

Visualizing Aspect-Oriented Goal Models with AoGRL

Gunter Mussbacher, Daniel Amyot João Araújo, Ana Moreira Michael Weiss
 SITE, University of Ottawa Univ. Nova de Lisboa Carleton University
 (gunterm | damyot)@site.uottawa.ca (ja | amm)@di.fct.unl.pt weiss@scs.carleton.ca

October 15, 2007

uOttawa

Motivation

- As goal models can be large and complex even for small problems, it is often a challenge for requirements engineers to aptly visualize them and to efficiently structure them for maintenance and reuse activities
 - Goal-oriented Requirement Language (GRL)
 - i*
 - NFR framework
- Aspects have the potential of improving the modularity, understandability, reusability, scalability, and maintainability of goal models
- Add support for aspect-oriented modeling to GRL
 - Aspect-Oriented GRL (AoGRL)
- Assess the benefits of AoGRL over GRL by evaluating the above-mentioned qualities with the help of metrics adapted from literature

How ?

Therefore

uOttawa

Table of Contents

- Aspects
- Goal-oriented Requirement Language (GRL)
- Aspect-Oriented GRL (AoGRL)
- Poster – Show and Tell
- Introduction to YKeyK Case Study
- Summary of Evaluation
- Conclusion & Future Work

uOttawa

Aspects

- Aspects address the problem of one concern **crosscutting** other concerns in a system or model
- Aspects can encapsulate concerns even if they are crosscutting

Without Aspects

Concern A Concern B Concern C

Tangling

Scattering

With Aspects

Concern A Concern B Concern C

Aspect1 Aspect2 Aspect3

(each aspect contains a **composition rule** illustrated by the arrows that defines where to add the aspect)

... 3 Crosscutting Concerns (Aspect1, Aspect2, Aspect3)

uOttawa

User Requirements Notation (URN)

- URN is the **first** and **currently only** standardization effort which explicitly addresses goals in addition to scenarios in a graphical way in one unified language (International Telecommunication Union, ITU-T Z.150 series)
 - Non-functional requirements (goals) with GRL (Goal-oriented Requirement Language)
 - Functional requirements (scenarios) with UCMs (Use Case Maps)
 - jUCMNav ... URN editor, open source project
- URN allows traceability relationships to be established with URN links
 - For example, between GRL models and UCM models
- Aspect-oriented extensions are being added to URN (i.e. AoURN)
 - Aspect-Oriented Use Case Maps (AoUCM), first introduced at REV'06
 - Aspect-Oriented GRL (AoGRL), REV'07
 - One framework** combining goal-oriented, scenario-based, and aspect-oriented modeling

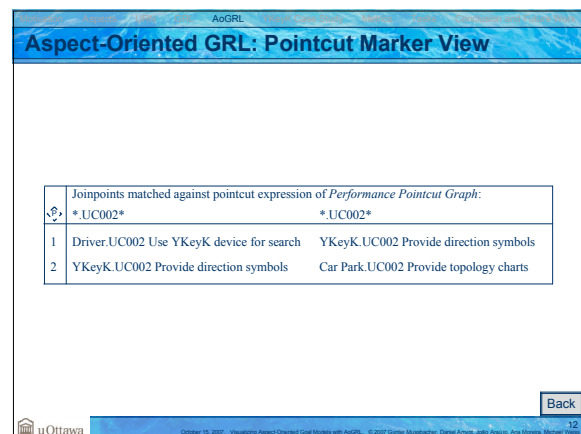
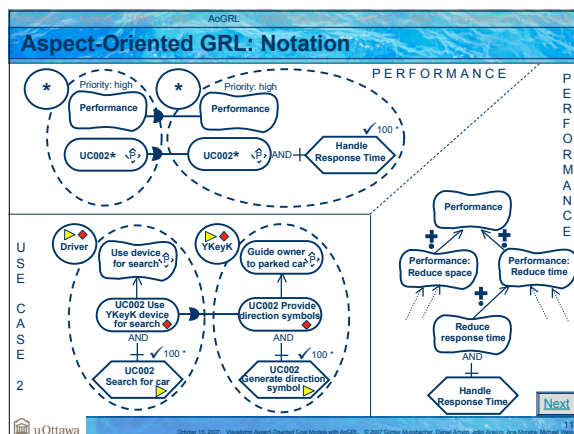
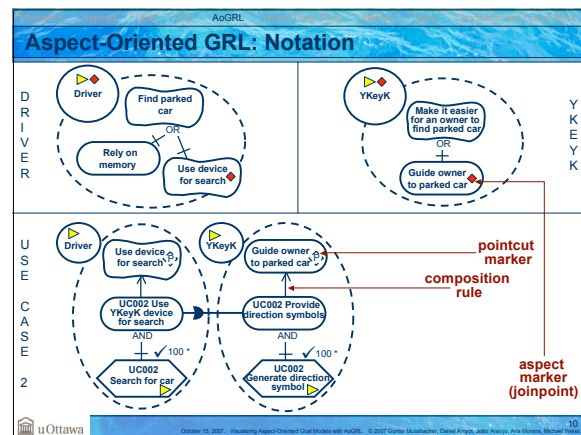
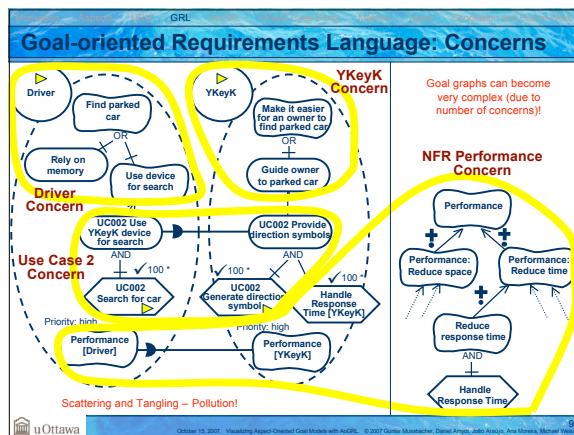
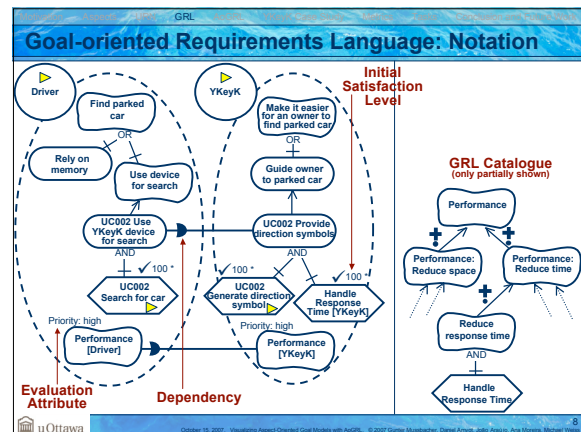
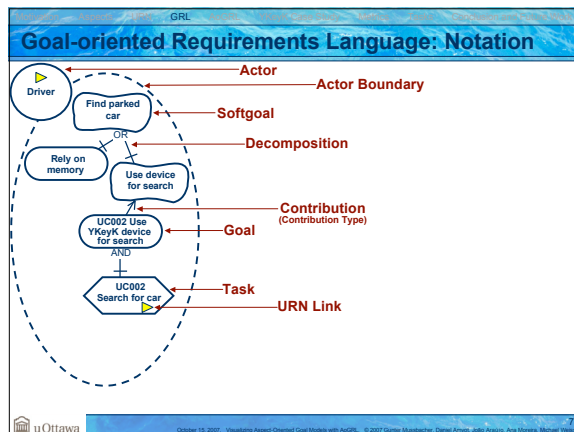
URN ... www.usecasesmaps.org/urn jUCMNav ... www.softwareengineering.ca/jucmnav Virtual Library ... www.usecasesmaps.org/pub

uOttawa


Goal-oriented Requirements Language: Overview


- The Goal-oriented Requirements Language (GRL) is based on ...
 - i* (concepts / syntax)
 - NFR Framework (evaluation mechanism)
- GRL is used to ...
 - Visually describe business goals, stakeholders' priorities, alternative solutions, rationale, and decisions
 - Decompose high-level goals into alternative solutions called tasks (this process is called **operationalization**)
 - Model positive and negative influences of goals and tasks on each other
 - Capture dependencies between actors (i.e. stakeholders)

uOttawa

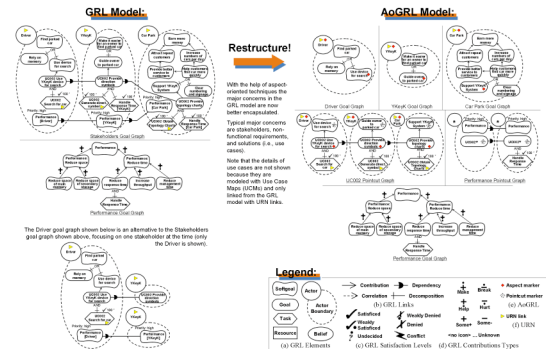


Aspect-Oriented GRL: Aspect Marker View

	Aspects that matched <i>Driver.UC002 Use YKeyK device for search</i>
1	Driver.UC002 Use YKeyK device for search ---- ----
	Pointcut Graph: Performance Pointcut Graph

	Aspects that matched <i>Driver.UC002 Use YKeyK device for search</i>	
1	Driver.UC002 Use YKeyK device for search --- ---	YKeyK.UC002 Provide direction symbols AND Decomposition into YKeyK.Handle Response Time
	Pointcut Graph: Performance Pointcut Graph	

Visualizing Aspect-Oriented Goal Models with AoGRL

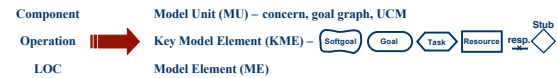


YKeyK Case Study: Three Models

- **YKeyK.A**
 - GRL model that contains only one goal graph with the complete system
 - Almost 300 model elements in a goal graph
 - Not scalable and mostly serves as a base to compare other alternatives
- **YKeyK.B**
 - Most likely used approach to describe the YKeyK system with GRL
 - Each stakeholder is described on its own goal graph, referencing goal graphs and use case maps for non-functional requirements and use cases
- **YKeyK.C**
 - AoGRL model of the YKeyK system
 - Each stakeholder, each non-functional requirement, and each use case is encapsulated by a concern and modeled on separate goal graphs
 - Pointcut graphs for each non-functional requirement and use case define how the crosscutting non-functional requirements or use cases are added to the model

Overview of Metrics

- Adapted from literature [Sant'Anna et al 2003]



- **Separation of Concerns Metrics**
 - CDMU
(Concern diffusion over MUs)
 - CDKME
(Concern diffusion over KMEs)
 - CDME (Concern diffusion over MEs – concern switches)
- **Coupling Metrics**
 - CBMU (Coupling between MUs)
- **Cohesion Metrics**
 - LCOCKME
(Lack of cohesion in KMEs)
- **Size Metrics**
 - VS (Vocabulary size – number of MUs)
 - NME (Number of MEs)
 - AvNME (Average Number of MEs)

[Sant'Anna et al 2003] ... On the Reuse and Maintenance of Aspect-Oriented Software: An Assessment Framework, SBES'03.

Summary of Metrics-Based Evaluation

- Overall, the AoGRL model performs significantly better than the two GRL models
- Some results for YKeyKA cannot be taken into account at all (see n/a)
 - Pathological situation with one graph
- The AoGRL model contains more but smaller goal graphs grouped by concerns
 - Trade complexity of stakeholder goal graphs with **localized** complexity of pinpoint graphs

Metric	GRL		AoGRL
	YKey _{K,A}	YKey _{K,B}	YKey _{K,C}
CDMU	n/a (14)	42	21
CDKME	111	154	127
CDME	98	117	0
CBMU	n/a (0)	34	7
LCOKME	n/a (0)	88	0
VS	n/a (1)	13	35
NME	285	320	264
AvNME	285	25	12

<i>AvNME for Goal Graphs</i>	<i>YKeyK.B</i>	<i>YKeyK.C</i>
Stakeholders	42	6
NFR / UC	10	10
Pointcuts	n/a	20

Summary of Task-Based Evaluation

- Common modeling tasks
 - Adding or changing a stakeholder, non-functional requirement, or use case
- YKeyK.A
 - Needle-in-the-haystack approach
- YKeyK.B
 - Changes distributed over many different goal graphs (e.g., change use case and then relevant locations in stakeholder goal graphs)
- YKeyK.C
 - Changes required for a **single** concern only (plus monitoring of any changes to matched joinpoints for pointcut expressions)

Conclusion and Future Work

- Results suggest that the case study's AoGRL model exhibits better modularity, understandability, reusability, and maintainability
- Trade-off complexity in stakeholder goal graphs with complexity in pointcut graphs
- Scalability is arguably improved by
 - Reduced complexity of goal graphs (see AvNME results)
 - Ability to group goal graphs with concerns
 - Encapsulation provided by concerns
 - Ability to use parameterized pointcut expressions in AoGRL
 - Simpler update tasks for AoGRL
- Future Work
 - Further controlled experiments with varied, real-world sized case studies
 - Implement enhancements in JUCMNav
 - Extend evaluation mechanism to AoGRL
 - How to best link, combine, and evolve goals and scenarios in single aspects