Outline

WQA: State of the art

The SMM

Empirical Validation of QM

An Integrated TQM Approach

WE TQM Automation

Conclusions
Web Engineering is gaining momentum among the research community
- RIA interfaces, process management, security, architecture, etc. are concerns that have been introduced in the WE community in the last couple of years

There are mature methodology proposals that come together with stable tools (OO-H, WebML)
- DEMO WebRATIO
However, the use of WE methodologies and tools is scarce in industry (Lang 2005).

In addition to learning curve and lack of communication between both communities, we do think that the WE community is failing to ‘prove’ practitioners that the use of methodologies bring in fact advantages in terms of cost and quality of the deployed application.

WE proposals have paid little attention to Quality in Use concerns, despite the fact that they are a continuous concern for Web developers, due to the necessity for this kind of applications to keep the audience coming back to the site [Fraternally and Paolini 2000].

Quality in use [ISO 9126]: efficiency, productivity, security and satisfaction with which users use the application to satisfy specific goals under specific conditions.
This goes in accordance with the traditional Web Quality assessment perspective, which consists on performing it once the WebApp has already been deployed (mass inspection, automated measures).

Before the advent of WE that was only logical, as creative approaches do not provide intermediate products.
The cost associated with fixing detected usability problems once the application has already been deployed can be up to 170 times greater than the cost of fixing the same problem had it been identified during the requirements phase [Moody 2003].

So, can we change this late assessment approach?

According to ISO and some empirical evidence, we can.

ISO: Several internal characteristics of the Web application are bound to influence this end-user quality perception, namely usability, functionality, reliability and efficiency

- We leave out maintainability and portability, relevant for other kinds of stakeholders

[Ivory and Hearst, 2001]: Quality assessment of Web interfaces with the help of internal measures on implementation artifacts (web pages) matches in some cases up to 80% of the results based on expert evaluation of the same Web pages
WE artifacts permit to assess quality at higher levels of abstraction
- From requirements to implementation
- Such assessment could even be extended to meta-model constructs
In order to perform such early quality assurance, we need to know **why**, **what**, **how** and **when** measuring at each level of abstraction... we need a WE Quality Model (QM) and a Quality Evaluation Process

- Quality Model [ISO 9126]: set of characteristics and the relationships between them which provide the basis for specifying quality requirements and evaluating quality

- Software Quality Evaluation Process [ISO 14198]: set of activities that must be carried out in order to evaluate software quality, together with the evaluation modules that help in the process
There are many standards that may help us to define a Quality Model to evaluate software products

- Product quality assurance: ISO 9126, ISO 9248, ISO 14102
- Some well-known quality models that make use of the ISO 9126 standard include
  - QUINT2 (Niessink, 2002): evaluation of software architectures
  - Franch and Carvallo (2003): evaluation of email systems
  - Botella et al. (2003): ERP systems selection
  - Díaz et al. (2004): Portlets usability
  - ... 
- However, most Quality Models specially devoted to the Web define their set of characteristics and subcharacteristics from scratch
  - WQM (Calero et al), PQM (Moraga et al), PDQM (Caro et al), BPQM (Cordoba et al.)
Existing QM proposals present some problems

- Terminology inconsistencies
- Missing elements
  - *e.g.* decision criteria, measure scales and units, ...
- Tangled coverage of concepts
  - Different stakeholders and different types of products not explicitly distinguished.
- Too cumbersome to apply (too many concepts and measures)
- Interdependencies between quality factors and/or measures not empirically established
  - coupling vs cohesiveness, learnability vs efficiency, ...
- Disregard for process quality
- Lack of integration with current WE practices
- Lack of tool support
There is no silver-bullet:
- Terminology inconsistencies
- Missing elements
  - e.g. decision criteria, measure scales and units, ...
- Tangled coverage of concepts
  - Different stakeholders and different types of products not explicitly distinguished.
- Too cumbersome to apply (too many concepts and measures)
- Interdependencies between quality factors and measures not empirically established
- Disregard for process quality
- Lack of integration with current WE practices
- Lack of tool support

WE-QM as instantiations of an Ontology-based Software Measurement Meta-Model with additional restrictions...
There is no silver-bullet:

- Terminology inconsistencies
- Missing elements: e.g., decision criteria, measurement scales
- Tangled coverage of concepts
  - Different stakeholders and different types of products not explicitly distinguished.
- Too cumbersome to apply (too many concepts and measures)
- Interdependencies between quality factors and/or measures not empirically established
- Disregard for process quality
- Lack of integration with current WE practices
- Lack of tool support

... one of such additional restrictions being that WE-QM must be devoted to a single stakeholder, another being that in WE-QM each EntityClass (type of product) represents a different WE artifact.
There is no silver bullet:

- Terminology inconsistencies
- Missing elements e.g. decision criteria, measure scales and units, …
- Tangled coverage of concepts
- Different stakeholders and different types of products not explicitly distinguished.
- Too cumbersome to apply (too many concepts and measures)
- Interdependencies between quality factors and/or measures not empirically established
- Disregard for process quality
- Lack of integration with current WE practices
- Lack of tool support

QM must be empirically validated to
✓ assure necessity, sufficiency and independence of QM concepts
✓ assure necessity, sufficiency and minimality of measures
✓ assess measures interdependences
Part 1: TQM approach for WE App development

- There is no silver-bullet:
  - Terminology inconsistencies
  - Missing elements
    - *e.g. decision criteria, measures*
  - Tangled coverage of concepts
    - *Different stakeholders and differences explicitly distinguished.*
  - Too cumbersome to apply (too many concepts and measures)
  - Interdependencies between quality factors and/or measures not empirically established
  - Disregard for process quality
  - Lack of integration with current WE practices
  - Lack of tool support
There is no silver-bullet:

- Terminology inconsistencies
- Missing elements
  - e.g. decision criteria, measure scales and units
- Tangled coverage of concepts
  - Different stakeholders and different types of products not explicitly distinguished.
- Too cumbersome to apply (too many concepts and measures)
- Interdependencies between quality factors and measures not empirically established
- Disregard for process quality
- Lack of integration with current WE practices
- Lack of tool support

Automation of the approach should be provided (also to maintain the semi-automatic nature of the WE process): follow the MDE paradigm.
Outline

- WQA: State of the art
- The SMM
- Empirical Validation of QM
- An Integrated TQM Approach
- WE TQM Automation
- Conclusions
A WE-SMM helps us to reduce:

- Terminology inconsistencies
- Missing elements
  - e.g. decision criteria, measure scales and units, ...
- Tangled coverage of concepts
- Different stakeholders and different types of products not explicitly distinguished.

Too cumbersome to apply (too many concepts and measures)

Interdependencies between quality factors not empirically established

Disregard for process quality

Lack of integration with current WE practices

Lack of tool support

WE-QM as instantiations of an Ontology-based Software Measurement Meta-Model with additional restrictions...

... one of such additional restrictions being that WE-QM must be devoted to a single stakeholder, another being that in WE-QM each EntityClass (type of product) represents a different WE artifact
The Software Measurement Meta-Model

GENERAL OVERVIEW

- Characterization and Objectives
- Measurement
- Measures
- Measurement approaches

Basic
### Characterization and Objectives

**Quality Model**
- **kind**: 1..n evaluates
- **defined for**: n

**Measurable Concept**
- **from Characterization and Objectives**: 1..n

**Entity Class**
- **has**: 1..n
- **belongs to**: 0..n includes

**Entity**
- **composed of**: 0..n

**Information Need**
- **from Characterization and Objectives**: 0..1

**Measurable Concept**
- **includes**: 0..1

**Attribute**
- **from Characterization and Objectives**: 1..n

### Terms and Definitions

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information Need</td>
<td>Insight necessary to manage objectives, goals, risks, and problems</td>
</tr>
<tr>
<td>Measurable Concept</td>
<td>Abstract relationship between attributes of entities and information needs</td>
</tr>
<tr>
<td>Entity</td>
<td>Object that is to be characterized by measuring its attributes</td>
</tr>
<tr>
<td>Entity Class</td>
<td>The collection of all entities that satisfy a given predicate</td>
</tr>
<tr>
<td>Attribute</td>
<td>A measurable physical or abstract property of an entity, that is shared by all the entities of an entity class</td>
</tr>
<tr>
<td>Quality Model</td>
<td>The set of measurable concepts and the relationships between them which provide the basis for specifying quality requirements and evaluating the quality of the entities of a given entity class</td>
</tr>
</tbody>
</table>
### SOFTWARE MEASURES

#### The Software Measurement Meta-Model

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measure</td>
<td>The defined measurement approach and the measurement scale. (A measurement approach is either a measurement method, a measurement function or an analysis model)</td>
</tr>
<tr>
<td>Scale</td>
<td>A set of values with defined properties</td>
</tr>
<tr>
<td>Type of Scale</td>
<td>The nature of the relationship between values on the scale</td>
</tr>
<tr>
<td>Unit of Measurement</td>
<td>Particular quantity, defined and adopted by convention, with which other quantities of the same kind are compared in order to express their magnitude relative to that quantity</td>
</tr>
<tr>
<td>Base Measure</td>
<td>A measure of an attribute that does not depend upon any other measure, and whose measurement approach is a measurement method</td>
</tr>
<tr>
<td>Derived Measure</td>
<td>A measure that is derived from other base or derived measures, using a measurement function as measurement approach</td>
</tr>
<tr>
<td>Indicator</td>
<td>A measure that is derived from other measures using an analysis model as measurement approach</td>
</tr>
</tbody>
</table>

#### Diagram

![Diagram](image.png)
### The Software Measurement Meta-Model

#### MEASUREMENT APPROACHES

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measurement Method</td>
<td>Logical sequence of operations, described generically, used in quantifying an attribute with respect to a specified scale. (A measurement method is the measurement approach that defines a base measure)</td>
</tr>
<tr>
<td>Measurement Function</td>
<td>An algorithm or calculation performed to combine two or more base or derived measures. (A measurement function is the measurement approach that defines a derived measure)</td>
</tr>
<tr>
<td>Analysis Model</td>
<td>Algorithm or calculation combining one or more measures with associated decision criteria. (An analysis model is the measurement approach that defines an indicator)</td>
</tr>
<tr>
<td>Decision Criteria</td>
<td>Thresholds, targets, or patterns used to determine the need for action or further investigation, or to describe the level of confidence in a given result</td>
</tr>
</tbody>
</table>
## The Software Measurement Meta-Model

### MEASUREMENT

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measurement Approach</td>
<td>Sequence of operations aimed at determining the value of a measurement result. (A measurement approach is either a measurement method, a measurement function or an analysis model)</td>
</tr>
<tr>
<td>Measurement</td>
<td>A set of operations having the object of determining the value of a measurement result, for a given attribute of an entity, using a measurement approach</td>
</tr>
<tr>
<td>Measurement Result</td>
<td>The number or category assigned to an attribute of an entity by making a measurement</td>
</tr>
</tbody>
</table>
Additional Restrictions on QM for WE

Need: Avoid missing concepts

Solution example:

Every measure must have a unit of measurement
Additional Restrictions on QM for WE

- **Need:** Keep the Quality Model simple and focused
- **Solutions:**
  - One WE QM for each stakeholder
    - **Customers** (quality as fulfillment of needs: end product)
    - **Analysts/Designers** (quality as conformance to specification: intermediate products - WE models)
    - **Developers/Maintainers** (quality as conformance to specification: intermediate products - App code)
    - **Final Users** (quality as fulfillment of needs: end product)
  - Each stakeholder-dependent WE QM divided into WE QM for each WE artifact (Product Type).
Additional Restrictions on QM for WE

- **Need**: Keep the Quality Model simple and focused
- **More Solutions**:
  - Only two levels of abstraction permitted in measurable concepts related to a given WE QM.
    - Plus one ‘context’ level that serves to establish the general goal of the quality model by means of the GQM template [Basili Rombach 2003].
  - Attributes for the different models with unique names
    - Name convention: attribute name=initials of model on which it is being measured (RM,DM, etc.)+attribute.
      - E.g. NM_StructuralComplexity
  - Every measurable concept connected to 0..1 information need.
  - Each information need satisfied by a single indicator.
  - Only measurable concepts of 1st level associated to information needs.

...
## The Software Measurement Meta-Model

### EXAMPLE: NAVIGATION USABILITY MODEL

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Subcharacteristic</th>
<th>Subsubcharacteristic</th>
<th>Structural Complexity</th>
<th>Consistency</th>
<th>Semantic Correctness</th>
<th>Grouping Density</th>
<th>Grouping Cohesiveness</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>NPC (Navigational pattern coherence)</td>
<td>Number of racing conditions in automatic links</td>
<td>Number of abstract pages</td>
</tr>
<tr>
<td>Readability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Completeness of Description (iso 9126-3)</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Familiarity</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>Brevity</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td>NO</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Learnability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Predictability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memorability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Controllability</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td></td>
<td>Number of manual service links</td>
<td>NO</td>
</tr>
<tr>
<td>Capability of Personalization (adaptability/adaptivity)</td>
<td></td>
<td></td>
<td></td>
<td>NO</td>
<td></td>
<td>Number of links with adaptation rules</td>
<td>NO</td>
</tr>
</tbody>
</table>
EXAMPLE: SMM INSTANTIATION

TQM approach for WE App development
An Integrated TQM Approach

Empirical Validation of QM
Measurable Concepts and Attributes in the WE QM must be sufficient, necessary and independent.

Measures in the WE QM must be empirical validated.

Measures must be sufficient, necessary and minimal.

There must be empirical evidence of the connections among concepts and/or between concepts and measures.

There must be empirical evidence of the connections between the quality of intermediate products and the quality in use of the final product (running application).
Outline

- WQA: State of the art
- The SMM
- Empirical Validation of QM
- An Integrated TQM Approach
- WE TQM Automation
- Conclusions

TQM approach for WE App development
TQM approach for WE App development
TQM approach for WE App development

OBJECTIVE: USABILITY IN USE
TQM approach for WE App development

WORKFLOWS

Requirements

CIM

Analysis

PIM

Conceptual Design

Requirements

Model2Model Transformation

T1

:Requirements Model

:Requirements Measurement Model

:Requirements Quality Model

Analysis

Conceptual Design

Requirements

Model2Model Transformation

T2

:Domain Model

:Domain Measurement Model

:Domain Quality Model

Detailed Design

PSM

Model2Model Transformation

T3

:Navigation Model

:Navigation Measurement Model

:Navigation Quality Model

Model2Model Transformation

T4

:Presentation Model

:Presentation Measurement Model

:Presentation Quality Model

Model2Text Transformation

T5

:Model for J2EE Platform

:Model for .NET Platform

OBJECTIVE:

USABILITY IN USE
An Integrated TQM Approach

CORRELATION BETWEEN ISO 14598 AND WE TQM PROCESS

- Additional advantage: our approach complies with ISO 14598-1 standard
  - Specification and evaluation of software conforms to ISO 14598 if it uses the process in clause 6 and a Quality Model…

1. Establish evaluation requirements
2. Specify the evaluation
3. Design the evaluation
4. Execute the evaluation
An Integrated TQM Approach

CORRELATION BETWEEN ISO 14598 AND WE TQM PROCESS

Establish evaluation requirements

Specify the evaluation

Design the evaluation

Execute the evaluation

Establish purpose of evaluation

Identify types of product(s)

Specify quality model

Quality Characteristics (in WE Quality Model)

Instantiate each WE Quality Model with the subset of relevant measurable concepts among those defined for each of the outgoing WE artifacts

QQM: analyzing the different WE Artifacts for the purpose of evaluating it with respect to the external quality of the Web application from the viewpoint of the end-user of the application in the context of testing environments:
- Requirements Model
- Domain Model
- Navigation Model
- Presentation Model
- Implementation Model
- Deployable code

TQM approach for WE App development
An Integrated TQM Approach

CORRELATION BETWEEN ISO 14598 AND WE TQM PROCESS

- Establish evaluation requirements
- Specify the evaluation
- Design the evaluation
- Execute the evaluation

Select measures

Establish decision criteria for measures

Establish indicators for assessment

Instantiate each WE Quality Model with the subset of relevant measures among those defined for each of the outgoing WE artifacts

Establish in the Instantiation of each WE Quality Model decision criteria according to the application family

Establish in the Instantiation of each WE Quality Model decision criteria according to the application family and the chosen measurable concepts

TQM approach for WE App development
An Integrated TQM Approach

CORRELATION BETWEEN ISO 14598 AND WE TQM PROCESS

1. Establish evaluation requirements
2. Specify the evaluation
3. Design the evaluation
4. Execute the evaluation

Produce evaluation plan

Evaluate each artifact as soon as it is generated in the development cycle

TQM approach for WE App development
An Integrated TQM Approach

CORRELATION BETWEEN ISO 14598 AND WE TQM PROCESS

- Establish evaluation requirements
- Specify the evaluation
- Design the evaluation
- Execute the evaluation
- Take measures
- Compare with decision criteria
- Assess results

Coded in Evaluation Transformation Rules
Coded in Evaluation Transformation Rules
Coded in Evaluation Transformation Rules

TQM approach for WE App development
TQM approach for WE App development
TQM approach for WE App development

WORKFLOWS

Requirements

Analysis

Conceptual Design

Detailed Design

Implementation

MDA

CIM

PIM

PSM

CODE

WEB APPLICATION

TQM approach for WE App development

OBJECTIVE:

USABILITY IN USE
WE TQM Automation

EXAMPLE: OO-H DOMAIN MODEL

Library
- name: String
  - addPublication

Publication
- title: String
- date: Date
  - addAuthor
  - addArticle
  - addKeyword

Author
- name: String
- e-mail: String

Article
- title: String
- abstract: String
  - complete: URL
  - associateAuthor
  - associateKeyword

Keyword
- word: String

Library
- library
- l2p
- publications

Author
- authors

Article
- authors
- articles

Keyword
- keywords

TQM approach for WE App development
Designing and maintaining Web applications is one of the major challenges for the software industry of the year 2000. In this paper we present Web Modeling Language (WebML), a notation for specifying complex Web sites ...
TQM approach for WE App development

Designing and maintaining Web applications is one of the major challenges for the software industry of the year 2000. In this paper, we present Web Modeling Language (WebML), a notation for specifying complex Web sites...

Conceptual Modeling of Device-Independent Web Applications

...Authors

1. Jaime Gómez
   jgomez@dsi.ua.es

2. Cristina Cachero
   ccachero@dsi.ua.es

3. Oscar Pastor
   opastor@dsic.upv.es

Keywords

MDD Articles
Navigability Articles
WebE Articles
...

Publications

IEEEMultimedia Articles
IWWOST Articles

Authors

1. Jaime Gómez
   jgomez@dsi.ua.es

2. Cristina Cachero
   ccachero@dsi.ua.es

3. Oscar Pastor
   opastor@dsic.upv.es

TQM approach for WE App development
Imagine that we have empirically validated that the DCNM measure is a valid indicator of the final usability of the navigation structure of the Web Application.

- Percentage of domain relationships which, having already been defined as the conceptual relationships in which a certain user type is interested, can in actual fact be navigated by such a user.

- Users may expect to find in the Web application the same relationships that exist among concepts in the problem space.
  - Not finding these relationships in the application may therefore diminish their general satisfaction with the application.
  - Users are likely to describe this phenomenon as a problem with the navigability of the application.
### Example: Navigation Usability Model

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Subcharacteristic</th>
<th>SubsubCharacteristic</th>
<th>Structural Complexity</th>
<th>Consistency</th>
<th>Semantic Correctness</th>
<th>Grouping Density</th>
<th>Grouping Cohesiveness</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Usability</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Understandability</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Readability</strong></td>
<td></td>
<td></td>
<td>NO</td>
<td>NPC (Navigational pattern coherence)</td>
<td>Number of racing conditions in automatic links</td>
<td>Number of abstract pages</td>
<td>% of origin links that are supported by an structural filter</td>
</tr>
<tr>
<td><strong>Completeness of Description</strong></td>
<td></td>
<td>NSDC: Number of supporting domain classes</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td><strong>Familiarity</strong></td>
<td></td>
<td>DCNM</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>% of origin links that are supported by an structural filter</td>
</tr>
<tr>
<td><strong>Brevity</strong></td>
<td></td>
<td>Depth of a Navigational Map</td>
<td>NO</td>
<td>NO</td>
<td></td>
<td>Number of links in origin</td>
<td>NO</td>
</tr>
<tr>
<td><strong>Learnability</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Predictability</strong></td>
<td></td>
<td>Number of links with Structural Filters</td>
<td>NPC (Navigational pattern coherence)</td>
<td>Number of racing conditions in automatic links</td>
<td>NO</td>
<td>% of origin links that are supported by an structural filter</td>
<td></td>
</tr>
<tr>
<td><strong>Memorability</strong></td>
<td></td>
<td>Number of links with Structural Filters</td>
<td>NPC (Navigational pattern coherence)</td>
<td>Number of racing conditions in automatic links</td>
<td>Number of links in origin</td>
<td>% of origin links that are supported by an structural filter</td>
<td></td>
</tr>
<tr>
<td><strong>Operability</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Controllability</strong></td>
<td></td>
<td>NO</td>
<td>Operation consistency</td>
<td>Number of manual service links</td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td><strong>Capability of Personalization</strong></td>
<td></td>
<td>Number of links with adaptation rules</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
</tbody>
</table>
Integration of usability models in WE practices:

Measurement Models are new models in WE. An example of an instantiation of a TQM approach for WE app development is shown below:

**EXAMPLE: SMM INSTANTIATION**

- **WebConceptualModel:** EntityClass
  - Description: "Abstract model that comprises all PIMs of a Web Application"

- **DomainModel:** EntityClass
  - Uses: **Nav_L**: Indicator
    - Description: "Nav_L"
    - Calculated with:
      - **AM_NavL**: AnalysisModel
        - Description: "AM_NavL"
      - **DC_NavL**: DecisionCriteria
        - Description: "DCNM, DCNM_L=Not Acceptable, DCNM>80 -> DCNM_L=Acceptable"

- **NavigationalModel:** EntityClass
  - Uses: **StructuralComplexityNM**: Attribute
    - Defined for:
      - **StructuralComplexityDM**: Attribute
        - Defined for:
          - **DomainModel**: EntityClass
            - Includes:
              - **Nav_L**: Indicator
                - Defined for:
                  - **AM_NavL**: AnalysisModel
                    - Description: "AM_NavL"
                  - **DC_NavL**: DecisionCriteria
                    - Description: "DCNM, DCNM_L=Not Acceptable, DCNM>80 -> DCNM_L=Acceptable"

- **StructuralComplexityNM Attribute**
  - Defined for:
    - **DomainModel**: EntityClass
      - Includes:
        - **Nav_L**: Indicator
          - Defined for:
            - **AM_NavL**: AnalysisModel
              - Description: "AM_NavL"
              - Uses:
                - **DC_NavL**: DecisionCriteria
                  - Description: "DCNM, DCNM_L=Not Acceptable, DCNM>80 -> DCNM_L=Acceptable"

- **DC_NavL**: DecisionCriteria
  - Description: "DCNM, DCNM_L=Not Acceptable, DCNM>80 -> DCNM_L=Acceptable"
  - Uses:
    - **DCNM**: Derived Measure
      - Description: "DCNM=NRR*(NDR)*100"
      - Defined for:
        - **DCNMFunction**: MeasurementFunction
          - Description: "(NDR/NRR)*100"
WE TQM Automation

EXAMPLE: MEASURING ON THE OO-H WE META-MODEL
TQM approach for WE App development

EXAMPLE: MEASURE AUTOMATION STEPS (0/3)
When = nm1.tLinks.coveredAssociations ->asSet () ->size ()/DomainModel.associations ->size ()*100<=80

Where = CheckNonNavigatedAssociations (associations , tLinks,nm1 )

CheckDCNM(Indicator,DomainModel , nm1)
TQM approach for WE App development

Example: Measure Automation Steps (2/3)

Evolution Rule (1)

When \( \neg (nA = n1 \text{ and } nB = n2) \) or \( nA = n2 \text{ and } nB = n1 \)

Where = DefineNewTraversalLink(Association, newTLink )
TQM approach for WE App development
TQM approach for WE App development
TQM approach for WE App development

EXAMPLE: TOOL SUPPORT

WE TQM Automation


UPT Tool
On-line MDA tool

Library System

What do you want?
- New Model
- View Models
- New UPT Transformations
- View UPT Transformations
- New Execution
- View Executions

Resume

Source Models
- Navigational Model
- MDomain Model
- DCNM Measurement Model

UPT Transformations
- DCNM Transformation

Target Models
- Navigational Model

Library System

In this case study, the library keeps track of a set of publications, each one having a set of keywords by which it is characterized. Each publication is composed of a set of articles.
We have provided all the elements that make up a Quality Evaluation Process in Web Engineering (ISO 14598)
- A Quality Model
- A Method of Evaluation
- A Software Measurement Process
- Supporting Tools

Every step is reusable among WE methodologies
Every Quality Artifact (except for specific transformations) is reusable among WE methodologies
- Even such transformations could be reused if we agreed on a common meta-model for each level of abstraction
QM proposals for each level of abstraction
Empirical Validation of assumptions traditionally made in WE methodologies
MDE Standardized support for WE processes (including evaluation issues)
Agreement on common meta-models for WE
Empirical validation of semantic equivalences/differences among WE proposals
Towards the TQM in WE

Thank you very much for your attention!

Speaker:
Cristina Cachero
ccachero@dlsi.ua.es