A Java based Implementation of a Network Level Information Model for the ATM / Frame Relay Interconnection

Radu State, Emmanuel Nataf, Olivier Festor

HAL Id: inria-00107843

https://inria.hal.science/inria-00107843

Submitted on 19 Oct 2006

A Java based Implementation of a Network Level Information Model for the ATM / Frame Relay Interconnection

R. State, E. Nataf ,O. Festor
INRIA-LORIA
615, Rue de Jardin Botanique – B.P.101
Villers-les-Nancy, 54600
France
{state, nataf, festor}@loria.fr

1.   Overview

In this presentation, we illustrate the use of the Java technology to implement a network information model which captures the essence of the ATM and Frame Relay Interconnection. We show how the TMN concepts stay valid and how the existing and extended information models can be reused and implemented according to the TMN principles in a full Java environment.

Keywords

Network Information Model, Tina-C, ATM, Frame Relay, Java, UML

2.   Information model

In order to model both ATM/Frame Relay interconnection scenarios, namely Network and Service PVC interworking, we started with the model introduced in [2], which is closely related to the TINA-C NRIM model. The power of expression correlated with its simplicity and versatility were very promising for its potential reuse in the context of ATM and Frame Relay interconnection. The generic model, introduced in [2] in the context of ATM/SDH/WDM transport networks, generalized straightforwardly to the PVC network interworking. Additionally, we had to extend it, by adding classes and relationships in order to represent the service PVC interworking scenario. We designed these additional classes, using the ATM and Frame Relay interconnection MIBs, developed within the IETF.

The model was specified in the Unified Modeling Language (UML), the standard language to specify object oriented systems, using the Rational Rose toolkit [3].

3. Implementation architecture
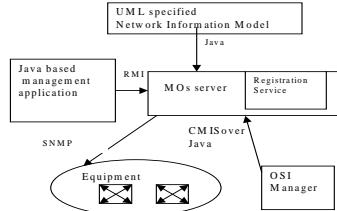
Figure 1. The Java management architecture

As illustrated in figure 1, the architecture contains the following components.
- A MO Server exhibiting 2 interfaces: one RMI interface with an application specific Java API and one CMIS interface through COJ [1].
- A registration service which enables each class and attribute to know its associated Object Identifier and an ASN.1 compatibility interface.
- Two managers (Management applications): one based on Java RMI, the other one based on a standard CMIS stack.

4. Conclusions

A Java based implementation of this model has been successfully realized, demonstrating the viability and usability of the model. The implemented architecture illustrates especially that one can easily combine the use of UML development tools and implement the system in Java so that multiple views of its contained managed objects (i.e. a standard CMIS interface as well as a complete application specific RMI interface), are offered. We have also shown that a UML design of the information model can serve both pure Java implementation of a MO server and for CMIS compliant interfaces by making an explicit separation between the MOs, the external hierarchical view and the naming issues.

References

[1] *L. Andrey, O. Festor, N.B. Youssef, "*COJ: A free CMIS Compliant Java API and its Various Implementations", to appear in Proc. IFIP/IEEE "NOMS 2000", Hawaii, 10-14 April, 2000.

[2] C.-C. Shen and J. Y. Wei, "Network-Level Information Models for Integrated ATM/SONET/WDM Management," presented at IEEE/IFIP 1998 Network Operations and Management Symposium, New Orleans, USA, 1998.

[3] Rational Rose, www.rational.com