# Deriving Message Sequence Charts from Use Case Maps Scenario Specifications*

Andrew Miga[1], Daniel Amyot[2], Francis Bordeleau[1], Donald Cameron[3], and Murray Woodside[1]

[1] Carleton University, Ottawa, Canada
{miga, cmw}@sce.carleton.ca, francis@scs.carleton.ca
[2] Mitel Networks, Kanata, Canada
Daniel_Amyot@Mitel.com
[3] Nortel Networks, Ottawa, Canada
dcameron@NortelNetworks.com
http://www.UseCaseMaps.org

**Abstract.** A set of scenarios is a useful way to capture many aspects of the requirements of a system. Use Case Maps are a method for scenario capture which is good for describing multiple scenarios, including scenario interactions, for developing an architecture, and for analysing architectural alternatives. However, once a component architecture is determined, Message Sequence Charts are better for developing and presenting the details of interactions, and provide access to well-developed methodologies and tools for analysis and synthesis. This paper considers what must be specified in UCM scenarios and the architecture to make it possible to derive MSCs automatically, and it describes our experience in executing these transformations within a prototype tool, the UCM Navigator.

## 1 Introduction

A scenario specification technique called Use Case Maps [11,12] is part of a new proposal to ITU-T for a User Requirements Notation (URN) [13]. The role of the UCM notation is to capture functional requirements and it has been baptized URN-FR, while another and complementary component for non-functional requirements [14] is called URN-NFR. UCMs capture functional requirements in terms of *causal scenarios* that link sequences of responsibilities to (external) events. These scenarios may also be bound to underlying abstract components.

UCMs have been useful in describing a wide range of systems, including Wireless Intelligent Networks [2,16,26], agent systems [15], Wireless ATM [7], GPRS [3], and others discussed in [5,25]. As suggested by the I.130 and Q.65

* This research was supported by Nortel Networks, Mitel Networks, and by the Industrial Partnerships program of the Natural Sciences and Engineering Research Council of Canada. Discussions with John Visser, Jim Hodges, Jacques Sincennes, Luigi Logrippo and Gunter Mussbacher were helpful in developing the ideas.

R. Reed and J. Reed (Eds.): SDL 2001, LNCS 2078, pp. 268–287, 2001.

methodologies [17,20] and several UML-based approaches [4], the process of creating specifications and standards is generally composed of three major stages. At Stage 1, services are described from the user's point of view in prose form and with use cases. The focus of the second stage is on control flows between the different entities involved, represented using Message Sequence Charts (MSCs) [19]. Finally, Stage 3 aims to provide (informal) specifications of protocols and procedures. Formal specifications are sometimes provided (in SDL [18] for example), but overall they still suffer from a low penetration, especially in North-America.

In such methodologies, scenarios are often used as a means to model system functionality and interactions between the entities such that different stakeholders may understand their general intent as well as technical details [27]. Use Case Maps are used in Stage 1, and to bridge the conceptual gap into Stage 2 descriptions [5]. UCMs are used to capture user (functional) requirements when very little design detail is available, *without reference to messages or component states*. In Stage 1 documents, UCM scenarios may or may not be bound to any particular components for execution. The organization and architecture of components can be introduced into the map when moving towards Stage 2 documents. One of the strengths of UCMs at this level is their ability to show a number of scenarios together, and to reason about architecture and behaviour over a set of scenarios. Once appropriate architectural decisions are taken, UCMs can be used to guide the generation of MSCs to complete Stage 2 descriptions. In turn, MSCs can be used for the synthesis and the validation of component-based behavioral models in SDL or similar languages [1,22]. Many such synthesis techniques are studied and compared in [6].

This paper builds on previous work [2,9] and describes research on a well-defined transformation from a subset of the UCM notation to MSC-96, along with preliminary results in implementing the transformation. This transformation enables the rapid and consistent generation of MSCs from UCMs, and the extraction of simple end-to-end scenarios from complex multilevel UCMs. These MSCs can further be refined for Stage-2 like documents (where the specifics of messages becomes more relevant), used for system understanding, and used for functional testing of more detailed models and of implementations. The UCM notation is briefly reviewed and illustrated in Sect. 2. The UCM/MSC relationship is further studied in Sect. 3, with a particular emphasis on scenario variables in Sect. 4. Section 5 explains the proposed transformation, which is then illustrated with an example in Sect. 6. Finally, Sect. 7 presents our conclusions.

## 2 Use Case Maps

Use Case Maps visually describe causal relationships between *responsibilities* superimposed on organizational structures of abstract *components*. Responsibilities represent generic processing (actions, activities, operations, tasks, etc.). Components are also generic and can represent software entities (objects, processes, databases, servers, etc.) as well as non-software entities (actors or hardware). The relationships are said to be causal because they link causes (preconditions