

# Geleitwort von Richard Mark Soley

## zur 1. Auflage

Laut Flugzeugbauer Boeing hat das Flugzeug vom Typ 747-400 ein maximales Bruttostartgewicht von fast 400.000 kg (einschließlich einer durchschnittlichen Anzahl von 416 Passagieren, einer Frachtladung von 171 Kubikmetern und über 200.000 kg Sprit). Vier Riesenmotoren treiben den Vogel mit bis zu 88% Schallgeschwindigkeit über unglaubliche Entfernungen – bis zu 13.500 km – an, ohne nachtanken zu müssen. Allein die Länge des Flugzeugs (45 m) ist länger als der gesamte Erstflug der Gebrüder Wright.

Doch diese erstaunlichen Zahlen, die nach 30 Jahren durch noch größere Passagierflugzeuge in den Schatten gestellt werden, sind nichts im Vergleich zur Komplexität des Systems bestehend wiederum aus Systemen (*System of Systems*), aus denen sich die Boeing 747-400 zusammensetzt. Das Elektrosystem des Flugzeugs umfasst etwa 274 km Kabel. Die hochfesten Aluminium- und Titanteile arbeiten sowohl im Stillstand als auch unter Hitzeeinwirkung durch das enorm schnelle Durchströmen der Luft. Backup-Systeme halten die Navigation und lebenswichtige Systeme in Gang für den Fall, dass das Hauptsystem ausfällt. Sogar das Unterhaltungselektroniknetz in der Kabine gehört zu den komplexeren Systemen auf der Welt. Die Boeing 747-400 umfasst etwa sechs Millionen Teile, wovon fast die Hälfte einfache Befestigungselemente wie Schrauben und Muttern sind. Kein Wunder, dass ein Boeing-Sprecher einmal witzelte: »Wir betrachten eine 777 als Sammlung von Teilen, die im geschlossenen Verband fliegen.« Als Vielflieger hoffe ich inständig, dass dieser »geschlossene Verband« eingehalten wird!

Alle diese Fakten spiegeln eine Tatsache wider, die jeder Ingenieur kennt: Komplexität ist schwierig zu handhaben, und die meisten interessanten Systeme sind komplex. Noch schlimmer: Die Zusammensetzung von Systemen zu Systemen, die wiederum aus Systemen bestehen, z. B. die Elektro-, Hydraulik-, Antriebs-

(Hub-, Schub- und Dauerbetrieb), Navigations- und weiteren Systeme des Flugzeugs in unserer Analogie, tendieren dazu, neue Komplexitäten in Form unerwarteter Teileüberlappung und unerwarteten Verhaltens von zuvor sich wohlverhaltenden Systemen einzuführen.

Als Folge der steigenden Komplexität von Systemen gibt es einen dringenden Bedarf, Komponentensysteme so beschreiben zu können, dass sich die Designs leicht unter Designerteams austauschen lassen. Eine gemeinsame Sprache ist eine entscheidende Komponente der Design-Methodologie eines jeden Systems oder Prozesses. Dabei spielt es keine Rolle, ob es sich um ein elektrisches, mechanisches oder chemisches System oder um Software handelt. Und wenn diese gemeinsame Sprache auf einer bereits existierenden basiert, wird das Team, das die Designs gemeinsam nutzt, rascher in der Lage sein, die Designaufgaben – und letztlich die noch anspruchsvolleren, langfristigen Wartungs- und Integrationsaufgaben – besser, schneller und billiger zu bewältigen.

Dieser Denkprozess führte die Object Management Group (OMG) mit ihren Hunderten von Mitgliedsfirmen aus aller Welt zur Entwicklung und Pflege der Systems Modeling Language (SysML). Der offene Standard weist eine Reihe positiver Merkmale auf:

- ❑ Die Sprache basiert auf einer Erweiterung der OMG-eigenen Unified Modeling Language (UML). Softwareentwickler, die mit UML vertraut sind, können daher leicht auf SysML umsteigen, und Werkzeugentwickler, die mit UML-Werkzeugen arbeiten, können SysML problemlos unterstützen.
- ❑ Die Sprache ist grafisch. Mich verblüfft es immer wieder, dass Softwareentwickler ihr Design mit Hilfe von Grafikelementen austauschen, dann aber z. B. »C++-Code« schreiben, um dieses Design auf den Computer zu übertragen. Ähnlich arbeiten Fertigungstechniker die Einzelheiten einer Produktfamilie mit Hilfe von Graphen, Kästen und Linien aus, um sie dann letztlich in Textform zu umreißen, z. B. mit PDES/STEP. Grafische Sprachen eignen sich naturgemäß für Designs und werden seit Tausenden von Jahren benutzt, wie etwa alte Gebäude- und Schiffsblaupausen, elektrische Schaltbilder usw. belegen.
- ❑ Die Sprache hat viele Implementierungen. Internationale Standards sind einfach nicht das Papier wert, auf dem sie gedruckt werden (oder den Festplattenplatz, den sie belegen), falls man sie nicht implementiert. Noch bevor SysML Anfang

2006 in der endgültigen Spezifikation verfügbar war, kündigten bereits mehrere Firmen die Implementierung des Standards an.

Wichtig ist dabei, dass es sich bei SysML nicht einfach nur um eine weitere Modellierungssprache für die Softwareentwicklung handelt. Software ist heute in nahezu allen komplexen Systemen zwar eine wichtige Komponente, aber fast nie die einzige. Flugzeuge, Züge und Autos beinhalten Software ebenso wie Gebäuderegelungseinrichtungen oder Chemiewerke. Sie alle umfassen aber auch u. a. Haustechnik-, Hydraulik- und Elektrosysteme, die es aufeinander abzustimmen und übergreifend zu entwerfen gilt. Ebenso müssen die komplexen Systeme, die von Menschen und automatisierten Prozessen genutzt werden sollen, auch entsprechend geplant werden. Nur so lassen sie sich sachgerecht und sinnvoll nutzen (sowie warten und im Hinblick auf neue Anforderungen in künftige Systeme integrieren). Bei all diesen designübergreifenden Fragen muss berücksichtigt werden, dass alle komplexen Systeme in mehreren Konfigurationen verfügbar sind, je nach Anforderung des Einsatzumfelds, der Mitarbeiterqualifikation und zahlreicher weiterer Variablen.

Das ist der Fokus von SysML, einer Sprache zur Modellierung all dieser unterschiedlichen Faktoren im Verlauf der Entwicklung komplexer Systeme und aus Systemen bestehender Systeme. Der Zweck ist dabei nicht das Verbergen der Komplexität jener Systeme, sondern vielmehr ihre Offenlegung und Kontrolle angesichts sich ständig ändernder Geschäftsanforderungen und Infrastrukturen.

SysML wird wahrscheinlich nicht die letzte Sprache für die Entwicklung komplexer Systeme sein. Durch den schnellen Wandel in den meisten Entwicklungs- und Konstruktionsbereichen (und der damit einhergehenden Einführung völlig neuer Bereiche, wie z. B. der Biotechnik) dürfte sich auch die Spezifikationssprache selbst ändern. Für einen Ingenieur sollte das keine große Überraschung sein, denn viele langlebige Spezifikationssprachen haben sich im Lauf der Zeit gewandelt. Ein einfaches Beispiel: Gebäudebaupläne gibt es schon seit mehreren hundert Jahren, diejenigen aus der Zeit bis etwa 1880 hatten aber kein internationales Symbol für elektrische Anschlüsse, da dieses Merkmal damals noch nicht existierte. Sprachen ändern sich, um sich an geänderte Konstruktionsanforderungen anzupassen. Vorteilhaft bei SysML ist, dass die Sprache auf einer Metamodell-Infrastruktur basiert – der Meta Object Facility (MOF), die ebenfalls von der OMG stan-

dardisiert wurde – und folglich für künftige Änderungen ausgelegt wurde. Die Technologie kann auch benutzt werden, um bestehende Designs, die in älteren Modellierungssprachen, wie etwa IDEF (eine häufig in der Luftfahrt und vielen anderen Bereichen verwendete Sprache), geschrieben wurden, umzuwandeln und zu integrieren.

Womit sich der Kreis schließt: Wir sind wieder angelangt bei der Sammlung von sechs Millionen Teilen, die im geschlossenen Verband fliegen. Die Aussicht, eine derart große Teilesammlung mit dem entsprechenden Bedarf an Menschen und Einrichtungen, Hardware und Software, Hydraulik, Elektrik und Navigation ohne eine gemeinsame Designsprache entwickeln zu können, die den konstanten Wandel von Komponentensystemen berücksichtigt, ist gleich null. SysML wurde eigens für diesen Bedarf ausgelegt. Die Sprache bewährt sich bereits in der Praxis für winzige, eingebettete Controller für Robotersysteme bis hin zu riesigen Industriekomponenten.

Lernen Sie also die Sprache und haben Sie Spaß dabei! Sie ist zwar nicht ganz einfach – komplexe Probleme bedürfen eben komplexer Lösungen –, aber logisch und beruht auf guten Designpraktiken. Und sorgen Sie letztlich dafür, dass jene Teile im geschlossenen Verband fliegen!<sup>1</sup>

*Richard Mark Soley, Ph.D.  
Chairman and Chief Executive Officer  
Object Management Group, Inc.  
30. Mai 2006*

---

<sup>1</sup>Das englische Original ist auf der englischen Webseite des Buches <http://www.system-modeling.com/foreword> zu finden.