the service treats the data, so that only authorized principals can access or modify the data [26]. | 45% | 86% |

# Customizing a domain QM

ELSEVIER

Information and Software Technology

Volume 62, June 2015, Pages 101-123


## Operationalised product quality models and assessment: The Quamoco approach

Stefan Wagner [a], Andreas Goeb [b], Lars Heinemann [b], Michael Kläs [c], Constanza Lampasona [c], Klaus Lochmann [d], Alois Mayr [e], Reinhold Plösch [e], Andreas Seidl [f], Jonathan Streit [g], Adam Trendowicz [c]

Show more

https://doi.org/10.1016/j.infsof.2015.02.009



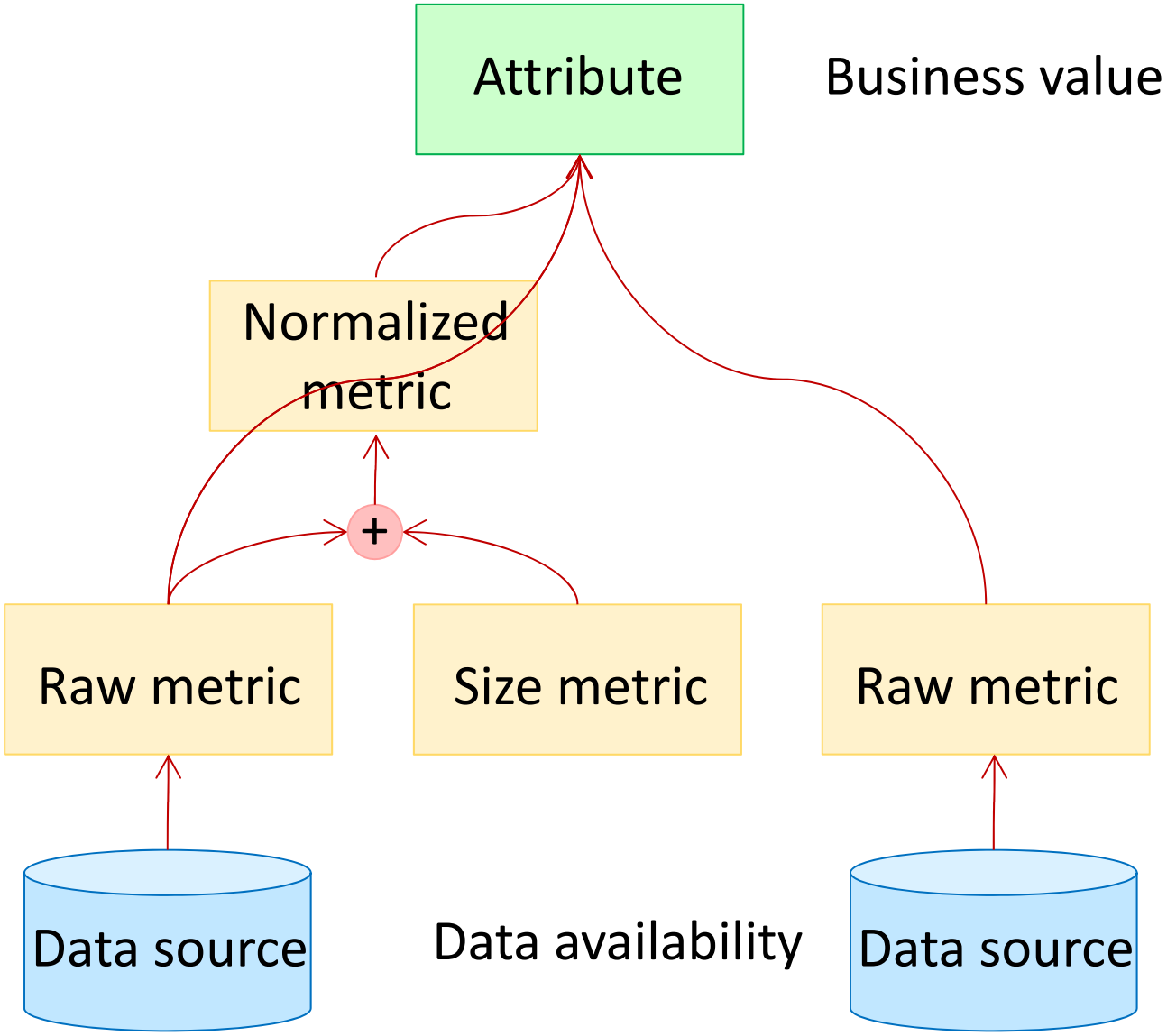
- Use their own metamodel
- Focus on internal quality
- Emphasis on modularity

I present some simplification/customization of the method

# 1. Definition of metrics

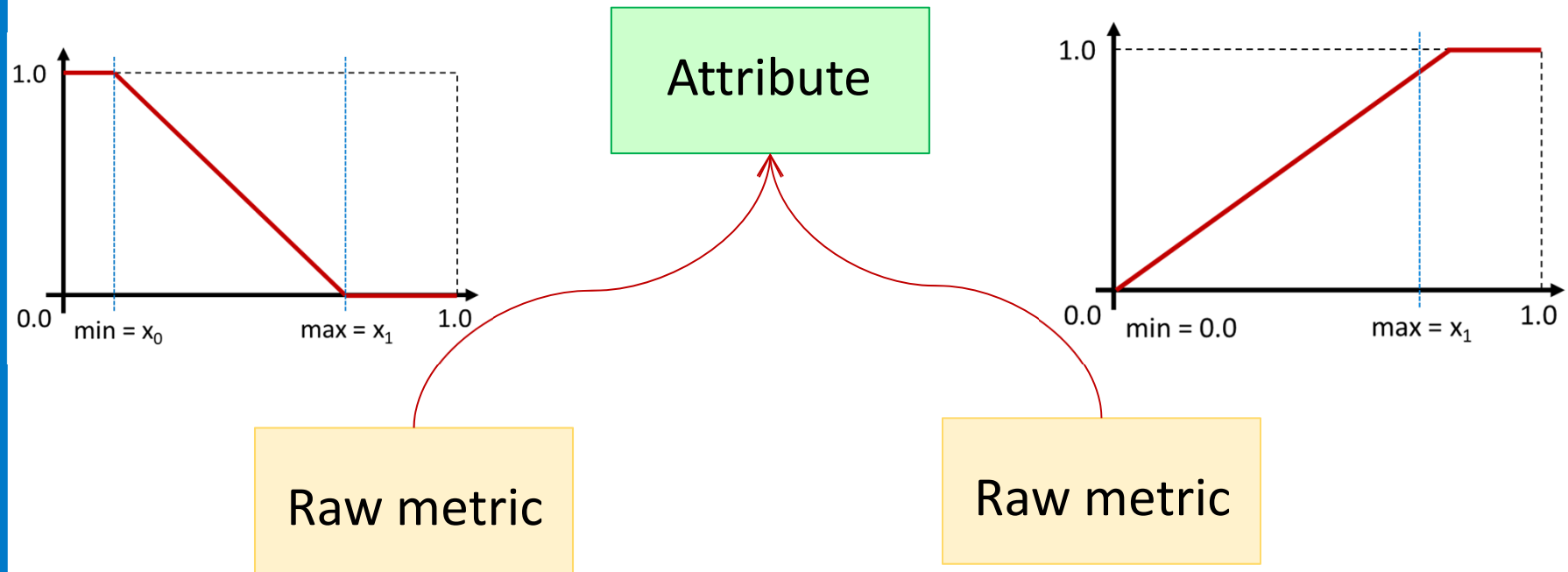
- In the basis of:
  - which are the business priorities of the project
  - what data is available (i.e., can be effectively used)

# 1. Definition of metrics



# 2. Evaluation of attributes

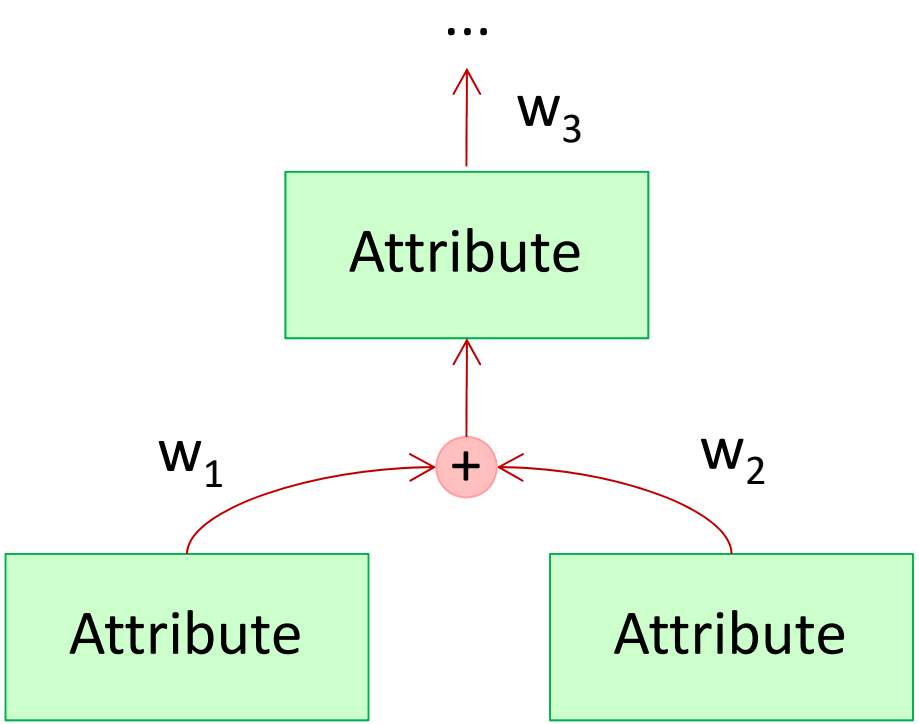
- Determination metrics' impact onto attributes
  - Use of (linear) utility functions
    - Utility: strength of preference a decision-maker has among alternatives regarding a specific quality aspect




Thresholds coming from benchmarking

# 3. Aggregation of factors

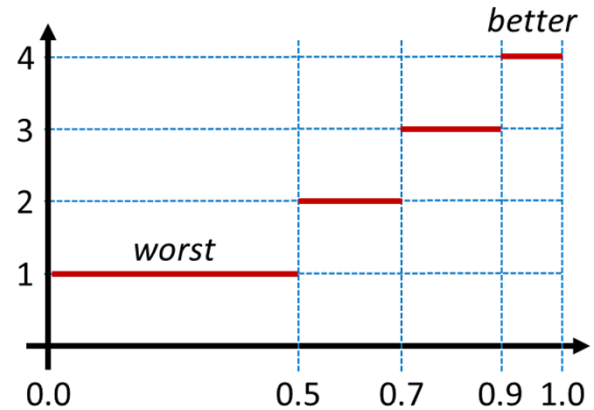
- Synthesis of individual utilities into global one
  - Weights for contributing factors

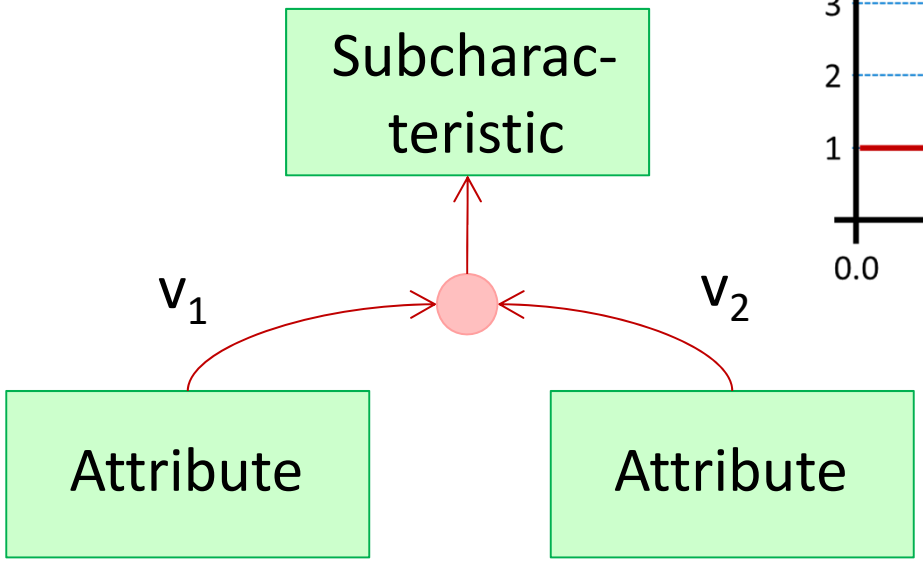



Weights coming from existing data or expert opinion

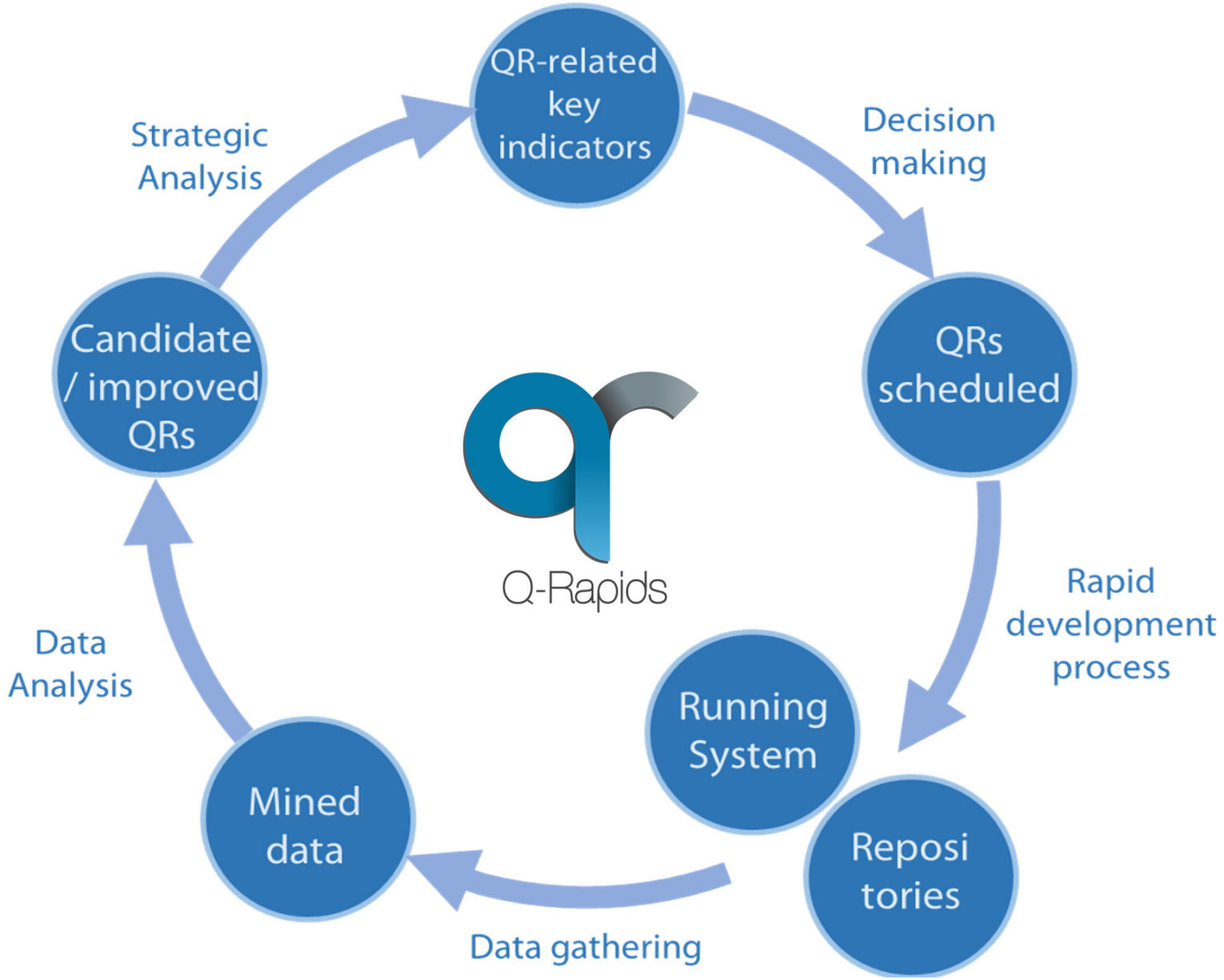
# 4. Interpretation

- Translation of utility values into a QA value
  - Interpretation model transforming from ratio to ordinal
  - Could be done at the level of general quality, a characteristic or a subcharacteristic
  - E.g., analogy with school grade

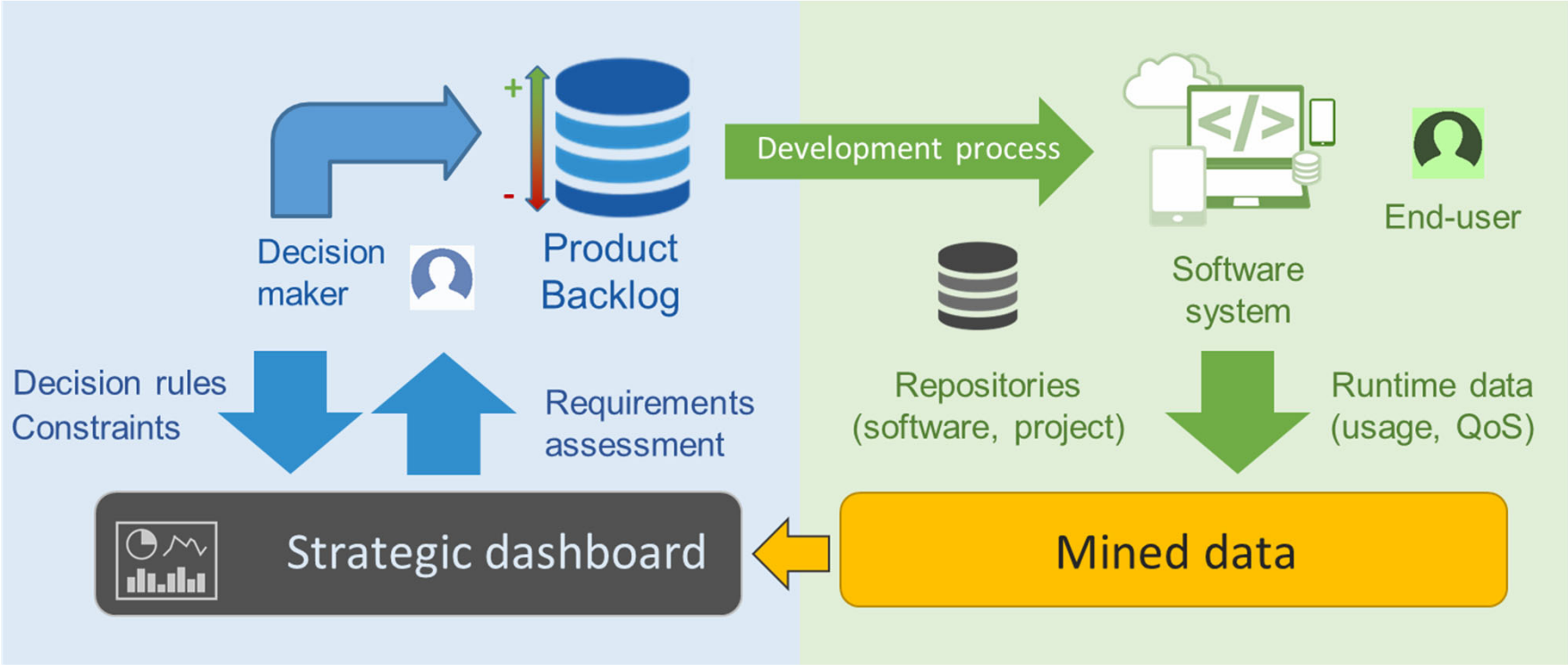


# Example: The Q-Rapids project



# Example: The Q-Rapids project




International Working Conference on Requirements Engineering: Foundation for Software Quality

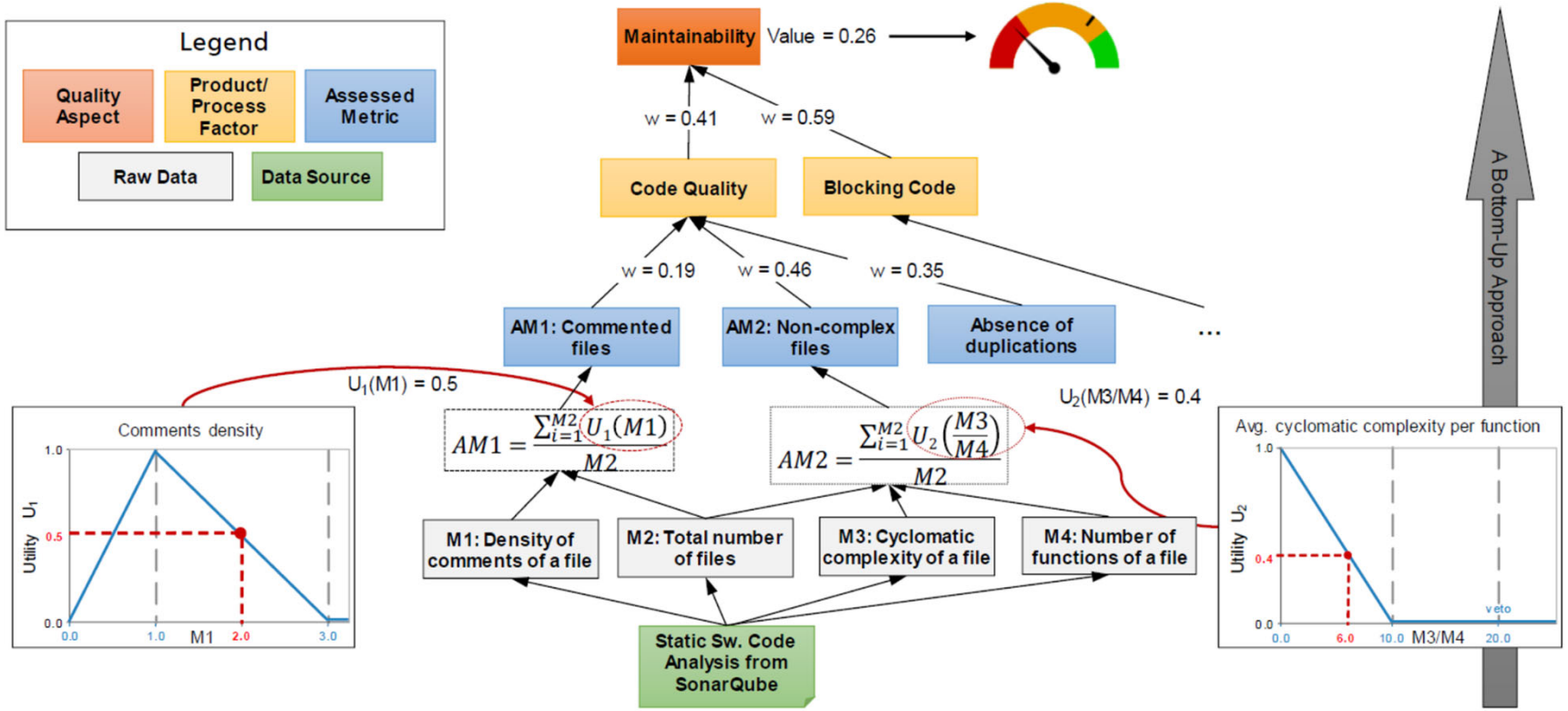
REFSQ 2017: Requirements Engineering: Foundation for Software Quality pp 167-173 | Cite as

How Can Quality Awareness Support Rapid Software Development? – A Research Preview

Authors | Authors and affiliations

Liliana Guzmán, Marc Oriol, Pilar Rodríguez, Xavier Franch, Andreas Jedlitschka, Markku Oivo

# Overall view




IEEE Xplore® Digital Library

Institutional Sign In

IEEE

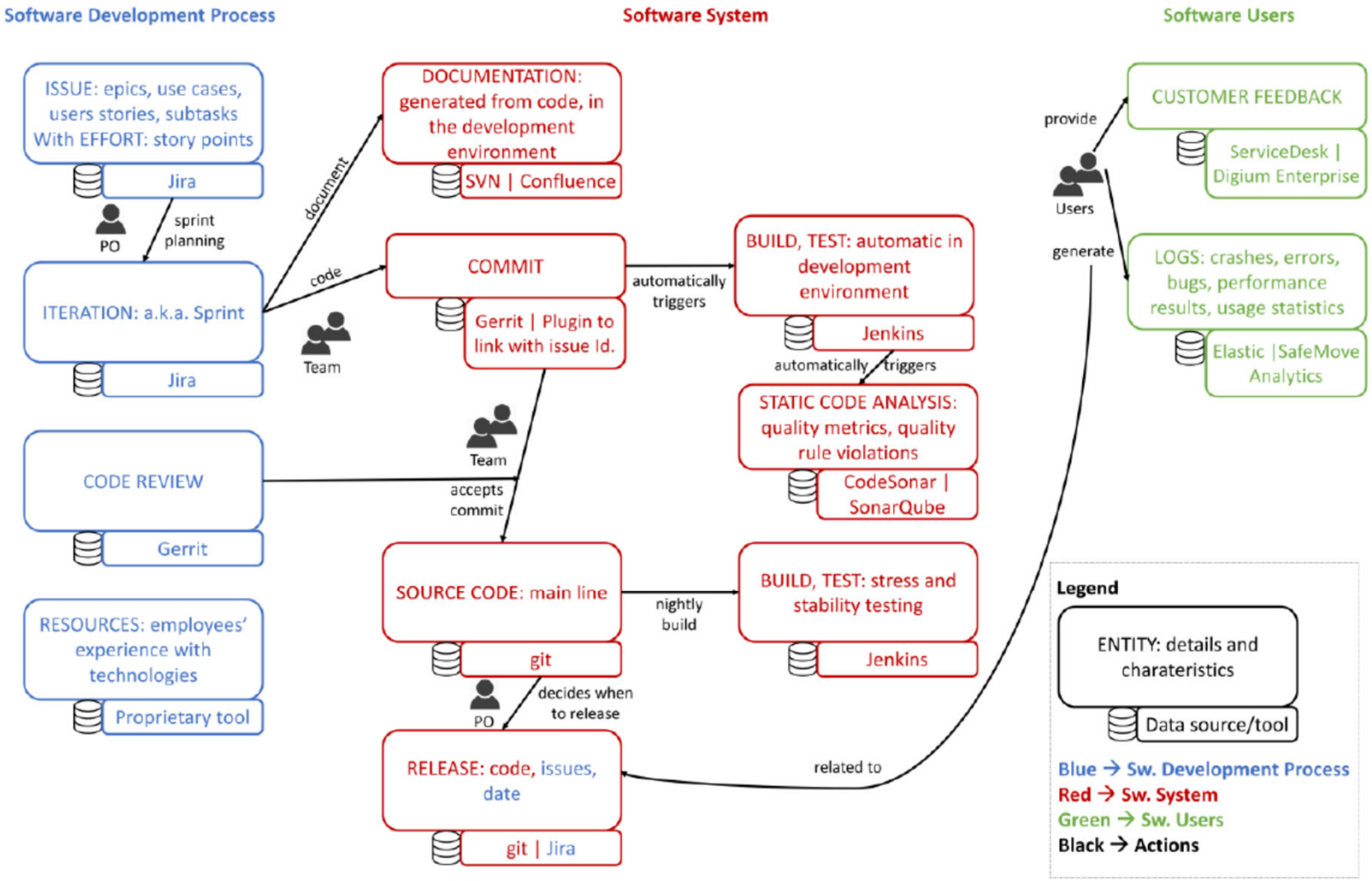
Conferences > 2018 44th Euromicro Conferenc...

## A Quality Model for Actionable Analytics in Rapid Software Development

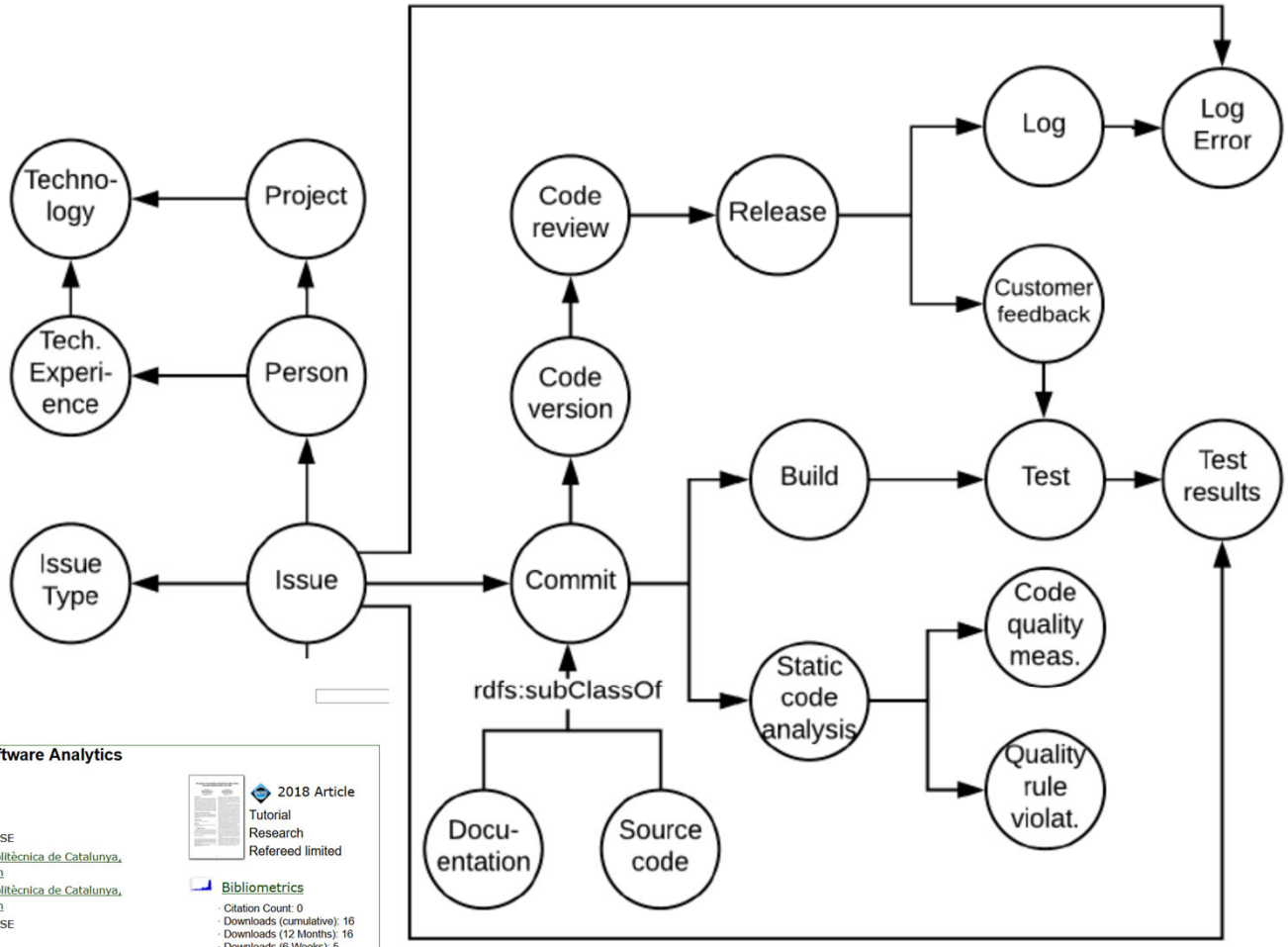
4 Author(s) Silverio Martínez-Fernández ; Andreas Jedlitschka ; Liliana Guzmán ; Anna Maria Vollmer View All Authors

UPC

# Complex ecosystem



# Data Integration ontology




ACM DL DIGITAL LIBRARY

**Towards Automated Data Integration in Software Analytics**

Full Text: PDF Get this Article

Authors:

| | |
|---|---|
| Silverio Martínez-Fernández | Fraunhofer IESE |
| Petar Jovanovic | Universitat Politècnica de Catalunya, BarcelonaTech |
| Xavier Franch | Universitat Politècnica de Catalunya, BarcelonaTech |
| Andreas Jedlitschka | Fraunhofer IESE |

2018 Article
Tutorial
Research
Refereed limited

Bibliometrics
- Citation Count: 0
- Downloads (cumulative): 16
- Downloads (12 Months): 16
- Downloads (6 Weeks): 5

Published in:
- Proceeding
BIRTE '18 Proceedings of the International Workshop on Real-Time Business Intelligence and Analytics
Article No. 6

# Example of application

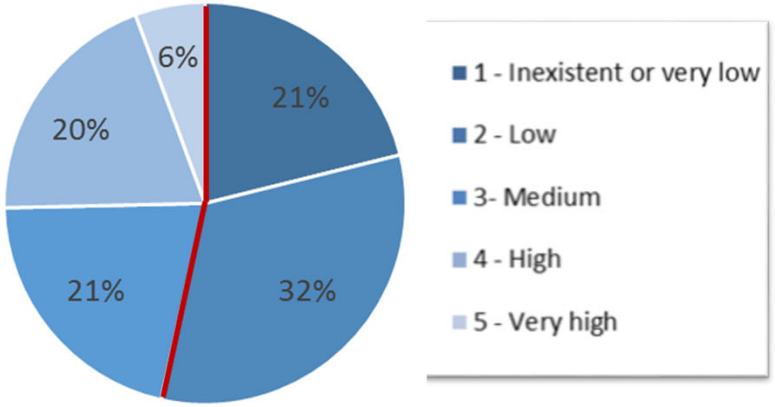

Managing Requirements Knowledge pp 95-116 | Cite as

## Constructing and Using Software Requirement Patterns

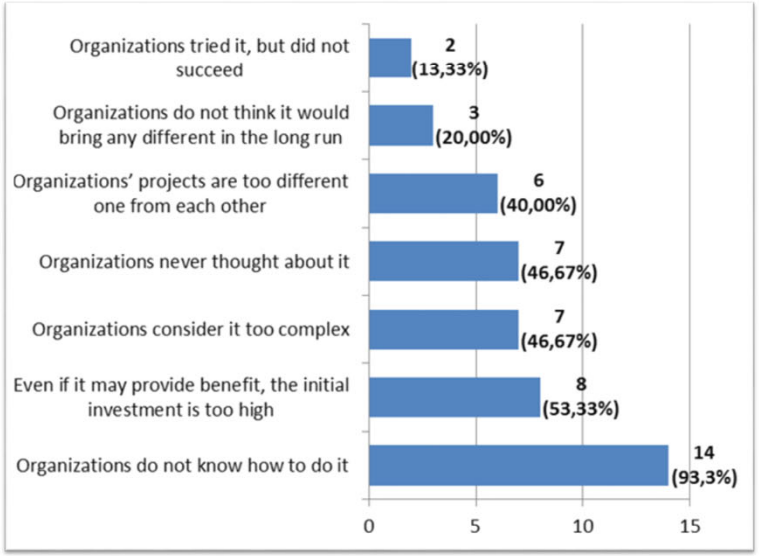
Authors Authors and affiliations

X. Franch, C. Quer, S. Renault, C. Guerlain, C. Palomares

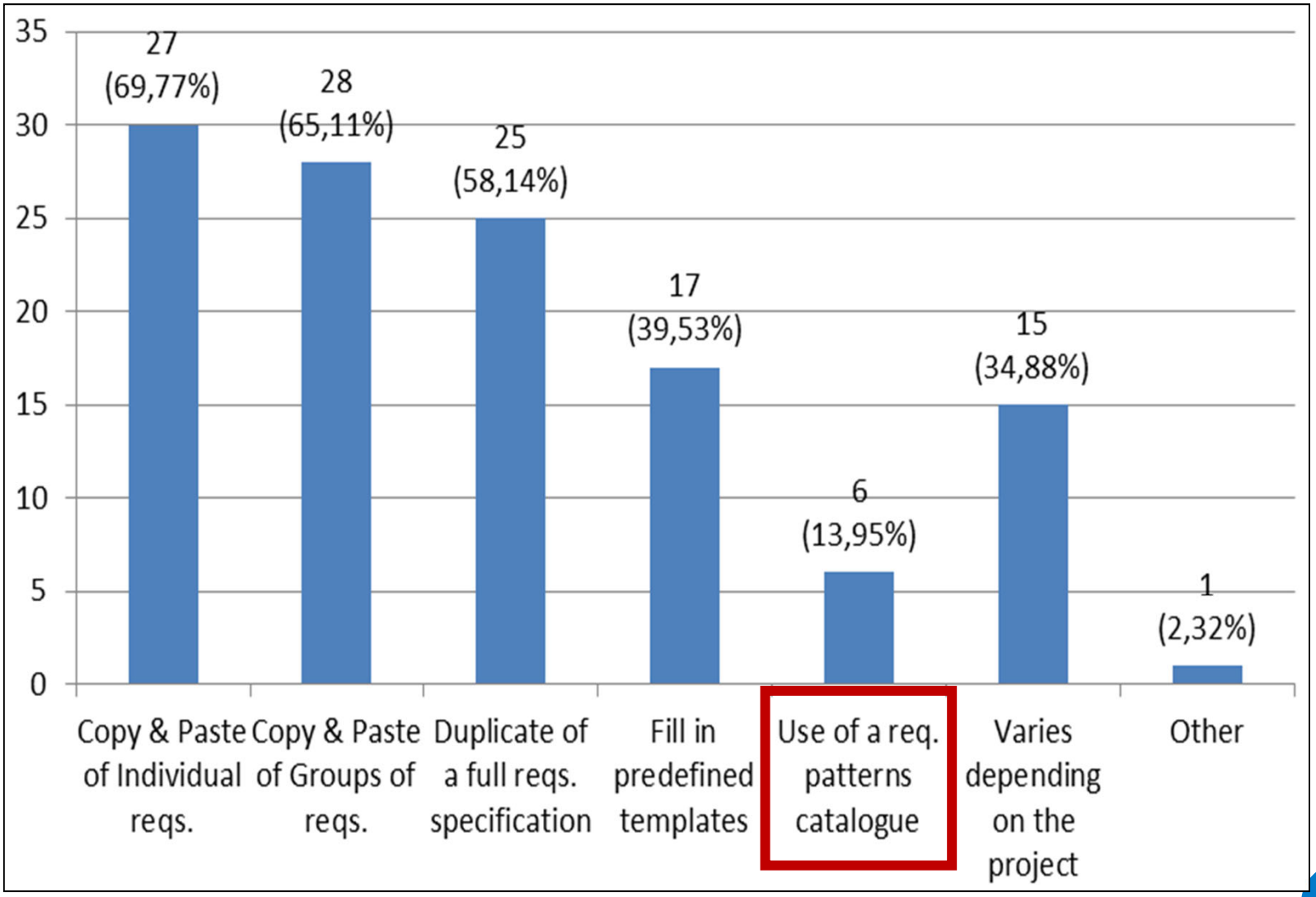
# Reuse adoption in industry



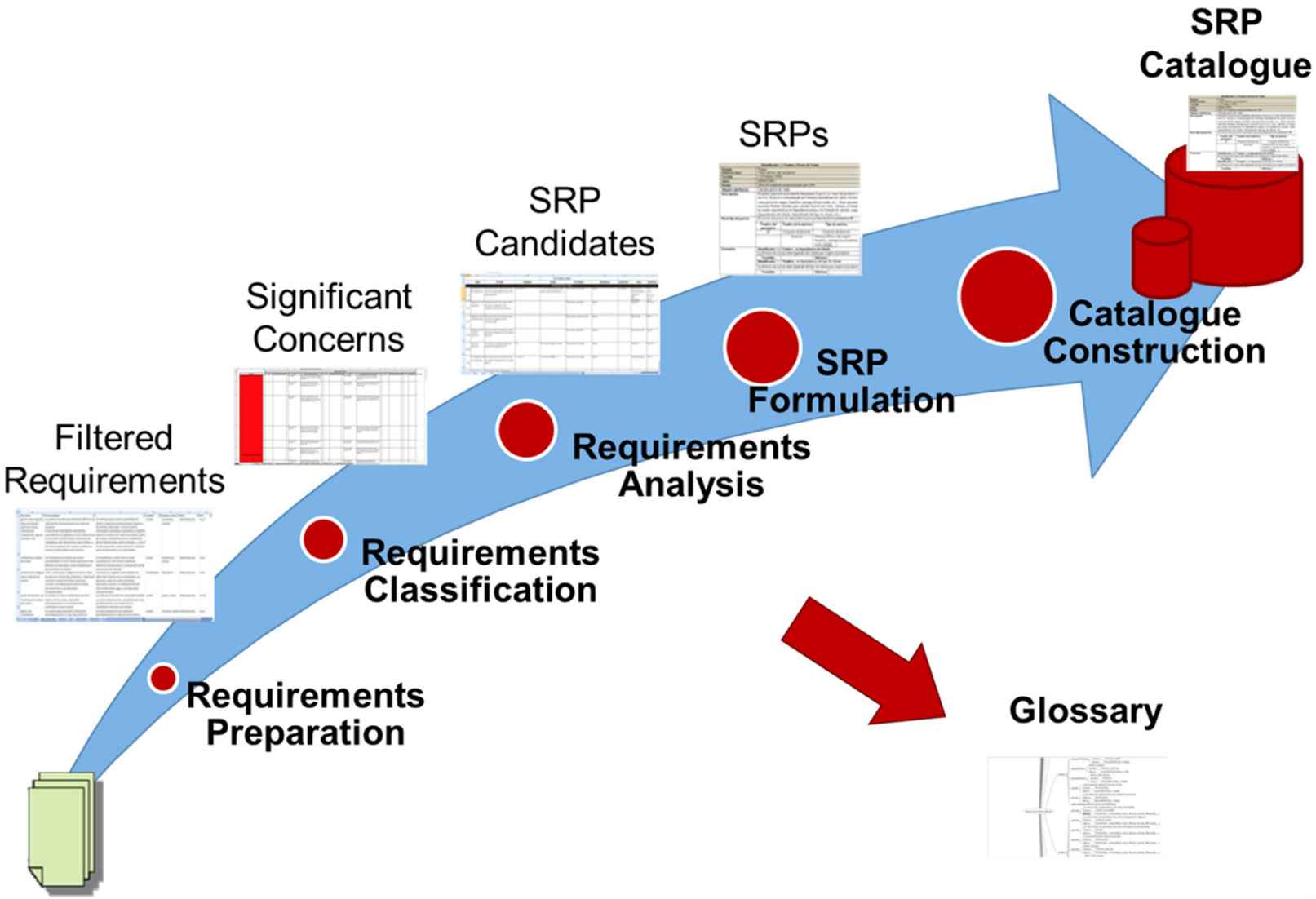




# Approaches to reuse



# Pattern approach to reuse



# Requirement patterns

## SRP: Supplier Economic Information

**Goal:** ***Assessing the economic situation of the supplier***

***Does the Customer requires specific conditions on the economic situation of the supplier?***

**No** — SRP Form

### Economic Situation Information Form

**SRP Parts**

- **Fixed**: The supplier shall provide economic information of its company.
- **Extended 1**: The supplier shall provide information of it's company's turnover on the last ***amountOfTime*** ***timeUnit***.
- **Extended 2**: The supplier shall provide information of its company's net income on the last ***amountOfTime*** ***timeUnit***.

**Constraints**

- **Fixed** part cannot be applied more than once.
- **Extended 1** and **Extended 2** parts cannot be applied more than once.

**Yes** — SRP Form

### Economic Situation Prerequisites Form

**SRP Parts**

- **Fixed**: The supplier shall fulfill some economic situation prerequisites.
- **Extended 1**: The supplier's company shall have a minimum net income of ***amount*** ***currencyUnit*** on the last ***amountOfTime*** ***timeUnit***.
- **Extended 2**: The supplier's company shall have a minimum turnover of ***amount*** ***currencyUnit*** on the last ***amountOfTime*** ***timeUnit***.

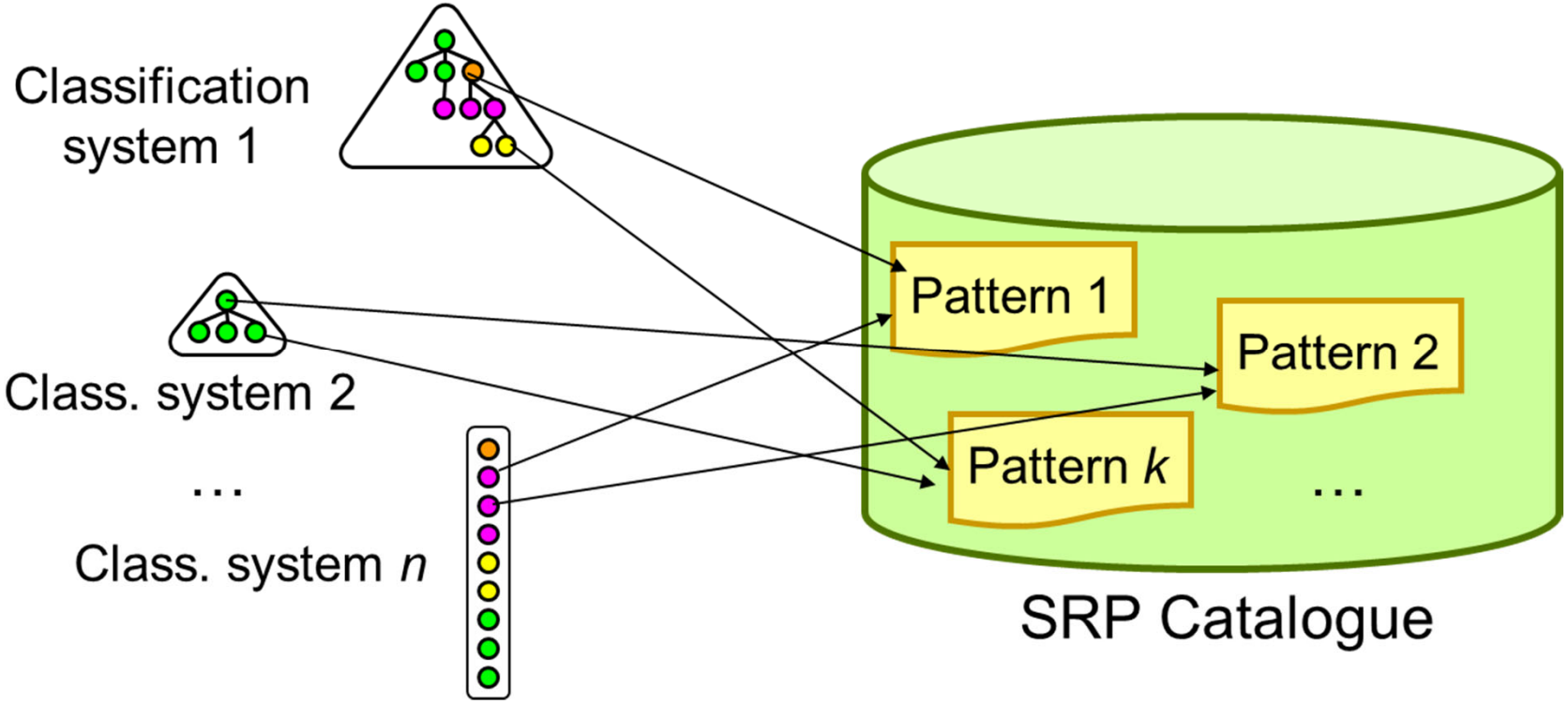
**Constraints**

- **Fixed** part cannot be applied more than once.
- **Extended 1** and **Extended 2** can be applied more than once if they are applied with disjoint values for the parameters (***amount***, ***amountOfTime***).
- **Soft Constraint**. **Extended 1** and **Extended 2** applications should use the same ***currencyUnit*** and ***timeUnit***.

### Parameter Metrics

**amount**, **amountOfTime**: integer (**inv** integer must be greater than 0)
**timeUnit**: domain {years, months, ...}
**currencyUnit**: domain {USD, EUR, GBP, JPV, ...}

# Use of the taxonomy



# Expressing an ontology

Different formalisms have been used:

OWL

SPARQL

RDF

DL

XML

*ad hoc*

SWRL

ER

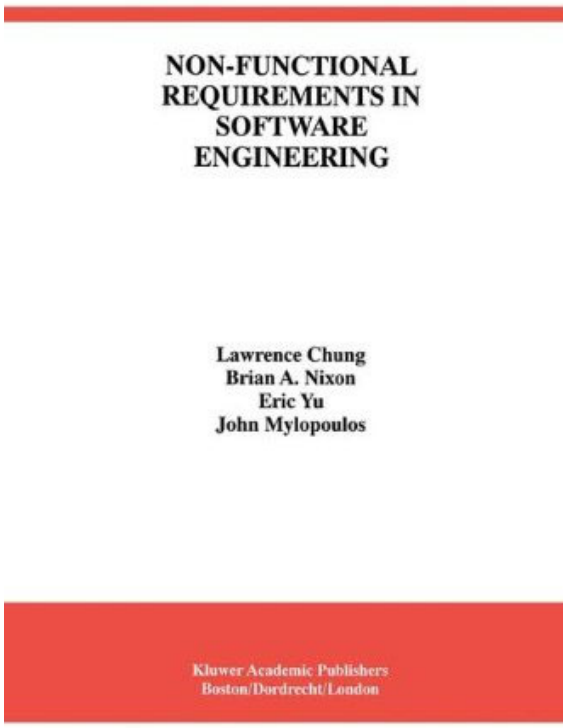
FOL

UML

# Example *ad hoc*: The NFR framework

The NFR framework was delivered in year 2000 after some efforts starting in early nineties:

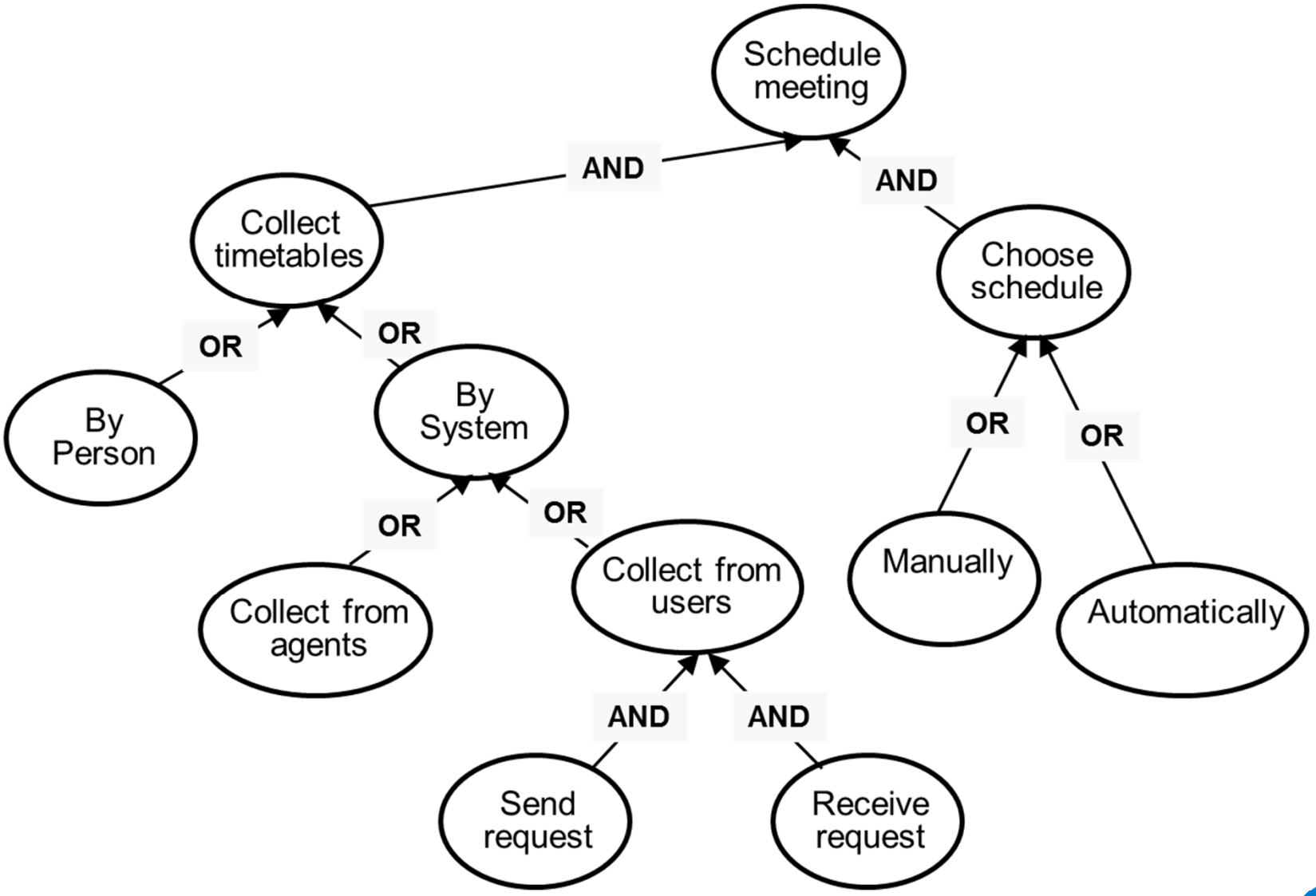
- created by J. Mylopoulos and team
- emerged in parallel to the *i** framework (Yu 1995)
- has become the most popular approach to NFR modeling and reasoning



# Main characteristics

- acquiring or accessing knowledge about the domain
  - functional requirements, priorities, …
- identifying particular NFRs for the domain
  - use of NFR type catalogues
- decomposing NFRs
- identifying feasible operationalizations, and selecting them
- dealing with ambiguities, conflicts and dependencies
- support decisions with design rationale
- evaluating the impact of decisions

# The basis: goal decomposition



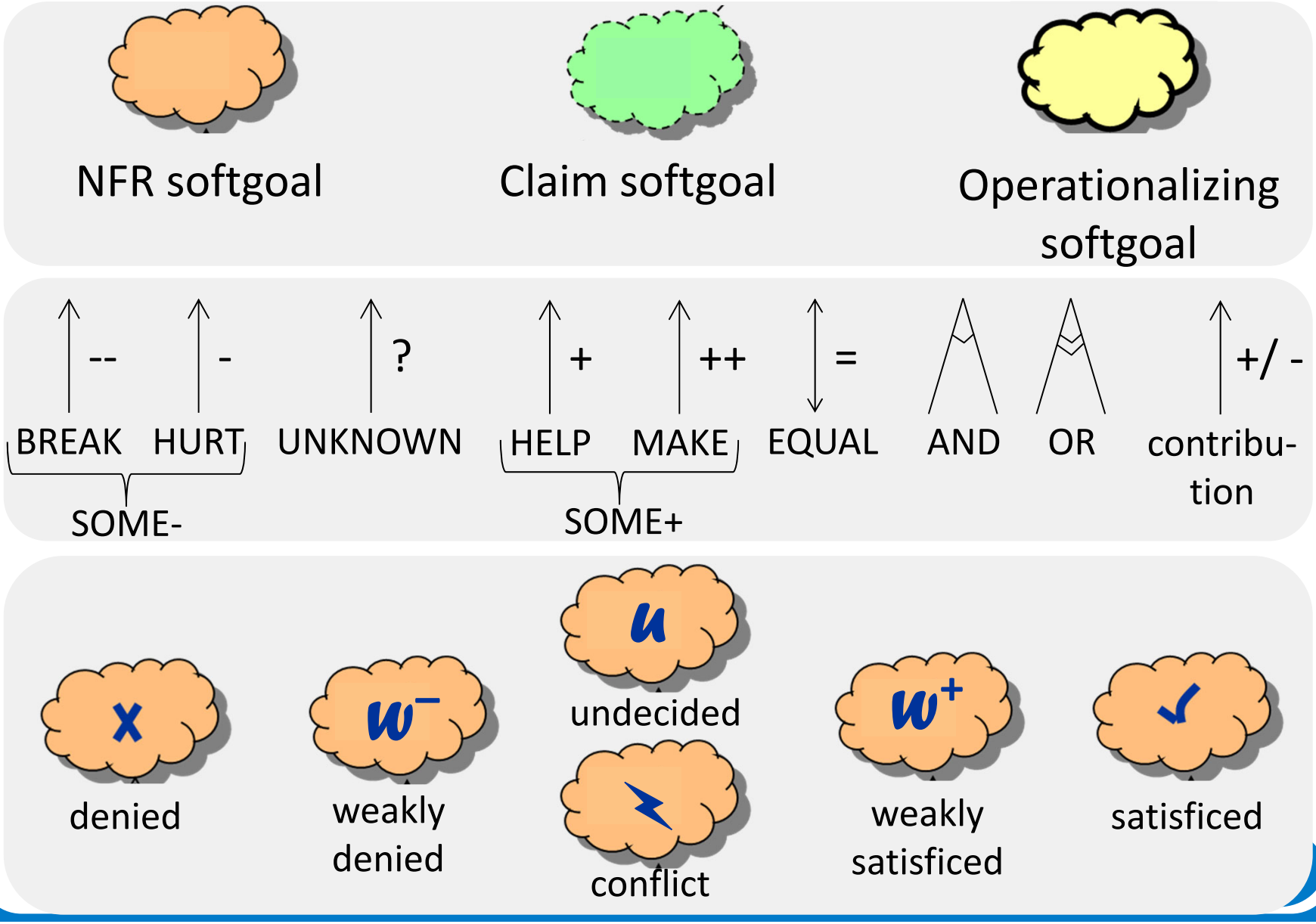
# Keypoint: notion of softgoal

- NFR Softgoal : high-level, non-operationalized, NFR
  - Softgoal can rarely be completely satisfied, hence goal is regarded satisfied (or satisfice) within acceptable limits
- Operationalizing Softgoal: possible solutions or design alternatives which help achieving the NFRs
- Claim Softgoal: justify the rationale and explain the context for a softgoal or interdependency link
- Softgoal consist of:
  - NFR Type (e.g. Security; Authentication)
  - One or more topic to indicate meaning and information item (e.g. [CardData]; [Account]).

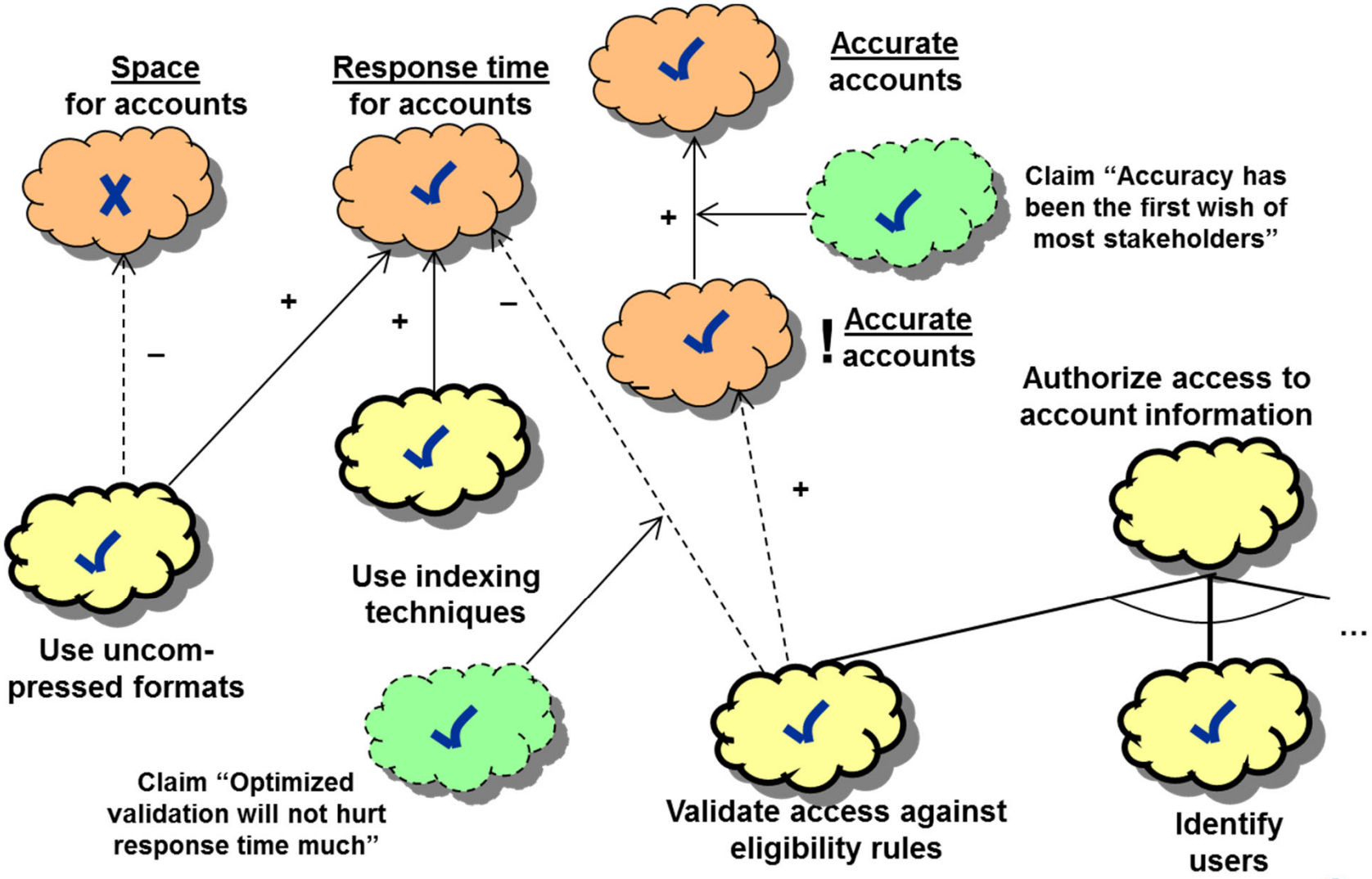
# Interdependency graph

- Interdependencies: refinement of softgoals and the contributions of offspring softgoals towards towards the achievement of its parent
- Softgoal Interdependency Graph (SIG): graph where softgoals and their interdependencies are represented

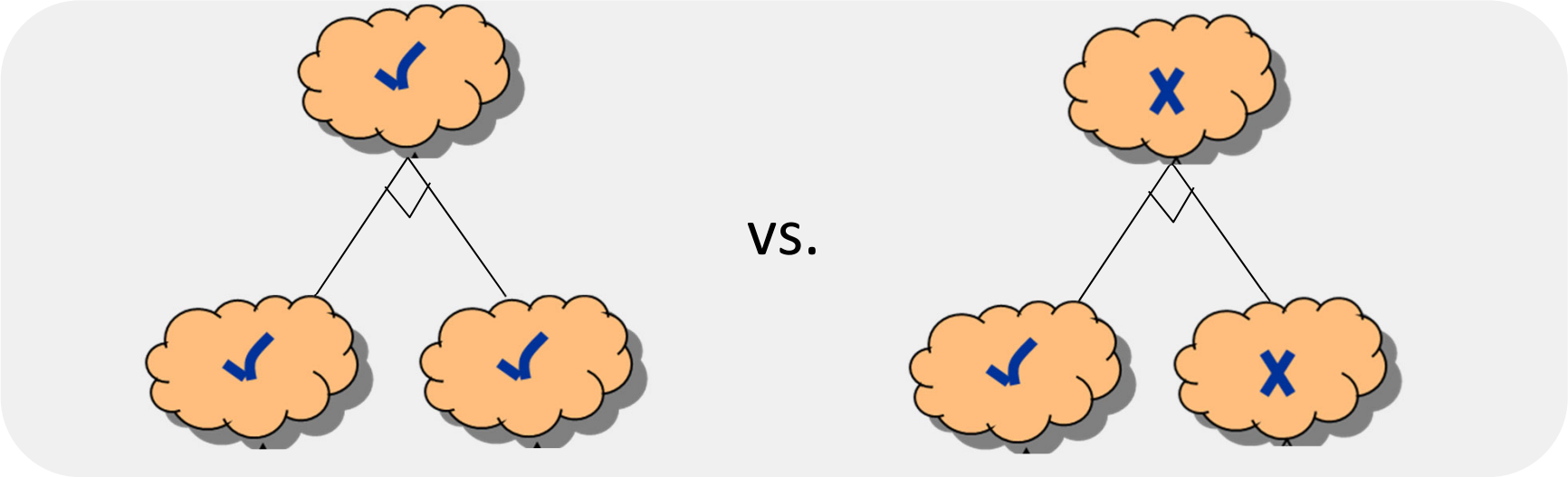
# Language

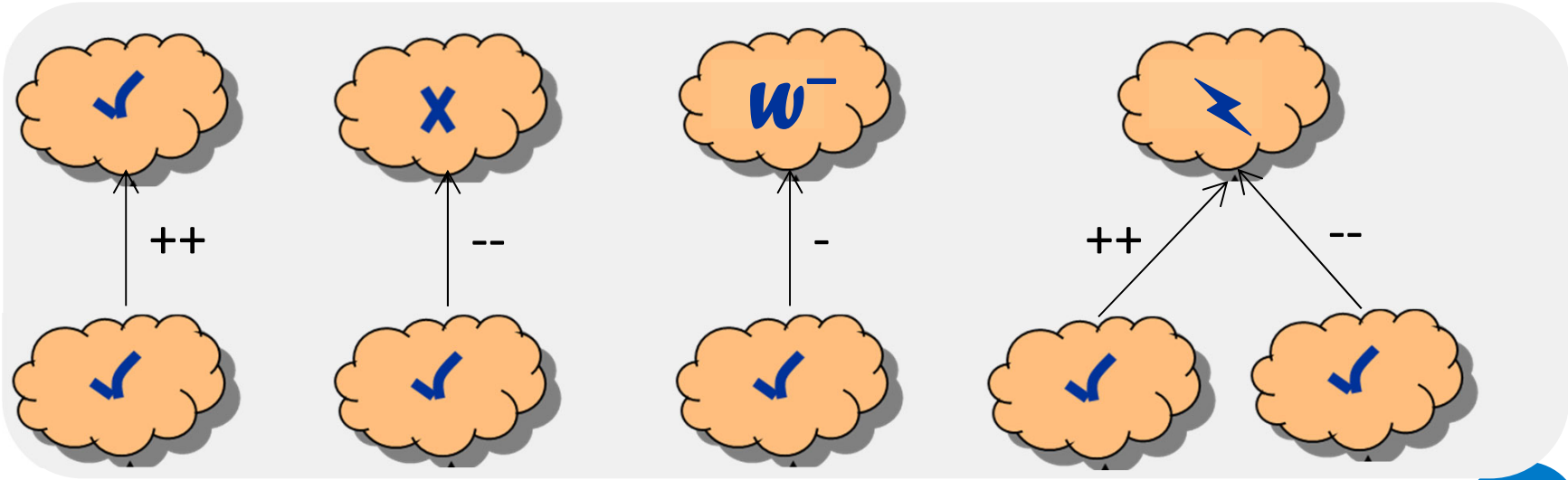

NFR softgoal
Claim softgoal
Operationalizing softgoal
--
-
?
+
++
=
+/ -
BREAK
HURT
UNKNOWN
HELP
MAKE
EQUAL
AND
OR
contribu-tion
SOME-
SOME+
x
w−
u
w+
✓
denied
weakly denied
undecided
conflict
weakly satisficed
satisficed

# Example of model



# Evaluation procedure (excerpt)




(Chung et al. 2000)

# Summary



SERC TALKS

# Upcoming Events

SYSTEMS ENGINEERING RESEARCH CENTER

SERC TALKS

# UPCOMING TALKS: 2019

**Themes:**

February, April, June 2019:

***Continuous Delivery and Deployment of Cyber-Physical-Human Systems***

August, October, December 2019:

***Dependable Autonomy***

Please visit the SERC Talks page for more information and updates.

SYSTEMS ENGINEERING RESEARCH CENTER

SERC TALKS

# Thank you for joining us!

Please check back on the SERC website for today's recording and future SERC Talks information!

Subscribe and follow SERC on these channels:

**CONTACT**

**Editor-in-Chief: Dr. Barry Boehm, University of Southern California – boehm@usc.edu**

**Webinar Coordinator: Ms. Mimi Marcus, Stevens Institute of Technology – SERCtalks@stevens.edu**