

Testen CORBA-basierter Systeme mit TTCN-3

Theofanis Vassiliou-Gioles¹, Ina Schieferdecker²

¹Testing Technologies IST GmbH,
Oranienburger Str. 65, D-10117 Berlin, Germany
vassiliou@testingtech.de

²Fraunhofer Research Institute for Open Communication Systems (FOKUS),
Kaiserin-Augusta-Allee 31, D-10589 Berlin, Germany
schieferdecker@fokus.fhg.de

Abstrakt

CORBA-basierte Systeme werden im TK und IT Sektor zur Realisierung verteilter Anwendungen wie im E-Business oder für Internet-Lösungen genutzt. Mit dem zunehmenden Einsatz CORBA-basierter Systeme in geschäfts- und sicherheitskritischen Bereichen nimmt der Bedarf an zuverlässigen Testlösungen zu. Mit TTCN-3 – der Testing and Test Control Notation – und der Anbindung von TTCN-3 an IDL – der Beschreibungssprache für Schnittstellen CORBA-basierter Systeme – steht nun eine Testlösung für systematische Tests auf Funktionalität, Leistungsfähigkeit, Sicherheit, etc. von CORBA-basierten Systemen zur Verfügung.

1 Ein Überblick zu TTCN-3

TTCN-3 – die Testing and Test Control Notation [1][5][6][7] – wurde auf der Grundlage jahrzehntelanger Erfahrungen beim Testen von Telekommunikationssystemen mit dem Anspruch entwickelt, das systematische Testen für verschiedenste Systeme und Arten von Tests zu unterstützen. TTCN-3 ist die bei ETSI (European Telecommunication Standards Institute) entwickelte und auch von der ITU-T (International Telecommunications Union – Telecommunications Sector) standardisierte Testspezifikations- und –implementierungssprache. TTCN-3 kann u.a. für funktionale, Integrations-, System-, Interoperabilitäts-, Konformitäts-, Skalierungs- und Lasttests eingesetzt werden, so dass mit einer Testmethode ein großes Spektrum der während des Systementwurfs notwendigen Tests unterstützt wird. TTCN-3 ist nicht nur eine Testmethode für Abnahmetests gegenüber Systembeschreibungen und/oder Standards, sondern kann somit während des gesamten Systementwurfs eingesetzt werden.

TTCN-3 wurde für das Paradigma reaktiver Systeme und Systemkomponenten entwickelt, deren Dienste über wohldefinierte Schnittstellen angeboten und genutzt werden können. Reaktive Systeme werden beispielsweise in der Telekommunikation, im Internet, in eingebetteten Systemen in der Automobil- und Flugzeugtechnik und in E-Business Systemen eingesetzt. TTCN-3 basiert auf dem so genannten Konzept des Black-Box-Testen, wobei das zu testende System als Black-Box betrachtet wird und nur über seine Schnittstellen stimuliert und beobachtet werden kann. Die Korrektheit des zu testenden Systems wird mittels eines Vergleichs von erwarteten und beobachteten Reaktionen des Systems auf Stimulationen an seinen Schnittstellen wie z.B. zur Anmeldung für einen Dienst oder bei der Nutzung eines Dienstes bewertet.

Während des Testens wird eine Reihe von sogenannten Testfällen ausgeführt. Testfälle sind Testzwecken zugeordnet, die beschreiben, welche Systemeigenschaften mit Hilfe der Tests nachzuweisen sind. Testfälle definieren detailliert die Prozeduren, die notwendig sind, um den Nachweis von Systemeigenschaften zu führen. Dabei werden mit TTCN-3 Testfälle abstrakt, also unabhängig von der Testplattform und den Testgeräten beschrieben. Werkzeuge wie Compiler und Ausführungsumgebung unterstützen die automatische Überführung der abstrakten Testfälle in ausführbare, die dann auf das System angewendet werden und die Testergebnisse liefern.

Die Testfälle werden durch Testkomponenten ausgeführt. Es gibt eine ausgewählte Testkomponente – die Master-Testkomponente – die erzeugt wird, wenn immer ein Testfall gestartet wird. Diese Master-Testkomponente führt das Testverhalten aus, das für den Testfall definiert wurde. Jede Testkomponente kann weitere parallele Testkomponenten kreieren, so dass für verteilte Tests beliebig viele Testkomponenten genutzt werden können, um das Systemverhalten zu bewerten. Testkomponenten können über verbundene Schnittstellen (die sogenannten Ports) Informationen austauschen bzw. mit dem zu testenden System interagieren. Die Menge der Testkomponenten und ihre Verbindungen untereinander und

zum testenden System werden als Testkonfigurationen bezeichnet. Die Testkonfigurationen sind in TTCN-3 dynamisch, da während der Testausführung Testkomponenten kreiert und terminiert als auch Verbindungen auf- und abgebaut werden können. Dynamische Testkonfigurationen sind für die heutigen zu testenden Systeme, da auch diese i.allg. während ihrer Laufzeit die Konfiguration dynamisch ändern, so dass das Testsystem im Verlaufe eines Tests angepasst werden muss.

Jede Testkomponente bewertet das zu testende System auf der Grundlage ihres lokalen Wissens. Ein Testfall terminiert, wenn die Master-Testkomponente beendet wird. Im Zuge dessen werden auch alle anderen Testkomponenten terminiert und die individuellen Testergebnisse zum Gesamtergebnis zusammengefasst.

Die Definition von Testfällen erfolgt in TTCN-3-Modulen, wobei über ein vollständiges Modulkonzept der flexible Import von Definitionen anderer Module erlaubt ist. Bestandteile eines Moduls sind Deklarationen von Konstanten, Typen, Daten, Funktionen, Testverhalten, Testfällen und die Kontrollfunktion. Module können parametrisiert werden, um sie an spezielle System- und Testkontexte anpassen zu können.

TTCN-3 Testspezifikationen können in verschiedenen Syntaxen – den sogenannten Präsentationsformaten – erarbeitet und dokumentiert werden. Die Präsentationsformate wurden definiert, um den Bedürfnisse verschiedene TTCN-3 Nutzer während des Testprozesses beim entwicklungsbegleitenden und Abnahme-Test entgegenzukommen. Die Core Notation von TTCN-3 ist textuell und an moderne Programmiersprachen angelehnt und auf Testentwickler und Programmierer ausgerichtet. Das graphische Präsentationsformat von TTCN-3 basiert auf Message Sequence Charts (MSCs) und visualisiert insbesondere das Testverhalten. Es erleichtert das Lesen und Dokumentieren von Testprozeduren, das Diskutieren darüber und das Repräsentieren und Analysieren von Testläufen. Das tabellarische Präsentationsformat von TTCN-3 hebt insbesondere strukturelle Aspekte von Typen und Daten hervor. Wenn Bedarf besteht, können weitere Präsentationsformate zum Herausstellen z.B. domainspezifischer Aspekte von Testprozeduren entwickelt werden und nahtlos in TTCN-3-basierte Testumgebungen integriert werden.

2 Testen von CORBA-basierten Systemen: die Abbildung von IDL nach TTCN-3

Die Common Object Request Broker Architecture (CORBA) ist eine weit verbreitete offene, verteilte, objektbasierte Infrastruktur, die durch die Object Management Group (OMG) standardisiert und in der mittlerweile in der Version 3.0 vorliegt. Die Interface Definition Language (IDL) dient dabei zur Definition der Schnittstellen von CORBA-basierten Systemen und den an diesen Schnittstellen zur Verfügung gestellten Operationen und Attributen. Die IDL Spezifikationen sind Ausgangspunkt für die Code-Generierung von CORBA-basierten Systemen.

Um IDL Spezifikationen in TTCN-3 Modulen zugänglich zu machen, ist eine Abbildung auf TTCN-3 [11] erforderlich. TTCN-3 verfügt über ein eigenes Typsystem mit Basisdatentypen wie Boolean, Integer, Floats und Strings und strukturierten Datentypen wie Records, Sets und Unions. Darüber hinaus erlaubt es das Referenzieren von Objekten, die in einer anderen Sprache definiert sind. Dazu muss eine in TTCN-3 äquivalente Darstellung für das fremde Objekt gefunden werden. Objekte aus anderen Sprachen können über eine explizite Abbildung nach TTCN-3 mittels eines vorgeschalteten Übersetzungsvorgangs eingebunden werden, oder aber diese Abbildung erfolgt implizit und wird durch die TTCN-3-basierte Testumgebung unterstützt. Dabei werden alleinig die TTCN-3 fremden Objekte per Import (unter Hinweis auf die Sprache der fremden Objekte wie ASN.1 (Abstract Syntax Notation One) oder IDL (Interface Definition Language) in ein TTCN-3 Module eingebunden. Compiler und Laufzeitumgebung realisieren die Abbildung nach TTCN-3 implizit – der Nutzer kann auf die fremden Objekte in ihrer ursprünglichen Form zugreifen.

Die Abbildung von IDL nach TTCN-3 wurde bei ETSI als separate technische Spezifikation [4] und als Ergänzung zu den TTCN-3 Standards verabschiedet. Die Abbildung generiert je IDL Schnittstelle die Port-Definitionen und je Operation und Attribut die Zugriffsmethoden als Signaturen in TTCN-3.

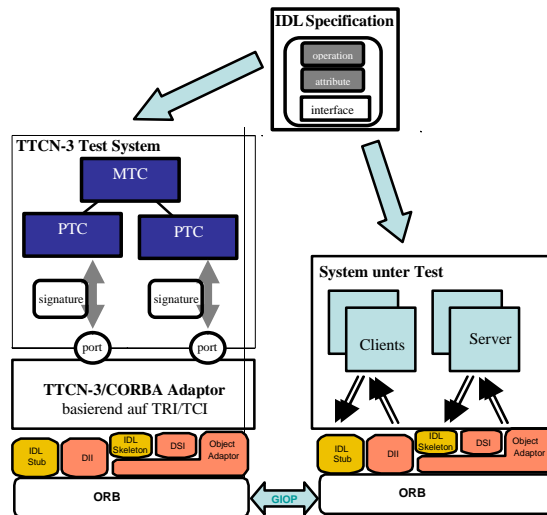


Bild 1: Testen CORBA-basierter Systeme mit TTCN-3 Tests

3 Werkzeug-Unterstützung

Damit wie in Bild 1 dargestellte Testumgebungen genutzt werden können, sind als Werkzeuge im Wesentlichen ein TTCN-3 Compiler mit einer Erweiterung um IDL und eine TTCN-3 Laufzeitumgebung mit einem Adaptor zwischen den TTCN-3-basierten Testsystemen und den auf CORBA basierenden Systemen unter Test notwendig. Entsprechend der Architektur von TTCN-3 basierten Testsystemen nutzt dieser Adaptor die Ausführungsschnittstellen von TTCN-3: das TTCN-3 Runtime Interface (TRI) und die TTCN-3 Control Interfaces (TCI).

Testing Technologies IST GmbH bietet den TTCN-3 Compiler TTthree [13] inklusive Laufzeitumgebung an und verfügt über ein β -Release für die CORBA-Anbindung. Erste Testsysteme für CORBA-basierte Systeme wurden aufgesetzt und demonstrierten den Vorteil spezifikationsbasierter, systematischer Tests im CORBA Umfeld. Mit derselben Testtechnologie lassen sich funktionale Tests, Interoperabilitätstests, Leistungs- und Skalierungstests entwickeln, dokumentieren und ausführen. Wesentliche Vorteile liegen dabei in der Homogenität des Testansatzes bei der Testentwicklung und den Werkzeugen der Testplattform als auch in der Möglichkeit zur Wiederverwendung von Testprozeduren für andere Testziele oder in anderen Testkontexten. Beispielsweise können funktionale Tests für Lasttests wieder verwendet werden.

4 Literatur

- [1] ETSI ES 201 873 – 1 , v2.2.1: *The Testing and Test Control Notation TTCN-3: Core Language*, Oct. 2002.
- [2] ETSI ES 201 873 – 5 , v1.0: *The TTCN-3 Runtime Interface (TRI); Concepts and Definition of the TRI*, Oct. 2002.
- [3] ETSI ES 201 873 – 6 , v1.0: *The TTCN-3 Control Interfaces (TCI); Concepts and Definition of the TCI*, Mar. 2003.
- [4] ETSI TS V1.0: *The IDL to TTCN-3 Mapping*. July 2003.
- [5] J. Grabowski, D. Hogrefe, G. Rethy, I. Schieferdecker, A. Wiles, C. Willcock: *An Introduction into the Testing and Test Control Notation (TTCN-3)*. - Computer Networks Journal, Vol.42, Issue 3, 2003.
- [6] I. Schieferdecker, A. Ulrich: *Die Testbeschreibungssprache TTCN-3*. – OBJEKTspektrum, Mai/Juni 2003, Heft 3/2003.
- [7] I. Schieferdecker, T. Vassiliou-Gioles: *Sicher wie ein Telefon - TTCN-3: Eine neue Testmethode für IT-Systeme*. - iX 11/01, Seite 56 ff, 2001.
- [8] S. Schulz, T. Vassiliou-Gioles: *Implementation of TTCN-3 Test Systems using the TRI*. IFIP 14th International Conference on Testing of Communicating Systems (TestCom 2002), Berlin (Germany), March 2002.
- [9] I. Schieferdecker, T. Vassiliou-Gioles: *Realizing distributed TTCN-3 test systems with TCI*, IFIP 15th Intern. Conf. on Testing Communicating Systems - TestCom 2003, Cannes, France, May 2003.
- [10] I. Schieferdecker, S. Pietsch, T. Vassiliou-Gioles: *Systematic Testing of Internet Protocols - First Experiences in Using TTCN-3 for SIP*. 5th IFIP Africom Conference on Communication Systems, Cape Town, South Africa, May 2001.
- [11] M. Ebner, A. Yin, M. Li: *Definitions and Utilization of OMG IDL to TTCN-3 Mappings*. IFIP 14th International Conference on Testing of Communicating Systems (TestCom 2002), Berlin (Germany), March 2002.
- [12] A. Wiles, T. Vassiliou-Gioles, S. Moseley, S. Mueller: *Experiences of Using TTCN-3 for Testing SIP and OSP*, 1st ATS Conference, ACATS Forum Conference, Athens, March 2002.
- [13] Testing Technologies' TTCN-3 Tools: *The TT Tool Series*, <http://www.testingtech.de/products>