An XML Description Language for Web-based Network Simulation

R. Canonico, D. Emma, G. Ventre
Università di Napoli "Federico II", Dipartimento di Informatica e Sistemistica
Via Claudio, 21 – I-80125 – Napoli (Italy)
roberto.canonico@unina.it, demma@napoli.consortio-cini.it, giorgio.ventre@unina.it

Abstract

Simulation of large scale network scenarios is a challenging task and requires a great amount of computational power. Hence, realizing web-accessible simulation servers is a key step for the success of network simulation as a useful instrument in the context of network administration and capacity planning. This paper presents an XML-based description language for describing network simulation scenarios. For the proposed language, we also present an XSL translation process that can be used to automatically translate a simulation scenario into a simulation script for a well known network simulator. This work is part of a larger project, aimed at implementing a cluster-based network simulation server to be integrated in a distributed system for QoS monitoring, SLA validation and measurement-based modeling in an inter-domain environment.

1. Introduction

In recent years, simulation has become an indispensable technique to evaluate the performance of communication protocols for computer networks [1]. Quite a large number of network simulators have been developed over the past decade [2, 3, 4, 5, 6, 7]. More recently, simulation has been proposed as an useful tool for network planning and what-if analysis [8, 9, 10, 11]. Packet-level simulation of large scale computer networks is a challenging task, due to the great amount of computation it requires. To overcome such a difficulty, researchers have investigated in two directions: parallelism and abstraction. Parallel simulators [12, 13] rely on multiprocessors and clusters of workstations to achieve the computation power and memory necessary to run large scale simulations. Abstraction, on the other hand, is aimed at reducing the unnecessary details of packet level simulation, in order to obtain a significant picture of the network behavior with a significant reduction of computation time [14, 15]. These two approaches are complementary, and can be exploited at the same time to scale up the potential of network simulation. The combination of analytical models with classical discrete event simulation has lead to so-called hybrid simulation techniques.

In the framework of the European INTERMON Research Project [16], the authors are investigating the problem of realizing a powerful network simulation server, based on a cluster of PCs. Our simulation server can be accessed from remote users through a web based interface. This component is integrated in a more general architecture [17], aimed at providing network administrators with a powerful set of tools that can be used for capacity planning and Service Level Agreement tuning and validation. The whole INTERMON architecture relies upon a distributed database, which is updated by proper measurement tools deployed within the real network. To integrate our simulation server with the rest of the INTERMON architecture, it was necessary to define a precise service interface, so that simulation scenarios could be automatically generated from other components and submitted to the server. These reasons have motivated our work, i.e. the use of XML to describe network simulation scenarios that can be automatically translated into simulation scripts, for a variety of network simulators.

The rest of this paper is organized as follows. In section 2 we describe the architecture of a network simulation server based on a cluster of PCs that provides a network simulation service that can be accessed through a web-based interface. In section 3, we describe an XML description language that we have defined to describe network simulation scenarios in a simulator-neutral way. In section 4, we present how we managed to automatically translate an XML simulation scenario into an ns-2 simulation script [6]. Finally, in section 5, we present some concluding remarks and future work.

2. A Web Based Network Simulation Server

In the context of the INTERMON research project [16], we are developing a web-accessible network simulation server. Our server has been designed according to a 3-tier architecture, as illustrated in Figure 1.
3. Network simulation scenarios in XML

A simulation scenario is a description of all the network components that need to be modeled when the behavior of a given network needs to be simulated.

Basically, a network scenario is composed of:
- a description of the network topology;
- a description of the traffic flows that will be generated during the simulated time;
- a description of particular events (e.g. link failures) that need to be modeled in the course of the simulation;
- a description of the expected simulation output.

In this section we describe the basic concepts that make a network simulation scenario. To better illustrate the logical relations among the various components of a simulation scenario, we make use of UML class diagrams. In a second step, we illustrate how a simulation scenario can be described by means of an XML document [19]. To describe the structure of such an XML document, we make use of XML Schema [20,21]. In the process of translating the UML description into an XML document, we had to face the lack of typical O-O paradigms in XML.

Figure 3 illustrates the main components of a Simulation Scenario in terms of abstract classes.
Basically, Figure 3 shows that a Simulation Scenario is made of three parts:
- a Network Description
- a Traffic Description
- a set of Simulation Commands

The Network Description is a static representation of the network topology. Hence, it is a collection of Nodes and Links. Here, the term “node” stands for “router or end-system”. In a node, one or more instances of Network Protocol may be present.

Since modeling large scale networks is usually based on network partitioning at a domain level, the most important network abstraction is the “Autonomous System”. Hence, at a topological level, a simulated network should be represented as a collection of inter-connected Autonomous Systems. However, we also take into account the possibility of simulating small networks in which the notion of AS is not present. For these smaller networks, the network description is simply made of network nodes connected by links.

Figure 4 shows the structure of a Network Description in greater detail. This picture shows that a Network Description is made of ASes, each of which is identified by its own ID. Autonomous Systems are interconnected by Links.

The internal structure of an AS is described by its AS model(s). The simplest description is by means of a “Topological Model”, in terms of Nodes and Links. However, an AS can also be described by other formalisms, that we call “Analytical Models”. All these formalisms have in common an “AS Node”, embedding the formal representation of the AS behavior. Figure 4 shows that the concept of AS Node is derived from a generic “Node” class, which can also be used to model a generic network router and/or an end-system.

The Traffic Description component (Figure 5) contains the information required to describe the traffic flows that will be generated during the simulation. Each flow may be characterized by a proper Traffic Model, which can be:
- an analytical model, describing the stochastic properties of the traffic flow;
- an application model, describing the specific traffic patterns generated by well-known applications (e.g. Telnet, HTTP, FTP, …);
- an exact traffic trace, describing the traffic flow packet by packet.

The latter part, the Simulation Command component, contains time-dependent events, such as variations in the network topology, in the routing policies, and so on.

![Figure 4: UML Class Diagram for the Network Description component](image)

![Figure 5: UML Class Diagram for the Traffic Description component](image)

### 3.1 XML Schemas for the proposed description language

In this subsection we illustrate how we have translated the previous class diagrams in a XML format, that we describe by means of XML Schemas. Figure 6 shows the format that we have defined for an XML document describing a network simulation scenario. Such a Schema is stored in a XSD file, which refers to three external XSD files: NetworkDescription.xsd, Traffic.xsd, and SimulationCommand.xsd. These three files contain the three components of a simulation scenario that we have described above. Figure 7 shows the structure of the Network Description part of a simulation scenario. Again, this XSD document refers to external files (e.g. Node.xsd, Link.xsd, AS.xsd, …) that here we do not include, for the sake of brevity.
Just to illustrate how this process of description needs to be further refined, we present in Figure 8 the Node.xsd Schema, describing the format of the XML description of a simulated network node (e.g. a router or an end-system). As illustrated in Figure 4, the Node element may be specialized into a “ASNode” element, describing a whole network domain. This is useful for hybrid kinds of simulation, in which a simplified analytical model is used to describe the behavior of a complex domain (such as an Autonomous System) [15].

Figure 6: Simulation Scenario XML Schema

Figure 7: Network Description XML Schema

Figure 8: Node Description XML Schema

4. Translating an XML scenario in ns-2 simulation scripts

The XSD Schemas we have presented in the previous section may be used to describe a complete simulation scenario in a simulator-independent way. Besides the XML Schemas, we have also defined a set of precise translation rules that can be used to produce a simulation script for the well known ns-2 network simulator [6], starting from an XML simulation scenario. These translation rules have been expressed by means of the
XSLT transformation language [22] and have been implemented by a JAVA servlet.
The XSLT transformation document has been organized in independent files, according to the structure of the Schemas presented in Section 3.

Figure 9 presents the root XSLT document, which includes other XSLT documents and translates the root element of the XML simulation scenario. The XSLT files contain ns-2 specific code and translates the XML simulation scenario in a script that can be executed by a ns-2 simulator.

5. Production of XML simulation scenarios

The creation and manipulation of complex network scenarios for simulation is a not trivial activity [23]. Simulation of large scale networks cannot rely on manual generation of the XML simulation scenarios. To perform this task we are currently developing a graphical tool which extends the basic functionality of the applet mentioned in Section 2.

Such a tool provides users with the following features:
• integration of existing topology generators;
• localization and visualization of sets of nodes on large network topologies according to different selection criteria (e.g. mutual distance, AS identifier);
• instantiation of network components of any type on the nodes of a given node set;
• customization of network components;
• definition of new network components (e.g. web caches).

6. Conclusions and future work

In this paper we have presented an XML description language that can be used to describe network simulation scenarios in a simulator-neutral way. Such an approach can be useful to allow interoperability of different network simulators, especially in the context of distributed network simulation. We have defined this language as a way to submit network simulation scenarios to a simulation server through a web-based component implemented as a middle-tier for a cluster of Linux PCs running the ns-2 simulator. Such a simulation server is a component of a more complex distributed system for QoS monitoring, SLA validation and measurement-based modeling in an inter-domain environment. We are currently investigating the difficulty of translating simulation scenarios formatted according to our specification in simulation scripts suitable for different network simulators.

7. Acknowledgements

This work has been carried out under the financial support of the European Community in the framework of the INTERMON-ISTF-2001-3412 Research Project. It has also been partially funded by MIUR, Ministero della Istruzione, dell’Università e della Ricerca, in the framework of the FIRB project “Middleware for advanced services over large-scale, wired-wireless distributed systems” (WEB-MINDS), and of the project “Scalability and Quality of Service in Web Systems”.

Figure 9: Root XSLT transformation document
8. References


