

# Testen mit TTCN-3

## Vorteile und Nutzen einer mächtigen und zudem standardisierten Testtechnologie

Dieser Artikel beleuchtet technische Grundlagen sowie praxisrelevante Hintergründe, die 3GPP, WiMAX Forum und ETSI dazu veranlassten, im Rahmen ihrer Testaktivitäten TTCN-3 einzusetzen. Der Leser kann sich anhand des Artikels schnell einen Überblick über TTCN-3 basierte Testspezifikationen und Testwerkzeuge verschaffen.

Autoren: Theofanis Vassiliou-Gioles, Prof. Dr. Ina Schieferdecker

### TTCN-3, was ist das?

In den letzten 24 Monaten wählten verschiedene Standardisierungsinstitute des Telekommunikationssektors TTCN-3 als ihre Testspezifikationssprache. Neben Konformitätstests spezifizieren sie zum Beispiel Benchmarks mit Hilfe dieser modernen Testsprache.

Die „Testing and Test Control Notation“ (TTCN-3) ist eine relativ neue Sprache für die Spezifikation und Implementierung von Tests. Sie unterstützt jegliche Art von Tests für kommunikationsbasierte, verteilte Systeme, wie z.B. funktionale Tests, Leistungstests, Skalierungstests und Interoperabilitätstests.

Die Sprachentwicklung wurde 1998 initiiert. Eine vollständige Version stellte das European Telecommunications Standards Institute (ETSI) erstmals 2002 vor. Seither wird diese Version kontinuierlich auf Grundlage von Verbesserungsvorschlägen aus der Industrie weiterentwickelt.

TTCN-3 stellt eine vollständige Überarbeitung des alten TTCN-Standards dar (bekannt noch unter dem Namen *Tree and Tabular Combined Notation*), der insbesondere Testern aus dem Bereich GSM sowie ISDN bekannt ist. TTCN-3 besteht aus einer textuellen Kernsprache mit einer Schnittstelle zu verschiedenen Datenbeschreibungssprachen. Standardisiert sind zum Beispiel die Schnittstellen nach ASN.1 und IDL, jedoch unterstützen die Werkzeuge zunehmend auch weitere Datenbeschreibungssprachen wie zum Beispiel XML und WSDL zum Testen von Web Services.

TTCN-3 bietet neben der Möglichkeit, formalisierte Testspezifikationen in verschiedenen Präsentationsformaten zu beschreiben (textuell a la Skriptsprache oder graphisch in Form von Sequenzdiagrammen), auch eine Referenzimplementierungsarchitektur. Sie ermöglicht es Testlösungsanbietern, z.B. Testgeräteherstellern, diese Teststandards effizient zu implementieren und Nutzern zur Verfügung zu stellen.

Obwohl im gegenwärtigen TTCN-3 Standard neben der Skriptsprache auch zwei weitere Präsentationsformate definiert sind, verwenden Testentwickler und Standardisierungsgremien hauptsächlich die Skriptform. Das standardisierte grafische Format visualisiert Testsequenzen durch eine spezielle Form von Sequenzdiagrammen, wie sie z.B. aus UML bekannt sind. Es wird bevorzugt zur Dokumentation der Testskripte verwendet. Das tabellarische Format, welches hauptsächlich aus Kompatibilität zur Vorgängerversion spezifiziert wurde, hat keine praktische Relevanz. Da neben den standardisierten Formaten auch die Möglichkeit existiert, eigene Präsentationsformate zu definieren, ist TTCN-3 an verschiedene Anwendungskontexte anpassbar und flexibel einsetzbar.

Trotz ihrer Wurzeln in der Telekommunikationswelt ist die ausdrucksstarke Sprache TTCN-3 nicht nur dort einsetzbar, sondern überall, wo reaktive Systeme mit vielfältigen, lokalen oder verteilten Kommunikationsschnittstellen zu testen sind.

Neben der Unterstützung von nachrichten-orientierten Schnittstellen wird auch das Testen von prozedural-orientierten Schnittstellen unterstützt. Bei der ersten Klasse von Schnittstellen erfolgt der Austausch von Informationen mit dem zu testenden System anhand von asynchronen Nachrichten. Ein typisches Beispiel hierfür sind Protokolltests, wie das Testen von Internet-Protokollen, die in Voice-over-IP Lösungen vorkommen. Prozedurale Kommunikation ermöglicht dienstorientiertes Testen, z.B. von Web Services, Middleware-Plattformen und von objekt- und komponentenbasierten Applikationen. Eine der Stärken von TTCN-3 liegt in der Fähigkeit, beide Kommunikationsprinzipien gleichzeitig mit der gleichen Mächtigkeit unterstützen zu können. Dies ermöglicht das Testen heterogener Schnittstellen - wie sie in vielen integrierten Systemen in IT und Telco vorkommen - von einer zentralisierten Testlogik aus.

## Warum TTCN-3?

Die heutigen Testmethoden werden den Anforderungen heterogener, verteilter, integrierter Systeme nur noch schwerlich gerecht. Nicht nur das funktionale Testen einzelner Schnittstellen, sondern auch Aspekte der Integration bis hin zu vollständigen Ende-zu-Ende Systemtests stehen auf der Tagesordnung der Qualitätssicherung. Der Anspruch an eine höhere Testeffizienz und Testwiederwendung verlangt nach Testautomation. Spätestens, wenn einem Testsystem darüber hinaus noch Verteilungsaspekte von Testkomponenten und Skalierungs- und Lasttests abverlangt werden, stoßen Eigenentwicklungen häufig an ihre Grenzen.

TTCN-3 bietet für all diese Aspekte verschiedene Konzepte an, die es ermöglichen, einer vielfältig einsetzbaren Testplattform einen guten Schritt näher zu kommen. Neben einer fortschrittlichen Testtechnologie bedarf es letztendlich aber auch die Verfügbarkeit von Werkzeugen, die diese Konzepte effizient unterstützen – sowohl für die Entwicklung als auch für die Ausführung automatisierter Tests. Insbesondere in diesem Bereich hat es in den letzten Jahren signifikante Fortschritte gegeben. So wird der industrielle Einsatz in Unternehmen nun auch von zahlreichen Aktivitäten zur Teststandardisierung mit TTCN-3 in starkem Maße unterstützt.

## TTCN-3 in der Praxis

Die Vorteile von TTCN-3 liegen vor allem in der Möglichkeit, einfach und effizient Testarchitekturen zu beschreiben. Stimuli des Testsystems an das System unter Test und dessen Antworten werden über definierte Schnittstellen ausgetauscht. Das Testsystem kann dafür eine Anzahl von nebenläufigen Testkomponenten nutzen. Nebenläufige Testkomponenten führen ihr Testverhalten unabhängig von einander aus und können dabei mit dem zu testenden System oder untereinander interagieren.

Das folgende kleine Beispiel beleuchtet kurz die ausgewählten Eigenschaften von TTCN-3. Listing 1 zeigt den Ausschnitt einer WSDL-Definition, die den Zugang zum Google-Suchdienst beschreibt.

```
WSDL Datei
<message name="doGoogleSearch">
  <part name="key" type="xsd:string"/>
  <part name="q" type="xsd:string"/>
  <part name="start" type="xsd:int"/>
  <part name="maxResults" type="xsd:int"/>
  <part name="filter" type="xsd:boolean"/>
  <part name="restrict" type="xsd:string"/>
  <part name="safeSearch" type="xsd:boolean"/>
  <part name="lr" type="xsd:string"/>
  <part name="ie" type="xsd:string"/>
  <part name="oe" type="xsd:string"/>
</message>

<message name="doGoogleSearchResponse">
  <part name="return" type="typens:GoogleSearchResult"/>
</message>

<portType name="GoogleSearchPort">
  ...
  <operation name="doGoogleSearch">
    <input message="typens:doGoogleSearch"/>
    <output message="typens:doGoogleSearchResponse"/>
  </operation>
</portType>
```

**Listing 1 - Ausschnitt der WSDL Datei des Google-Suchdienstes**

Dieser Dienst wird nun daraufhin getestet, ob er verfügbar ist und ob er innerhalb einer vorgegebenen Zeit die Anfrage wie erwartet bedient.

Listing 2 zeigt einen einfachen Testfall, mit dem der Suchdienst entsprechend der WSDL-Definition abgefragt werden kann. Hierbei soll der (Test-)Client über den öffentlichen Web Service Port **GoogleSearchPort** eine Suchanfrage starten und dabei den Rückgabewert überprüfen.

```
GoogleSearchTest.ttcn3
 testcase TestSearch() runs on Client system SUT {
  map(self:gp, system:gp) ;
  # googleSearchPort.call(doGoogleSearch(qs, 2,0) {
  #   [!googleSearchPort.getReply(gs value expectedResults) {
  #     setverdict(pass) ;
  #   }[!googleSearchPort.getReply(gs value ?) {
  #     setverdict(fail) ;
  #   }[!googleSearchPort.catch {
  #     setverdict(fail) ;
  #   }
  # }
```

**Listing 2 - Ein einfacher TTCN-3 Testfall**

Der Testfall namens **TestSearch** führt dabei das folgende Verhalten aus.

Zuerst verbindet sich der Client mit dem Dienst (❶). Aus Testsicht bedeutet dies nichts anderes, als dass der Client nun beginnen kann, mit dem Server zu kommunizieren. In einem nachrichtenbasierten Szenario könnte der Client nun nicht nur Nachrichten senden, sondern auch empfangen.

Im zweiten Schritt (❷) wird der eigentliche Ruf des Web Services ausgeführt. Auf dem Kommunikationsport **googleSearchPort** wird der Aufruf **doGoogleSearch** initiiert. Hierbei wird maximal zwei Sekunden auf die Antwort gewartet.

Da die Reaktion des getesteten Systems bewertet werden soll, wird den möglichen Antworten eine besondere Bedeutung im Testfall zuteil: Aus Testsicht kann das System Unter Test (SUT) nun mit einer der drei Optionen antworten:

- a) Es antwortet mit der erwarteten Antwort.
- b) Es antwortet nicht mit der erwarteten Antwort.
- c) Es antwortet gar nicht innerhalb der spezifizierten Zeit.

Diese unterschiedlichen Reaktionen werden in so genannten Alternatives, eingeleitet durch [], beschrieben und bewertet. Die Alternative in Schritt ❸ beschreibt das Testfallverhalten, wenn das erwartete Ergebnis (**expectedResults**) eintritt. In diesem Fall wird das Verdict, das Testfallergebnis, auf **pass** gesetzt (❹). Im Falle, das System antwortet mit anderen als den erwarteten Rückgabewerten (❺), wird das Testfallergebnis auf **fail** gesetzt (❻). Ebenso wird ein **fail** zugewiesen (❾), wenn das System nicht innerhalb der geforderten zwei Sekunden antwortet und wie in Schritt ❷ beschrieben der timeout eintritt.

Die vollständige Behandlung des Beispiels, die an dieser Stelle nicht möglich war, kann bei den Autoren erfragt werden. Dennoch zeigen die hier präsentierten Aspekte bereits, mit welchen Mitteln TTCN-3 den Testentwickler aktiv unterstützt: die testrelevanten und immer wiederkehrenden Aktivitäten wie Datenvergleiche, Management des Verdicts etc. übernimmt die Sprache. Somit kann sich der Tester auf das Wesentliche konzentrieren, auf das „Was möchte ich testen“. Das technische „Wie“ übernimmt die Testumgebung.

## Standardisierung

Hier diskutieren wir Standardisierungsgremien und Industiekonsortien, die im Rahmen ihrer Testaktivitäten TTCN-3 einsetzen. So benutzt 3GPP (Third Generation Partnership Project, verantwortlich für UMTS) für alle neu zu entwickelnden Testsuiten ausschließlich TTCN-3. Diese Entscheidung wurde auch auf neue, noch in frühen Phasen der Standardisierung befindliche Technologien wie LTE (Long Term Evolution) ausgeweitet.

Dabei wird unter anderem von den Erfahrungen, die das WiMAX Forum im Rahmen seiner Zertifizierungsaktivitäten sammeln konnte, profitiert. Das erste Gremium, welches TTCN-3 basierte Testsuiten veröffentlichte, war zwar ETSI (für SIP, IPv6, und andere), jedoch nutzte das WiMAX Forum als erstes Gremium die Technologie TTCN-3 als Basis für sein Zertifizierungsprogramm: TTCN-3 wird hierbei zur formalisierten Beschreibung aller Testszenarien für die WiMAX-Protokollkonformität verwendet. Diese setzten wiederum Testgerätehersteller wie Aeroflex um und Zertifizierer wie AT4wireless ein.

Ein weiteres Gremium im Telekommunikationsumfeld, das mit TTCN-3 liebäugelt, ist die Open Mobile Alliance (OMA), die den Einsatz von TTCN-3 als Testsprache für das Testen von Enablers in Betracht zieht. Ein erstes Pilotprojekt evaluiert derzeit, wie der OMA Enabler BCAST mittels TTCN-3 getestet werden kann. Die OMG ging jedoch noch einen weiteren Schritt, um der großen UML Community TTCN-3 näher zu bringen: Sie band mit dem UML 2.0 Testing Profile TTCN-3 in ihre Modellierungssprachen ein.

In Hinblick auf die steigende Akzeptanz von TTCN-3 sticht besonders eine Standardisierungsaktivität außerhalb des Telekommunikationsumfeldes hervor. Das AUTOSAR Konsortium, welches sich der Standardisierung einer Softwareplattform für die Automobilindustrie verschrieben hat, evaluierte TTCN-3 im Rahmen eines umfangreichen Pilotprojektes. Im Ergebnis steht der Einsatz von TTCN-3 für die Spezifikation und Implementierung der AUTOSAR Validierungsplattform.

## Werkzeuge

Die Akzeptanz der Sprache im Testautomatisierungsbereich lässt sich auch an der Vielfalt der zur Verfügung stehenden TTCN-3 Werkzeuge ablesen. Acht verschiedene Anbieter von TTCN-3 Werkzeugen werden derzeit auf der offiziellen TTCN-3 Seite unter [www.ttcn-3.org](http://www.ttcn-3.org) gelistet. Dabei reicht die Spannbreite von Editoren über Compilern mit Laufzeitumgebung bis hin zu vollständigen, integrierten Entwicklungs- und Ausführungsumgebungen. Die Entwicklungsumgebungen enthalten dabei Editoren für den textuellen oder grafischen Entwurf von Testsuiten, Compiler zur Übersetzung der TTCN-3-Testsuiten in ausführbaren Code (zumeist Java oder C) sowie Testumgebungen zum Ausführen von Testsuiten (*siehe Abbildung 1*) Features wie Debugger oder auch verteilte Laufzeitumgebungen erhöhen dabei die Produktivität bei der Erstellung und Wartung von TTCN-3 basierten Testsystemen signifikant.

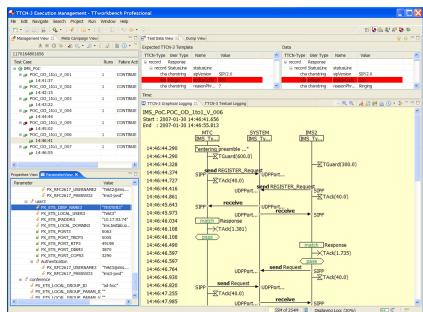


Abbildung 1 - TTworkbench, eine TTCN-3 IDE

Bei der konkreten Implementierung von TTCN-3 setzen insbesondere die Entwicklungswerkzeuge auf die Referenzimplementierungsarchitektur sowie deren zwei standardisierten Schnittstellen TRI (TTCN-3 Runtime Interface) und TCI (TTCN-3 Control Interface).

Diese Schnittstellenbeschreibungen zielen darauf ab, die einfache Anpassbarkeit einer TTCN-3 Testsuite an eine gegebene Testinfrastruktur zu gewährleisten. Dabei passt TRI die Kommunikationsschnittstellen des Systems unter Test an das Testsystem an und stellt Operationen zur Implementierung von Timer-Operationen zur Verfügung. TCI hingegen verwaltet die Ausführung der TTCN-3-Testsuite, realisiert die Verteilung der Testkomponenten und übernimmt die Testlaufprotokollierung sowie das Kodieren und Dekodieren der Testdaten (*siehe Abbildung 2*).

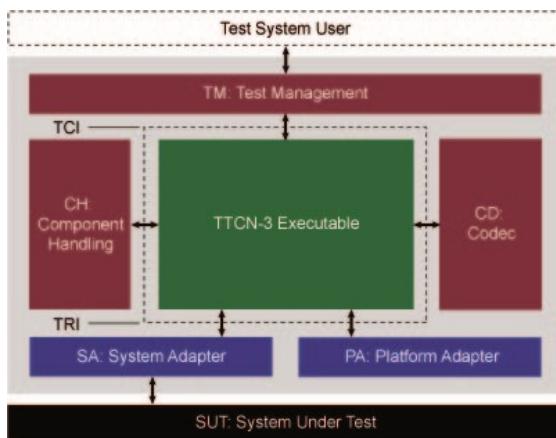


Abbildung 2 – Die TTCN-3 Referenzimplementierungsarchitektur

Neben reinen entwicklungsbegleitenden Werkzeugen bieten Testgerätehersteller wie Tektronix, Aeroflex oder auch Catapult Communications zunehmend sowohl TTCN-3 Unterstützung als auch TTCN-3 basierte Lösungen an. Gerätehersteller der Telekommunikations- und Automobilindustrie, wie Nokia, Ericsson, Siemens oder Daimler-Chrysler, haben in Pilotprojekten erfolgreich TTCN-3 basierte Tests aufgesetzt und nutzen sie mittlerweile im Rahmen der industriellen Software-Produktion. Die Standardisierung von TTCN-3, die ETSI Hersteller-übergreifend betreibt, ermöglicht TTCN-3 Anwendern eine freie Wahl der Testwerkzeuge und sichert die Nachhaltigkeit ihrer Testinvestitionen.

Mit der domänen- und technologieübergreifenden Ausbreitung von TTCN-3 (auch durch Standardisierungsgremien vermittelt), wächst der Nutzerkreis kontinuierlich, was sich wiederum nutzbringend auf die Weiterentwicklung der Werkzeuge und der Testtechnologie auswirkt. Neben der Verfügbarkeit von Werkzeugen ist die Verfügbarkeit von Ausbildungs- und auch Zertifizierungsmöglichkeiten für TTCN-3 von großer Bedeutung, insbesondere für einen breiten industriellen Einsatz. Das German Testing Board setzte hierzu in Kooperation mit ETSI und iSQI als Zertifizierer das TTCN-3 Certificate® auf, welches sich als anerkanntes Ausbildungsschema und -nachweis für TTCN-3 durchsetzt. Mit dem TTCN-3 Certificate können Ingenieure ihr TTCN-3 Wissen unter Beweis stellen und Firmen gezielter qualifiziertes Personal akquirieren.

## Zusammenfassung / Ausblick

Nach ersten punktuellen Erfolgen registriert TTCN-3 in den letzten fünf Jahren eine kontinuierlich wachsende Nutzergemeinde und eine Vielzahl erfolgreicher Roll-Outs in die Industrie. Neben der Standardisierungs- und Zertifizierungswelt im Telekommunikationssektor setzen immer mehr industrielle Anwender auf TTCN-3, um ihre Testautomatisierungsprozesse zu vereinheitlichen. Aus der Resonanz zahlreicher Unternehmen lässt sich eine Anzahl von mehr als 10.000 Nutzern der TTCN-3 Technologie ableiten.

Diese Entwicklung ist dabei nicht auf Europa begrenzt, sondern ebenso im asiatischen wie auch im nordamerikanischen Raum zu beobachten. So findet beispielsweise im Oktober dieses Jahres die erste TTCN-3 User Conference (T3UC-Asia) in Peking, China statt. Als Spin-Off der in diesem Jahr bereits zum vierten Mal ausgerichteten T3UC in Europa wurde sie auf expliziten Wunsch der Nutzergemeinde ins Leben gerufen.

TTCN-3 hat insbesondere im Telekommunikationsbereich eine hohe Nutzbarkeit erreicht. Testgerätehersteller, industrielle Anwender sowie die wachsende Zahl an Werkzeuganbietern arbeiten gemeinsam daran, den Standard weiter zu etablieren und dem Anspruch einer domänenübergreifenden Technologie gerecht zu werden. Darüber hinaus weisen Erfolge, beispielsweise in der Automobilbranche, darauf hin, dass richtige Weichen auch in anderen Bereichen gestellt wurden.

### Links

[www.ttcn-3.org](http://www.ttcn-3.org)  
[www.testingtech.com](http://www.testingtech.com)  
[www.etsi.org](http://www.etsi.org)  
[www.german-testing-board.info/en/ttcn3\\_certificate.shtml](http://www.german-testing-board.info/en/ttcn3_certificate.shtml)

## Email der Autoren

Theofanis Vassiliou-Gioles [vassiliou@testingtech.com](mailto:vassiliou@testingtech.com)

Prof. Dr. Ina Schieferdecker: [schieferdecker@fokus.fraunhofer.de](mailto:schieferdecker@fokus.fraunhofer.de)

### Der TTCN-3 Standard

Der TTCN-3-Standard von ETSI trägt die Nummer ES 201 873 und ist über die ETSI Webadresse <http://www.ttcn-3.org> beziehbar. Die International Telecommunication Union (ITU) hat den Standard unter der Bezeichnung ITU-T Recommendation Z.140 Series übernommen.

Im Rahmen der Standardisierung von UML 2.0 durch die OMG wird TTCN-3 auch als UML Testing Profile (U2TP) dem wachsenden UML-Anwenderkreis zugänglich gemacht.