

Einführung der komponentenbasierten Anwendungsentwicklung - Auswirkungen auf die Organisation

Autor: Dieter E. Jenz, Febr. 1999

Die komponentenbasierte Anwendungsentwicklung eröffnet konkrete Chancen zur Steigerung der Software-Entwicklungsproduktivität. Dies äußert sich auf vielerlei Weise, so vor allem:

- in der Verkürzung der Entwicklungszeit neuer Anwendungssysteme und Anwendungen,
- in geringerem Personalbedarf im Entwicklungsprozeß,
- in der Verminderung des Finanzbedarfs für Entwicklung und Wartung.

Neben den ökonomisch orientierten Faktoren besteht außerdem die Chance, die Interoperabilität der Software zu verbessern, einen höheren Standardisierungsgrad der Software und einen höheren Qualitätsstandard zu erreichen. Schließlich können Mitarbeiter leichter in verschiedenen Projekten tätig werden, ohne sich jedesmal vollständig neu einarbeiten zu müssen. In der Summe kann sich das Anwenderunternehmen durch den intelligenten Einsatz der Informationstechnologie einen Wettbewerbsvorsprung erarbeiten.

Die genannten Argumente sind nicht neu und kaum jemand wird sich sträuben, diese anzuerkennen. Auch ist längst bekannt, daß etwa die Hälfte bis zwei Drittel der schöpferischen Design-Leistung ohne adäquaten Nutzen erbracht werden, da über sämtliche Anwendungen hinweg viele Gemeinsamkeiten hinsichtlich Architektur, Infrastruktur, Design, Funktionen und Code bestehen. Trotzdem wurden nur in wenigen Unternehmen wirklich ernsthafte Anstrengungen unternommen, die komponentenbasierte Anwendungsentwicklung konsequent anzugehen.

Hindernisse sind nicht in der Technologie zu suchen, sondern eindeutig in den über lange Zeit eingeschliffenen Denk- und Arbeitsmustern. Der schwarze Peter wird jedoch oft allein der Technologie zugeschoben, völlig zu unrecht. So wird allenthalben darüber räsoniert, daß die Objekttechnologie die in sie gesetzten Erwartungen nicht erfüllt habe. Die versprochene Wiederverwendbarkeit sei ausgeblieben. Natürlich bleibt Wiederverwendbarkeit aus, wenn sie nicht gezielt gefordert wird und als Qualitätsmaßstab für erfolgreiche Projekte gilt.

Die Technologie ist nur einer von drei Faktoren, die für Erfolg oder Mißerfolg der komponentenbasierten Anwendungsentwicklung verantwortlich sind. Die beiden anderen Faktoren sind Methoden und Verfahren sowie Organisation. In der Wertigkeit der Faktoren kommt der Technologie die geringste Bedeutung zu. Selbstverständlich gelten auch in technologischer Hinsicht einige Rahmenbedingungen, um die komponentenbasierte Anwendungsentwicklung zum Erfolg zu führen. Die zugrundeliegenden Technologien müssen eine hochgradige Stabilität erreicht haben und die Middleware, die zugleich systemorientierte Infrastruktur bereitstellt, muß hochgradig standardisiert sein. Die Aufwärtskompatibilität muß unbedingt gewährleistet sein, da nachträgliche Schnittstellenänderungen mit neuen Produktversionen auch zu Eingriffen in Anwendungs-Code führen können und auf diese Weise eine Vielzahl von Folgeaktivitäten auslösen.

Außerdem muß eine Anwendungsarchitektur entwickelt werden, die auf Basis bekannter Design Patterns eine Schichtenstruktur realisiert. Die Anwendungsarchitektur ordnet sämtliche Dienste, Schnittstellen und Protokolle in einen Kontext ein. Entweder kann eine vorhandene Architektur übernommen werden, die bereits auf die Branche zugeschnitten ist (z. B. Anwendungsarchitektur für Versicherungsunternehmen) oder die Anwendungsarchitektur muß selbst entwickelt werden. Frameworks basieren auf der Anwendungsarchitektur und stellen Architektur und Design für sämtliche Anwendungen bereit.

Die Methoden sowie Design- und Entwicklungsprozesse sind bis heute noch nicht ausreichend auf die komponentenbasierte Anwendungsentwicklung ausgerichtet. Auch die meisten CASE-Werkzeuge sind auf die projektzentrierte Entwicklung fokussiert und unterstützen die komponentenbasierte Anwendungsentwicklung nicht. Als Folge wird "das Rad immer wieder neu erfunden". Erst jetzt werden auch die Software Engineering-Prozesse so angepaßt, daß die komponentenbasierte Anwendungsentwicklung tatsächlich gefördert wird. Im Schlepptau dieser Entwicklung werden endlich auch die CASE-Werkzeuge überarbeitet. Der Transformationsprozeß hat jedoch eben erst begonnen.

Die Organisation ist als wichtigster Faktor entscheidend für das Gelingen der komponentenbasierten Anwendungsentwicklung. Wiederverwendbarkeit von Funktionen (prozedurale Denkweise) bzw. Objekten (objektorientierte Denkweise) und komponentenbasierte Anwendungsentwicklung verfolgen prinzipiell dieselben Ziele. Gleichwohl zwingt die Bildung von Komponenten nicht unbedingt zur Wiederverwendung. Eine

Komponente kann durchaus nur ein einziges mal genutzt werden. Andererseits liegt es selbstverständlich nahe, eine Komponente so auszulegen, daß sie möglichst oft wiederverwendet werden kann. Im Vergleich mit den bisherigen Möglichkeiten, etwa der Entwicklung von Klassen-Bibliotheken und Frameworks ergeben sich schon auf der technischen Ebene durch die externe Konfigurierbarkeit wesentlich weitreichendere Möglichkeiten.

Die komponentenbasierte Anwendungsentwicklung ist ohne Wiederverwendbarkeit von Software somit nicht denkbar und bleibt ohne Sinn. Wie bisherige Erfahrungen zeigen, ist Wiederverwendbarkeit kein Abfallprodukt, sondern muß mit gezielten Maßnahmen hart erarbeitet werden. Wiederverwendbarkeit muß als eigenständige Disziplin gemanaged und zu einem natürlichen Teil des Entwicklungsprozesses werden. Das bisherige Denk- und Arbeitsmuster der Software-Entwicklung, die bisherige projektzentrierte Entwicklungskultur, kann nicht beibehalten werden. Wiederverwendbarkeit verlangt einen projektübergreifenden Ansatz und erfordert tiefgreifende Eingriffe in die Aufbau- und Ablauforganisation. Ohne gezielt unterstützende Maßnahmen bleibt auch die komponentenbasierte Anwendungsentwicklung ohne Erfolgchance und nur so kann der Transformationsprozeß, vom Software-Entwickler zum Komponentenentwickler und Anwendungsassemblierer gelingen.

Die komponentenbasierte Anwendungsentwicklung darf nicht nur als technologische Strategie verstanden werden, sondern muß als langfristige Geschäftsstrategie aufgefaßt werden. Die Verankerung dieser Strategie im Management und dessen rückhaltlose Unterstützung ist unerlässlich, da es mit Sicherheit Ausbremsversuche geben wird. Die unterschweligen Argumente sind längst bekannt: das fehlende Vertrauen in die Arbeit anderer, die projektzentrierte Sicht, Sorge um möglichen Verlust an Einfluß und Macht und viele andere mehr. Auch an sachlich begründbaren Argumenten besteht kein Mangel: mangelhafte Unterstützung der Software Engineering-Methoden, wenige Werkzeuge, keine unternehmensweiten Software Engineering-Standards usw.. Zweifellos werden Projektleiter versuchen, nach wie vor ihr eigenes Süppchen zu kochen und die Wiederverwendungsstrategie zwar nicht grundsätzlich, jedoch punktuell zu unterlaufen. Deshalb muß Wiederverwendung vorhandener Komponenten als Qualitätsmerkmal eines "guten" Projekts festgeschrieben werden.

Eine Wiederverwendungskultur, die in jedem Fall notwendig ist, läßt sich nicht im Vorbeigehen schaffen. Jeder Eingriff in gewohnte Denk- und Arbeitsmuster verlangt sehr hohen Aufwand und braucht Zeit. Mit einem unternehmensinternen Marketing- und Werbetrommelfeuer muß die Änderung der Entwicklungskultur über längere Zeit unterstützend begleitet werden. Im Wertesystem muß Wiederverwendung einen hohen Rang einnehmen.

Wiederverwendung darf nicht dem einzelnen Projekt überlassen bleiben. Aus Projektsicht ist Wiederverwendbarkeit ein notwendiges Übel und verursacht lediglich zusätzliche Kosten. Das Projektbudget kann schnell überschritten werden, wobei dem zusätzlichen Aufwand kein Ertrag gegenüber steht. Das Projekt investiert nur, bekommt jedoch keine Belohnung für den Dienst, den es für andere erbringt. Dergestalt kann Wiederverwendung nicht funktionieren.

Es liegt nahe, Wiederverwendung auf der Unternehmensebene vorzusehen. Dieser Ansatz verlangt jedoch auch unbedingt eine unternehmensweite Standardisierung von Methoden, Verfahren und Werkzeugen. Es ist leicht einzusehen, daß Anwendungssysteme, die auf klar definierten Standards gründen, auch aus Komponenten zusammgebaut werden können. Sofern diese Voraussetzungen erfüllt sind, ist dies ein großer Vorteil. Allerdings existieren in vielen Unternehmen noch keine entsprechenden Standards auf Unternehmensebene. Deshalb muß Wiederverwendung häufig auf die nächst tiefere Ebene, die Ebene der Anwendungsdomänen (z. B. Finanzbuchhaltung, Materialwirtschaft usw.) bezogen werden. Gleichzeitig sollte jedoch so schnell wie möglich versucht werden, unternehmensweite Standards zu erarbeiten und einzuführen.

Auch die Aufbauorganisation muß auf die Unterstützung von Wiederverwendung ausgerichtet werden. Ein Wiederverwendungs-Manager (Reuse Manager) muß eingesetzt werden, der über die Grenzen von Fachabteilungen hinweg die Wiederverwendbarkeit durchsetzen kann und über den Rückhalt des Management verfügt. Er muß mit den erforderlichen Befugnissen ausgestattet sein, damit er Wiederverwendbarkeit auch durchsetzen kann und in Konfliktfällen zwischen den streitenden Parteien nicht zermalen wird. Eine Organisationseinheit wird innerhalb der IT-Abteilung gegründet, die sich ausschließlich mit Design und Entwicklung wiederverwendbarer Komponenten beschäftigt. Die Anwendungsentwicklung bisheriger Prägung evolviert zur Anwendungsmontage.

Die komponentenbasierte Anwendungsentwicklung muß im Vorgehensmodell verankert sein und muß auch bei der Aus- und Weiterbildung der Mitarbeiter immer wieder betont werden. Schließlich muß die Komponentenorientierung auch Maßstab für die Projektqualität sein. Kurzum, der gesamte Software-Entwicklungsprozeß muß auf Komponentenorientierung ausgerichtet sein.

Selbstverständlich muß auch eine geeignete Infrastruktur geschaffen werden. Sämtliche Anstrengungen wären umsonst, wenn Wiederverwendung technisch nicht unterstützt würde. Wiederverwendet werden kann nur, was auch wiedergefunden wird. Ein Repository, mit dem Komponenten verwaltet werden können, ist deshalb eine unabdingbare Notwendigkeit, ebenso auch die erforderlichen Verwaltungsfunktionen.

Eine Komponentisierung um jeden Preis ist keinesfalls sinnvoll. Immer muß auch eine wirtschaftliche Betrachtung unterlegt werden, um in der Frage "Komponente oder nicht" die richtige Entscheidung zu treffen. Die bisher gültigen Maßstäbe gelten auch weiterhin, gemäß denen ein wiederverwendbares Objekt im allgemeinen um den Faktor 5-7 teurer ist als wenn auf Wiederverwendbarkeit verzichtet würde. In Einzelfällen kann durchaus auch Faktor 10 und mehr erreicht werden, während andererseits in manchen Fällen auch Faktor 5 deutlich unterschritten werden kann.

Bei der Überlegung, welche wiederverwendbaren Komponenten entwickelt werden sollen, helfen einige einfache Leitlinien. Kriterien sind die strategische Wichtigkeit einer Komponente für das Unternehmen, die Höhe des zu erwartenden Return on Investment und außerdem bereits gemeldeter Bedarf. Mit diesen drei Kriterien ist ein zielorientiertes Vorgehen gewährleistet. Außerdem wird zu einem relativ frühen Zeitpunkt der Erfolg der Wiederverwendungsstrategie nachgewiesen, nicht nur für das Management erforderlich, sondern auch für die Entwickler. Sofern Komponenten selbst entwickelt werden, wird sich im ersten Projekt noch kein Erfolg der Wiederverwendungsstrategie zeigen. Erst ab dem zweiten Projekt werden die Früchte geerntet, wobei dann bei fachlich verwandten Projekten schon deutliche Erfolge erzielbar sind.

Der Übergang zur komponentenbasierten Anwendungsentwicklung erfordert zunächst erhebliche Initialinvestitionen und braucht Zeit. Die "Kulturfragen" sollten zuerst angegangen werden, dann erst die technischen. Der zeitliche Vorlauf, bis mit der komponentenbasierten Anwendungsentwicklung tatsächlich begonnen werden kann, beträgt mindestens ein halbes Jahr. Je nach der Anzahl der Entwickler und den bisherigen Software Engineering-Praktiken kann es mehrere Jahre benötigen bis sich eine Wiederverwendungskultur etabliert hat. Vor allem ist wichtig, den Transformationsprozeß in jeder Hinsicht zu unterstützen und ihm eine hohe mentale Priorität beizumessen. Eine halbherzige Wiederverwendungsstrategie kann scheitern und einen wahren Scherbenhaufen hinterlassen. Beispiele aus der Vergangenheit gibt es zur Genüge. Der Return on Investment, der einer konsequent durchgesetzten Wiederverwendungsstrategie folgt, lohnt jedoch auch den hohen Aufwand allemal.