Assuring Standard Conformance of Partial Interfaces

[Extended Abstract] *

Hardi Hungar
Institute of Transportation Systems
German Aerospace Center (DLR)
Braunschweig, Germany
hardi.hungar@dlr.de

ABSTRACT

A current standardization effort for track-side equipment in German railways faces the difficulty of having to proceed incrementally. This means that only some of the interfaces of complex entities like interlocking controllers are specified while others remain under the control of the diverse manufacturers. As these other interfaces are necessary for testing the specified ones for standard conformance, a specific approach has to be devised to be able to achieve this goal. This paper presents the problem in its practical setting and the way it is intended to be solved.

Categories and Subject Descriptors

C.0 [**Systems Specification Methodology**]: Interfaces; D.2.5 [**Testing and Debugging**]: Testing tools

General Terms

Verification and Validation

1. PROBLEM OVERVIEW

Notwithstanding the by now more than twenty years of efforts of standardizing railway control systems in Europe, proprietary interfaces and resulting incompatibilities between equipment components are still abundant. Any attempt at improving the situation faces the difficulty that the necessity of keeping the railway system in operation—defective equipment has to be replaced—and political demands—e.g. timeframes for new lines are set by third parties—that often a compromise between systematic and pragmatic solutions has to be found. Adding to this are the high costs of bying equipment and bringing it into operation. An approach currently employed by German railways is to incrementally specify the interface behavior of equipment components. I.e., only some of the interfaces of an interlocking system are specified (and shall be tested), while others remain to be considered somewhen in the future.

*This current version is a draft of a more complete position paper to be submitted in due course.

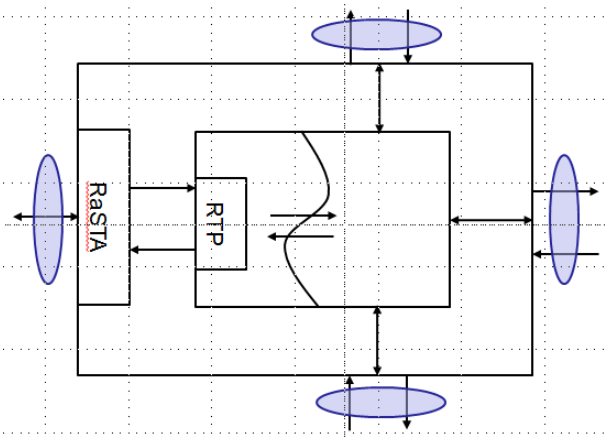

Figure 1: Schema of a system with four interfaces, of which one is to be specified.

From a system-theoretic point of view, this becomes problematic when moving from specification to testing. To drive the specified interface (and observe the correct interpretation of messages received over it), it is usually necessary to have access to (all the) other interfaces. Specification is easier by far—one can "internalize" the uncontrolled interfaces by subsuming everything in internal behavior of a specification automaton.

Fig. 1 shows a schematic view óf a system where one of the interfaces is to be specified. The specification shall address the functional level of the *Rail Technical Protocol* (RTP) and abstract from the concrete implementation of communication through the *Rail Safe Transport Application* (RaSTA) utilizes an ethernet connection.[1]

2. DETAILED SETTING AND SOLUTION APPROACH

2.1 Specification

The specification of the *focus interface* follows an operational approach, employing UML state machines. The specifying state machines serve to emit and accept telegrams over the focus interface, while internally switching states and operating on other auxiliary variables. Actions happening

[1]The technical details of the differences between RaSTA and RTP are of course important in practice but will not be discussed in depth in this version of the paper.

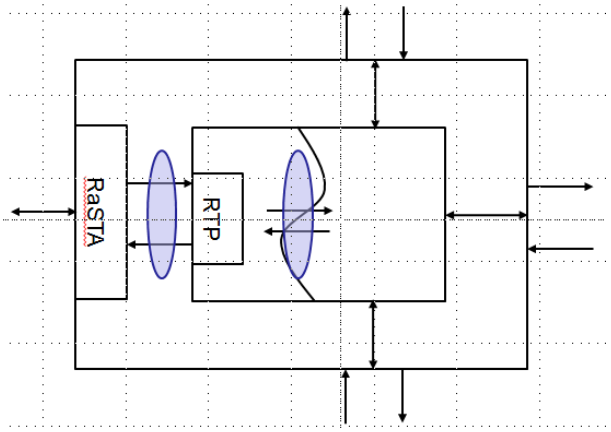

Figure 2: The specification view on the system.

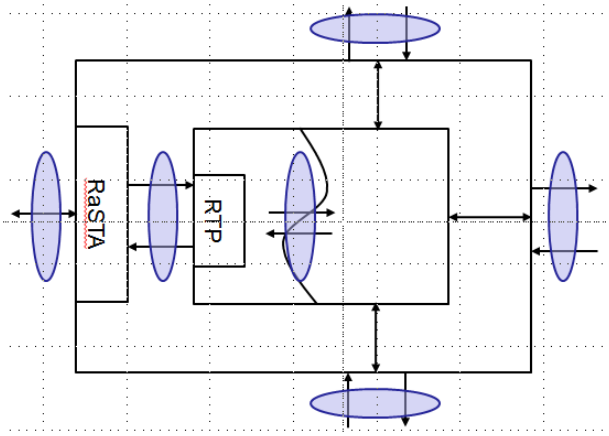

Figure 3: The combined testing and specification view.

on the other (*masked*) interfaces are reflected in *messages* (incoming) and *commands* (outgoing). The names of these messages and commands point to their meaning without any formal relation to actual actions. Fig. 2 depicts this view in introducing a virtual internal interface. The state machine does not have an explicit second interface. In particular, the "messages" are generated by the state machine and do not correspond to inputs.

2.2 Testing Problem

Testing the focus interface for conformance with the specification can of course not be done in terms of the internal specification interface but needs the real behavior on the masked interfaces. I.e., test cases and operation have to take the view of Fig. 1, while their derivation must refer to Fig. 2. The combined view is given in Fig. 3. The problem is acerbated by the unavailability of a precise relation between internal and masked interfaces. In current practice, such a relation does not even exist: There are considerable differences between the masked interfaces (whose standardization is yet to be initiated) in the different manufacturers' implementations of the devices.

2.3 Solution Approach

Differences between the solutions of different manufacturers call for integrating them into the test process in some way. Our solution relies on the assumed ability of the manufacturers of bridging the gap between (virtual) internal messages and commands and externals. The envisaged test architecture is depicted in Fig. 4.

The test rack adds two components to the test object:

Adapter internal-external: The manufacturer shall provide a module which translates between internal and masked interfaces. For its realization, interface drivers, simulators, or existing test interfaces accessing internals of the device may be used.

Adapter RaSTA-RTP: This module must be provided by the test laboratory.

The test rack serves to provide the test object with an interface which is on the same level of abstraction as the specification. The remaining components of the test architecture are rather standard.

Test Execution Kernel: The kernel controls the test execution, i.e., it initializes the test objects, starts test sequences (including parameter completion in advanced scenarios), protocols the results, performs corrective actions (breaks and restarts if necessary) and generally monitors the execution. The kernel will be partially automatized.

Test Sequences: A data base with test sequences.

Test Report: A data base for detailed result data and accumulated reports.

2.4 Validation

To be able to make qualified assertions of standard conformance, several arguments have to spelled out. On the one hand, the correctness and completeness, resp. sufficient coverage, of the test cases wrt. the specification has to be checked. This involves techniques and methods form the domain of model based testing. Currently, manually derived test suites are evaluated for their suitability. In future enhancements of the overall approach, also test case generation from the specification models is intended to be considered.

Adapter design and validation will have to cope with the common problems of crossing abstraction levels (namely atomicity and timing issues as well as value concretizations). For the internal-external adapter a monitoring concept which observes its operation dynamically is envisioned. The user interface of an interlocking systems provides many informations about internal states and thus qualifies as an adequate point of observation.

3. CONCLUSION

An approach to solve the problem of checking the standard conformance of complex rail devices wrt. partial interface specifications has been presented. The standardization concerns the functional interface aspect of the systems, including those real-time properties which are relevant to the systems' function (and safe behavior). The goal is to assert

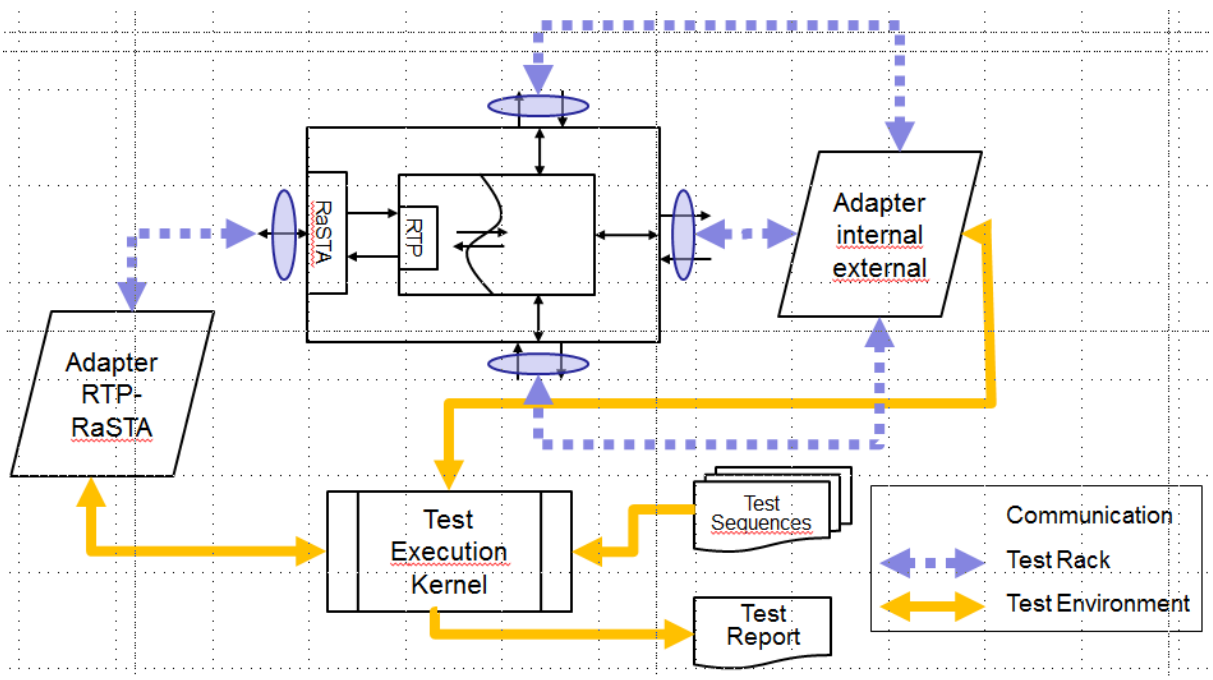

Figure 4: Components of the test architecture.

that system passing the test will be compatible in operation. The German Aerospace Center is involved in several ongoing activities which relate to the specific topic described here.