

Modeling and Analysis of URN Goals and Scenarios with jUCMNav

Gunter Mussbacher, Sepideh Ghanavati, Daniel Amyot

SITE, University of Ottawa, 800 King Edward, Ottawa, ON, K1N 6N5, Canada

{ gunterm | sghanava | damyot }@site.uottawa.ca

Abstract

In November 2008, the User Requirements Notation (URN) was approved as a standard by the International Telecommunication Union (ITU-T). jUCMNav is the most comprehensive tool available to date that supports the definition, analysis, transformation, and management of URN requirements engineering models. URN is the first standardized framework unifying modeling concepts and notations for goals and intentions (mainly for non-functional requirements, quality attributes, and reasoning about alternatives) and scenarios (mainly for operational/functional requirements and reasoning about scenario interactions, performance, and high-level architecture). jUCMNav has been and continues to be instrumental in validating key concepts for the current standard as well as prototyping new concepts.

1. Introduction

jUCMNav [3] is the most comprehensive tool available to date that supports the *User Requirements Notation* (URN) standard [2]. jUCMNav is an open-source Eclipse plug-in for URN's two complementary sublanguages – GRL (the *Goal-oriented Requirement Language*) models goal-oriented/intentional concepts and UCM (*Use Case Maps*) model scenario concepts.

URN supports the elicitation, analysis, specification, and validation of requirements. URN is the first standardized language explicitly unifying goals, scenarios, and the links between them in a visual way. URN models focus on exposing “why” certain choices for behavior and/or structure were introduced, combined with an abstract view of “what” capabilities and architecture are required. Detailed specifications of “how” functionalities are to be supported are not a primary concern of URN. Modeling goals and scenarios is complementary and may aid in identifying additional or spurious goals and scenarios, thus contributing to the completeness and accuracy of requirements.

URN has concepts for the specification of stakeholders, goals, non-functional requirements, rationales, behavior, structure, and scenarios. URN is suitable for describing most types of service-oriented, concurrent, distributed, reactive, and information systems, covering abstractions from business goals and requirements description to high-level design. For more details, the reader is referred to jUCMNav's website [3] and the (220+) publications and theses available at the URN Virtual Library [8].

2. Features of jUCMNav

jUCMNav's **basic model definition features** are based on the URN standard and ensure that syntactically correct URN models are created and that definitions of URN model elements can be shared among multiple interlinked GRL and UCM diagrams.

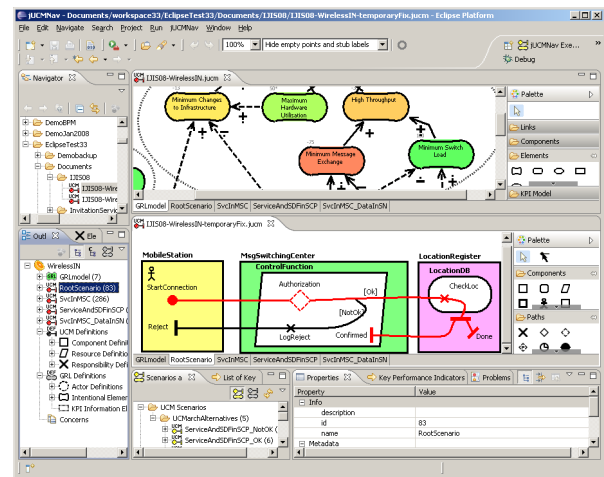


Fig. 1. URN modeling with jUCMNav

The two key URN **analysis features** supported by jUCMNav are GRL model evaluation with strategies and UCM model interpretation with scenario definitions and the path traversal mechanism. jUCMNav has an extensible architecture that allows customized GRL and UCM analysis techniques to be added to the tool with little effort. jUCMNav's **GRL evaluation**

mechanisms give immediate visual feedback for global tradeoffs among conflicting stakeholder goals (see Fig 1). Currently, the three sample GRL evaluation mechanisms in the URN standard are supported (quantitative, qualitative, as well as a mixed analysis), plus a fourth one proposed in [7].

jUCMNav's **path traversal mechanism** conforms to the requirements defined in the URN standard and traverses a UCM model based on a scenario definition provided by the modeler. The path traversal mechanism enables the (regression) testing and validation of the UCM model, as well as the detection of undesired scenario interactions. A scenario definition specifies scenario start points, preconditions, expected scenario end points, postconditions, and initializations of global variables. The actual path to be traversed is determined by the initial, user-defined values of these global path variables and the changes to their values during the traversal. The URN data model contains a simple action language that supports variable assignments, expressions, and conditional statements and is fully supported by the jUCMNav tool. Various errors and warnings are reported if the traversal cannot proceed, when non-deterministic choices are encountered, or when preconditions, postconditions, or expected scenario end points are violated.

jUCMNav also uses scenario definitions to highlight, in red, the paths traversed while running the scenario on a UCM model (see Fig 1). This improves the understandability of complex UCM models by focusing on a particular path through the model. In addition, strategies and scenario definitions can be combined for an integrated analysis of a URN model and their results can influence each other [4]. jUCMNav automatically creates UCM integer variables for each GRL graph element. The current satisfaction levels of such elements can hence be used to influence the selection of UCM paths. Also, new values can be assigned to these variables during the traversal, hence influencing the propagation of satisfaction levels in a GRL graph.

Recently, jUCMNav has been extended with the concept of **Key Performance Indicators (KPIs)** [6]. KPIs represent real metrics of a system that can be retrieved from external sources of information (e.g. data warehouses or performance management tools). KPIs are used in URN as input for the GRL evaluation mechanism. Therefore, KPIs may change the satisfaction level of high-level goals which in turn may cause changes to the parts of the UCM model that are linked to the goal model. A feedback-based framework for online monitoring of system metrics and runtime adaptation of scenario models is therefore established.

The **transformation features** of the jUCMNav tool support advanced functionalities such as scenario

export to MSC [4] with an integrated MSC viewer, export of performance models based on UCM performance annotations, and integration with the IBM/Telelogic DOORS requirements management system [6] (for requirements and model evolution, and for compliance). The tool can also generate reports and export diagrams in various formats.

3. Conclusion

jUCMNav is the most popular and feature-rich modeling tool for the User Requirements Notation, a goal and scenario modeling standard recently approved by ITU-T. jUCMNav is currently used at a dozen universities and in several companies around the world for education, research, and industrial modeling purposes. We ourselves have been using it successfully for 4 years in our undergraduate Software Requirements Analysis and graduate Software Engineering courses.

jUCMNav has been and continues to be very useful in validating new ideas and concepts for URN and to demonstrate the usefulness of the notation. Future versions of jUCMNav will focus on better supporting new URN constructs recently standardized as well as concepts and capabilities from **advanced research** in the areas of business process monitoring / runtime adaptation [6], composition of aspect-oriented models [5], OCL-based semantic model verification [1] and model metrics, as well as advanced workflow patterns.

References

- [1] Amyot, D. and Yan, J.B.: Flexible Verification of User-Defined Semantic Constraints in Modelling Tools. *CASCON 2008*, Toronto, Canada, October 2008.
- [2] ITU-T – International Telecommunications Union: *Recommendation Z.151 (11/08), User Requirements Notation (URN) - Language Definition*. Geneva, Switzerland, 2008.
- [3] jUCMNav 3.2.1, University of Ottawa, acc. May 2009, <http://jucmnav.softwareengineering.ca/jucmnav/>
- [4] Kealey, J. and Amyot, D.: Enhanced Use Case Map Traversal Semantics. *13th SDL Forum (SDL'07)*, Paris, France, LNCS 4745, Springer, 133-149, September 2007.
- [5] Mussbacher, G.: Aspect-Oriented User Requirements Notation: Aspects in Goal and Scenario Models. *Models in Software Eng.: Workshops and Symposia at MoDELS 2007*, Giese, H. (Ed.), LNCS 5002, Springer, 305-316, June 2008.
- [6] Pourshahid, A., Chen, P., Amyot, D., Forster, A.J., Ghanavati, S., Peyton, L., and Weiss, M.: Business Process Management with the User Requirements Notation. To appear in: *Electronic Commerce Research*, Springer, 2009.
- [7] Roy, J.-F., Kealey, J., and Amyot, D.: Towards Integrated Tool Support for the User Requirements Notation. *SAM 2006*, Kaiserslautern, Germany, May 2006.
- [8] User Requirements Notation Virtual Library, accessed May 2009, <http://www.usecasemaps.org/pub/>