Towards systematic interoperability testing of distributed systems

Thomas Rings
Software Engineering for Distributed Systems Group,
Institute for Computer Science, University of Göttingen,
Goldschmidtstr. 7, 37077 Göttingen, Germany.
rings@cs.uni-goettingen.de

Abstract

This work suggests an approach towards a systematic methodology of developing and applying interoperability tests for distributed systems. The methodology includes interoperability assessment and dynamic interoperability testing.

1. Introduction

Interoperability testing checks if different systems can communicate properly with each other. Interoperability means that systems are able to communicate [8] whereas interoperation is "The use of interoperable systems".

In practice, interoperability testing is done in an ad hoc fashion by plugging systems in a pragmatic manner together. An example for interoperability testing events is the Plugtests™ [1] event organized by *European Telecommunications Standards Institute* (ETSI). Automation and formalization of the testing process in such events may increase the testing effectiveness. This paper discusses an approach for establishing and applying a systematic process for interoperability testing.

2. Systematic interoperability testing

A systematic process for interoperability testing should follow formal guidelines and standard procedures. It should be applicable to different systems (generic) and no manual work should be required for test execution. The process itself comprises interoperability assessment, interoperability achievement, and dynamic testing. First, a method for interoperability assessment is required to evaluate if certain conditions for interoperability are fulfilled. If yes, dynamic testing can be applied. Otherwise, further strategies for interoperability achievement have to be investigated.

3. Assessing interoperability

We compare interfaces of systems to assess their interoperability, and advice how interoperability can be achieved. Therefore, we analyze the system capabilities, e.g., security or data transfer. System capabilities can be defined by different specifications. For example, the specifications of GridFTP [7] and of OGSA-ByteIO [10] belong both to the capability data transfer. Furthermore, different implementations of the same specification may exit due to several implementation options offered by the specification. For each implementation, a detailed capability description has to be added to a model, which in the following is called capability-model. The analysis of the capability-model leads to interoperability gaps, but also points to common agreed standards.

The capability-model will be enhanced with information about the behavior at the interfaces of the systems. The allowed behavior of an interface can be described by an automaton, in the following called interface-automaton. To assess interoperability, the relation between the interface-automata needs to be investigated. A prerequisite of interoperability of two systems is joined behavior of their interface-automata. In a later stage, the joined behavior may be used for test selection, or even the automatic generation of interoperability tests.

We studied the assessment of grid interoperability in the project "Study of ICT Grid interoperability gaps" [4] funded and organized by ETSI and the *European Commission* (EC). In the grid domain, different communities implementing their own grid systems coexist. To allow their comparison, the capability-model has to be instantiated. However, even if it seems that implementations interoperate because they implement the same specification, they may not be interoperable. For example, the capability for security in grid systems usually relies on the X.509 certificate and their instantiations in form of proxy certificates. The certificates may have varying lifetimes and can, therefore, lead to interoperability problems. As ongoing work at ETSI, we identify interoperability gaps and relevant, stable standards in grid systems. Furthermore, an abstract grid testing framework is under development [5].

The capability-model can be enhanced with solutions that overcome interoperability gaps. These include gateways, adapters, or common interfaces [6]. A gateway is a specific service that bridges between grid infrastructures so that one of them looks like a single resource to the other infrastructure. An adapter translates events of one interface into events that can be handled by another interface. The common interfaces approach requires that the systems implement the same specification. But especially in grid systems, interfaces are rarely standardized so that each system implements its own specifications. Grid systems are running stable and the willingness to change the interfaces to standardized ones is low. Therefore, we argue that the application of adapters is currently the most suitable approach to achieve interoperability between grid infrastructures. As a consequence of this observation, we started to investigate the automatic generation of adapters from the capability-model.

4. Dynamic interoperability testing

Theoretically, interoperability testing of several systems requires that each system is tested against all other systems. This approach is cost and time intensive and is, therefore, not commonly applied. Another approach is to certify a reference system, also called *Qualified Equipment* (QE). It is then required to test the systems that should be interoperable against the QE. Unfortunately, such certifications are also problematic.

As a pragmatic way, interoperability tests with conformance checks can be deployed. The test configuration is depicted in Figure 1. Complete interoperability testing of two systems can to be undertaken while conformance at the common interface is monitored [3].

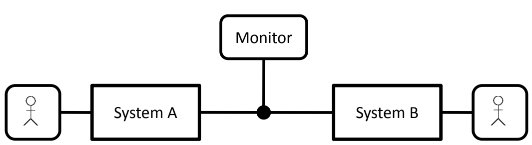

Figure 1. Interoperability testing with conformance checks

The focus of our work is to formalize and automate the interoperability testing process. This includes a formalization of the capability-model, the automatic generation of adapters from the capability-models as well as test selection or even the automatic generation of interoperability tests based on the joined behavior of interface-automata.

As a case study, we started to develop an adapter for interoperation of the grid infrastructure UNICORE [11] and the grid-like system Amazon elastic compute cloud [9]. An interoperability test environment based on *Testing and Test Control Notation* (TTCN-3) [2] will be developed.

Acknowledgments This work is supported by ETSI TC GRID STF331 and the European Commission under contract SA/ETSI/ENTR/000/2006-10.

References

[1] ETSI. Plugtest - Interop Event. [Online; `http://www.etsi.org/PLUGTESTS/` fetched on 01/16/09].

[2] ETSI. ETSI ES 201 873 V3.2.1: The Testing and Test Control Notation version 3; Parts 1-8. European Telecommunications Standards Institute (ETSI), Sophia-Antipolis, France, also published as ITU-T Recommendation series Z.140, 2007.

[3] ETSI. ETSI ES 202 237: Methods for Testing and Specification (MTS);Internet Protocol Testing (IPT);Generic approach to interoperability testing. European Telecommunications Standards Institute (ETSI), Sophia-Antipolis, France, 2007.

[4] ETSI. ETSI TR 102 659: GRID; Study of ICT Grid interoperability gaps. Technical report, European Telecommunications Standards Institute (ETSI), Sophia-Antipolis, France, 2008.

[5] ETSI. ETSI TR 102 766: GRID;ICT Grid Interoperability Testing Framework and survey of existing ICT Grid interoperability solutions. European Telecommunications Standards Institute (ETSI), Sophia-Antipolis, France, 2009.

[6] L. Field and M. Schulz. Grid interoperability: The interoperations cookbook. *Journal of Physics: Conference Series*, 119(1), 2008.

[7] GridFTP. [Online; `http://dev.globus.org/wiki/GridFTP` fetched on 03/20/09].

[8] IEEE. Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries, 1990.

[9] A. W. S. LLC. [Online; `http://aws.amazon.com/ec2/` fetched on 03/20/09].

[10] OGSA-ByteIO. [Online; `http://forge.ggf.org/projects/ByteIO-wg` fetched on 03/20/09].

[11] A. Streit, D. Erwin, T. Lippert, D. Mallmann, R. Menday, M. Rambadt, M. Riedel, M. Romberg, B. Schuller, and P. Wieder. UNICORE - From Project Results to Production Grids. *CoRR*, 2005.