



□ Dr. Frank Simon

(frank.simon@bluecarat.de)

hat Informatik studiert und im Bereich der Qualitätssicherung großer IT-Systeme promoviert. Heute ist er in der Geschäftsleitung der BLUECARAT AG und für das Business Development verantwortlich. Er ist Vorsitzender des BITKOM Lenkungsausschuss Software sowie im Vorstand des German Testing Boards (GTB).

I14Y-Testing: Warum Interoperabilitätstesten schwierig auszusprechen, aber absolut notwendig ist

Ungeachtet technischer Details von Internet of Things (IoT), Machine-to-Machine (M2M) und Industrie 4.0 bleibt ein Aspekt gleich: Die Anzahl von Schnittstellen vergrößert sich täglich. Und diese Schnittstellen sind Fluch und Segen zugleich: Segen, da sie neue Geschäftskanäle öffnen können (z. B. für Partnerportale) und Geschäftsprozesse unternehmensübergreifend einfach zusammen orchestriert werden können. Fluch, da sie natürlich auch Einfallstor von Angriffen, Orte von Falschbenutzungen (z. B. Robustheit) und Kristallisationspunkte guter Nutzbarkeit und Dokumentation sind. Das Ziel dieser Schnittstellen ist Interoperabilität (engl. Interoperability) oder kurz I14y (vgl. [wik]). Interoperabilität ist dabei nicht automatisch gegeben, sondern muss ebenso wie andere Qualitätseigenschaften anderer IT-Artefakte geplant und getestet werden. Und dieser Test ist mindestens ebenso nicht-trivial, wie das Aussprechen der entsprechenden Testdisziplin „Interoperabilitätstesten“. Dieser Artikel gibt einen Überblick über das Testen von Interoperabilität, dessen Wichtigkeit in Zukunft von IoT und M2M deutlich zunehmen wird.

Die Schnittstellenflut

Was haben IoT, M2M und Industrie 4.0 gemeinsam? Die IT verlässt ihren schützenden Bereich von Host, Desktoprechnern und Laptops und integriert Kühlschränke (Internet der Dinge), Sensoren (Maschine-zu-Maschine-Kommunikation) und ganze Produktionsstraßen (Internet 4.0). Je nach Analystenhaus wird aktuell für 2020 mit 26 bis 40,9 Milliarden verbundener Geräte gerechnet (vgl. [Venk14]). Jedes dieser Geräte hat Schnittstellen, über die es mit anderen Systemen Informationen austauscht.

Neben zusätzlichen Schnittstellen durch zusätzliche Geräte gibt es noch eine viel „leisere“, aus dem Business heraus aber mindestens genauso starke Quelle für neue Schnittstellen, die häufig unter dem Begriff „API-Management“ subsumiert wird (vgl.

z. B. [PoSi14]). Hierbei wird das Konzept der „SOA-fizierung“ auf unternehmensübergreifende Kommunikation ausgeweitet, d. h. bisher monolithische Systeme werden künstlich aufgebrochen, um an den Bruchstellen über neue Schnittstellen mit anderen Unternehmen als Provider zusätzliches Business zu generieren: „API Management enables new business models that generate gains in market share“ [PoSi14].

Die Anzahl solcher Schnittstellen nimmt bereits aktuell rasant zu: So ist alleine die Anzahl von kommerziellen Web-APIs zwischen März und Oktober 2013 um über 3000 zusätzliche Schnittstellen gewachsen (vgl. [Prog13]).

Der Erfolg all dieser Schnittstellen hängt im Wesentlichen von zwei Faktoren ab, die jeweils eine Sicht auf Schnittstellen repräsentieren:

- **Schnittstelle als Informations-Provider:** Hier wird eine Schnittstelle angeboten, über die bestimmte Informationen beziehbar sind. Je zahlreicher/intensiver/profitabler die Schnittstelle genutzt wird, desto besser für den Informations-Provider.
- **Schnittstelle als Informations-Consumer:** Hier wird eine Schnittstelle für das eigene Business genutzt, um bestimmte Informationen darüber zu beziehen. Je einfacher/zuverlässiger/profitabler die Schnittstelle genutzt wird, desto besser für den Informations-Consumer.

Die Funktion einer Schnittstelle ist also in jedem Fall der (richtungsunabhängige) Informationsaustausch darüber. Die Disziplin, die sich genau mit diesem Phänomen

auseinandersetzt, ist die Interoperabilität. Die ISO 25010 versteht unter Interoperabilität „degree to which two or more systems, products or components can exchange information and use the information that has been changed“ (vgl. [ISO25010]).

Jede Schnittstelle hat eine gewisse Interoperabilität und in Zeiten der zunehmenden Vernetzung wird diese Ausprägung mehr und mehr zum marktbeherrschenden Distinktionsmerkmal. Der Definition entsprechend genügt hier aber nicht nur eine rein technische Machbarkeit des Informationsaustausches. Vielmehr bedeutet das „can exchange“ und „can use the information“ den gesamten Qualitätskanon der ISO 25010 für Schnittstellen.

Die drei wesentlichen Herausforderungen sind hierbei:

- **Funktionalität:** Die Schnittstelle muss funktional verwendbar sein, d. h. die übertragenen Informationen müssen funktional korrekt, vollständig und passend sein.
- **Performance:** Die Schnittstelle muss performant sein, d. h. die übertragenen Informationen müssen in angemessener Zeit und mit angemessener Ressourcen-Nutzung verfügbar sein.
- **Sicherheit:** Die Schnittstelle muss sicher sein, d. h. die Daten müssen vertraulich, integer und zugreifbar sein, die Nutzung muss nachvollziehbar und authentifiziert sein.

Je nach Schnittstellenart können auch Usability, Portabilität, Wartbarkeit und Zuverlässigkeit noch wichtig sein.

Um die Interoperabilität einer jeden Schnittstelle, heruntergebrochen in die Untermerkmale, sicherzustellen und jederzeit Transparenz über den aktuellen Status quo zu haben, bedarf es ihres dedizierten Testens.

Interoperabilitätstesten

Resultiert aber nun daraus, dass die neuen Trends IoT, M2M und Industrie 4.0 die Motivation für Interoperabilitätstesten sind, zwangsläufig ein neues Testen?

Aus einer naiven Gesamtsystemsicht heraus kann der Interoperabilitätstest als rein fakultativ aufgefasst werden: Denn wenn der End-to-End-Test erfolgreich verläuft, scheinen mögliche involvierte Schnittstellen funktioniert zu haben, unabhängig ihrer Interoperabilität. Den

End-to-End-Tester mit seiner Black-Box-Sicht interessiert eventuell überhaupt nicht, ob Schnittstellen involviert sind oder nicht.

Die Interoperabilität wird demzufolge über den End-to-End-Test implizit mitgetestet. Grundsätzlich hat diese Sichtweise zwei Defizite:

- **Ignorieren früher Teststufen:** Bereits der vom ISTQB geprägte Testbegriff beinhaltet explizit „all life cycle activities [...] concerned with planning, preparation and evaluation of software products“ [ISTQB]. Grundsätzlich gilt, dass Fehler am effizientesten und effektivsten zu dem Zeitpunkt identifiziert und behoben werden können, an dem sie in das Produkt induziert wurden. Im Idealfall werden also im Systemabnahmetest keine Fehler mehr gefunden, die bereits früher im Lebenszyklus hätten gefunden werden können. Dieses Grundparadigma wird heute häufig als „Shift-Left“ bezeichnet (vgl. z. B. eine der frühesten Begriffsnutzungen in [Smit01]).

Der Interoperabilitätstest ist demzufolge eine frühe Teststufe und identifiziert Fehler, die bereits auf Schnittstellenebene, d. h. bevor der Service in größere zusammenhängende Prozessketten orchestriert wird, zu Tage treten. Wird er ignoriert, fallen Fehler – wenn überhaupt – erst im nachgelagerten Systemabnahmetest auf, in dem die Behebung allerdings deutlich aufwendiger ist und der meist zeitlich sehr kurz vor einem geplanten „Go-Live“ liegt, sodass wenig Möglichkeiten der Kompensation bestehen.

- **Ignorieren geänderter Qualitätseigenschaften:** Auch wenn der Qualitätskanon für Schnittstellen grundsätzlich derselbe ist, wie er für ganze Systeme anwendbar ist, so können die Gewichte deutlich variieren: So kann eine reine End-to-End-Testbetrachtung, bei der sicher ist, dass beide Enden innerhalb des Unternehmens lokalisiert sind, den Bereich der Sicherheit herunterpriorisieren.

Dabei wird aber u. U. übersehen, dass für die Bereitstellung des Dienstes andere Schnittstellen außerhalb der Unternehmung genutzt werden und so das Thema Sicherheit erst aus dieser White-Box-Analyse heraus relevant wird. Auch scheinbar datenschutzir-

relevante End-Funktionen können im Hintergrund sehr wohl datenschutzrechtliche Aspekte streifen, die es über Tests sicherzustellen gilt.

Zum Interoperabilitätstest gibt es also keine Alternative, wenn das System wirtschaftlich sinnvoll und architektonisch vollständig getestet werden soll. Die entlang des ISTQB dafür geeignete Teststufe ist der „Integrationstest“ mit dem Ziel, „Fehlerzustände in den Schnittstellen und im Zusammenspiel zwischen integrierten Komponenten aufzudecken“ [ISQTB].

Der Interoperabilitätstest ist dabei nichts fundamental Neues: Es handelt sich hierbei „lediglich“ um eine spezifische Art von Testobjekten mit spezifischen Testkriterien, die erfüllt sein müssen. Wie für jeden anderen Test auch bedarf es dafür eines Testplans, einer Analyse und Design-Phase, einer Durchführung und einer Auswertung. Auch die konkreten Testverfahren (von adhoc über explorativ zu äquivalenzklassenorientiert) unterscheiden sich nicht vom klassischen Testen, gehören aber für den Integrationstest neu angewendet (z. B. eben ein exploratives Testen einer WSDL-Schnittstellenbeschreibung).

Neue und alte Skills

Und trotzdem birgt der Interoperabilitätstest neue Herausforderungen an den Tester. Die wesentlichen Anforderungen hierbei sind:

- **Techniknähe:** Wie bereits im agilen Testen festgestellt (vgl. z. B. [Lin13]), haben die früheren Teststufen naturgemäß eine größere Techniknähe: Die beiden Extrempositionen sind hierbei auf der einen Seite der klassische End-to-End-Test auf Basis einer GUI-Nutzung und auf der anderen Seite der strukturorientierte, programmierte Unit-Test.

Der Interoperabilitätstest liegt dazwischen: Die Schnittstellen werden heute meist als Web-Schnittstellen verfügbar gemacht, sind über spezielle Sprachen bekannt zu machen (z. B. WSDL) und nutzen TCP/IP-basierte Kommunikation. Technisch kommen dabei SOAP und REST-Technologien zum Einsatz, die allesamt eher technikorientiert sind. Auch die Datenformate reichen von reinen Binärformaten, über XML bis zu JSON und sind ebenfalls eher technikorientiert.

Über diese Technologien sind sowohl Eingabe- als auch Ausgabewerte zu bedienen, was vom Tester ein entsprechendes Verständnis erfordert. Entlang des ISTQB-Kanons vertritt der Technical Test Analyst auf dem Advanced Level den für diese neue Techniknähe relevanten Syllabus.

- **Unvollständige Testumgebung:** Wenn ein System zum Funktionieren aus vielen über Schnittstellen miteinander kommunizierenden Teilsystemen besteht, stellt sich natürlich die Frage der Integrationsreihenfolge. Es ist fast nicht vermeidbar, dass bestimmte Komponenten noch nicht verfügbar, aber für den Test anderer Komponenten notwendig sind.

Ein Interoperabilitätstest liegt mitten in dieser Problematik, und es muss, um nicht in ein fortwährendes Warten oder sogar in eine Deadlock-Problematik zu kommen, nach Alternativen gesucht werden. Ansonsten kommt es zu den 51 % der Projekte, die „regular or frequent delays of testing cycles due to unavailable dependencies“ [Voke12] aufzeigen.

In der Vergangenheit wurden Begriffe wie *Mock-Objekte*, *Stubs* oder *Fakes* geprägt, denen allesamt gemeinsam ist, dass mittels separater Codierung versucht wird, das Fehlen einer Komponente durch eine Minimalimplementierung zu umgehen. Hier wird sich im Kontext von IoT, M2M und Industrie 4.0 eine Professionalisierung durchsetzen müssen, da bereits heute jedes System über 30 externe solcher Abhängigkeiten besitzt (vgl. [Voke12]): Dezentrale Individuallösungen via *Mocking* und *Faking* sind nicht nur ineffizient, sondern lösen auch nicht die Frage nach repräsentativen, wiederverwendbaren Testdaten, die über die simulierte Schnittstelle abgearbeitet werden sollen. Hier wird die Technologie der Ser-

vice-Virtualisierung verstärkt Einzug halten und damit den Interoperabilitätstest weiter beflügeln.

Ausblick

Dem Interoperabilitätstesten wird in Zukunft eine deutlich stärkere Bedeutung zukommen. In einem ersten Schritt handelt es sich hierbei lediglich um ein besonderes Testobjekt, das mit der gewünschten Interoperabilität ein besonderes Testkriterium erfüllen muss. Das Testgeschirr für einen derartigen Interoperabilitätstest ist bereits heute vorhanden.

In Zukunft wird die Interoperabilität von Schnittstellen allerdings noch ganzheitlicher aufzufassen sein: Da es immer mehr Schnittstellen für ähnliche Funktionalitäten gibt, werden diejenigen Schnittstellen an Marktbedeutung zulegen, die besonders gut dokumentiert sind, für die es bereits eine breite aktive und betreute Anwender-Community gibt, die neben der

Schnittstelle auch weiterführende Informationen wie Testdaten, Service-Virtualisierungen und typische Laufzeitverhalten angeben und die sich in bestehende Sicherheitskonzepte (z. B. OAuth als API-Autorisierung) effektiv einklinken.

Die heutigen kommerziellen API-Managementplattformen (vgl. z. B. von Microsoft und Axway in [PoSi14]) zeigen deutlich, wie ein effektives API aussehen kann. Interoperabilitätstesten wird entsprechend ganzheitlich aufgestellt sein und neben neuen Testobjekten wie API-Portalen, API-Beschreibungen und API-Testinfrastrukturen auch weitere Qualitätsattribute wie Usability, Wartbarkeit und Portabilität (z. B. von einem API zu einem anderen) sicherstellen.

Das klassische Testen nach ISTQB ist hier nach wie vor gültig und dient als Universalwissen, das allerdings durch spezifische Aspekte (z. B. UX-Testing) erweitert werden muss. ■

Referenzen:

[ISO25010] International Standard 25010: „Software engineering - Systems and Software Quality Requirements and Evaluation – System and software quality models“.

[ISTQB] ISTQB Glossary: “Standard glossary of terms used in Software Testing”, Version 2.3, 13. April 2014, verfügbar unter www.german-testing-board.info.

[Linz13] Tilo Linz: “Testen in Scrum-Projekten: Leitfaden für Softwarequalität in der agilen Welt”; dPunkt-Verlag, Heidelberg 2013.

[PoSi14] Uta Pollmann, Frank Simon: “Interoperabilität über Unternehmensgrenzen hinweg: Von SOA zum API-Management“, in Andreas Schmietendorf, Frank Simon (Hrsg.): „BSOA/BCloud 2014“, 9. Workshop, Shaker Verlag, Berlin, 2014.

[Prog13] ProgrammableWeb Research Center: „Growth in Web APIs from 2005 to 2013“, verfügbar unter: <http://www.programmableweb.com/api-research>.

[Smit01] Larry Smith: „Shift-Left Testing“, in Dr. Dobbs Magazine, verfügbar unter: <http://www.drdobbs.com/shift-left-testing/184404768>.

[Venk14] Archana Venkatraman: “IoT-enabled devices to top 40 billion by 2020, say researchers”, in ComputerWeekly.com, verfügbar unter: <http://www.computerweekly.com/news/2240227341/IoT-enabled-devices-to-cross-40-billion-by-2020>.

[Voke12] Voke Research: Market Snapshot report: Service Virtualization, by Threesa Lanowitz, Lisa Dronzek, December 11, 2012.

[wik] <http://de.wikipedia.org/wiki/Interoperabilität>