

# „Wir wollten ein Schätztool, das die Kunden-Lieferanten-Zusammenarbeit unterstützt“

**Software-Projektmanagement-Pionier Barry W. Boehm über sein COCOMO-Modell**

Siegfried Seibert

*Fast täglich hören wir Meldungen über Projekte, bei denen Kosten und Termine nicht eingehalten werden. Mehr als die Hälfte der Softwareprojekte überschreitet ihre Termine und ihre Kosten um mehr als 50 %. Auch wenn einzelne dieser Überschreitungen nicht vorhersehbar sind, so können doch die meisten Fehleinschätzungen durch fundierte Schätzmethoden vermieden werden. Eines der bekanntesten Modelle zur Schätzung der Kosten von Softwareprojekten ist COCOMO („COConstructive COst Model“). Es wird von Tausenden von Projektmanagern weltweit eingesetzt, um die Kosten- und Terminalsituation von Softwareprojekten zu analysieren. Entwickelt und vorangetrieben wurde COCOMO von Professor Barry W. Boehm, einem der einflussreichsten und am häufigsten zitierten Fachleute im Software-Projektmanagement. projektMANAGEMENTaktuell sprach mit Barry Boehm über den Nutzen des COCOMO-Modells, über dessen Problembereiche und über die zukünftige Weiterentwicklung.*

**C**OCOMO ist ein Modell, mit dem Kosten, Aufwand und Termine von Softwareprojekten geschätzt werden können. Im Gegensatz zu anderen Kostenschätzsystemen ist COCOMO ein „offenes“ Modell, für das alle Einzelheiten veröffentlicht wurden, einschließlich der zugrunde liegenden Schätzgleichungen und jeder Annahme und jeder Definition, die den Gleichungen zugrunde liegen. Das Modell beruht auf der Auswertung von rund 250 Projekten in mehreren bekannten amerikanischen Unternehmen. Weil COCOMO genau spezifiziert ist und weil es nicht auf geheim gehaltenen Algorithmen beruht, weist es für seine Benutzer folgende Vorteile auf:

- COCOMO-Schätzungen sind objektiver und nachvollziehbarer als Schätzungen, die auf intuitiven Methoden oder auf (teureren) geschützten Modellen beruhen.
- Die COCOMO-Gleichungen können an die speziellen organisatorischen Randbedingungen eines Unternehmens angepasst (kalibriert) werden, um genauere Schätzungen zu ermöglichen.
- COCOMO ist bei kleinen Projekten einfach anwendbar, gleichzeitig aber mächtig genug, um damit auch sehr große Projekte zu planen und zu steuern.
- COCOMO kann bereits zu frühen Zeitpunkten eingesetzt werden, wenn noch kein abgesicherter Projektstrukturplan für eine detailliertere Projektkalkulation vorliegt.

Wegen dieser Vorteile ist COCOMO heute das bekannteste Softwarekostenschätzmodell weltweit. Es wurde von Boehm erstmals 1981 veröffentlicht. Damals war es eine kleine Sensation in der IT-Industrie. Seither haben sich die Softwareentwicklungsmethoden allerdings dramatisch ver-

ändert. Um diesem Wandel Rechnung zu tragen, wurde im Jahr 2000 ein komplett überarbeitetes und zukunftsweisendes Modell COCOMO II veröffentlicht.

**Herr Boehm, was waren Ihre wichtigsten Ziele, als Sie COCOMO entwickelten?**

Wir entwickelten COCOMO bei meinem damaligen Arbeitgeber TRW in den späten 70er Jahren, um bessere und realistischere Schätzungen für unsere Softwareprojekte zu erhalten. Außerdem wollten wir die Wirtschaftlichkeit der Softwareentwicklung erhöhen. Als eine der größten Softwareorganisationen arbeitete TRW damals bereits mit vielen externen Unternehmen zusammen. Ein wichtiges Ziel dabei war, mit einem solchen Modell die Zusammenarbeit von Kunden und Lieferanten zu unterstützen. Jede der beteiligten Parteien sollte das gleiche Tool zur Verfügung haben, um den Projektaufwand zu schätzen und zu analysieren, insbesondere bei der Diskussion und Verhandlung von Änderungsanträgen.

**Und dieses Ziel wurde erreicht?**

COCOMO war ein großer Erfolg. Viele Unternehmen in der ganzen Welt benutzen es und entwickelten eigene Anpassungen wie Ada COCOMO (US-Verteidigungsministerium), REVIC (Revised COCOMO), Incremental COCOMO oder SICOMO (Siemens). In jüngster Vergangenheit kamen zusätzliche Impulse für die systematische Kostenschätzung aus den Anforderungen des Capability Maturity Models CMML. Stufe 3 dieses Modells erfordert, dass die betreffende Organisation systematische Kostenschätzungen auf Basis quantitativer Erfahrungsdaten durchführt.



Foto: University of Southern California

Seit knapp 50 Jahren hat Professor Dr. Barry W. Boehm als Wissenschaftler, Manager und Hochschullehrer gearbeitet. In den 60er Jahren war er Leiter des Information Sciences Department der Rand Corporation, eines „Think Tanks“ der US-Regierung. In den 70er und 80er Jahren war er Chefwissenschaftler der Defense Systems Group bei TRW, einem kalifornischen Automotive- und IT-Unternehmen. Später arbeitete er als Direktor des Information Sciences and Technology Office im US-Verteidigungsministerium. Seit 1993 ist er Professor für Software Engineering und Direktor des Zentrums für Software Engineering an der University of Southern California (USC). Er hat im Beirat mehrerer wissenschaftlicher Magazine sowie in vielen Beratungsgremien und Komitees mitgewirkt und eine große Zahl nationaler und internationaler Ehrungen und Auszeichnungen erhalten. Boehm hat das Software Engineering und Management maßgeblich beeinflusst und gestaltet. Er war der Erste, der in den 60er Jahren die Software als den Hauptkostenfaktor zukünftiger Computersysteme identifizierte. Er entwickelte das COCOMO-Kostenschätzmodell. Er ist Erfinder des Spiralmodells für risikogetriebene, iterative Softwareprojekte, das später als Grundlage für den Rational Unified Process RUP diente und die derzeit intensiv diskutierten agilen Projektmodelle beeinflusste. Und er entwickelte bereits vor mehr als 15 Jahren den „Win-win-Approach“, mit dem die systematische Stakeholderanalyse in Softwareprojekte integriert wurde.

#### **Können Sie uns interessante Projekte nennen, in denen COCOMO erfolgreich eingesetzt wurde?**

Ja, davon gibt es eine ganze Reihe. Beispielsweise hatten die US Air Force und das Unternehmen General Dynamics (heute Lockheed Martin) lange Zeit Probleme, den Leistungsumfang neuer Software Releases für das Kampfflugzeug F-16 festzulegen. Jedes Jahr forderten Piloten, Betriebs- und Wartungspersonal viel mehr Ver-

besserungen, als mit dem vorhandenen Budget und Personal machbar waren. Irgendwann wurde es dem Einkaufsmanager der Air Force zu bunt und er lud die Verhandlungspartner von General Dynamics zu sich nach Hause ein. Dort sollten beide Seiten eine realistische Schätzung für den Änderungs- und Weiterentwicklungsumfang des nächsten Jahres erarbeiten. Wie bei der Papstwahl durften die Beteiligten das Haus nicht verlassen, bevor die gemeinsame Schätzung unter Dach und Fach war. Dazu benutzten sie das COCOMO-Modell. Es dauerte zwei Tage, bis sie zu einer übereinstimmenden Beurteilung kamen. Aber die erwies sich als außerordentlich realitätsnah und robust. Das Verfahren wenden sie nun schon seit mehr als zehn Jahren an, um die im nächsten Jahr zu realisierenden, neuen Anforderungen zu verhandeln.

#### **COCOMO scheint demnach besonders nützlich bei Projekten für öffentliche Auftraggeber?**

Nicht allein. Vor kurzem hatten wir ein Projekt, bei dem eine Bank versuchte, ihre Softwareentwicklungsproduktivität zu steigern. Der Chief Information Officer (CIO) war der Meinung, dass er mit einem einzigen linearen Prozentsatz das Produktivitätsverbesserungspotenzial und die erreichbaren Einsparungen abschätzen könne. Mit Hilfe des COCOMO-Modells zeigten wir ihm, dass Produktivitätsverbesserungen von den Anforderungen und Randbedingungen in jedem einzelnen Projekt abhängen. Wenn Mitarbeiter komplexere Software oder Software mit sehr hohen Echtzeitanforderungen entwickeln, produzieren sie weitaus weniger Befehlszeilen pro Personenmonat als bei kaufmännischen Anwendungen. Statt einfacher Produktivitätsprozentsätze wurden die Kostentreiber jedes einzelnen Projekts mit COCOMO geschätzt. Diese wurden dann als Basis für Programm- und Multi-projektschätzungen und -vergleiche genutzt. Mit diesen Schätzwerten kommt die Bank sehr gut voran.

#### **Wird COCOMO auch in kleineren Unternehmen für kleinere Projekte genutzt?**

COCOMO wird hauptsächlich in größeren Unternehmen der Luft- und Raumfahrt, der Telekommunikation, des Automobilbereiches sowie im öffentlichen und im Verteidigungssektor eingesetzt. In kleineren Unternehmen ist es nicht so stark verbreitet.

#### **Welche Arten von Schätzungen kommen in kleineren Unternehmen zum Einsatz?**

Meistens irgendwelche Analogieschlüsse. Die Schätzer fragen sich dann: Wie haben wir es letztes Mal gemacht, und welchen Aufwand haben wir dafür benötigt? Wenn Unterschiede zwischen dem früheren und dem neuen Projekt bestehen, werden entsprechende Zu- und Abschläge gemacht.

#### **In jüngster Vergangenheit kamen zusätzliche Impulse für die systematische Kostenschätzung aus den Anforderungen des Capability Maturity Modells CMMI**

#### **Kann ein parametrisches System wie COCOMO auch bei solchen Analogieschätzungen helfen?**

Ja, alle COCOMO-Kostentreiber können als Analogierelationen dienen, um die Auswirkungen geänderter



Foto: Lockheed Martin

Mit mehr als 4.200 Einheiten ist die F-16 das weltweit am weitesten verbreitete Mehrzweck-Kampfflugzeug. Software-Änderungen werden dort seit vielen Jahren erfolgreich mit dem COCOMO-Modell geschätzt. Der Systemumfang ist dabei exponentiell angestiegen: Der Kernspeicher der neuesten F-16-Generation ist 2.000-mal so groß und der Durchsatz 260-mal so hoch wie in der ersten F-16-Version im Jahr 1978.

Projektrandbedingungen abzuschätzen. Wir hatten einmal sogar ein Computertool entwickelt, mit dem COCOMO für Analogieschätzungen eingesetzt werden konnte.

*Im Sommer 2000 haben Sie ein komplett erneuertes COCOMO II veröffentlicht. Was wollten Sie damit erreichen?*

Bereits seit Anfang der 90er Jahre hatte die 1981er-Version von COCOMO zunehmend Probleme, mit einer Reihe neuer Entwicklungen in Softwareprojekten Schritt zu halten. COCOMO 81 beruhte auf Befehlszeilen (Source Lines of Code SLOC). Aber vielen Leuten fällt es schwer, den Umfang der Befehlszeilen für ein neues Projekt zu schätzen. Anwenderbezogene Schätzgrößen, wie die Funktionspunkte, gewannen an Bedeutung und wurden immer besser standardisiert. Für bestimmte Aufgaben, wie den Entwurf grafischer Benutzeroberflächen, waren Befehlszeilen sogar mehr oder weniger irrelevant. Die Entwicklungsprozesse in Projekten änderten sich. Das ursprüngliche COCOMO funktionierte sehr gut mit Wasserfallprozessen. Im Spiralmodell, im Rational Unified Process und in ähnlichen iterativen Projektabläufen werden die Meilensteine und die Projektpunkte aber anders gesetzt. COCOMO 81 war darauf nicht ausgerichtet. Außerdem gab es ganz neue Kostentreiber, wie die Entwicklungsprozessreife, neue Erkenntnisse zur Software-Wiederverwendung und Ähnliches mehr. Mit COCOMO II wollten wir ein Schätzmodell schaffen, das stärker auf die Anforderungen moderner Projekte ausgerichtet ist als auf die Randbedingungen der Vergangenheit.

*Sind die Funktionspunkte damit zur wichtigsten Messgröße für den Umfang von Softwareprojekten geworden?*

Funktionspunkte sind die wichtigste Messgröße im Bereich von Geschäftsanwendungen und Informationssystemen. Bei wissenschaftlichen Systemen und Echtzeitsystemen messen Funktionspunkte das, was sich innerhalb der Software abspielt, nicht besonders gut. Außerdem sind Funktionspunkte abgeschlossener Projekte bisher nur aufwändig „von Hand“ zu zählen. Wir experimentieren daher auch mit anderen Größenmaßen, wie den Konstrukten der Unified Modelling Language UML. Diese Größen, beispielsweise die Anzahl der Use Cases, der Klassen und Objekte einer Anwendung, sind mit den Funktionspunkten verwandt, können aber automatisch gezählt werden.

*Kann man das bereits praktisch einsetzen?*

Wir arbeiten darauf hin, haben aber ähnliche Hürden zu überwinden, wie sie die Funktionspunkteleute früher hatten. Jeder hat eine andere Vorstellung, was ein Use Case genau ist oder was eine Aktionsfolge ist. Was wir dringend benötigen, sind allgemein akzeptierte Standards zur Definition planungsbasierter und entwurfsbasierter Use Cases. Wenn man mehrere Projekte vergleicht, haben diese oft unterschiedliche Detaillierungsgrade in ihren Planungsdokumenten. Nicht selten nimmt dann die Anzahl der Use Cases zu, ohne dass sich am Umfang der Anwendung wirklich etwas geändert hat. In einem Forschungsprojekt haben wir zwischen der Anzahl der Use Cases und der Anzahl der Befehlszeilen lediglich Korrelationskoeffizienten von 0,4 bis 0,5 gemessen.



## COCOMO-Kostentreiber

### Organisationsfaktoren

- Erfahrung im Produktbereich
- Entwicklungsflexibilität
- Ausgereiftheit des Entwurfs
- Stakeholder-Zusammenhalt
- Software-Prozessreife

### Personalfaktoren

- Systemanalysefähigkeiten
- Programmierfähigkeit
- Anwendungserfahrung
- Plattformerfahrung
- Sprach- und Toolererfahrung
- Personelle Kontinuität

### Produktfaktoren

- Erforderliche Zuverlässigkeit
- Datenbankgröße
- Produkt-/Modulkomplexität
- Wiederverwendbarkeit
- Dokumentationsumfang

### Plattformfaktoren

- Rel. Rechnerzeitnutzung
- Rel. Hauptspeichernutzung
- Plattform-Änderungsdynamik

### Projektfaktoren

- Nutzung von SW-Tools
- Standortübergreifende Teamarbeit
- Verfügbare Projektdauer

Wir entwickeln COCOMO II laufend weiter, um mit dem Modell auch die Schätzanforderungen der kommenden Jahre und Jahrzehnte bearbeiten zu können. Hierzu wurden eine Reihe von Zusatzmodulen für spezielle Anwendungen und Projektanforderungen entwickelt. Beispielsweise gehört dazu das Modell COCOTS, mit dem die Kosten der Evaluation, der betriebspezifischen Anpassung und Integration bei der Anschaffung von Standardsoftware (COTS: Commercial of the Shelf) geschätzt werden können. In COCOMO II war dieser Projekttyp nicht gut genug abgebildet. Das Modell CORADMO ist auf schnell abzuwickelnde RAD-Projekte (Rapid Application Development) ausgerichtet. Die Teamstärke ist in solchen Projekten ganz anders als in Projekten, bei denen mehr Wert auf die Optimierung von Aufwand und Kosten gelegt wird. COCOMO und die meisten anderen Kostenschätzmodelle gehen standardmäßig davon aus, dass die Projektdauer (in Monaten) bei ungefähr dem dreifachen Wert der Kubikwurzel des Aufwands (in Personenmonaten) liegt. Bei RAD-Projekten liegt die Projektdauer eher bei der Quadratwurzel des Projektaufwands. Ein Projekt mit 25 Personenmonaten Aufwand benötigt dann als RAD-Projekt fünf Monate lang ein Team von fünf Leuten. Als „Normalprojekt“ würden wir nur drei Leute einplanen, dies allerdings für eine Laufzeit von etwa neun Monaten.

### *Sind diese Erkenntnisse noch im Forschungsstadium?*

Ja, das ist teilweise der Fall. Für Standard-COCOMO hatten wir ursprünglich Datenpunkte von rund 160 Projekten, die jetzt auf knapp 250 Projekte angewachsen sind. Für COCOTS haben wir 20 Projekte, für CORADMO erst knapp zehn Projekte.

### *Gibt es noch andere neue Entwicklungen?*

Seit das COCOMO-II-Buch veröffentlicht wurde, haben wir noch das Zusatzmodell COSYSMO entwickelt. Damit soll bei sehr großen Projekten der Aufwand zur Integration von Systemen geschätzt werden, die selbst wiederum aus größeren, bereits integrierten Teilsystemen bestehen. COSYSMO schätzt neben dem Aufwand zur

Integration der individuellen Teilsysteme auch den Zusatzaufwand, um diese Teilsysteme zu einem Gesamtsystem zu integrieren.

### *Benötigen moderne agile Prozessmodelle auch speziell darauf ausgerichtete Schätzmodelle?*

Ja, das tun sie sicher.

### *Insbesondere, wenn sie keine vollständige oder fast vollständige Anforderungsliste haben. Wie können Sie da überhaupt die Größe eines Projekts schätzen?*

Im Prinzip können wir hier eine vereinfachte Funktionspunktschätzung verwenden. Oder wir können Analogieschlüsse, Befehlszeilen- und Funktionspunktschätzungen kombinieren. Das ist zwar alles nicht perfekt, aber es ist das Einzige, was zu diesem Zeitpunkt bei einem solchen Vorgehen möglich ist.

*Herr Boehm, vielen Dank für das Gespräch.* ■

### Hauptveröffentlichungen von Barry Boehm zur Kostenschätzung

- [1] *Software Engineering Economics*, Prentice Hall, 1981.
- [2] *Software Risk Management*, IEEE Computer Society Press, 1989.
- [3] *Software Cost Estimation with COCOMO II*, Prentice Hall, 2000.
- [4] *Balancing Agility and Discipline: A Guide for the Perplexed*, Addison-Wesley 2003.

### Kontakt

Prof. Dr. Barry W. Boehm  
 University of Southern California  
 518 Adelaide Dr  
 Santa Monica, CA 90402  
 E-Mail: boehm@sunset.usc.edu  
 www.sunset.usc.edu/cse/