Graphical Test Specification – The Graphical Format of TTCN-3

Paul Baker[1], Ekkart Rudolph[2], and Ina Schieferdecker[3]

[1] Motorola Labs, Basingstoke, Hampshire, UK
paul.baker@motorola.com
[2] Technische Universität München, Institut für Informatik
Arcisstraße 21, D-80290 München, Germany
rudolphe@informatik.tu-muenchen.de
[3] GMD FOKUS, Berlin, Germany
schieferdecker@fokus.gmd.de

Abstract. Recently, the European Telecommunications Standards Institute (ETSI) approved the third edition of the Tree and Tabular Combined Notation (TTCN-3) as a requirement to modernise and widen its application beyond pure OSI conformance testing. As part of this evolution, TTCN is embracing Message Sequence Charts (MSCs) as a natural notation for specifying and visualising test suites. This paper defines the role of MSCs during test development, and more specifically introduces an MSC profile called the Graphical Format for TTCN (GFT) that facilitates the effective specification of TTCN-3 test suites.

1 Motivation

TTCN is a language used for test specification. However, experience has shown that the second edition [1, TTCN-2], a semi-graphical representation by means of a tabular format, has turned out not to be very intuitive for behaviour description, even if tools are used. For example, within TTCN-2 tables are used for the graphical representation of test cases, where statements are written on successive lines with either successively incremented indentations to indicate subsequent statements, or with equal indentations to indicate alternatives. In the case of highly nested alternatives, such a notation becomes very user-unfriendly. Consequently, with the third edition [2,3, TTCN-3] a textual language was developed that now looks more like a common programming language e.g., C or C++ or Java. Even though TTCN-3 makes the description of complex distributed test behaviour much easier there is still a requirement from the TTCN user committee to provide a visualisation means.

Message Sequence Charts (MSC) appeared to be a particularly attractive candidate as a graphical means for visualising TTCN. Therefore, in addition to the pure textual core language, the definition of other presentation formats has been admitted within TTCN-3. At present, two presentation formats are defined: a tabular conformance format [4] that resembles the tabular format of TTCN-2, and an MSC presentation format denoted as the Graphical Format for TTCN-3

R. Reed and J. Reed (Eds.): SDL 2001, LNCS 2078, pp. 148–167, 2001.

(GFT) [5]. GFT supports the presentation and development of TTCN-3 test descriptions on an MSC level. Thereby, the TTCN-3 core language may be used independently of GFT, but GFT cannot be used without the core language. Use and implementation of GFT shall be done on the basis of the core language. In the following, the TTCN-3 core representation is denoted briefly as TTCN-3.

GFT is based on the ITU Recommendation Z.120 for Message Sequence Charts [6] using a subset of MSC with test specific extensions, as well as extensions of a general nature. A main advantage of the MSC language is its clear graphical layout, which immediately gives an intuitive understanding of the described behaviour. Within the area of conformance testing, MSC is already well established for the specification of test purposes, and as such for the automatic generation of TTCN test cases [10]. Beyond that, MSCs have been proposed for a selected visualisation of TTCN descriptions by means of simulation techniques [7]. Although MSC has been used for limited test specification in the past, the latest version of the language now contains constructs that make the comprehensive MSC specification of test suites feasible. Such language constructs include MSC composition, object oriented modelling, as well as data. However, it should be pointed out that GFT is not intended as a standalone language, but as a basis for the generation of TTCN-3 descriptions. It may be possible that hybrid representations may turn out to be most effective, where only the main parts of the test description are visualised by means of MSCs, whilst the remaining parts are provided in the form of TTCN descriptions. Such a hybrid representation appears to be ideally tailored for a smooth transition from an MSC test specification to TTCN test case descriptions.

The possibility to clearly discriminate between different parts of a test description, and between different language constructs is one of the main points that are strongly in favour of using GFT.

The second advantage of GFT in comparison with TTCN-3 refers to the description of the communication behaviours between test components and their ports, and between test components via connected ports, and between test components and the system under test via mapped ports. Note that within GFT all ports may be represented by different port instances. Consequently the test events belonging to different ports are clearly separated visually, in contrast to TTCN-3 where all events appear in a mixed form. Within TTCN-3, the communication between test components via connected ports is provided in a fairly indirect manner. In contrast to that, GFT has the possibility to show the communication via connected ports in a more explicit manner either by means of 'MSC connectors' (which are introduced as an extension of the MSC language), or even by means of the explicit message flow for selected cases. The combined use of GFT and TTCN-3 can be compared in this respect with the combined use of MSC and SDL. Both SDL and TTCN-3 are component oriented, whereas MSC is communication oriented.

Using MSC as a presentation format for TTCN-3 may considerably improve the readability of test cases and make them more understandable. At the same time, MSC in the form of Sequence Diagrams forms a central constituent of UML