Model-Based Prototyping of an Interoperability Protocol for Mobile Ad-Hoc Networks

Lars M. Kristensen[1,⋆], Michael Westergaard[1], and Peder Christian Nørgaard[2]

[1] Department of Computer Science, University of Aarhus,
IT-parken, Aabogade 34, DK-8200 Aarhus N, Denmark
{kris, mw}@daimi.au.dk
[2] Ericsson Danmark A/S, Telebit,
Skanderborgvej 222, DK-8260 Viby J, Denmark
Peder.Chr.Norgaard@ericsson.com

Abstract. We present an industrial project conducted at Ericsson Danmark A/S, Telebit where formal methods in the form of Coloured Petri Nets (CP-nets or CPNs) have been used for the specification of an interoperability protocol for routing packets between fixed core networks and mobile ad-hoc networks. The interoperability protocol ensures that a packet flow between a host in a core network and a mobile node in an ad-hoc network is always relayed via one of the closest gateways connecting the core network and the mobile ad-hoc network. This paper shows how integrated use of CP-nets and application-specific visualisation have been applied to build a model-based prototype of the interoperability protocol. The prototype consists of two parts: a CPN model that formally specifies the protocol mechanisms and a graphical user interface for experimenting with the protocol. The project demonstrates that the use of formal modelling combined with the use of application-specific visualisation can be an effective approach to rapidly construct an executable prototype of a communication protocol.

Keywords: Model-driven prototyping; animation; Coloured Petri Nets; mobile ad-hoc network.

1 Introduction

The specification and development of communication protocols is a complex task. One of the reasons is that protocols consist of a number of independent concurrent protocol entities that may proceed in many different ways depending on when, e.g., packets are lost, timers expire, and processes are scheduled. The complex behaviour makes the design of protocols a challenging task. Protocols operating in networks with mobile nodes and wireless communication present an additional set of challenges in protocol engineering since the orchestration of realistic scenarios with many mobile nodes is impractical, and the physical characteristics of wireless communication makes reproduction of errors and scenarios almost impossible.

⋆ Supported by the Danish Natural Science Research Council.

J. Romijn, G. Smith, and J. van de Pol (Eds.): IFM 2005, LNCS 3771, pp. 266–286, 2005.

We present a case study from a joint research project [15] between the Coloured Petri Nets Group [6] at University of Aarhus and Ericsson Danmark A/S, Telebit [7]. The research project applies formal methods in the form of Coloured Petri Nets (CP-nets or CPNs) [13,16] and the supporting CPN Tools [5] in the development of Internet Protocol Version 6 (IPv6) [12] based protocols for ad-hoc networking [24]. An ad-hoc network is a collection of mobile nodes, such as laptops, personal digital assistants, and mobile phones, capable of establishing a communication infrastructure for their common use. Ad-hoc networking differs from conventional networks in that the nodes in the ad-hoc network operate in a fully self-configuring and distributed manner, without any preexisting communication infrastructure such as base stations and routers.

CP-nets is a graphical discrete-event modelling language applicable for concurrent and distributed systems. CP-nets are based on Petri nets [27] and the programming language Standard ML (SML) [30]. Petri nets provide the foundation of the graphical notation and the basic primitives for modelling concurrency, communication, and synchronisation. The SML programming language provides the primitives for the definition of data types, modelling data manipulation, and for creating compact and parameterisable models. CPN models are executable and describe the states of a system and the events (transitions) between the states. CP-nets includes a module concept that makes it possible to organise large models into a hierarchically related set of modules. The CPN modelling language is supported by CPN Tools and have previously been applied in a number of projects for modelling and validation of protocols (see, e.g., [17, 8, 9, 23]).

The use of formal modelling languages such as CP-nets for specification and validation of protocols is attractive for several reasons. One advantage of formal models is that they are based on the construction of executable models that make it possible to observe and experiment with the behaviour of the protocol prior to implementation using, e.g., simulation. This typically leads to more complete specifications since the model will not be fully operational until all parts of the protocol have been at least abstractly specified. A model also makes it possible to explore larger scenarios than is practically possible with a physical setup. Another advantage of formal modelling is the support for abstraction, making it possible to specify protocols while ignoring many implementation details.

From a practical protocol engineering viewpoint, the use of formal modelling also have some shortcomings. Even if the modelling language supports abstraction and a module concept there is in most cases an overwhelming amount of detail in the constructed model. This is a disadvantage, in particular when presenting and discussing the design with colleagues unfamiliar with the applied modelling language. This means that a formal specification in many cases is accompanied by informal drawings being developed in parallel. The level of detail can also be a disadvantage when exploring the protocol design via, e.g., simulation. Furthermore, even if a model is executable, it still lacks the application- and domain-specific appeal of a conventional prototype.

The contribution of this paper is to present a model-based prototyping approach where formal modelling is integrated with the use of an animation GUI