Bank2010: Eine fachliche und technische Vision

Dr. Sanjay Dewal, Ludger Schnichels

debis Systemhaus DL GmbH
BU Banken Düsseldorf/Stuttgart
Emanuel-Leutze-Str. 4
40457 Düsseldorf
Tel.: 0172 / 944 1429
E-Mail: lschnich@debis.com

Abstract. Der innerdeutsche Bankenmarkt befindet sich derzeitig im Umbruch. Fusionen zur Konzentration sind dabei in jüngster Zeit vermehrt zu beobachten. Gestützt auf die Globalisierung, EU-Harmonisierung und der gestiegenen Informationstransparenz versuchen die Banken als Full-Service-Anbieter, dem möglichen Markteintritt der Non- und Nearbanks sowie angelsächsischer Spezialinstitute (z.B. Wertpapierhäuser) entgegenzuwirken. Aber um dem internen Kostendruck sowie dem externen Margenverfall entgegenzuwirken, ist mittel- bis langfristig auf dem deutschen Bankenmarkt eine Abkehr vom Universalbankensystem hin zur Bündelung von Kernkompetenzen zu erwarten.Die Autoren zeigen im folgenden auf, wie sich diese Konzentration auf Kernkompetenzen in Zukunft in der Bankenlandschaft darstellen wird. Dabei wird explizit auf die fachliche Darstellung und der damit einhergehenden technischen Umsetzung abgehoben.

Einleitung

Die derzeitigen Entwicklungen am innerdeutschen Bankenmarkt sind durch Fusionstendenzen im Sinne von Abwehrstrategien gekennzeichnet. So gingen jüngst die HVB (München) und die LBW (Stuttgart) aus Fusionsaktivitäten hervor, um so politisch- oder managementmotiviert einen größeren Marktanteil zu generieren. Als Folge dessen resultieren daraus laterale Universalbanken, die durch komplexe Organisationen und Wertschöpfungsketten gekennzeichnet sind.
Um im Zuge sinkender Margen durch interne und externe Kosteneinflüsse die Profitabilität auch mittel- bis langfristig aufrechterhalten zu können, ist eine ablauf- und aufbauorganisatorische Neuausrichtung der Universalbanken unabdingbar.
Die Autoren zeigen in der folgenden Darstellung auf, welche Umstrukturierungen der Universalbanken zukünftig zu erwarten sind. Überdies wird explizit dargestellt, inwieweit diese strukturellen Modifikationen durch DV-Architekturszenarios begleitet werden bzw. unbedingt unterstützt werden müssen, weil die Veränderungen nur mittels dv-technischer Unterstützung zu einer langfristigen Flexibilität und Kostenreduktion führen.

Die Rahmenbedingungen der klassischen Universalbank

Die typische deutsche Universalbank versteht sich traditionell dem Kunden gegenüber als Full-Service-Anbieter. Für das **Frontoffice** bedeutet dies, daß jeder Kundenbetreuer über ein umfassendes Vertriebs- und Produkt-Know-How verfügen muß, um dem Universalbankanspruch - "den Kunden kompetent in allen Fragen zu beraten" - gerecht zu werden. Die Vielzahl der einzelnen Produkte und Produktvarianten macht es für den einzelnen Kundenberater aber nahezu unmöglich, dem Kunden ein optimales Produktangebot zu unterbreiten.
Aus diesem Grund haben die Universalbanken ihre Kunden in Segmente, die von unterschiedlichen Einheiten betreut werden, eingeteilt. Die Universalbank kann somit relativ effizient Spezial-Know-How vorhalten, allerdings nur zu dem Preis, daß der Kunde bei unterschiedlichen Problemen verschiedene Ansprechpartner hat. Dies widerspricht aber prinzipiell ihrem "Full-Service-Anbieter"-Anspruch, bei dem implizit suggeriert wird, der Kunde habe für alle seine Probleme einen einzigen Ansprechpartner in der Bank. Dieser „One-Face-To-The-Customer"-Ansatz kann im besten Fall dadurch eingehalten werden, daß der Kundenberater als Lotse des Kunden im Universalbankvielerlei agiert.
Betrachtet man beispielsweise den Bankenmarkt in Deutschland, so ist davon auszugehen, daß die Kunden künftig verstärkt auf ein gutes Preis-Leistungs-Verhältnis achten werden. Banken wird es infolgedessen zunehmend schwerer fallen, ihre getätigten Investitionen auf ihre Kunden umzulegen. So wird zum einen durch den zunehmenden Trend hin zum Home- und SB-Banking die Vergleichbarkeit von Bankprodukten und –dienstleistungen in Zukunft weiter steigen. Zum anderen drängen durch die Liberalisierung des EU-

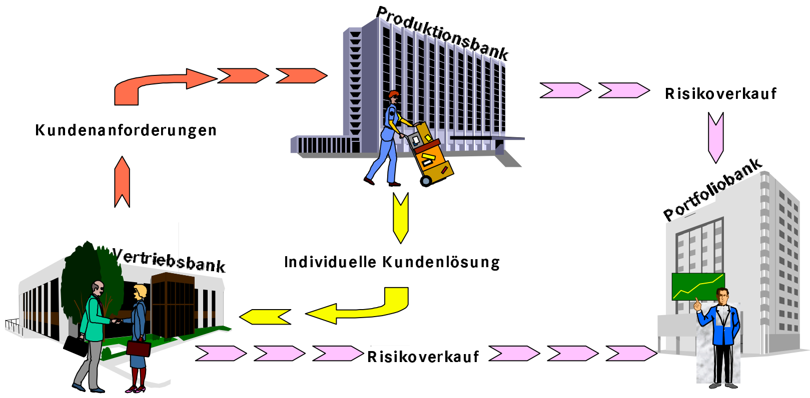
Binnenmarktes verstärkt Non- und Nearbanks[1] und ausländische Wettbewerber verstärkt in die klassischen Domänen deutscher Universalbanken ein. Aufgrund dessen können bereits heute Veränderungen in der klassischen Universalbank beobachtet werden (siehe verworfene Fusionspläne der Deutschen Bank und der Dresdner Bank).
Zur ersten Gruppe zählen **Maßnahmen zur Reorganisation der Geschäftsprozesse**. Dabei sind vor allem die Verschlankungen der Sachbearbeitungs- und Abwicklungsaktivitäten und die Verlagerung von Aktivitäten aus dem Backoffice in das Frontoffice zu nennen. Als konsequenteste Form dieser Reorganisationsmaßnahmen ist das Outsourcen von Aktivitäten zu nennen, bei denen entweder eigene Gesellschaften, z.T. mit anderen Partnern zusammen, gegründet oder an externe Anbieter vergeben werden. Die drastischste Form der Reorganisation ist die in allen Bankengruppen zu beobachtende Freisetzung von Personal, was bisher, bezogen auf die Öffentlichkeit, relativ still vollzogen wurde.
Zur zweiten Gruppe zählen **Maßnahmen zur Marktneuausrichtung**. Hauptansatzpunkt ist dabei die Überprüfung der bestehenden Vertriebswege. So wurde und wird immer noch die Filialstruktur neu auf die Kundenbedürfnisse ausgerichtet.
Außerdem versuchen die großen Bankenkonzerne, ihr Portfolio durch die Gründung von spezialisierten Tochterunternehmen wie z.B. Direktbanken oder den Kauf von in das Bankportfolio passenden Spezialinstituten wie z.B. Investmentbanken sowohl quantitativ als auch qualitativ sinnvoll zu ergänzen.
Überdies sind derzeit auch schon Bestrebungen zu erkennen, die aufgrund des zunehmenden Wettbewerbdrucks auf eine stärkere Fokussierung der für die Kunden der einzelnen Teilportfolien notwendigen Produkte abzielen. Ein Schritt in diese Kundenorientierung ist die systematische Erfassung, Strukturierung und Verbesserung der Produktpalette mit Hilfe von Produkt- und Leistungsdatenbanken.

Banking 2010 – Eine Vision

***Erfolgreiches Banking in der Zukunft ist nur möglich,
wenn die interne Strukturen effizient und flexibel an die
exogenen Markteinflüsse angepaßt werden können und die Struktur über eine Spezial-
isierung effizient und überschaubar wird.***

Aufsetzend auf der obigen These geht das Autorenteam davon aus, daß eine Universalbank ihre drei Kernkompetenzen „Vertrieb", „Produktion" und „Portfoliomanagement" in einzelne Kompetenzcenter aufteilt, am besten unserer These gerecht werden kann.
Der **Vertriebsbank** fällt dabei die Rolle des Spezialisten für die Kundenwünsche zu. Analog zum Kundenbetreuer der klassischen Universalbank führt sie als Navigator den Kunden durch den <u>Markt für Finanzdienstleistungen</u>. Dabei liegt die Betonung auf Markt, d.h. im Gegensatz zum klassischen Kundenbetreuer der Universalbank hat die Vertriebsbank das Recht bzw. die Pflicht, dem Kunden das für ihn optimale Produkt bzw. die optimale Dienstleistung aus dem Angebot aller Anbieter herauszusuchen.
Der **Produktionsbank** fällt die Aufgabe der kostengünstigen Produktion bzw. Bereitstellung von Finanzprodukten und –dienstleistungen zu. Als Serviceprovider bietet sie ihre Produkte und Dienstleistungen den am Markt agierenden Vertriebseinheiten an. Die konkrete Ausgestaltung der Produktionsbank hinsichtlich ihres Produktportfolios ist dabei an die individuellen Möglichkeiten hinsichtlich Produktbreite und –tiefe anzupassen.
Die **Portfoliobank** ist Spezialist für das Management von Markt- und Kreditrisiken. Sie kauft von der/den Vertriebs- und gegebenenfalls von der/den Produktbanken Risiken an und managt diese im Sinne ihrer Strategie und der gesetzlichen Rahmenbedingungen. Durch ihre Dienstleistung „befreit" sie die Vertriebs- und Produktionsbank(en) vom zeit- und kostenaufwändigen Risikomanagement, so daß sich diese voll auf ihre Kernkompetenzen konzentrieren können.
Ihr Know-How und die Vielfalt der von ihr angekauften Risiken ermöglichen es ihr, das Portfolio unter Chance-Risiko-Aspekten optimal zu gestalten. Dieses läßt sich grafisch wie folgt veranschaulichen:

[1] Eine Non-Bank ist ein Industrieunternehmen, welches keine Dienstleistungen einer Bank erbringen, aber Töchter gründen, die als Konkurrenten zu Banken auftreten (DaimlerChrysler mit MBLF). Eine Near-Bank bietet Teile der Dienstleistungen einer Vollbank an (z.B. Broker, Wertpapierhaus, Leasinghaus).



Durch die enge Verzahnung untereinander können sich die drei Spezialbanken voll auf ihre individuellen Stärken ausrichten. Dabei besteht zwischen den einzelnen Kompetenz-Centern ein Lieferanten- und Abnehmerverhältnis, welches aber, im Gegensatz zur Universalbank, nicht ausschließlich auf die „Schwestereinheiten“ beschränkt ist. Der große Vorteil dieser Ausrichtung läßt sich auf zwei Perspektiven konkretisieren:
Aus **Kostenrechnungssicht** befinden sich alle Front- und Backoffice-Einheiten der ursprünglichen Universalbank nunmehr in einer unmittelbaren marktlichen Wettbewerbssituation. Dadurch werden alle Einheiten an wirklichen Marktpreisen gemessen, was allgemein zu effizienten Kostenstrukturen führt, weil in den einzelnen „Kompetenz-Centern“ die vorhandenen Kapazitäten einer optimaler Betriebsgröße vorhalten werden kann.
Aus **Kundensicht** wird dieser im ursprünglichen Universalbanksinne ganzheitlich mit den für ihn besten Produkten und Dienstleistungen zu attraktiven Konditionen bedient. Anders formuliert bedeutet das, daß eine profitable Gestaltung des individuellen Kundenbedürfnisses dadurch gewährleistet wird, daß die individuellen Kernkompetenzen der einzelnen „Bankeinheiten“ entlang der gesamten Wertschöpfungskette gebündelt werden, ohne dabei einen zusätzlichen Koordinierungsaufwand zu erfordern und ohne daß den Kunden diese Struktur beeinträchtigt.
Als Folge dessen betrachten es die Autoren als Zukunftsvision, daß die oben beschrieben Einheiten in einem nächsten Schritt als wirtschaftlich selbständige Einheiten weltweit auf dem Markt operieren. Somit werden im gesamten Wertschöpfungsprozeß die einzelnen Einheiten nicht mehr das Angebot der Schwestereinheiten wahrnehmen, sondern das unter Margen- und Risikogesichtspunkten optimale Angebot eines global transparenten Marktes generieren.
Im Zuge des kompletten Redesigns der aufbau- und ablauforganisatorischen Umgebung müssen von vornherein auch die Anforderungen an die zukünftige DV-Landschaft Berücksichtigung finden. Denn jedes Prozeßdesign ist nur so gut, wie es später funktional abgebildet werden kann. So sind im Vorfeld die Anforderungen an die Systemlandschaft zu definieren, auf denen zukünftige Strukturen arbeiten werden.

Anforderungen an die DV-Unterstützung

Aus den Überlegungen des vorangegangenen Abschnittes ergeben sich eine Reihe von strategischen und fachlichen Aspekten, aus denen die Anforderungen für die DV-Unterstützung abgeleitet werden müssen.

Fachliche Anforderung
Unterstützung verschiedener Vertriebswege[2]: Künftig ist davon auszugehen, daß der Bankkunde über verschiedene Wege seine Bankgeschäfte initiieren und durchführen wird. Die DV muß deshalb das Hinzufügen neuer Vertriebswege problemlos ermöglichen.
Beschleunigung der Geschäftsprozesse: Der Einsatz von Workflow-Managementsystemen (WfMS) und Dokumenten-Managementsystemen (DMS, elektronische Kundenakte) reduziert die Durchlauf-, Liege- und Rüstzeiten und vermeidet Medienbrüche in den Prozessen. Darüber hinaus können die Geschäftsprozesse durch den Einsatz von Entscheidungskomponenten hinsichtlich Bearbeitungs- und Durchlaufzeiten weiter optimiert werden. Dadurch kann die Diskriminanz erhöht und der Einkauf von Stornogeschäft bereits im Vorfeld reduziert werden.
Potentialausschöpfung. Zur Erkennung der vielfältigen Kundenwünsche ist es gerade im Finanzumfeld wichtig, die Aspekte bzw. Produkte herauszufiltern, die den Kunden interessieren. Heutzutage wird das Neugeschäft z.B. durch Kampagnenmanagement schwerpunktmäßig durch die Bank gesteuert. Über virtual Communities, Portale oder Content Management (siehe [7]) besteht die Möglichkeit, daß Bankkunden diese

[2] Beispiele für Vertriebswege sind Homebanking, Telefonbanking, Filiale, POS, SB, etc.

Fachliche Anforderung
Informationen durch thematische Chat-Rooms und Foren aktiv übermitteln.
Maßgeschneiderte Produkte für den Kunden. Der Kunde will optimal beraten werden und bedient sich dabei verschiedenener Vertriebswege. Hierzu sind intelligente DV-Systeme erforderlich, die Kundenanforderungen mit Produkten und Konditionen in Einklang bringen. Als Basistechnologie sind dabei Übereinstimmungs-Algorithmen (siehe [10,11]) einzusetzen. Der effiziente Einsatz dieser Methoden erfordert geeignete Produkt- und Konditionendatenbanken. Bezogen auf die Dreiteilung der Universalbank bedeutet dies, daß eine Vertriebsbank die Möglichkeit haben muß, auf Produkte und Konditionen anderer Produktionsbanken über sogenannte intelligente Agenten (siehe [9]) automatisch zugreifen zu können.
Flexibilisierung von Bankstrategien. Die zu erwartende zunehmende dv-technische Durchdringung der Geschäftsprozesse führt dazu, daß immer mehr Informationen in Form von Entscheidungsbäumen und Kundendaten elektronisch vorgehalten werden. Konzepte wie unternehmensweites Datenmodell (siehe [3,6]) und Datawarehouse sowie Analysen mit Datamining-Techniken (siehe [14]) sind dazu zwingend erforderlich.
Nutzung von Funktionalitäten zur Risikominimierung. Im Zuge der Entwicklung standardisierter Risikokomponenten werden die zuvor beschriebenen Entscheidungsbäume und vorhandenen Kundeninformationen zur Steuerung des kundenspezifischen Kreditrisikos bei der Antragstellung herangezogen. Überdies wird die vorhandene Datenbasis zur Integration von maschinellen Lernverfahren und Credit-Value-at-Risk-Methoden zur Steuerung des Gesamtbankrisikos in Zukunft von großer Bedeutung sein. Über die funktionale Unterstützung verbunden mit der entsprechenden Datenbasis wird somit die Steuerung von Einzel- und Portfoliorisiken neue Dimensionen erlangen.

Aus den Anforderungen resultiert letztlich die Forderung nach **einer höheren Flexibilität in der Aufbau- und Ablauforganisation**, damit auf die sich ändernden Markterfordernisse effizient und zeitnah reagiert werden kann. Die benötigten DV-Systeme müssen dafür hinsichtlich der Geschäftslogik durch Komponententechnologie[3] gekapselt werden.
Ferner müssen **übergreifende Geschäftsprozesse über die Vertriebs-, Produktions- und Portfoliobank** unterstützt, organisiert und gesteuert werden, um die erforderliche Effizienz und Geschwindigkeit bei der Zusammenarbeit der einzelnen Banken umzusetzen. Daraus resultiert die Anforderung nach der Verteilung von Komponenten im WAN.
Nicht jede fachliche bzw. strategische Anforderung führt zu neuen dv-technischen Aspekten. Zur besseren Darstellung sind in der folgenden Tabelle die konkreten dv-technischen Anforderungen zusammengestellt.

dv-technische Anforderung
Komponententechnologie: Trennung von Geschäftslogik und Präsentationslogik; Vertriebsweg als Komponente; Unterschiedliche Endgeräte; Verteilung von Komponenten
Geschäftsprozeß als Komponente: Automatisierung der Geschäftsprozesse durch Einsatz von Entscheidungskomponenten; Parametrisierbarkeit von Geschäftslogik
Datenbanken: Leistungsfähige und verteilte Datenbanken; Aufbau von Konditionen- und Produkt-, Kompentenzdatenbanken; Unternehmensweites Datenmodell; Datamining-Techniken für Auswertungen
Sicherheit durch Middleware: Transaktionsschutz für Vertriebswege; Auto-Restart
Nutzung von Workflow- und Dokumenten-Managementsystemen, Virtual Communities, Einsatz von intelligenten Agenten

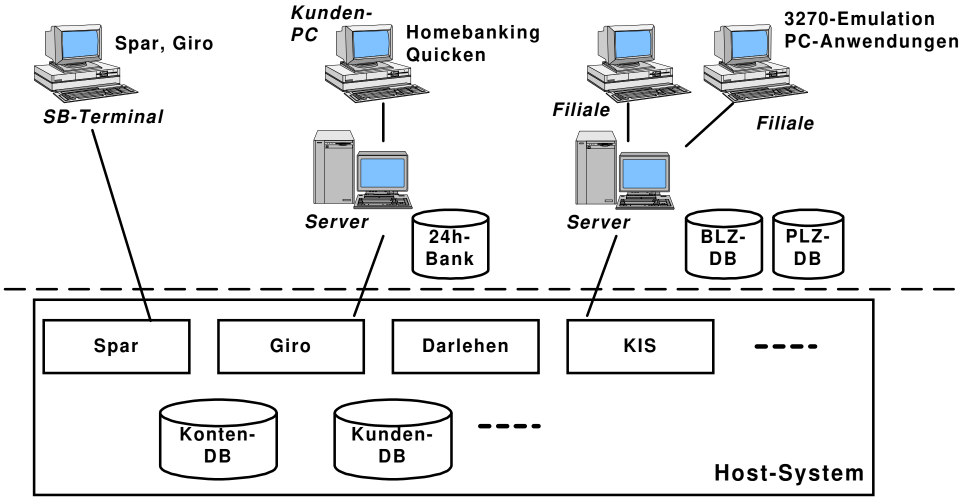
Die oben genannten dv-technischen Anforderungen machen deutlich, daß die Komplexität von DV-Anwendungen hinsichtlich Daten und Komponenten sowie der Anzahl von Komponenten erheblich zunimmt. Außerdem ist die zu erwartende Evolution der Komponenten zu berücksichtigen. Im Ergebnis bedeutet dies, daß DV-Systeme (vorgegebene Konfigurationen von Komponenten) ebenso Änderungen unterliegen (siehe [13]). Während durch die Komponententechnologie die Evolution von Komponenten durch Kapselung und standardisierte Schnittstellen auf diese selbst beschränkt bleibt (d.h. Lokalität von Änderungen und Seiteneffektfreiheit) trifft dies auf die DV-Systeme nicht zu. Deshalb ist es für DV-Systeme unabdingbar, daß Rahmenbedingungen in Form von DV-Architekturen definiert werden, die diese Evolution unterstützen. Somit ist die Konzeption von Architekturen eine wesentliches Merkmal der zukünftigen Banken. Aus den Überlegungen zur Evolution ergeben sich grundlegende Anforderungen zur Offenheit und Erweiterbarkeit hinsichtlich neuer fachlicher wie technologischer Komponenten sowie zur Reduzierung der Wartungs- und Pflegekosten.

[3] Komponententechnologie ist die Zerlegung eines komplexen DV-System in Module. Dabei sind die einzelnen Module bzw. Komponenten als Black Box zu verstehen, die nur über die definierte Schnittstelle angesprochen werden kann. Für die Spezifkation von Komponenten kann das Paradigma der Objekt-Orientierung gewählt. Dies bedeutet, daß Alt-Systeme nur durch Wrapping-Techniken als Komponente bereitgestellt werden können.

Analyse des Ist-Zustandes bei den Banken

Beschreibung des Ist-Zustandes

In den letzten Jahren haben die Autoren in Kundenprojekten eine Vielzahl von DV-Systemen und Architekturen bei verschiedenen Banken kennengelernt. Trotz bankindividueller Abwicklungssysteme ist festzuhalten, daß die Architektur, die den DV-Systemen zugrunde liegt, über die verschiedenen Banken hinaus durchaus vergleichbare Strukturen aufweist (siehe nachfolgende schematische Abbildung).



Nach wie vor spielt in den Banken der Großrechner (Host) die dominierende Rolle. Einziges Unterscheidungsmerkmal ist die Wahl des Herstellers bzw. des Betriebssystems (im wesentlichen werden IBM mit VSE, MVS oder OS/390 bzw. Siemens mit BS2000 eingesetzt). Auf diesem Host-System sind alle Abwicklungssysteme installiert. Ferner werden die gesamten juristisch relevanten Daten in den entsprechenden Host-Datenbanken (d.h. ISAM, VSAM, DB2, Leasy, SESAM) abgespeichert.
Dezentral prägen derzeit 3270- bzw. 9570-Front-Ends das Erscheinungsbild. Diese sind meist über eine Windows-Emulation auf den PCs zugänglich. Teilweise wurden "GUIfizierungen" vorgenommen, um das "Look-and-Feel" von Windows-Applikationen zu geben.
Client-/Server-Architekturen sind dadurch verwirklicht, daß auf dem Client verschiedene Plausibilitätschecks bei der Eingabe und einfache Funktionalitäten implementiert wurden. Ferner sind häufig generell genutzte und selten veränderte Tabellen wie PLZ, BLZ, Branchenschlüssel, Adreßverzeichnisse, etc. auf Servern bzw. auf Clients ausgelagert.
Aufgrund der sich verändernden Kundenbedürfnisse, genannt seien z.B. SB- und Homebanking, haben die Banken die existierenden Abwicklungssysteme durch dezentrale Systeme und Datenbanken (z.B. 24h-Bank) erweitert, um die Verfügbarkeitsbeschränkungen des Host (insbesondere beim Nachtbetrieb) ausgleicht. Dies bedeutet auch, daß Kundentransaktionen erst nach einem gesonderten Batchlauf im Host-Abwicklungssystem der Bank juristisch wirksam werden.
Als Kommunikationsplattform werden bei den meisten Banken nach wie vor SNA-Protokolle wie LU0 oder LU6.2 verwendet. TCP/IP zum Host ist recht selten zu finden.
Wesentlicher Vorteil der hostbasierten Architektur ist die Sicherheit und Verfügbarkeit der Host-Systeme, schließlich erreichen dezentrale Netzwerke keine vergleichbaren Verfügbarkeiten. Auch die geringeren Betriebskosten sind bei diesen Architektur geringer. Dezentrale Servern oder Clients sind insbesondere wegen der Anzahl dieser Systeme hinsichtlich der Betreuung kostspieliger.

Bewertung der Ist-Architektur hinsichtlich der künftigen Anforderungen

Es stellt sich nun die Frage, wie die im vorherigen Abschnitt aufgestellten Anforderungen mit Hilfe der existierenden DV-Systeme und Architekturen abgedeckt werden können. Es zeigt sich sehr schnell, daß viele dieser Anforderungen während der Entwicklungsperiode der Altsysteme gar keine Relevanz hatten. Beispielsweise spielt bei monolitischen Systemen Komponententechnologie, die Trennung von Geschäftsprozessen, Workflow- und Dokumentenmanagement, virtual Communities, intelligente Agenten keine sonderliche Rolle. Derartige Anforderungen ergeben sich erst durch verteilte Systeme und wechselnde Vorgaben oder sie werden erst in Zukunft relevant.
Zusammenfassend lassen sich die folgenden Nachteile bei den heutigen Architekturen herauskristallisieren:

- **Hohe Redundanz bei Daten und Funktionen**. Durch die Abbildung der fachlichen Organisation auf die DV (es gibt für jeden Fachbereich auch eine entsprechende DV-Gruppe) ergeben sich hohe Redundanzen hinsichtlich der Daten und Funktionen, weil jede DV-Gruppe für "ihren" Fachbereich die gewünschten DV-Systeme mehr oder weniger eigenständig entwickelt.
- **Hohe Wartungs- und Entwicklungsaufwände** bei den heutigen Bankanwendungen. Weitere Wartungsaufwände entstehen dadurch, daß für jede fachliche Anforderung (z.B. Vertriebswege) eigene Server mit entsprechender Anwendungslogik und Host-Anbindung installiert werden mußten. Dies führt insbesondere bei den fachlichen Funktionen zu Redundanzen, so daß bei Änderungen auf dem Host-System auch viele Komponenten auf den Servern anzupassen sind.
- **Mangelnde Integriertheit.** Die einzelnen Abwicklungssysteme sind in sich integriert. Allerdings fehlt die Integration hinsichtlich eines Gesamtprozeßbetrachtung über den Lebenszyklus von Produkten. Dabei werden in der Regel, weil einzelne Prozeßschritte durch verschiedene Systeme unterstützt werden, Daten wiederholt neu eingegeben oder gar überprüft.
- **Mangelnde Offenheit und Erweiterbarkeit**, weil proprietäre Systeme und Schnittstellen zum Einsatz kommen, die automatisch zu hohen Aufwänden bei der Integration neuer Komponenten führen.
- **Geringe Flexibilität**, da die Komponenten alle monolithisch aufgebaut sind und auf dem Host sind.

Mehrstufige Architektur für die Bank 2010

Die derzeitigen Architekturen erfüllen bestenfalls in Ansätzen die Anforderungen der skizzierten Bank 2010, so daß die aufgezeigten fachlichen Aspekte dv-technisch überhaupt nicht umgesetzt werden können. Desweiteren werden Technologie-Trends immer unvorhersehbarer und die Innovationszyklen von neuen Technologien immer kürzer. Daraus abgeleitet läßt sich ***die*** zentrale Frage formulieren:

Wie müssen die DV-Architekturen aussehen, die den künftigen DV-Anforderungen einer Bank genügen.

Als wesentliche Aspekte und Kriterien, für eine derartige Architektur sind Flexibilität, Erweiterbarkeit und Offenheit.

Zwingende Rahmenbedingungen für Architekturen

Auch die künftige Architektur unterliegt selbst wiederum bestimmten Rahmenbedingungen, die von der Bank vorgegeben werden. Diese resultieren sowohl aus strategischen als auch auf rechtlichen Gegebenheiten.

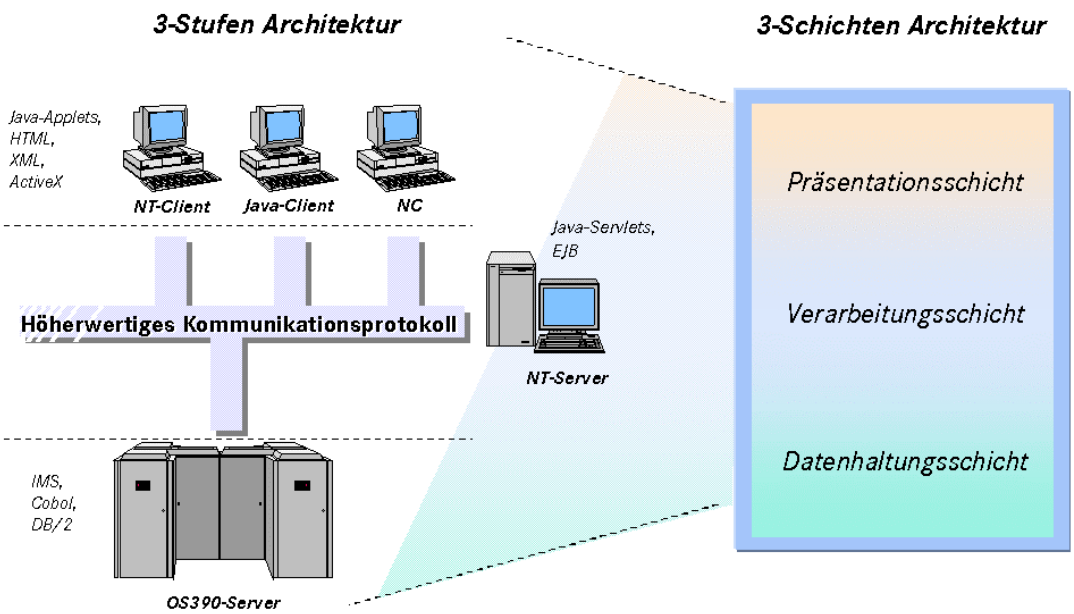
- **Juristische Datenhaltung**. Unsere Projekterfahrungen bei Entwicklung von DV-Systemen für Banken zeigen, daß die Banken auch künftig den Host für diese Aufgabe favorisieren. Daraus resultiert eine zentrale Datenhaltung, die im Hinblick auf Sicherheit und Konsistenz der einzelnen Datenbanken, nur zu begrüßen ist.
- **Abwicklungssysteme**. Mit der juristischen Datenhaltung verbleiben auch die zentralen Abwicklungssysteme auf dem Host.
- **Berechnungmodule**. Es gibt verschiedenste bankinterne oder jurististische Vorgaben, die bestimmte Qualitätsansprüche an Berechnungsmodule stellen. So ist die Effektivzinsberechnung im Sinne des Verbraucherkreditgesetzes auf die erste Kommastelle hin genau auszuweisen. Dieses Beispiel hat neben der gesetzlichen Vorgabe auch betriebswirtschaftliche Gesichtspunkte: Eine minimale Abweichung in der Zinsberechnung führt unter Umständen einerseits zu hohen Zinsverlusten über das gesamte Geschäftsvolumen der Bank und anderseits u.U. zu hohen Erstattungsaufwänden bei der Abrechnung von Konten. Deshalb führt die Dezentralisierung derartiger Module zum Teil zu erheblichen Testaufwänden, um die Korrektheit der Berechnung nachzuweisen. Daher sollten Berechnungsmodule, soweit wie möglich, auf dem Host verbleiben.
- **Bestehende Netze**. Wichtig bei jeglichen Konzeptionen ist die Nutzung der existierenden Bandbreiten und Kapazitäten des WAN bzw. LAN. Denn auch wenn diese erweitert werden, werden die erforderlichen Bandbreiten für moderne Technologien auf lange Sicht nicht bereitgestellt werden können.

Eine Architekturvision für die Bank 2010

Die Rahmenbedingungen aus dem vorherigen Abschnitt bilden die wesentlichen Eckpfeiler einer Vision, deren wesentliche Kriterien eine Architektur ist, die die obigen Anforderungen unterstützt:

- Die sich dynamisch verändernde Bedürfnisstruktur der Bankkunden bedingt, daß unterschiedlichste Vertriebswege für die Produkte angeboten werden müssen, um besser auf das Verhalten der Kunden eingehen zu können,
- 3-Schichten-Architektur, d.h. Trennung von Präsentations-, Anwendungs- bzw Geschäfts- und Datenhaltungslogik zur besseren Erweiterbarkeit und Offenheit.
- 3-Stufen-Architektur, d.h. Verteilung der 3-Schichten auf verschiedene Rechnerstufen zur besseren Beherrschbarkeit und höheren Verfügbarkeit.
- Kapselung von Systemen auf ihre Kernkompetenzen, d.h. Zerlegung der DV-Systeme in Komponenten, die für einen fest definierten Umfang Funktionen bereitstellen. Dadurch wird die Erweiterbarkeit und Beherrschbarkeit komplexer DV-Systeme unterstützt (getreu dem Software-Entwicklungsprinzip: divide-and-conquer).
- Nutzung standardisierter Schnittstellen, d.h. die gekapselten Komponenten werden über Schnittstellen spezifiziert und angesprochen, die einem entsprechenden Industrie- oder Marktstandard genügen. Dadurch sind einzelne Komponenten veränderbar und austauschbar, ohne daß das Gesamtsystem beeinträchtigt wird.
- Reduzierung der Redundanzen hinsichtlich Daten und Funktionen durch Komponententechnologie und ein unternehmensweites Datenmodell.

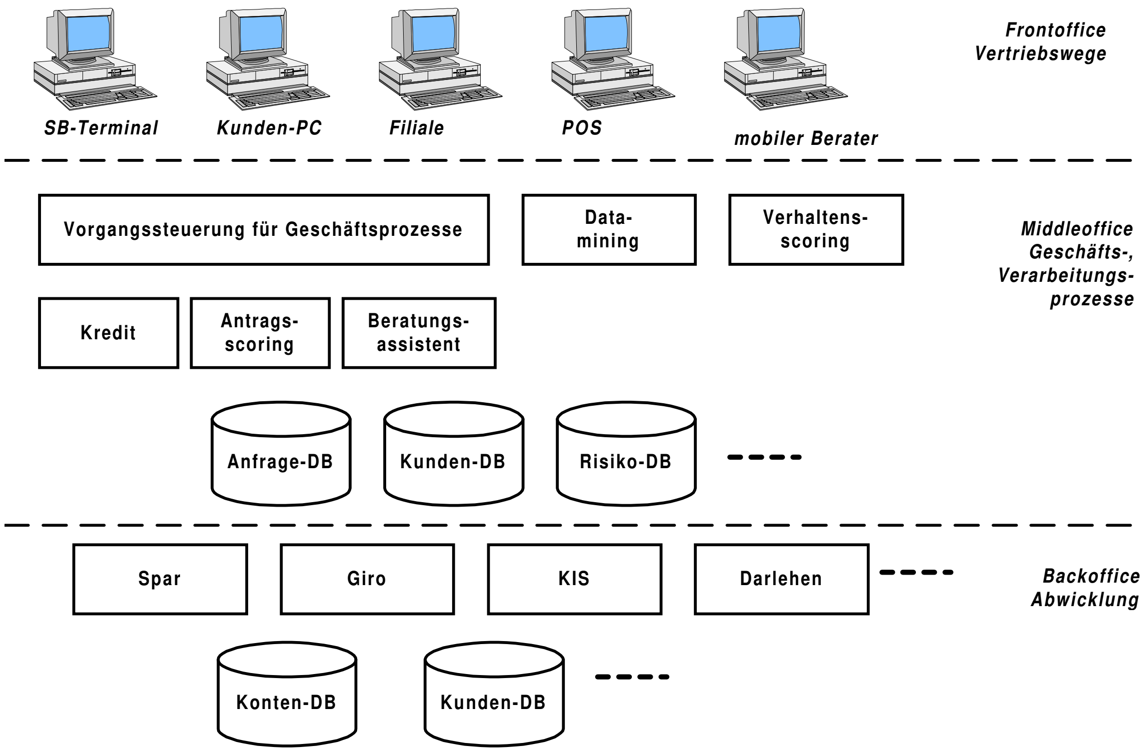
Im folgenden wird die resultierende Architektur konkretisiert. Wie die Verteilung der 3-Schichten auf die 3-Stufen erfolgt (siehe nachfolgende Abbildung), ist nicht zu verallgemeinern sondern hängt von den Gegebenheiten und Anforderungen der Bank ab. Hierzu zwei Beispiele:



- Beispiel 1: Ein erste Kunde möchte die Präsentationsschicht mittels Java-Applets und HTML realisieren. Dabei sollen die Applets klein gehalten werden, um die Übertragungszeiten zu reduzieren.. Dementsprechend ist der Kunde bereit, auf Informationen eine gewisse Zeit zu warten. Aus diesen Vorgaben resultiert, daß lediglich einfache Plausibiltätsüberprüfungen im Applet durchgeführt werden.
  Die Stufen / Schichten-Zuordnung sieht in diesem Fall so aus, daß auf dem Client Präsentationslogik und einfache Elemente der Applikationslogik abgelegt werden.

- Beispiel 2: Wie schon im vorherigen Beispiel möchte der zweite Kunde auch die Präsentationsschicht mittels Java-Applets und HTML realisieren. Allerdings möchte der Kunde aufwendige Berechnungen und daraus resultierende Grafiken online (d.h. in Echtzeit) umsetzen. Diese Vorgabe führt dazu, daß neben der Präsentationsschicht und den einfachen Plausibilitätschecks auch Berechnungsfunktionalitäten der Applikationsschicht auf dem Client abgelegt werden.

In der nachfolgenden Abbildung ist eine derartige Verteilung von Komponenten auf die verschiedenen Stufen grafisch dargestellt. Dabei sind die Stufen als Front-, Middle- und Backoffice bezeichnet worden.



Im **Frontoffice**[4] sind die Arbeitsplätze (entspricht der Architekturstufe "Client") der verschiedenen Vertriebswege lokalisiert. Dies bedingt, daß entsprechende Präsentationsoberflächen für die verschiedenen Nutzergruppen realisiert und angeboten werden, da es signifikante Unterschiede für diese hinsichtlich der Bedienbarkeit von Anwendungen gibt. So wird ein Mitarbeiter in der Filiale oder Zentrale andere Anforderungen stellen als ein Kunde, der von zu Hause aus Produkte der Bank nutzen will, oder ein Kunde, der vor einem SB-Terminal steht.

Im **Middleoffice** (entspricht der Architekturstufe "Server") ist in erster Linie die zentralisierte Geschäftslogik lokalisiert. Mit dieser Logik werden die Abläufe für die verschiedenen Produkte der Bank zunächst einmal unabhängig vom Vertriebsweg spezifiziert. Durch entsprechende Werkzeugunterstützung bei der Steuerung der Geschäftsprozesse (z.B. Workflow-Managementwerzeuge) wird die Flexibilität hinsichtlich der Veränderungen gewahrt.

Desweiteren befinden sich im Middleoffice Teile der Verarbeitungslogik. Es sind alle erforderlichen Komponenten zu finden, die zur Entscheidungsfindung benötigt werden. Diese Komponenten können auch Datenhaltungssysteme umfassen, die die erforderlichen Daten für eine Entscheidung bereitstellen.

Sind juristische Daten erforderlich oder gilt es Vorgänge juristisch wirksam werden zu lassen, so werden entsprechende Aktivitäten im Backoffice angestoßen und das Middleoffice fungiert dann sozusagen als "Durchreiche" vom Front- zum Backoffice.

Das **Backoffice** (entspricht der Architekturstufe "Host") umfaßt die klassischen Abwicklungssysteme und die juristischen Daten.

Durch die Implementierung dieser drei Stufen wird gewährleistet, daß Veränderungen in einer Stufe nur auf die benachbarte Stufe Auswirkungen haben. Im Idealfall sollte durch eine adäquate Kapselung der Komponenten in den einzelnen Stufen erreicht werden, daß Veränderungen in einer Stufe ohne sofortige Auswirkungen auf andere Stufen bleiben. So ist es beispielsweise möglich, einen neuen Vertriebsweg im Frontoffice für ein existierendes Produkt der Bank zu realisieren, ohne daß sich notwendigerweise im Backoffice etwas verändert. Im Middleoffice müssen lediglich die entsprechenden Kompetenzen für den neuen Vertriebsweg eingetragen werden. Nur wenn der neue Vertriebsweg auch neue Produkte und damit eine neue Geschäftslogik bedingt, sind im Middleoffice Veränderungen erforderlich. Entsprechendes gilt auch für Veränderungen im Backoffice. So können sich beispielsweise Berechnungsmodule der Abwicklungssysteme ändern, ohne daß sofortige Änderungen im Frontoffice erforderlich werden.

[4] Die Begriffsverwendung im technischen Umfeld (die Trennung in Front-, Middle- und Backoffice den verschiedenen Schichten wie Präsentation, Applikation und Datenhaltung) entspricht nicht der Einteilung aus fachlicher Sicht (siehe Abschnitt “Rahmenbedingungen der klassischen Universalbank”).

Die Komponenten jeder Stufe werden durch Techniken wie abstrakte Datentypen und Information-Hiding gekapselt. Dadurch wird erreicht, daß Modifikationen einer Komponente nur beschränkte Auswirkungen (Seiteneffektfreiheit) auf das Gesamtsystem haben.
Ein funktionsfähiges DV-System wird immer auf proprietäre Teilsysteme wie Datenhaltungssystem, Oberfläche, Betriebssystem zurückgreifen müssen. Die Kapselung dieser proprietären Teilsysteme durch standardisierte Schnittstellen stellt zusätzlich sicher, daß die Unabhängigkeit vom Hersteller bzw. Anbieter gewahrt bleibt.
Durch Verwendung von standardisierten Schnittstellen ist es darüber hinaus mit geringem Aufwand möglich, Komponenten[5] auszutauschen bzw. neue zu integrieren. Ferner ist eine Verteilung der Komponenten über mehrere Stufen möglich, was unbedingt erforderlich ist, um z.B. bei einer Internet-Anbindung aufgrund der zu realisierenden Sicherheitsmaßnahmen eine Mehrstufigkeit zu gewährleisten.
Für die Unterstützung der verschiedenen Vertriebswege aber auch zur Kapselung der Backoffice-Systeme ist es notwendig, die Geschäftslogik von den Dialogarbeitsplätzen und von den Abwicklungssystemen zu trennen.
Wesentlicher Nachteil der 3-stufigen Architektur ist die Verteilung der Komponenten. Zwar wird ein System durch Verteilung skalierbarer, aber daraus ergeben sich Probleme hinsichtlich der Transaktionssicherheit, Verfügbarkeit und Beherrschbarkeit. Diese Aspekte können mit herkömmlichen Kommunikationsprotokollen wie LU0, LU6.2, TCP/IP nicht gelöst werden sondern erfordern erheblichen Realisierungsaufwand. Daher ist es unbedingt erforderlich, daß ein höherwertiges Kommunikationsprotokoll in Form einer Middleware als Object Transaction Monitor oder Object Request Broker zum Einsatz kommt. Die Middleware stellt dann über entsprechende Mechanismen wie Transaktionsklammer, Rollback (Zurücksetzen von Transaktionen), Auto-Restart (automatisches Wiederaufsetzen von abgebrochenen Transaktionen, etc.) die Transaktionssicherheit, Verfügbarkeit und Beherrschbarkeit sicher. Die Autoren gehen davon aus, daß künftig Middleware als Basistechnologie von Komponenten angeboten wird, so daß keine Eigenentwicklungen erforderlich sind.

Bewertung der Architekturvision

Die im vorherigen Abschnitt dargestellte Architektur erfüllt die Anforderungen der Bank 2010. Durch die Kapselung und die Verwendung von standardisierten Schnittstellen ist Evolution als wesentlicher Aspekt in der Architektur inhärent. Dementsprechend können auch über die gestellten Forderungen neue Technologietrends und Kundenanforderungen hinzugefügt werden.
Bezogen auf die Vision "Bank 2010" bzw. die angedachten drei Teilbanken ist es wichtig, daß alle drei Banken diese Architektur als konzeptionelle Basis verwenden. Durch eine leistungsfähige Middleware und die Nutzung von standardisierten Schnittstellen lassen sich dann auch zwischen den Banken Funktionalitäten, Produkte und Konditionen nutzen bzw. bereitstellen, weil eine Verteilung von Komponenten und Daten im WAN unterstützt wird. Die Midlleware ist somit die zentrale Integrationsplattform im Sinne eines "Querschnittsbrokers", um Komponenten verfügbar zu machen.
Neben den üblichen Transaktionseigenschaften ergibt sich an die Middleware eine weitere Anforderung hinsichtlich der Autonomie der an einem Prozeß beteiligten Banken. Denn obwohl ein Geschäftsprozeß künftig über die Banken integriert abläuft, behält jede ihre Eigenständigkeit. Die Banken können also selbst entscheiden, wie und wann eine Transaktion abgewickelt wird. Grundsätzlich wird die Autonomie durch die Komponententechnologie, die Workflowsteuerung und das Berechtigungskonzept schon gewahrt. Allerdings müssen leistungsfähige Rollback-Mechanismen für den Fehlerfall von der Middleware bereitgestellt werden, um die langen Transaktionen abzusichern.
Darüber hinaus ist über die Schnittstellen auch eine Anbindung von „dritten" Banken möglich. Allerdings ist es von deren Architektur abhängig, inwieweit Funktionen und Daten ausgetauscht werden können.

Implikationen für künftige DV-Projekte bei Banken

"Big-Bang" oder Migration in Schritten

Für die Banken bedeuten die fachlichen Trends die Migration ihrer host-basierten DV-Landschaft hin zu einer komponentenbasierten, offenen Architektur. Idealerweise wäre es sinnvoll, diese Migration in einem Schritt durchzuführen, was auch den Vorteil hätte, daß die DV-Systeme neu entwickelt werden und somit dem aktuellen Stand der Technik entsprechen. Allerdings ist das Risiko, alles auf einen Schlag zu verändern, viel zu

[5] Dies bedeutet, daß eine Komponente, obwohl sie über die standardisierte Schnittstelle integrierbar ist, in der Implementierung proprietär sein kann.

groß, weil die Komplexität des zu realisierenden Systems hoch ist und somit ein Fehlschlag wahrscheinlich wäre (siehe [1,2] sowie die Probleme bei der Jahrtausend- und EURO-Umstellung).
Daraus resultiert zwingend, daß nur eine stufenweise Migration ein sinnvoller Weg ist. Durch die Definition einzelner Migrationsschritte werden die einzelnen Veränderungen beherrschbar.
Die einzelnen Migrationsstufen umfassen folgende Inhalte:

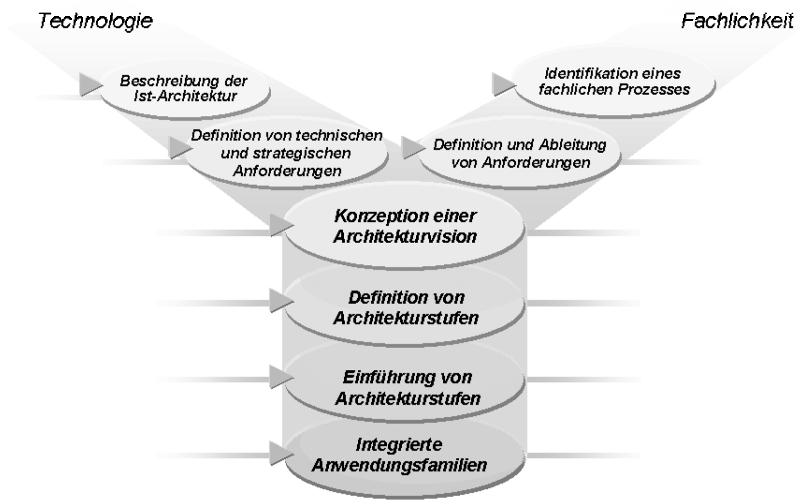
- Standardisierte Schnittstellen einführen und so die existierenden Bausteine kapseln. Denn durch die Schnittstelle werden alle Zugriffe auf die Bausteine gefiltert (vereinfacht läßt sich eine derartige Schnittstelle mit den Programmbibliotheken für bestimmte Anwendungsbereiche wie mathematische Funktionen vergleichen; für den Programmierer war es nur interessant wie die Funktion lautete; die Implementierung war dem Programmierer nicht bekannt). Änderungen der Bausteine haben somit nur noch Auswirkungen auf die Schnittstelle selbst, solange keine Funktionalität benötigt wird.
  Durch die Verwendung von standardisierten Schnittstellenbeschreibungssprachen (IDL von CORBA oder DCOM) wird die Spezifikation programmiersprachen- und auch protokollunabhängig. Insbesondere werden spätere Portierungen vorbereitet und vereinfacht.
  Ein weiterer Vorteil der Schnittstellen ist die Möglichkeit Bausteine mit geringerem Aufwand auszutauschen. Denn bislang bedeutete der Austausch eines Bausteins immer umfangreiche Anpassungen in allen Moduln, die diesen Baustein benutzen. Dies wird auf die Anpassung der Schnittstelle reduziert. Durch die Kapselung wird erreicht, daß ein Unternehmen unabhängiger hinsichtlich proprietärer Komponenten wird, die in einem Anwendungssystem immer erforderlich sind (beispielsweise die Oberflächen, Datenbanken).
- Trennung von Präsentations-, Applikations- und Datenhaltungslogik, um eine Zerlegung der Monolithen zu erreichen. Dies ist außerordentlich wichtig, weil durch die 3-Schichten-Architektur eine höhere Flexibilität erreicht wird. So kann durch die Abtrennung der Applikationslogik der Einsatz von Workflow-Management-Werkzeugen unterstützt werden.
  Mit der Trennung der Datenhaltungslogik (siehe als Beispiel [12]) kann auch eine Portierung der Daten auf relationale bzw. objekt-orientierte Datenbanken durchgeführt werden. Ferner erleichtert die Datenhaltungsschicht wegen der Kapselung eine spätere Migration.
- Zerlegung des monolithischen Hostsystems in Komponenten ist erforderlich, um die Evolution zu unterstützen. Es ist einfacher kleinere Komponenten zu verändern, die unabhängig voneinander sind, als ein komplexes System. Ferner wird in Zusammenhang mit den standardisierten Schnittstellen die Möglichkeit geschaffen, daß Komponenten ausgetauscht werden können, ohne die Verfügbarkeit des Gesamtsystems zu beeinträchtigen. Für die Zerlegung sollte die Technologie der Objekt-Orientierung angewendet werden.
- Aufbau eines Datenmodells für das gesamte Unternehmen führt dazu, daß die Datenhaltung vereinheitlicht wird. Das Datenmodell gibt vor, wie Daten strukturiert werden und welche Typen ihnen zugrunde liegen. Auf Basis des Datenmodells lassen sich dann auch Datensammlungen in Form eines Dataware-Houses anlegen, die dann mit entsprechenden Datamining-Techniken bearbeitet und kondensiert werden können. Ob ein relationales oder objekt-orientiertes Datenmodell angewendet wird, ist im Einzelfall zu entscheiden. Für mengenorientierte Zugriffe (Ermittlung aller Konten eines Kunden) ist ein relationales Datenmodell geeignet, während bei navigierendem Zugriff (Ermittlung der Kreditnehmereinheiten) ein objekt-orientiertes vorzuziehen ist.

Die Inhalte sind nicht losgelöst voneinander sondern haben gegenseitige Abhängigkeiten. Unsere Erfahrungen zeigen, daß in einem Schritt nicht alle Teilaspekte berücksichtigt werden sollten, was dem "Big-Bang"-Ansatz entsprechen würde. Vielmehr kann der Fokus durch fachliche Projekte beschränkt werden, um die Migrationskomplexität zu reduzieren und damit die Umsetzbarkeit zu vereinfachen (siehe nachfolgenden Abschnitt).

Definition der Migrationsstufen durch fachliche Themen

Eine Architekturmigration ist mit erheblichen Kosten und Aufwänden verbunden, wobei der Nutzen kaum monetär zu bewerten ist, da sich dieser erst in Form von reduzierten Wartungs- und Pflegeaufwänden bzw. durch schnellere Entwicklung zukünftiger Komponenten darstellt.
Daher empfehlen wir, daß bankfachliche Projekte als "Treiber" der Architekturmigrationen dienen. Denn derartige Projekte haben nach Einführung des entwickelten DV-Systems einen signifikanten Nutzen durch Prozeßverbesserungen und können somit die Kosten für die technische Migration teilweise mittragen. Außerdem werden durch das fachliche Projekt auch die Inhalte und Reihenfolge der Migrationsstufen vorgegeben.
Um die Kopplung von fachlichen und technischen Themen schon zu Projektbeginn zu verdeutlichen, empfehlen wir, mit einem Initialisierungsprojekt "Voruntersuchung" noch vor dem eigentlichen Projekt zu beginnen. In der Voruntersuchung wird aus fachlicher Sicht die Thematik näher spezifiziert und mit der technischen Sichtweise abgestimmt. Für die technischen Aspekte wird eine Architekturstudie im Rahmen des Initialisierungsprojektes durchgeführt, für die die Autoren ein detailliertes Vorgehensmodell entwickelt haben (siehe [5]), welches in verschiedenen Projekten bereits mit Erfolg angewendet wurde. Dieses Modell umfaßt

neben der detaillierten Beschreibung der einzelnen Aktivitäten Methoden und Techniken, die aus unserer Erfahrung angewendet werden sollten, auch Checklisten für die Aktivitäten sowie Hinweise auf Best Pratices.
Es würde den Rahmen dieses Papiers übersteigen, detailliert auf das Modell einzugehen. Daher beschränkt sich das Autorenteam auf die Nennung der Aktivitäten im Rahmen der Voruntersuchung, die zeitlich befristet ist (max. 3 Monate): (1) Identifikation eines bankfachlichen DV-Projektes, (2) Definition der fachlichen Inhalte und der strategischen und technischen Anforderungen, (3) Festlegung der Realisierungsstufen für die fachlichen und technischen Belange und (4) Ermittlung der Kosten. Die ersten drei Aktivitäten sind in der nachfolgenden Abbildung grafisch dargestellt. Es ist wichtig, daß die fachlichen und technischen Teilprojekte eng miteinander verzahnt sind.



Resultat der Architekturstudie ist die Definition von Stufen für die Umsetzung von Architekturbestandteilen. Allerdings sollten die Stufen auch die Architekturvision berücksichtigen, um zu vermeiden, daß Tatsachen geschaffen werden, die spätere Migrationen behindern. Bei der Stufendefinition ist neben fachlichen und technischen Aspekten natürlich auch der Zeit- und Budget-Rahmen des Projektes zu beachten.
Ein weiteres Ergebnis der Voruntersuchung ist die Ermittlung der Kosten für die fachlichen und technischen Projektinhalte. Die Kosten müssen auf die Projekt- und Architekturkosten aufgeteilt werden. Es ist dabei wichtig, daß die Architekturkosten nur bedingt dem bankfachlichen Projekt anzulasten sind, da die Erkenntnisse projektübergreifend sind.

Auswirkungen einer Migration auf die Bank

Ein Migrationsprojekt wird letztlich von der Bank und somit den Mitarbeitern der Bank durchgeführt. Daher ist die Mitarbeit und Unterstützung von Key-Playern der Bank unabdingbar, da diese letztlich den Erfolg und Mißerfolg des Projektes entscheidend beeinflussen. Daher ist unbedingt erforderlich, daß die Bankmitarbeiter frühzeitig informiert und für die Ideen begeistert und entsprechend der künftigen Anforderungen geschult werden müssen.
Auf Geschäftsführungsebene bedeutet dies, daß ein Migrationsprojekt hin zu einer komponentenbasierten Architektur als ein strategisches Unternehmensziel definiert werden muß (siehe [8]). Nur so ist sicherzustellen, daß zum einen die notwendige Motivation bei allen Bankmitarbeitern sichergestellt wird und zum anderen gewährleistet wird, daß projektspezifische und technische Interessen sich nicht gegenseitig behindern.

Abschließende Bemerkungen

Es mag vielleicht vermessen klingen, daß die Autoren gerade in einer Zeit, wo Banken stärker miteinander fusionieren, Trends aufzeigen, die eher das Gegenteil bewirken. Denn die Fusionsbestrebung sind aus der Notwendigkeit der Marktverteilung im Zuge der Globalisierung zu sehen, um einerseits ausländischen Übernahmen entgegenzuwirken und andererseits den Marktzutritt globaler Anbieter abzuwenden. Dabei muß berücksichtigt werden, daß die Fusionsbestrebungen der vergangenen Jahre vor allem politisch und unter Managementgesichtspunkten motiviert waren[6]. Allerdings zeigen bereits heutige Bestrebungen wie die Einrichtung von Kompetenzcentren innerhalb von Großbanken, daß die Universalbank nur noch auf Zeit existieren wird.

[6] Das beste Beispiel ist die Fusion der LG, SüdWEST LB und der L-Bank zu Landesbank Baden-Württemberg. Die drei Banken sind in Ihrer Geschäftstruktur sehr heterogen und die Fusion wurde vom Landesvater Teufel forciert.

Die Autoren sind der Überzeugung, daß es in 5 - 10 Jahren Spartenbanken wie Vertriebs-, Produktions- und Portfoliobanken geben wird, um die Anforderungen des Marktes und des Kunden am besten abzudecken.
Neben den organisatorischen Umstrukturierungen ist künftig eine höhere Flexibilität bei den Prozeßabläufen und den Produkten gefragt, welche nur mit adäquater DV-Unterstüt-zung umsetzbar ist. Den Anforderungen werden heutige DV-Systeme nicht gerecht, weil sie auf geschlossenen Architekturen beruhen. Nur ein Umstieg auf komponentenbasierte Architekturen ermöglicht die Konzeption von DV-Systemen, die trotz stetig neuer fachlicher und technologischer Herausforderungen offen und erweiterbar sind.
Der Umstieg von einer hostbasierten Architektur auf eine komponentenbasierte ist allerdings komplex und sollte in mehreren Schritten/Stufen durchgeführt werden. Das Stufenkonzept orientiert sich dabei an konkreten DV-Projekten, um die resultierenden Implikationen auf die Organisation und das soziale Umfeld besser zu beherrschen (die Erkenntnisse aus [4] sind auch auf Architekturen übertragbar). Dieser langfristige Umstieg versetzt die Bank in die Lage, zeitnah und flexibel auf die sich ändernden fachlichen und technologischen Markterfordernisse zu reagieren und damit wettbewerbsfähig zu bleiben.

Literatur

[1] L. Bass, P. Clements, R. Kazman: "Software Architectures in Practice", Addison Wesley Longman, 1998

[2] M. L. Brodie, M. Stonebraker: "Migrating Legacy Systems: Gateways, Interfaces & the incremental Approach", Morgan Kaufmann Publisher, San Francisco, 1995

[3] M. Cook: "Building Enterprise Information Architectures", Prentice Hall, 1996

[4] S. Dewal: "CASE Deployment: Improving Software Development by Transfer of Technology“, In: Software Engineering Environments Conference, Reading, UK, Proceedings, Los Alamitos, CA: IEEE Computer Society Press, Jul‘1993

[5] S. Dewal: “Voruntersuchung: Konzept für die Vorghensweise bei Architekturstudien”, interner Bericht, debis Systemhaus Dienstleistungen GmbH, BU Banken Düsseldorf, Jun’1999

[6] T. Fischer, A. Rothe: "Industrielle Fertigung von Bankensoftware", erschienen in J. Moormann, T. Fischer: "Handbuch Informationstechnologie in Banken", Gabler, 1999

[7] J. Hagel, A. G. Armstrong: "Net Gain : Expanding Markets Through Virtual Communities", Harvard Business School Press, 1997

[8] M. Keil, P. E. Cule, K. Lyytinen, R. C. Schmidt: "A Framework for identifying software projekt risks", CACM, Vol. 41, No. 11, pp 76 - 83, Nov'98

[9] D. Lange, M. Oshima: "Programming and Deploying Java Mobile Agents with Aglets", Addison Wesley Longman, 1998

[10] Mehlmann, Landvogt, Jameson, Rist, Schäfer; Einsatz Bayes'scher Netze zur Identifikation von Kundenwünschen im Internet; in: KI (Künstliche Intelligenz), Heft 3/98, Sept. 1998, S.43 ff

[11] Mehlmann, 0.; Multimediale Marktplätze - ein Medium zum elektronischen Handel; in: Manfred Nagl (Hrsg.); Online '97 Congressband VI; Online GmbH Velbert 1997; S. C611.01 ff

[12] N. Schumacher, M. Buchheit, K. Teille: "Design und Implementierung einer Objektzugriffsschicht für verteilte Systeme", Objektspektrum, Jan/Feb'99

[13] C. Szyperski: "Components vs. Objects vs. Component Objects", Proceedings of OOP, München, Jan'99

[14] R. Tanler: "The Intranet Data Warehouse", Wiley & Sons, 1997