UNIVERSITÄT
FRANKFURT AM MAIN

A Topic Map Templates based Prototype for Software Development Support

M. Ueberall, O. Drobnik

Telematics Group, Institute of Computer Science
J. W. Goethe-University, Frankfurt/Main, Germany

2007 / 10 / 11

Motivation and Objective

Problems in Software Development:

- Participants (stakeholders, architects/designers, programmers) have different objectives/points of views as well as different domain knowledge
  → *consistent conceptualisations*
- Any loss of information is to be considered fatal
  → *traceability*
- *semantics-preserving phase transitions*

Our approach is based on

- use of light-weight representations (*Topic Maps*)
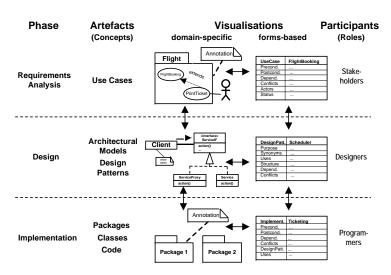- development process support by means of *templates*

TM Templates based Prototype

M. Ueberall, O. Drobnik

Introduction

User Support

Modeling

Versioning

Application Logics

Tools

Ongoing Work

Overview

Phase	Artefacts (Concepts)	Visualisations		Participants (Roles)
		domain-specific	forms-based	

Requirements Analysis — Use Cases

Flight / Annotation / FlightBooking extends / PrintTicket

UseCase	FlightBooking
Precond.	...
Postcond.	...
Depend.	...
Conflicts	...
Actors	...
Status	...

Stake-holders

Design — Architectural Models / Design Patterns

Client uses «interface» ServiceIF action() / Annotation / ServiceProxy action() / Service action()

DesignPatt.	Scheduler
Purpose	...
Synonyms	...
Uses	...
Structure	...
Depend.	...
Conflicts	...

Designers

Implementation — Packages / Classes / Code

Annotation / Package 1 / Package 2

Implement.	Ticketing
Precond.	...
Postcond.	...
Depend.	...
Conflicts	...
DesignPatt.	...
Uses	...

Programmers

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

User Support (GUI)

- **Roles:**
  - mainly used for access control (e.g., certain domain-/phase-specific concepts can only be annotated)

- **Views:**
  - based on roles; used to filter information and present results according to user's domain knowledge

- **Queries:**
  - filter/constraint mechanism; both pre-defined (for views) and user-adjustable

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

UNIVERSITÄT
FRANKFURT AM MAIN

Prototype
Who is it supposed to look like?

- Eclipse Plugin

- Based on the Eclipse GMF (Graphical Modeling Framework) Mindmap Example

    - Essentially, we're interested in "a kind of formalised mindmap using domain-specific visualisations"

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

Hierarchical Structuring

- A *faceted classification system* allows the assignment of multiple classifications to an object

- this enables the classifications to be ordered in multiple ways, rather than in a single, pre-determined, taxonomic order

- The most prominent use of faceted classification is in faceted navigation systems that enable a user to navigate information hierarchically, going from a category to its sub-categories, but choosing the order in which the categories are presented
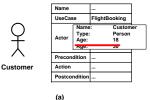
TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

Usage Restrictions for Templates
Constraints Example

(a)

(b)

(c)

(a) cardinality constraint for "Age"

(b) context-sensitive labeling

(c) occurrence-type cannot be used as template attribute

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

UNIVERSITÄT
FRANKFURT AM MAIN

Versioning
Metadata for Units of Information, LTM notation

```
// <occurrence | association> ~object-ref

// dc: cf. http://dublincore.org/documents/dces (Dublin Core)
{object-ref, dc:creation-date, [[2007-06-04T2359:59+01:00]]}
{object-ref, dc:version, [[version-id]]}
{object-ref, dc:author, [[user-id]]}
// skos: cf. http://w3.org/2004/02/skos (SKOS)
{object-ref, skos:changeNote, [[description]]}
...

is-replaced-by(object-ref₁ :old-obj, object-ref₂ :new-obj)
is-deprecated(object-ref₃ :obj)
is-deleted(object-ref₄ :obj)
```

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

Versioning
Granularity of Objects

Name	...
UseCase	FlightBooking
Description	Statement A Statement B Statement C
Precondition	...
Action	...
Postcondition	...

→

Name	...	
UseCase	FlightBooking	
Description		
	Description1	Statement A
	Description2	Statement B
	Description3	Statement C
Precondition	...	
Action	...	
Postcondition	...	

```
is-replaced-by(desc-ref₁ :old-obj, desc-ref₂ :new-obj)
part-of(desc-ref₂ :whole, description₁ :new-obj)
```

$\cdots$

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

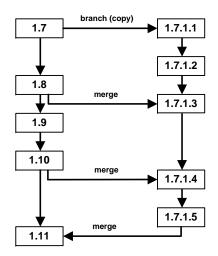User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

Versioning

Importance of the Directed Acyclic Graph structure

(cf. http://www.kerneltraffic.org/kernel-traffic/kt20030323_210.txt)

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

(Underlying) Application Logics

- **Dynamically loadable Java Classes ("modules"):**
  - constraints/templates can be added as topic map objects; however, executable (application/scenario-dependent) logics must be "hard-coded"; this mechanism allows at least to change resulting java classes on-the-fly

- **Versioning:**
  - pre-defined "module" – any topic map object will be annotated with metadata concerning, e.g., author, date, ancestor information

- **Meta Process Model:**
  - another pre-defined "module" which forces participants to re-concile definitions whenever they commit or checkout changes from/into their workspace

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

UNIVERSITÄT
FRANKFURT AM MAIN

Existing Tools/Libraries

- **TMAPI (Common Topic Map Application Programming Interface, `www.tmapi.org`):**
  - well-known programming interface for accessing and manipulating data held in a topic map
- **H2 Database Engine (`http://www.h2database.com`):**
  - very fast, free SQL database with a small memory footprint written in Java featuring a JDBC and (partial) ODBC API; in-memory mode also makes it an alternative to TinyTIM (http://tinytim.sourceforge.net)
- **Jakarta Commons Id Component (`http://commons.apache.org/sandbox/id/`):**
  - component used to generate identifiers which offers several different algorithms that are also suitable for distributed collaboration/development scenarios
- **ANTLR 3.x (`http://www.antlr.org`):**
  - parser generator that comes with ANTLRWorks, a *very* useful graphical tool to test your grammars

TM Templates
based
Prototype

M. Ueberall,
O. Drobnik

Introduction

User Support

Modeling

Versioning

Application
Logics

Tools

Ongoing Work

UNIVERSITÄT
FRANKFURT AM MAIN

Prototype: Ongoing Work

- modularised querying/filtering support for preliminary prototype
  - evaluation with regard to more complex development scenarios
  - combination with other Eclipse plugins, e.g., pattern scanners

- medium-term objective: support for coping with ontology changes and/or mergers within meta process model subphases

- long-term objective: support of operations as needed for decentralised version management (n-way merge)

UNIVERSITÄT
FRANKFURT AM MAIN

Thank you!

e-mail to:
ueberall@tm.informatik.uni-frankfurt.de