Einführung einer komponentenbasierten Informationsarchitektur bei der R+V Versicherung.

Stefan Langenfeld, R+V Versicherung
Abraham-Lincoln-Straße 34
65189 Wiesbaden

1. Zusammenfassung

Die folgenden Ausführungen beschreibt die Zielsetzung der R+V und die bereits unternommenen und beabsichtigten Schritte auf dem Weg zu einer komponentenbasierten Informationsarchitektur. Dabei werden allgemeine Konzepte, die für die R+V sinnvoll erscheinen und R+V spezifische Vorstellungen erläutert. Im Abschnitt 2 wird erläutert, warum die R+V als Finanzdienstleister die Notwendigkeit sieht, Wiederverwendung als wichtigstes Prinzip des Systementwicklungsprozesses zu etablieren. Der 3. Abschnitt beschreibt die notwendige Voraussetzung für die Gestaltung des Wiederverwendungsprozesses. Dabei wird zuerst erläutert, was die R+V unter Komponenten versteht und welche Arten von Komponenten unterschieden werden. Zum zweiten wird die Rolle einer Informationsarchitektur im Wiederverwendungsprozess beschrieben. Zum dritten wird eine von R+V entwickelte Ablauforganisation zur Entwicklung und Nutzung von Komponenten dargelegt. Der 4. Abschnitt widmet sich dem Status der Einführung der komponentenbasierten Informationsarchitektur und beschreibt Maßnahmen für die Umsetztung des Vorhabens. Im 5. Abschnitt wird ein Ausblick auf die folgenden Schritte gegeben.

2. Einleitung

Die Veränderung der Marktsituation und der zunehmend globalisierte Wettbewerb sowie der zunehmende Wettbewerb um Kunden und Partner, zwingen die deutschen Versicherer dazu, mit flexiblen und unverwechselbaren Wettbewerbsstrategien am Markt zu agieren [PO97]. Auch für die R+V besteht der Zwang zur eindeutigen strategischen Positionierung und zur Senkung der Kosten, auch im Informationsressort. Die individuellen Wünsche nach Risikovorsorge der R+V Kunden müssen durch flexible Produktangebote erfüllt werden [FAR97].

Die vorhandene Informationsarchitektur ist in diesem Zuge auf diese neue durch hohe Anforderungen an Flexibilität und Schnelligkeit gekennzeichnete Situation anzupassen. Die Leistungen der Informationssysteme müssen schneller verfügbar sein und flexibler auf Änderungen des Geschäftsumfeldes reagieren können. Die Entwicklung oder der Einkauf von Softwarekomponenten im Rahmen einer komponentenbasierten Entwicklung sind für die R+V ein erfolgversprechender Weg die starren, statischen Strukturen der heutigen Systeme aufzubrechen und die Einführungszeit von Informationssystemen abzukürzen.

Dieses Ziel will die R+V durch die Ausrichtung der Systementwicklungsprozesse hin zur Wiederverwendung und Kooperation erreichen. Die Einführung einer passenden Entwicklungsorganisation und -kultur sind dabei ebenso zu beachten, wie das Schaffen eines technologischen Umfeldes, das den Einsatz von Komponenten unterstützt. Die Erfahrungen der letzten Jahre zeigen, daß der zur Erfüllung der Anforderungen notwendige Wandel der Informationssysteme durch die Definition und Nutzung von Architekturen und durch das Etablieren einer systematisch geplanten und gesteuerten Wiederverwendung erreicht werden kann [UWM92][McC93]. Dies gilt sowohl für die Wiederverwendung durch R+V erzeugter , als auch für den Einsatz am Markt erhältlicher Systemkomponenten. So ist die Handlungsreihenfolge: "Wiederverwenden, Kaufen, Entwickeln in der IT/IS-Entwicklungsphilosophie des Ressorts " [R+V96] bei R+V verankert.

3. Komponentenbasierte Entwicklung für die R+V

Die Erfahrungen der R+V zeigen, daß die Erhöhung des Grads der Wiederverwendung durch ungeplant Ansätze wie das Kopieren von Quellcode oder die Nutzung von einfachen, natürlich wachsenden Programmbibliotheken nicht zum Erfolg führen. Aus den Mängeln der Ad Hoc-Wiederverwendung wurde die Notwendigkeit erkannt, Softwarebausteine von vornherein wiederverwendbar zu entwickeln. Welche Bauteile und welche Anforderungen an deren Leistung anzulegen sind, läßt sich aus einer unternehmensweiten fachlichen Architektur [UWM92] in Form eines Anwendungsportfolios ableiten. Notwendige Voraussetzung für eine erfolgreiche Gestaltung des Wiederverwendungsprozesses ist zum einen das Vorhandensein von **Standards** zur Entwicklung von Komponenten (syntaktische Ebene), zum zweiten das Vorhandensein einer umfassenden **Informationsarchitektu**r, aus der die betriebswirtschaftlich notwendigen Komponenten (semantische Ebene) abgeleitet werden können. Zum dritten ist es notwendig, die **Prozesse** der Entwicklung und Nutzung von Komponenten in der Organisation aufbau- und ablauforganisatorisch zu verankern.

3.1 Komponentenbasierte Entwicklung / Component Based Development (CBD)

CBD ist eine Methode, die bisherigen methodischen Vorgehensweisen der R+V wie Information Engineering [MA90] oder das objektorientierte Paradigma [Qu94] zu ergänzen. Sie basiert auf der Idee, in sich geschlossene Softwarekomponenten zu Anwendungen zusammenzubauen. Diese Bausteine werden durch eine auf Wiederverwendung eingerichtete Organisation auf der Grundlage einer abgestimmten Architektur realisiert oder aber vom Markt eingekauft. CBD ist somit eine Weiterentwicklung der bestehenden, modellbasierten Vorgehensweisen.

3.2 Definition Komponente

Eine Komponente ist ein speziell für die Wiederverwendung entwickelter Softwarebaustein, der eine Menge von Diensten zur Nutzung anbietet. Es ist ein auslieferbares, unabhängiges Paket von Softwareoperationen. Komponenten können im eigenen Unternehmen entwickelt, vom Markt eingekauft oder am Markt vertrieben werden. Komponenten müssen folgende Eigenschaften besitzen:

- **Kapselungsprinzip**
  Komponenten setzten das Kapselungsprinzip aus der Objektorientierung durch Verstecken von Realisierungsdetails um. Die Realisierungsdetails wie die zugrundeliegende Datenstruktur oder die Art der Programmierung bleiben dem Nutzer verborgen.
- **Interfaces**
  Jede Komponente hat ein oder mehrere wohldefinierte Interfaces. Funktionen der Komponente können nur über Services des Interfaces benutzt werden. Zur Beschreibung der Interfaces existieren bereits standardisierte Sprachen "Interface Definition Language" (IDL).
- **Aggregierbarkeit**
  Komponenten können in beliebigen Kombinationen zusammengesetzt werden. Die Art der Nutzung der Komponente ist unabhängig von der Leistungsfähigkeit der Komponente selbst. So sind beispielsweise Partnerverwaltungskomponenten in allen Branchen notwendig und einsetzbar.
- **Interoperabilität**
  Ein wichtiges Merkmal von Komponenten ist ihre Interoperabilität. Diese erlaubt den Einsatz der Komponente auf unterschiedlichen Systemplattformen. Der Baustein kann somit benutzt und eingebunden werden, unabhängig vom Adressraum, von der Lokation in Netzwerken, von Programmiersprachen, Betriebssystemen, Entwicklungtools und unabhängig von der Entstehungshistorie des nutzenden Bausteins.

Aus Sicht der R+V müssen Komponenten die oben erläuterten Eigenschaften besitzen. Die Interoperabilität ist für R+V besonders wichtig, da Systemleistungen sowohl auf den großrechnerbasierten Systemen, als auch auf den Notebooks des Außendienstes nutzbar sein müssen.

Zur Definition von Komponenten verwendet die R+V keine der Standartbeschreibungssprachen, sondern es wird eine durch die genutzten Softwareentwicklungswerkzeuge vorgegebene Beschreibungsmethodik verwandt [TI96]. Eine Komponente besteht demnach aus drei Teilen: der **Spezifikation**, der **Implementierung** und den **ausführbaren Programmen** (siehe Abbildung 1). Die Spezifikation beschreibt die Außensicht der Komponente. Für die R+V ist es zweckmäßig, daß die Spezifikation einer Komponente mehrere Interfaces für unterschiedliche Nutzersysteme bereitstellt. So stellt die Komponente "Produktmanagement" unterschiedliche Interfaces für einzelne Produktbereiche wie "Haftpflicht", "Leben", "Krankenversicherung" usw. bereit. Jedes Interface wird durch ein Typmodell in Form eines E/R-Diagrammes und die Liste der verfügbaren Operationen beschrieben.

Die Implementierung umfaßt die tatsächliche Umsetzung der spezifizierten Eigenschaften. Sie sollte im Allgemeinen für den Nutzer nicht sichtbar sein (Information Hiding). Die Implementierung selbst besteht aus der Definition *eines* logischen Datenmodelles, der Datenbankdefinition, der Modul- und der Programmbeschreibungen. Aus diesen kann mit Hilfe des verwendeten Werkzeuges der ausführbare Code generiert werden.

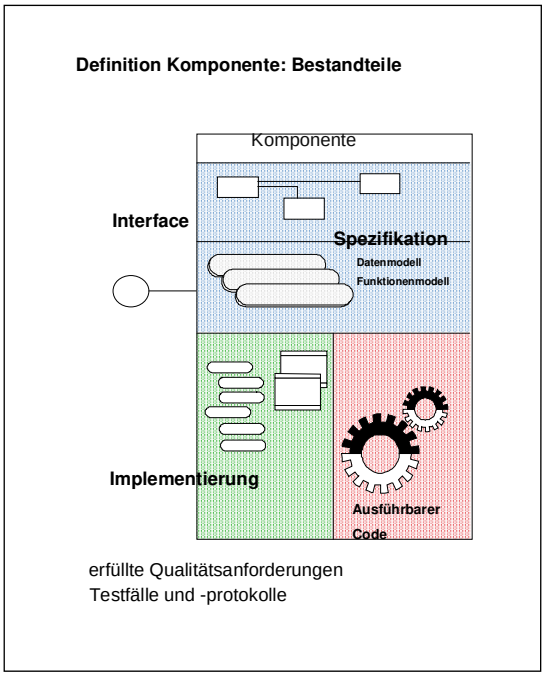

Abbildung 1: Bestandteile einer Komponente

Die Auslieferung einer Komponente geschieht entweder als ausführbares Programm oder als Gesamtbeschreibung mit Spezifikation, Implementierung und ausführbarem Programm. Die zweite Alternative erlaubt dem Nutzer die Anpassung der Implementierung [TI96].

In einem betriebswirtschaftlichen Umfeld ist es außerordentlich wichtig, daß Komponenten den definierten Anforderungen genügen. Die Qualitätssicherung hat einen wesentlichen Einfluß auf die Marktchancen einer Komponente. Somit ist es ratsam mit der Komponente Testfälle und Testprotokolle auszuliefern. Diese belegen die Erfüllung der Qualitätsanforderungen.

R+V unterscheidet drei Arten von Komponenten: **Basiskomponente**, **Anwendungskomponente und Infrastrukturkomponenten.**

Ein **Basiskomponente** ist eine Komponente, die ein Objekt der realen Welt repräsentiert, das für das geschäftliche Umfeld der R+V von Bedeutung ist. Es ist anwendungs- und geschäftsprozessunabhängig, beinhaltet fachliche Logik und zugehörige Regeln sowie persistente Daten. Beispiele für Basiskomponenten: Adresse, Vertragspartner, Versicherungsnehmer, Produkt, Vertrag, Fahrzeug usw.

Anwendungskomponenten gewährleisten die Einhaltung von übergreifenden Regeln zwischen den Basis- und Anwendungskomponenten und sorgen für eine konsistente, transaktionssichere Verarbeitung. Sie beschreiben und realisieren zusammengehörende, geschäftlich relevante Abläufe oder Teilabläufe. Anwendungskomponenten können Oberfläche besitzen. Sie nutzen die Leistungen von Basiskomponenten oder anderer Anwendungskomponenten. Beispiele für Anwendungskomponenten sind: Partnerverwaltung, Produktmanagement, Schadenabwicklung, Inkasso, Provisionierung, Antragsverwaltung usw.

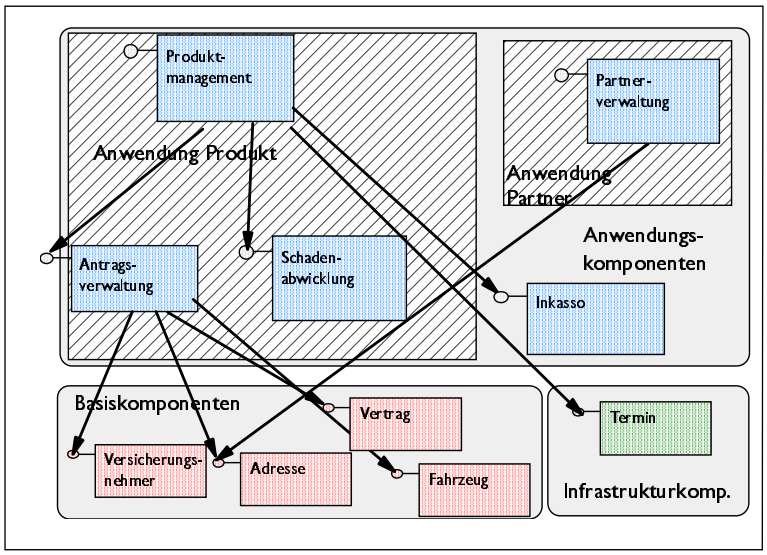

Abbildung 2: Zusammenhang zwischen Komponentenarten

Infrastrukturkomponenten stellen zentrale Services im Umfeld einer komponentenbasierten Anwendungsentwicklung zur Verfügung. Beispiele für Infrastrukturkomponenten: Kompetenzen/Berechtigungen, Postkorb, Notizblock, Transaktionsverwaltung, Selektionen mit den jeweiligen Services.

3.3 Die Informationsarchitektur der R+V

Shaw, Garlan [SG96] beschreiben, daß ein architekturorientierter, konstruktiver Ansatz der Schlüssel für die Erstellung von flexibler, langfristig erfolgreicher Software ist. Komponentenbasierte Entwicklung hat den Anspruch eine Methode zu sein, mit der dieses Ziel erreicht werden kann. Deshalb ist es für die Anwendung dieser Methode aus Sicht der R+V notwendig auf einer unternehmensweiten Informationsarchitektur zu basieren. Die von R+V entwickelte Informationsarchitektur beschreibt das Gesamtsystem "Unternehmen R+V" aus DV-Sicht [RV98-3]. Sie enthält eine **fachliche Architektur**, eine **technische Architektur**, die **Softwarearchitektur** und die Beschreibung der **Prozesse** der Systementwicklung und des Systembetriebes selbst (siehe Abbildung 3).

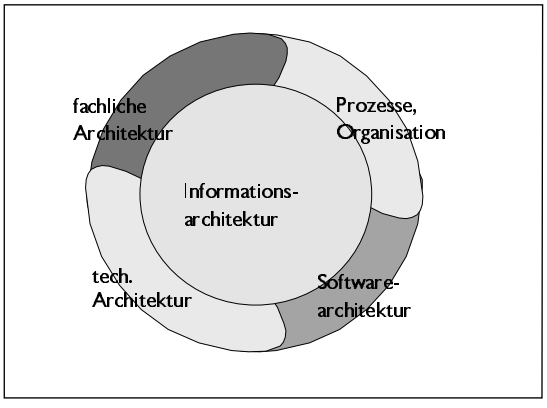

Abbildung 3: Bestandteile der R+V Informationsarchitektur

3.3.1 Die fachliche Architektur

Die fachliche Architektur besteht aus zwei Teilen, dem Komponentenmodell und dem Anwendungsportfolio: Das **Komponentenmodell** enthält die Beschreibungen (siehe Abbildung 1) aller Komponenten, die für das Kerngeschäft des Unternehmens notwendig sind (Basiskomponenten, Anwendungskomponenten, Infrastrukturkomponenten). Diese müssen flexibel an Änderungen der Geschäftsabläufe oder der Änderungen der Geschäftsstrategie anpaßbar sein. Diese Flexibilität wird entweder durch den Austausch einzelner Komponenten, oder aber die Erweiterung der Leistungsmerkmale der genutzten Komponenten erreicht.

Das Komponentenmodell liefert die Grundlage für die Anwendungsplanung und -erstellung sowie die Wiederverwendungsplanung und -steuerung. Das **Anwendungsportfolio** beschreibt das Wesen und den Inhalt einzelner Anwendungen, die im Rahmen von Projekten in Form von Komponenten aus Komponente montiert werden (siehe Abbildung 2). Die Komponenten werden aus dem vorhandenen Bestand entnommen, oder aber neu entwickelt. Ausgehend von der Geschäftsstrategie und der Gewichtung der Ziele läßt sich somit eine konsequente Vorgehensweise mit Terminen und notwendigen Ressourcen für die Entwicklung von Komponenten ableiten.

3.3.2 Die technische Architektur

Die technische Architektur beschreibt den informations- und kommunikationstechnologischen Rahmen für die Umsetzung der fachlichen Architektur. Dies beinhaltet alle für einen bestimmten Zeitraum und bestimmten Einsatzzweck ausgewählten und festgelegten Technologien, die notwendig sind, um die fachliche Architektur unter Nutzung der Softwarearchitektur effizient und effektiv in Anwendungssysteme umzusetzen. In diesem Teil der Architektur werden Managementdienste (Kommunikationsmodell, Datenbanksysteme, Transaktionsdienste usw.) festgelegt. Weiterhin werden die zu nutzenden Betriebssystemplattformen und Rechnerarchitekturen vorgegeben [R+V96].

Für die Entwicklung und die Nutzung von Komponenten hat das Kommunikationsmodell eine entscheidende Bedeutung. Es liefert eine standardisierte Infrastruktur und Dienste zur Implementierung von Komponenten und zur Unterstützung der Interaktion von Komponenten in einem heterogenen Umfeld.

Die Dienste des verteilten Komponentenmodells gewährleisten die Interoperabilität und die Aggregierbarkeit der Komponenten untereinander. In diesem Modell geht es insbesondere darum, einen entsprechenden Standard auszuwählen oder festzulegen. Es existieren drei konkurrierende Standards am Markt CORBA der OMG (Object Management Group), Java Beans und OLE/DCOM von Microsoft.

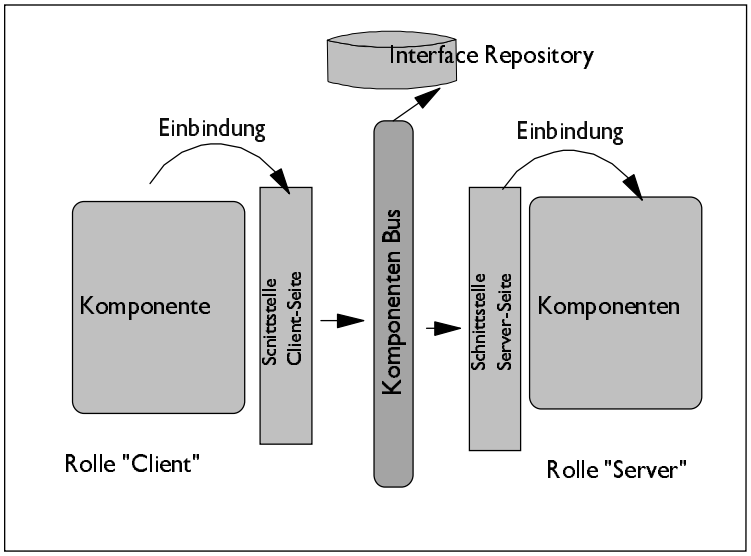

Abbildung 4: Prinzip des verteilten Komponentenmodelles

Wesentliche Bestandteile eines verteilten Komponentenmodelles sind der **Komponenten Bus** und das **Interface Repository.** Aufgabe des **Komponenten Bus** ist es, die Kommunikation zwischen den Komponenten transparent zu gewährleisten. Hierzu bedient er sich eines Interface Repositories, das alle Informationen über die aktuellen Komponenten besitzt.

R+V wird auf jeden Fall einen am Markt angebotene Standard nutzen. Für R+V ist zur Zeit eine Kombination aus nachrichtenorientierten Werkzeugen und einem verteilten Komponentenmodellen (DCOM/CORBA) sinnvoll .Mit welchem Standard für verteilte Komponentmodelle die R+V in Zukunft arbeitet, wird zur Zeit in einem Auswahlprozess geklärt.

3.3.3 Die Softwarearchitektur

Die Softwarearchitektur legt die Abbildungsregeln der fachlichen Architektur auf Anwendungssysteme fest. Es werden die methodischen und technischen Standards zur Strukturierung der Komponenten beschrieben. Sie beschreibt die innere Organisation der Komponenten. Alle Komponenten müssen diesem Organisationsprinzip gehorchen. Sie unterstützt die Bereitstellung und Wiederverwendung der Komponenten im Rahmen aller avisierten Zielplattformen durch formale Vorgaben und durch Bereitstellung von

Infrastrukturkomponenten. Für ein betriebswirtschaftliches Umfeld muß sie die Verfahren zur Einbindung von Altanwendungen unterstützen.

3.3.4 Organisation und Prozesse

Die wichtigste Einflußfaktore für eine erfolgreiche Einführung einer komponentenbasierten Entwicklung, ist die adäquate Gestaltung der Entwicklungsprozesse (Ablauforganisation) und die Ausgestaltung der Aufbauorganisation. Diese beschreiben und steuern die Art und Weise, wie Leistungen für die Kunden der Informatik erstellt werden. Je besser die Prozesse auf die übrigen Bestandteile der Informationsarchitektur abgestimmt sind, desto kürzer ist die benötigte Zeit, die Geschäftsstrategie mit entsprechenden Informationen und Systemen zu unterstützen.

Komponentenbasierte Entwicklung erfordert für die R+V einen umfassenden Veränderungsprozeß der Organisation. Diese Organisation war jahrzehntelang geprägt durch Projekte, die autarke, teilweise redundante Systeme für eine Plattform entwickelt haben (siehe Abbildung 5). Es fehlte das Vertrauen in die Arbeit anderer und deren Qualität.

Das Entwickeln eines Systems gestaltet sich in einer komponentenbasierten Umgebung anders als in einer auf autarken Projekten basierenden Organisation, wie sie über viele Jahre hinweg bei R+V vorzufinden war. Die Umwandlung der bestehenden, auf autarke Projektabwicklung ausgerichtete Organisation in eine komponentenbasierte Organisation, unter Berücksichtigung der spezifischen Unternehmenskultur bildet die größte Herausforderung der Einführung.

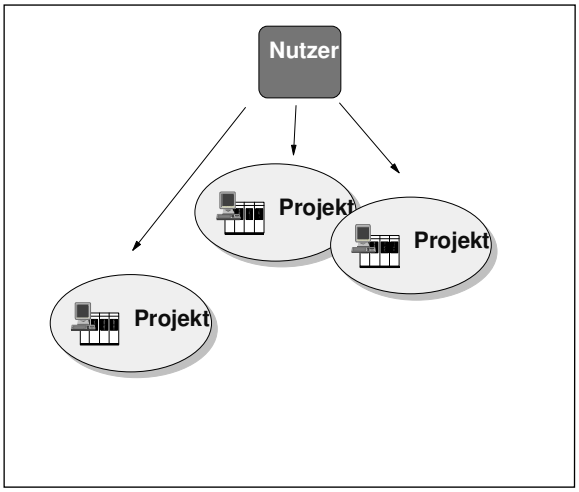

Abbildung 5 : Istzustand der R+V Systementwicklung.

Hierzu hat die R+V einen Ablauforganisation entwickelt (siehe Abbildung 6), die im Rahmen eines Erprobungszenarios erfolgreich getestet wurde. In diesem Ansatz lassen sich fünf Kernprozesse unterscheiden.

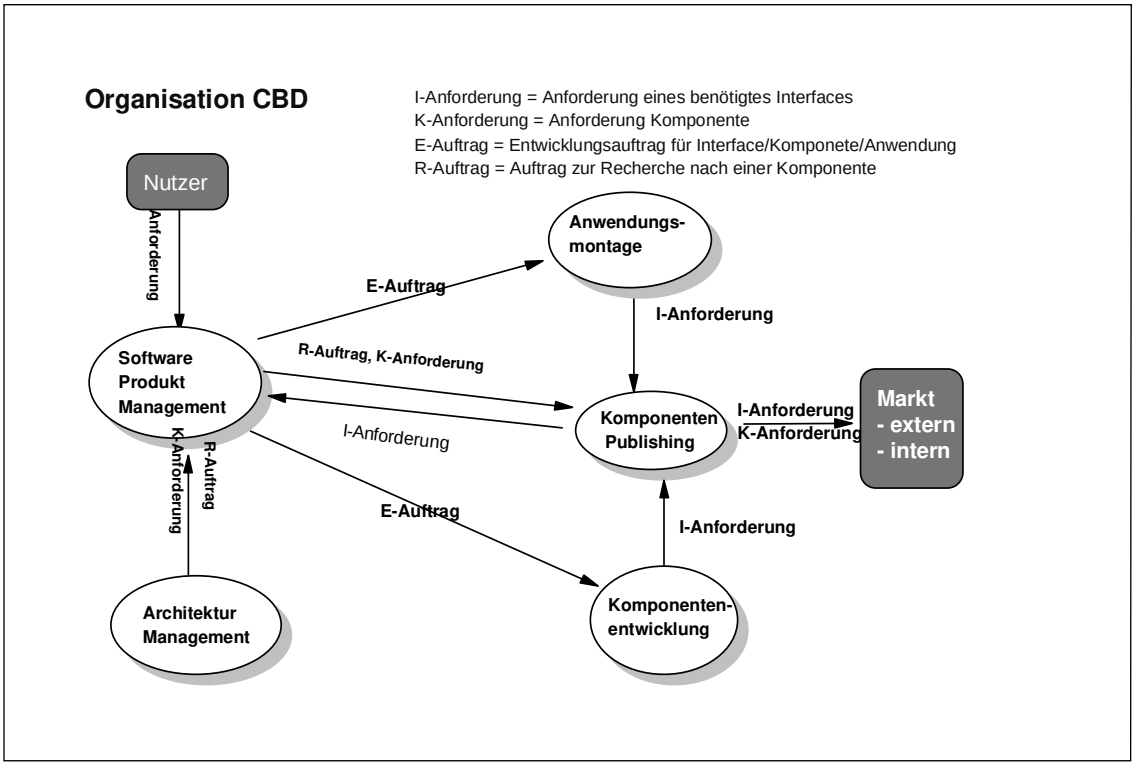

Abbildung 6 : Ablauforganisation einer CBD-Organisation

Das **Architekturmanagement** entwickelt und pflegt die fachliche Architektur, die technische Architektur, die Prozesse, Methoden und Werkzeuge für die komponentenbasierte Entwicklung und die Softwarearchitektur. Es bildet damit die Grundlage für die Anwendungs- und Projektplanung und die Projektdurchführung.

In einer zunehmend komplexen Umwelt hat ein übergreifendes **Softwareproduktmanagement** ein großes Gewicht. Basis- , Anwendungs-, Infrastrukturkomponenten und Anwendungen sind Softwareprodukte, die für Kunden der Informatik erstellt und gepflegt werden. Die angestrebte hohe Wiederverwendungsrate von Komponenten erfordert die Abstimmung der Lebenszyklen der Softwareprodukte aufeinander. Deshalb sind die Komponenten im Sinne eines Softwareproduktmanagements zu behandeln.
Dies umfaßt:

- die Analyse der Auswirkungen von Änderungen auf bestehenden Komponenten
- Abstimmung der Anforderungen während der parallelen Entwicklung von Softwareprodukten
- Konfigurations- und Änderungsverwaltung der Softwareprodukte
- Priorisierung der Anforderungen an die Softwareprodukte auf Basis der Unternehmenstrategie
- übergreifende Planung und Einsatz der Entwicklungsressourcen

Das bei R+V seit 1994 erfolgreich eingesetzte Modellmanagement [R+V98-2], das die Koordination der einzelnen Projektergebnisse auf Basis eines abgestimmten Unternehmensmodells zur Aufgabe hat, wird als fundierte Basis für das Softwareproduktmanagent genutzt werden.

Das **Komponenten-Publishing** unterstützt die übrigen Prozesse bei der Suche nach Komponenten und dem Erwerb solcher Softwareprodukte. Existieren keine passenden Komponenten, so gibt der Komponenten-Publisher dies intern an die Komponentenentwicklung oder extern in Auftrag. Wichtigste Aufgabe des Komponenten-Publishing ist es, für die Einhaltung der Qualitätsanforderungen an die Komponenten zu sorgen. Sinnvoll erscheint hier eine unternehmensübergreifende Zertifizierung von Komponenten auf Basis von branchenspezifischen Architekturen, wie sie von der OMG oder durch den Gesamtverband der Deutschen Versicherer [VAA97] angestrebt werden.

Die **Anwendungsmontage** ist der Prozeß, der notwendige Anwendungen durch Zusammenfügung von Basiskomponenten und Anwendungskomponenten erstellt.

Entwicklungs- oder Wartungsaufträge für Komponenten werden durch den Prozeß der **Komponentenentwicklung** umgesetzt. Der Auftrag wird vom Softwareproduktmanagement erteilt und setzt die eindeutige Beschreibung des Leistungsumfanges voraus. Die Entwicklung muß sich an die Vorgaben der Architektur, insbesondere der Softwarearchitektur halten. Die Art und Weise der Implementierung, also die innere Organisation, die Verwendung des Budgets, der Einkauf der Ressourcen und der Ort der Erstellung usw. bleibt dem jeweiligen Komponentenentwicklungsteam selbst überlassen. Die Trennung von Was und Wie ermöglicht somit die Realisierung des Gedankens einer "Softwarefactory".

Die Entwicklung von Anwendungskomponenten basiert auf der Verwendung vorhandener Komponenten. Voraussetzung dafür ist ein System zur Verwaltung des Angebotes an solchen Softwarebausteine. In dieses können neue Komponenten einfach eingebracht und vorhandene Komponenten mit vertretbarem Aufwand gesucht und wiederverwendet werden. Ein solches System wird als **Komponentenkatalog** bezeichnet. Dem Komponentenkatalog kommt entscheidende Bedeutung bei der Wiederverwendung zu. Er muß eine Reihe von Anforderungen erfüllen, um Wiederverwendung effektiv zu unterstützen. Neben Speicherung, Suche, Bewertung und Einbindung der Komponenten müssen Möglichkeiten der Konfigurations- und Änderungsverwaltung eingebunden sein [Li96]. Die Existenz unterschiedlicher Versionen einer Komponente muß abbildbar sein. [TI96].

4. Status bei R+V

4.1 Vorteile einer komponentenbasierten Entwicklung

Von der komponentenbasierten Entwicklung verspricht sich die R+V eine Methode, die die Beherrschung der zunehmenden Komplexität der Informationsverarbeitung betriebswirtschaftlich sinnvoll ermöglicht. Komponentenbasierte Entwicklung unterstützt die Handhabbarkeit von Softwareprodukten durch Unterteilen des Gesamtproblemes in unabhängige kleinere Einheiten. Sie läßt den Austausch von Teilen von Systemen durch leistungsfähigere Komponenten zu und erhöht damit die Flexibilität und Stabilität des Gesamtsystems. Durch die Plattformunabhängigkeit werden eine Vielzahl von Zugangswegen zu Kunden und Lieferanten möglich.

Ein weltweit entstehender Komponentenmarkt, der kostengünstige Softwarekomponenten bereitstellt, wird die Flexibilität und Produktivität weiter steigern. Die Fähigkeit, neue Wettbewerbsstrategien zu wagen, wird für Unternehmen zunehmen, die sich dieses Marktes bedienen können. Notwendig sind hierfür jedoch branchenspezifische Standards. Diese werden u.a. zur Zeit von der Finance Domain Task Force der OMG und durch den Gesamtverband der Deutschen Versicherer [VAA97] angestrebt.

Ein Vorteil der Anwendungsentwicklung aus Komponenten gegenüber rein objektorientierten Ansätzen besteht in der Möglichkeit eines schrittweisen Überganges von der bestehenden Software-Entwicklung zum

Objektparadigma. Ausgehend von den zentralen Elementen des Geschäftsbereiches können nach und nach mehr Komponenten erzeugt und in die Anwendungen integriert werden. Desweiteren entsteht kein gravierender Bruch in der Vorgehensweise der Entwicklung wie bei einem Übergang zur Objektorientierung in einem Schritt.

4.2 Nachteile der komponentenbasierten Entwicklung

Neben den Chancen, die sich aus der komponentenbasierten Entwicklung für die Informatik eines Unternehmen ergeben, sind mögliche Risiken zu bedenken. Der Übergang zu einer komponentenbasierten Vorgehensweise erfordert höhere Anfangsinvestitionen, einen hohen Reifegrad zugrundeliegender Technologien wie verteilte Komponententechnologien, die mittelfristige Stabilität der zu schaffenden Architekturen sowie ein adäquates Wissen und Fähigkeiten der Mitarbeiter.

4.3 Kritische Erfolgsfaktoren

Besonders wichtig ist die Änderung der bestehenden Entwicklerkultur, die gekennzeichnet ist durch das Überbetonen der Projektteilsicht und das unzureichende Vertrauen in die Qualität der Arbeit Anderer (siehe Abbildung 5). Die Kultur muß das Teilen von Wissen, Verantwortlichkeit und eindeutige Zuständigkeit fördern und die Sicht von der engen Projektsicht hin zu einer Sicht als Teil eines Gesamtvorhabens entwickeln.

Voraussetzung für das Funktionieren eines verteilten Komponentenmodelles ist das Vorhandensein einer performanten, standardisierten Middleware, die die Interoperabilität unterstützt und eines mächtigen Komponentenkataloges. Wichtig ist es auch, die Softwarearchitektur so zu gestalten, daß die enormen Investitionen in bestehende Systeme durch Umstrukturierung, Entkernung und Herauslösen von Komponenten gewinnbringend weiterverwandt werden können.

4.4 Maßnahmen zur Einführung

Die Einführung einer komponentenbasierten Entwicklung wird von R+V als ein vielversprechender Weg, die Probleme der bestehenden Informationsarchitektur zu bewältigen, angesehen. Den Entscheidungsträgern ist jedoch bewußt, daß der Übergang zu einer solchen Arbeitsweise einen tiefgreifenden Eingriff in die Aufbau- und Ablauforganisation bedeutet [RV98-1]. Aus diesem Grund wurde eine Vorgehensweise entwikkelt, die eine endgültige Entscheidung vom positiven Ausgang mehrerer Erprobungsszenarien abhängig macht.

Erprobungsszenario Fachliche Architektur

Das bei R+V seit 1993 erfolgreich eingesetzte Unternehmensmodell wird in Teilbereichen zu einem Komponentenmodell umgebaut, um die Tragfähigkeit der Beschreibungsregeln für Komponenten zu erproben (siehe Abbildung 7). Das Unternehmensmodell und der Prozeß der Verwaltung der einzelnen Projektmodelle im Modellmanagement [R+V98-1] haben die Wiederverwendung von Datenstrukturen massiv erhöht. So verwenden die in den letzten 4 Jahren begonnenen Projekte der R+V redundanzfreie Datenstrukturen. In Teilbereichen werden sogar schon einzelne Module basierend auf den Vorgehensweisen des Modellmanagements wiederverwendet.

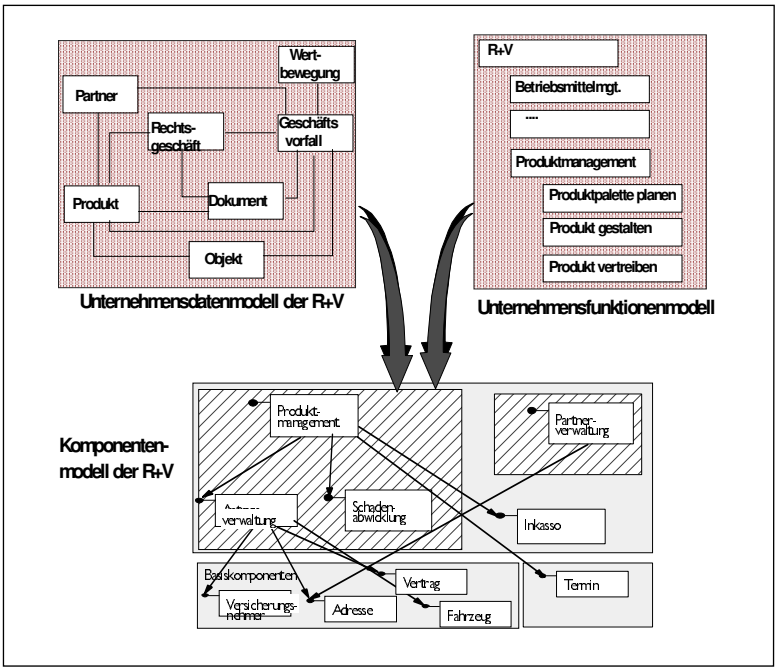

Abbildung 7 : Erprobungsszenario Fachlich Architektur

Erprobungsszenario Technische Architektur

Auf Basis der vorhandenen technischen Infrastruktur wurden Prototypen entwickelt, die alternative Realisierungsvarianten aufzeigten und die technische Machbarkeit bewiesen haben.

Erprobungsszenario Ablauforganisation
Die Tragfähigkeit der Aussagen zur Ablauforganisation (siehe 3.3.4 und Abbildung 6) wurde durch ein Projekt, das die fachlichen, die technischen Rahmenbedingungen und die Auswirkungen auf die Organisation gesamtheitlich erproben sollte, unter Beweis gestellt. In diesem Projekt wurden insbesondere die Prozesse einer komponentbasierten Entwicklung ausgetestet.

4.5 Fazit

Die Auswertung der Erprobungsszenarien und die Diskussionen mit den Betroffenen zeigen, daß es wichtig ist, die Denk- und Handlungsprozesse der Mitarbeiter zu verändern. Die Funktionstüchtigkeit der Methoden und der Technologie wurde nachgewiesen.
Für R+V heißt es noch in diesem Jahr zu entscheiden, ob der Übergang zu einer komponentenbasierten Entwicklung gewagt wird. Die Erprobungsszenarien haben gezeigt, daß das Scheitern der Umgestaltung der Unternehmenskultur die Einführung einer komponentenbasierten Entwicklungsumgebung verhindern kann. Aus diesem Grund wurden die Betroffenen frühzeitig in den Änderungsprozess durch Vorträge und Diskussionsforen eingebunden. Die Ideen können von jedem Mitarbeiter in einem Intranet-Forum eingesehen werden. Es hat sich gezeigt, daß die Einrichtung eines mit den notwendigen Kompetenzen versehenen Softwareproduktmanagement die höchste Hürde ist, da bisher autarke Projektleiter und Manager es nicht gewohnt sind die Gesamtsicht vor ihre Teilsicht zu stellen.
Ein erster Schritt zur Beschleunigung des Veränderungsprozesses wird es sein, eine "Softwarefactory" (siehe 3.3.4) als Gemeinschaftsunternehmen zu gründen. Diese soll von R+V spezifizierte Komponenten unabhängig, unter Verwendung der Aussagen der Informationsarchitektur realisieren.

Literaturverzeichnis

[WE97] Weede: Möglichkeiten der Software-Wiederverwendung durch komponentenbasierte Anwendungsentwicklung in einem Versicherungsunternehmen, Diplomarbeit, Universität Leipzig Institut für Informatik / R+V Allgemeine Versicherung AG, 1997

[FAR97] Farny: Versicherungsmarkt im Wandel: Von der Spartenorganisation zu kundenorientierten Geschäftsfelder in: Kompetenz 31 Diebold Management Report

[MA90] James Martin: Information Engineering: Book 1, 2, and 3

[Li96] Lindner : Massive Wiederverwendung: Konzepte, Techniken und Organisation; in: OBJEKTSpektrum 1/96, 10-17, Januar 1996

[McC93] McClure, Carma: Software-Automatisierung - Reengineering Repository Wiederverwendbarkeit; Eine Coedition der Verlage: Carl Hanser Verlag München Wien, Prentice-Hall International Inc. London, 1993

[PO97] Porter: Nur Strategie sichert auf Dauer hohe Erträge
Harvard Business review 3/97

[Qu94] Quibeldey-Cirkel: Das Objekt-Paradigma in der Informatik
B.G. Teubner, Stuttgart, 1994

[RV98-1] Dern, Kelter, Dr. Noack: "Component Based Development im ZI-Ressort der R+V Versicherung" , Wiesbaden 1998

[RV98-2] "Modellmanagement für Host-Anwendungen und GUI Anwendungen", Wiesbaden 1998

[RV98-3] Dern, Kelter, Langenfeld, Dr. Noack, Ricker: Komponentenbasierte Anwendungsarchitektur, Wiesbaden 1998

[R+V96] R+V: Systemplanung zur IT/IS-Landschaft der R+V

[SG96] Shaw, Garlan: Software Architecture: Perspectives of an emerging disciplin; Prentice Hall 1996

[TI96] Texas Instruments: Component-Based Development - Fundamentals; Texas Instruments Incorporated; Juli 1996

[TIIP] Texas Instruments: Component-Based Development - Component-Based Vision; Texas Instruments Incorporated Internal Paper; 1995

[UWM92] Dern, Haag, Kelter, Langenfeld: Unternehmensweites Modell der R+V-Versicherung, Wiesbaden 1992

[VAA97] Gesamtverband der Deutschen Versicherer: Versicherungs Anwendung Architektur VAA