Kopplung operativer (horizontaler)

Applikationen mit dem Data Warehouse

Verfasser: G. Auth, T. Frie

Lehrstuhl: Prof. Dr. R. Winter
Bericht Nr.: BE HSG/CC DW2/01
Datum: September 2001

Universität St. Gallen -
Hochschule für Wirtschafts-, Rechts-
und Sozialwissenschaften (HSG)

Institut für Wirtschaftsinformatik
Müller-Friedberg-Strasse 8
CH-9000 St. Gallen
Tel.: ++41 / 71 / 224 2934
Fax: ++41 / 71 / 224 2189

Prof. Dr. A. Back
Prof. Dr. W. Brenner
Prof. Dr. H. Österle
Prof. Dr. R. Winter (geschäftsführend)

Das Kompetenzzentrum Data Warehousing 2 (CC DW2)

In grösseren Unternehmen existieren eine Vielzahl verschiedener und häufig sehr heterogener Informationssysteme. Neue Problemstellungen und sich dynamisch verändernde Geschäftsmodelle machen es jedoch erforderlich, dass vorhandene Datenquellen auch integriert, das heisst unabhängig von ihrem operativen Einsatzzweck genutzt werden können. Die Zielvorstellungen reichen von einer einheitlichen Kundensicht bis hin zu schnell verfügbaren Führungsinformationen.

Während im technischen Bereich des Data Warehousing in den vergangenen Jahren erhebliche Fortschritte erzielt worden sind, fehlt es noch immer an gesicherten Erkenntnissen in Bereichen wie:

- Applikationsintegration,
- Entwicklung von Strategien zur operativen Nutzung des Data Warehouse,
- Integration des Operational Data Stores (ODS) ins Data Warehouse,
- Metadatenmanagement,
- Datenqualitätsmanagement und
- Datenschutz und –sicherheit beim Data Warehousing.

Im Rahmen des Kompetenzzentrums Data Warehousing 2 (CC DW2) entwickelt das Institut für Wirtschaftsinformatik der Universität St. Gallen (IWI-HSG) zusammen mit namhaften Partnerunternehmen Methoden und Referenzlösungen für die genannten Bereiche. Die folgenden Unternehmen sind Partner im CC DW2:

- ARAG Lebensversicherungs-AG (DE)
- Credit Suisse (CH)
- Mummert + Partner (DE)
- Rentenanstalt/Swiss Life (CH)
- SwissRe (CH)
- UBS AG (CH)
- Winterthur Versicherungen (CH)
- W & W AG (DE)

Durch die entstehenden dedizierten Problemlösungen kann das Data Warehouse als wichtige Komponente langfristig und in wirtschaftlicher Weise in das betriebliche Informationsmanagement integriert werden.

Der vorliegende Arbeitsbericht ist dem Themenbereich Applikationsintegration und Entwicklung von Strategien zur operativer Nutzung des Data Warehouse zuzuordnen.

Inhaltsverzeichnis

Das Kompetenzzentrum Data Warehousing 2 (CC DW2) I

Inhaltsverzeichnis III

Abbildungsverzeichnis V

Tabellenverzeichnis VI

1 Einleitung 1

1.1 Ausgangslage 1

1.2 Ziel des Arbeitsberichts 2

1.3 Aufbau der Arbeit 2

2 Auslöser für die Integration horizontaler Applikationen 5

3 Modell der betrieblichen Applikationslandschaft 9

4 Integrationskonzepte für horizontale Applikationen 13

4.1 Integration mittels Schnittstellen 13

4.2 Integration mittels Operational Data Store 14

4.3 Integration mittels Messaging 16

5 Architekturszenarien für die Anbindung von horizontalen Applikationen 19

5.1 Fallbeispiel 1: Kreditvergabe im Internet 19

5.1.1 Szenario und Prozessmodell 19

5.1.2 Datenkategorien 22

5.1.3 Informationssystemarchitektur zum Anwendungsszenario 23

5.2 Fallbeispiel 2: Adressänderung über ein Call Center 31

5.2.1 Szenario und Prozessmodell 31

5.2.2 Datenkategorien 33

5.2.3 Informationssystemarchitektur zum Anwendungsszenario 34

5.3 Fallbeispiel 3: Wertpapierhandel im Internet 39

5.3.1 Szenario und Prozessmodell 39
5.3.2 Datenkategorien 41
5.3.3 Informationssystemarchitektur zum Anwendungsszenario 43
6 Zusammenfassung und Ausblick 51
Literaturverzeichnis 55

Abbildungsverzeichnis

Abbildung 1: Vertikale Applikationen....9

Abbildung 2: Vertikale Applikationen und Querschnittsapplikationen....10

Abbildung 3: Applikationslandschaft mit horizontalen Applikationen....11

Abbildung 4: Schnittstellen zwischen vertikalen und horizontalen Applikationen....14

Abbildung 5: Integration horizontaler und vertikaler Applikationen mit Hilfe eines ODS 15

Abbildung 6: Messaging-basierte Kopplung operativer Systeme 17

Abbildung 7: Kombination aus ODS- und Messaging-Architektur 17

Abbildung 8: Prozessmodell zum Anwendungsszenario „Internet-Applikation“ 20

Abbildung 9: Datenflüsse zum Anwendungsszenario „Internet-Applikation“....24

Abbildung 10: Datenrückflüsse zum Anwendungsszenario „Internet-Applikation“....27

Abbildung 11: IS-Architektur zum Anwendungszenario „Internet-Applikation“ 29

Abbildung 12: Prozessmodell zum Anwendungsszenario „Call Center“....32

Abbildung 13: Datenflüsse zum Anwendungsszenario „Call Center“ 37

Abbildung 14: Datenrückflüsse zum Anwendungsszenario „Call Center“ 38

Abbildung 15: Prozessmodell zum Anwendungsszenario „Wertpapierhandel“ 40

Abbildung 16: Datenflüsse zum Anwendungsszenario „Wertpapierhandel“....44

Abbildung 17: Datenrückflüsse zum Anwendungsszenario „Wertpapierhandel“....47

Abbildung 18: IS-Architektur zum Anwendungsszenario „Wertpapierhandel“ 49

Tabellenverzeichnis

Tabelle 1: Datenkategorien zum Anwendungsszenario „Internet-Applikation“ 23

Tabelle 2: Bewertung der Datenflüsse 26

Tabelle 3: Bewertung der Datenrückflüsse 28

Tabelle 4: Entwurfsentscheidung Data Warehouse/ODS 30

Tabelle 5: Datenkategorien zum Anwendungsszenario „Call Center“ 34

Tabelle 6: Anforderungen und Charakteristika der Informationssystemarchitektur.....35

Tabelle 7: Datenkategorien, Zugriffsarten und Aktualitätsanforderungen 36

Tabelle 8: Datenkategorien zum Anwendungsszenario „Wertpapierhandel“.....42

Tabelle 9: Bewertung der Datenflüsse 46

Tabelle 10: Bewertung der Datenrückflüsse 48

1 Einleitung

1.1 Ausgangslage

Die Integration von Daten zu einer unternehmensweiten Informationslogistik steht für eine Vielzahl von Unternehmen mehr denn je im Mittelpunkt ihrer Informatikstrategie. Nachdem sich das Data-Warehouse-Konzept für die Integration von Daten zu analytisch-dispositiven Zwecken etabliert hat, wird nun in Wissenschaft und Praxis eine Diskussion über die operative Nutzung der integrierten Daten des Data Warehouse geführt. Die Erweiterung der Grenzen einer schwerpunktmässig auf Datenanalyse ausgerichteten Data-Warehouse-Konzeption hat zum Ziel, auch die Datenintegration in horizontaler Richtung, wie zum Beispiel für Call-Center- und E-Commerce-Aktivitäten erforderlich, durch das Data Warehousing zu unterstützen. Die bisherige konzeptionelle Trennung zwischen operativen Systemen zur Abwicklung von Geschäftsvorfällen und analytisch-dispositiven Systemen zur Unterstützung von Management-Entscheidungen wird dabei immer undeutlicher, da die Bedeutung analytischer Fragestellungen in operativen Systemen weiter zunimmt. Beispielsweise nutzen CRM-Anwendungen immer stärker Erkenntnisse über Kunden, die aus Fragestellungen mit analytischem Charakter entstehen [Forrester 2000].

Vor diesem Hintergrund werden zunehmend Anforderungen geäussert, die von der ursprünglichen Data-Warehouse-Konzeption nicht abgedeckt werden. Ein Beispiel hierfür ist der zeitnahe (real-time) Zugriff auf die Data-Warehouse-Datenbasis im Rahmen einer operativen Nutzung. Die operative Nutzung integrierter Daten steht im Mittelpunkt des Konzepts des Operational Data Stores (ODS). Im ODS werden operative Daten für horizontalen Applikationen zeitnah zusammengeführt. Im Unterschied zum Data Warehouse steht jedoch nicht allein die zeitintensive Analyse der Daten zur Entscheidungsunterstützung im Vordergrund. Durch die bereits aus dem Namen ersichtliche mehr operative Ausrichtung erscheint der Einsatz des ODS zur Integration heterogener Applikationen zu einer umfassenden Informationslogistik sinnvoll. Dabei wird beabsichtigt, Unterschiede zwischen Applikationen für die jeweils andere Applikation transparent zu machen und diese durch Zwischenschaltung des ODS voneinander zu entkoppeln. Der Aufbau eines ODS kann jedoch auch neue Probleme mit sich bringen, z.B. durch die redundante Speicherung von Daten im ODS und im Data Warehouse.

Neben dem ODS stehen allerdings weitere Integrationskonzepte zur Verfügung, die im vorliegenden Arbeitsbericht ebenfalls auf ihre Eignung für die Realisierung einer unternehmensweiten Informationslogistik überprüft werden.

1.2 Ziel des Arbeitsberichts

Grundlegendes Ziel dieses Arbeitsberichts ist es, Konzepte zur Applikationsintegration zu diskutieren und auf ihre Eignung für die speziellen Anforderungen von horizontalen Applikationen zu überprüfen. Da in vielen Unternehmen Integrationsbestrebungen zu analytischen Zwecken mit dem Aufbau eines Data Warehouses realisiert wurden, stellt sich die Frage, inwieweit diese vorhandene integrierte Datenbasis und die zugrunde liegende Systemarchitektur für die neuen Anforderungen genutzt werden können. Aus dem Blickwinkel des Data Warehousing heraus werden weitere Integrationskonzepte betrachtet und Überlegungen angestellt, wie diese mit dem Konzept des Data Warehouse zu kombinieren sind. Im Vordergrund der Betrachtungen steht eine umfassende Orientierung über mögliche Ansätze für die Realisierung einer Architektur zur Kopplung horizontaler Applikationen mit dem Data Warehouse.

Der vorliegende Arbeitsbericht wendet sich an Fach- und Führungskräfte, die sich mit der Planung und Weiterentwicklung einer komplexen Informationslogistik beschäftigen. Die Diskussion der Integrationskonzepte wird schwerpunktmässig unter datenorientierten Gesichtspunkten geführt und richtet sich damit an Entwickler, Anwender und Berater aus den Bereichen Daten- und Informationsmanagement. Hierzu wird sich dem Themengebiet aus einer praxisnahen Sicht genähert. Auf Basis der grundlegenden Betrachtungen konnte implizit ein Konzept für ein methodisches Vorgehen systematisiert werden, das bei der Entwicklung und Umsetzung von Architekturkonzepten für die Kopplung operativer horizontaler Applikationen mit dem Data Warehouse dabei helfen kann, existierende Informationssystemarchitekturen im Data Warehousing weiterzuentwickeln.

1.3 Aufbau der Arbeit

Die in dem vorliegenden Arbeitsbericht dokumentierten Ergebnisse sind im Rahmen der Zusammenarbeit von namhaften Unternehmen des Finanzdienstleistungssektors mit dem Kompetenzzentrum Data Warehousing 2 (CC DW2) am Institut für Wirtschaftsinformatik der Universi-

tät St. Gallen erarbeitet worden. Konzeptleitende Idee ist die Einführung eines neuen Geschäftsmodells im Unternehmen, z. B. die Ausweitung von Geschäftsprozessen ins Internet. Dazu werden zunächst beispielhafte Prozessabläufe identifiziert, die von der zu entwerfenden Systemarchitektur optimal unterstützt werden sollen. Anhand der Anforderungen des gewählten Geschäftsmodells und seiner Prozesse werden verschiedene Architekturszenarien zur Umsetzung entwickelt. Dabei werden vor allem mögliche Datenflüsse zwischen den Komponenten der zu erarbeitetenden Architekturvarianten untersucht, da das besondere Interesse hierbei der Datenintegration gilt. Zur Komplexitätsreduktion wird die Betrachtung auf einen Ausschnitt besonders relevanter Datenkategorien beschränkt. Dieses Vorgehen zielt auf eine Analyse der verschiedenen Integrationsformen und der mit ihnen verbundenen Koordinationsaufgaben aus dem besonderen Blickwinkel des Data Warehousing.

Für das Verständnis der Problematik ist es unumgänglich, sich zunächst mit der Entwicklung von Integrationskonzepten auseinander zu setzen. Dazu vermittelt Kapitel 2 einen Überblick über die Gründe für die ständig zunehmende Bedeutung der Integration für Informationssysteme. Hieraus werden grundsätzliche Anforderungen horizontaler Applikationen an die Informationslogistik abgeleitet und analysiert. Als theoretischer Bezugsrahmen für die weiteren Ausführungen werden in Kapitel 3 das Modell der betrieblichen Applikationslandschaft und das zugrunde liegende Begriffsverständnis vorgestellt. Darauf aufbauend werden in Kapitel 4 konzeptionelle Lösungsansätze zur Gestaltung einer unternehmensweiten, integrierten Informationslogistik skizziert. Damit gibt es einen umfassenden Überblick über wichtige Technologien zur Realisierung der Integrationsformen.

Kapitel 5 dokumentiert die im Rahmen des CC DW2 erarbeiteten Ergebnisse und vermittelt damit einen Eindruck, wie durch Kooperation von Wissenschaft und Praxis Lösungsansätze für ein zentrales Forschungs- und Praxisproblem erarbeitet worden sind. Dieses Kapitel stellt exemplarisch anhand von Fallbeispielen die zurzeit von den Unternehmen behandelten Problemstellungen zu Fragen der Integration vor. Anschliessend werden auf Basis der vorgestellten Prozesse mögliche Architekturansätze skizziert und die grundlegenden Datenflüsse zwischen den Komponenten der Architektur aufgezeigt.

2 Auslöser für die Integration horizontaler Applikationen

In der Literatur wird unter Integration „der Vorgang der Herstellung einer Einheit aus Teilen sowie der Einbeziehung eines Teils in ein grösseres Ganzes“ [Riehm 1997, S. 10] verstanden. Obwohl die Vorteile integrierter Anwendungssysteme sowohl in der Wissenschaft als auch in der Praxis allgemein anerkannt sind (vgl. bspw. [Heine 1999] u. [Mertens 2000]), befinden sich typische Applikationslandschaften grösserer Unternehmen von einem integrierten Zustand immer noch weit entfernt. Hauptursache ist der historische Entstehungsprozess der betrieblichen Anwendungssysteme. Die Bedeutung der Integration ist immer wieder Motivation für die Entwicklung teils eher theoretischer, teils mehr pragmatischer Lösungsansätze. Beispielsweise wurden in der Vergangenheit grosse Anstrengungen unternommen, um Integrationsansätze zu entwickeln, die eine unternehmensweite Modellierung von Daten und Funktionen als Basis einer integrierten Informationslogistik ermöglichen. Insbesondere die Diskussion unternehmensweiter Datenmodelle (UDM) kann in diesen Bereich eingeordnet werden (vgl. z. B. [Ferstl, Sinz 1993, S. 99 ff.]). Viele Unternehmen sind allerdings bereits während der Modellierung an der hohen Komplexität dieser Aufgabe gescheitert. In anderen Fällen werden die erstellten Modelle nicht mehr gepflegt oder wurden nicht konsequent implementiert.

Durch den Übergang von einer funktionsorientierten Organisation der betrieblichen Leistungserstellung zu einer prozessorientierten Anfang der 90er Jahre, wurde versucht, den immer schneller wechselnden Umweltbedingungen Rechnung zu tragen. Die Möglichkeit, die Geschäftsprozesse flexibel anzupassen, wurde zu einem wichtigen Erfolgsfaktor. Daraus entstand die Anforderung nach möglichst schneller Umsetzung der neu entworfenen Prozesse in das Informationssystem. Während man sich in den Unternehmen zunächst isoliert daten-, funktions- oder prozessorientierten Integrationsansätzen gewidmet hatte, wurden nun umfassende Ansätze zur Entwicklung von Informationssystemarchitekturen mit einer Vielzahl von Integrationssichten und -ebenen vorgestellt wie bspw. die Architektur integrierter Informationssysteme ARIS von Scheer (vgl. bspw. [Scheer 1998]).

Trotz der zunehmenden Verbreitung umfassender IS-Architekturen, wie z. B. durch ERP-Systeme teilweise implementiert, werden in vielen Unternehmen immer noch geschäftskritische Funktionen von individuell entwickelten, monolithischen Applikationen ausgeführt, die nicht oder nur unzureichend in die Applikationslandschaft integriert sind. Das Ablösen dieser Altsys-

teme durch moderne Systeme ist mit enorm hohen Kosten verbunden und vollzieht sich deshalb nur schrittweise. Abhilfe für die applikationsübergreifende Analyse der Daten schafft hier das Data Warehouse. Dabei wird nicht weiter versucht, die Applikationen selbst zu integrieren, sondern eine neue Applikation übernimmt die Integration der Daten, die bisher logisch und physisch voneinander getrennt waren.

Zu diesen gewachsenen Applikationsstrukturen kommen fortlaufend weitere neue Anwendungen hinzu, welche die aktuellen Technologieentwicklungen nutzbar machen oder neue Strategien unterstützen. Ein Schwerpunkt der Planung und Entwicklung von unternehmensweiten Applikationsstrukturen in jüngerer Zeit ist die Einbindung übergreifender, sog. horizontaler Applikationen, die Zugriff auf Daten aus einer Vielzahl von bereits existierenden operativen Applikationen benötigen. Auslöser für die Entwicklung neuer Applikationen und damit einhergehend Anstrengungen zur Datenintegration können sowohl geschäftlicher als auch technologischer Natur sein, wie folgende Beispiele zeigen:

Verlagerung von Geschäftsprozesse in das Internet

Eine Möglichkeit für die Abwicklung von Transaktionen über das Internet ist die Entwicklung sog. Front-End-Applikationen, die den Zugriff auf vertikalen Applikationen über das Internet ermöglichen. In vielen Fällen ist das Vorschalten eines Web-Front-Ends aber nicht ausreichend, da die benötigten Daten von den betroffenen Transaktionssystemen nicht in akzeptablen Zugriffszeiten geliefert werden können. Um das Informationssystem an die Anforderungen des E-Business anzupassen, ist es vielmehr erforderlich, die vorhandenen Applikationen umfassend zu integrieren, ohne dabei die Flexibilität zu verlieren, bei neuen Anforderungen Anpassungen schnell vornehmen zu können [Fleisch, Österle 2001]. Noch immer bieten viele Unternehmen über ihre Internet-Präsenz lediglich Informationen über ihre Organisation sowie Dienstleistungen oder Produkte an. Es fehlt aber die Möglichkeit, über das Internet echte Transaktionen wie z. B. den Kauf von Produkten abzuwickeln. Das Internet wird hier noch nicht als Vertriebskanal genutzt. Beispielsweise ist es nur bei sehr wenigen Herstellern der Automobil-Branche möglich, das Wunschauto über das Internet zu konfigurieren und auch gleich zu bestellen. Die Ursache liegt oftmals im Unvermögen, die existierenden Applikationen für die Transaktionsabwicklung über herkömmliche Vertriebskanäle zu einer Informationslogistik für das E-Business zu integrieren.

Customer Relationship Management (CRM)

Nachdem sich in den letzten Jahren bei einer Vielzahl von Unternehmen die Erkenntnis durchgesetzt hat, dass zum erfolgreichen Bestehen im immer härter werdenden Wettbewerb eine konsequente Ausrichtung auf die Kundenbedürfnisse unverzichtbar geworden ist, steht nun die Implementierung oder Verbesserung von konkreten Strategien für das Customer Relationship Management (CRM) an. CRM zielt darauf ab, ein möglichst umfassendes Wissen über den Kunden und die von ihm getätigten Transaktionen zu sammeln. Daraus entsteht ein beachtliches Potenzial für den Verkauf von ergänzenden Produkten oder Dienstleistungen (vgl. [Bach, Österle 2000], [Hofmann 2001]). Für das Erstellen eines möglichst detaillierten Kundenprofils ist es wichtig, Informationen aus allen Vertriebskanälen zusammenzuführen. Ob ein Kunde mit dem Unternehmen über das Internet kommuniziert, bei einem Call Center anruft oder von einem Aussendienst-Mitarbeiter besucht wird, muss zwar einerseits bei Auswertung unterschieden werden können, darf aber andererseits keinen Unterschied im Hinblick auf die Vollständigkeit der gespeicherten Informationen verursachen. Alle verfügbaren Informationen müssen integriert werden, um eine einheitliche, integrierte Kundensicht zu erhalten.

Aus Sicht des Kunden ist es von grosser Bedeutung, bei wiederkehrenden Kontakten eine einheitliche Unternehmensdarstellung anzutreffen. Der Kunde sucht nach einem verlässlichen Partner für seine Bedürfnisse und erwartet, dass der Partner ihn und seine Bedürfnisse erkennt. Entsteht dagegen beim Kunden der Eindruck vom Unternehmen als unkoordinierte Ansammlung von Abteilungen und Mitarbeitern, so wird das Vertrauensverhältnis gestört. Im Extremfall kann dies zum Verlust des Kunden führen. Daher zielt CRM auf das Festigen der Kundenbeziehungen, um die hohen Kosten für die Anwerbung von Neukunden zu senken. Für die Applikationen, mit deren Hilfe eine CRM-Strategie implementiert wird, ergibt sich daraus neben der Anforderung, integrierte und konsistente Informationen zu liefern, vor allem auch die Anforderung diese Informationen zeitnah bereitzustellen.

Supply Change Management (SCM)

Ähnliche Anforderungen erwachsen aus dem Bedürfnis nach einer verstärkten Integration mit Lieferanten und Partnern im Rahmen des Supply Change Managements (SCM). Ziel ist hierbei, den Informationsaustausch zwischen Unternehmen möglichst weitgehend zu automatisieren, um so Kosteneinsparungen zu realisieren. Dafür nötig sind standardisierte Schnittstellen und Austauschformate, die etwa mit Electronic Data Interchange (EDI) seit längerem existieren. Die Integration zweier Unternehmen über EDI ist allerdings sehr aufwendig und kostspielig, wodurch

Kosteneinsparungen wieder neutralisiert werden. Ein Beispiel für einen technologischen Auslöser ist die Entwicklung der Datenbeschreibungssprache Extensible Markup Language (XML). Mit XML ist seit kurzem eine kostengünstige, Web-basierte Alternative verfügbar, die SCM-Lösungen auch für kleinere Unternehmen attraktiv macht. Grundvoraussetzung bleibt aber auch hier die Möglichkeit, unternehmensweit integrierte, konsistente und hinreichend aktuelle Daten bereitstellen zu können (vgl. [Ruh, Maginnis, Brown 2001]).

Für die Integration der Daten zu analytisch-dispositiven Zwecken liegt mit dem Data-Warehouse-Konzept ein mittlerweile bewährter Architekturrahmen vor. Ausgehend von den erprobten Konzepten des Data Warehousing sind erweiterte Architekturen denkbar, die auch die Daten der horizontalen Applikationen miteinbeziehen und somit eine unternehmensweit integrierte Informationslogistik schaffen. Eine wichtige Rolle in solchen erweiterten Architekturen wird dabei immer öfter dem Operational Data Store zugeschrieben.

Ausgehend von diesen Überlegungen wird im folgenden Kapitel das Modell der betrieblichen Applikationslandschaft eingeführt, dass eine Klassifizierung und Strukturierung sämtlicher Typen von Applikationen im betrieblichen Umfeld zum Ziel hat.

3 Modell der betrieblichen Applikationslandschaft

Als erster Schritt von Intergrationsbemühungen empfiehlt sich eine Bestandsaufnahme der gegenwärtig im Unternehmen existierenden Applikationen und ihrer Beziehungen untereinander. Ein solches Modell der betrieblichen Applikationslandschaft vermittelt eine Gesamtsicht auf die Informationsverarbeitung im Unternehmen, ohne die eine auf die Unternehmensziele ausgerichtete Planung und Weiterentwicklung des betrieblichen Informationssystems kaum möglich ist (vgl. [Winter 2000b]).

Ein erstes, einfaches Modell der Applikationslandschaft ordnet Applikationen nach den Merkmalen „Funktion“, „Produkt(gruppe)“ und „Prozess“ ein. Damit lassen sich zunächst die traditionellen operativen Applikationen beschreiben, die zur Unterstützung der Geschäftsprozesse dienen und hauptsächlich auf die massenhafte Verarbeitung von Transaktionen ausgerichtet sind. Diese Applikationen umfassen verschiedene Komponenten, die in ihrer Gesamtheit alle funktionalen Aspekte und die vollständigen Geschäftsprozesse für eine bestimmte Produktgruppe abdecken. Für eine Versicherung wären dies beipielsweise die Anwendungssysteme zur Verwaltung der einzelnen Versicherungsprodukte wie Lebensversicherung oder Sachversicherung. In dem durch die drei Dimensionen Funktion, Produkt(gruppe) und Prozess aufgespannten dreidimensionalen Raum lassen sich die wegen des optischen Erscheinungsbildes im Modell auch vertikale Applikationen genannten Systeme wie folgt darstellen:

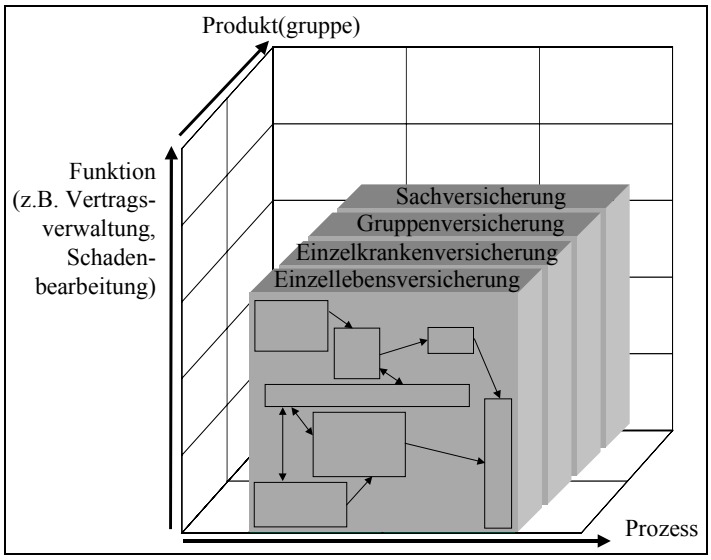


Abbildung 1: Vertikale Applikationen [Winter 2000a, S. 131]

Durch Zusammenführung bestimmter Querschnittsapplikationen in separate Applikationen entsteht ein neuer Typ von Applikationen, sog. Querschnittsapplikationen. Ein Beispiel ist die Auslagerung der Kundenverwaltung aus allen vertikalen Applikationen in eine einzige übergreifende Partnerapplikation. Hierdurch wird u. a. die Gefahr von Anomalien durch Redundanzen verringert. Obwohl die von Querschnittsapplikationen verwalteten Daten von allen anderen Applikationen gelesen werden, sind sie weiterhin als operative Daten zu betrachten. Die somit erweiterte Applikationslandschaft zeigt Abbildung 2:

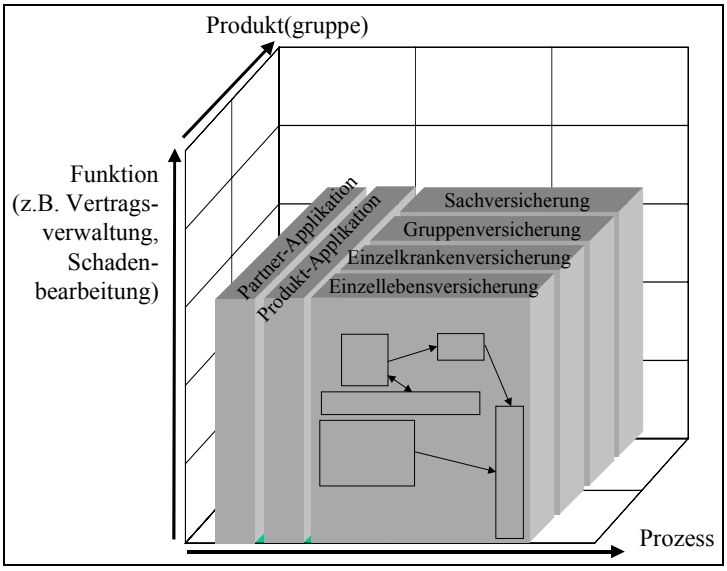


Abbildung 2: Vertikale Applikationen und Querschnittsapplikationen [Winter 2000a, S. 132]

Eine dritte Klasse von operativen Applikationen, die in jüngster Zeit durch die stark zunehmende Bedeutung elektronischer und telefoniebasierter Zugangskanäle entstand, sind die horizontalen Applikationen. In diesen dedizierten Anwendungssystemen werden bestimmte Funktionalitäten über alle Produkte bzw. Produktgruppen hinweg für einen spezifischen Abschnitt des Prozesses zusammengefasst. In der betrieblichen Applikationslandschaft lassen sich horizontale Applikationen wie folgt positionieren:

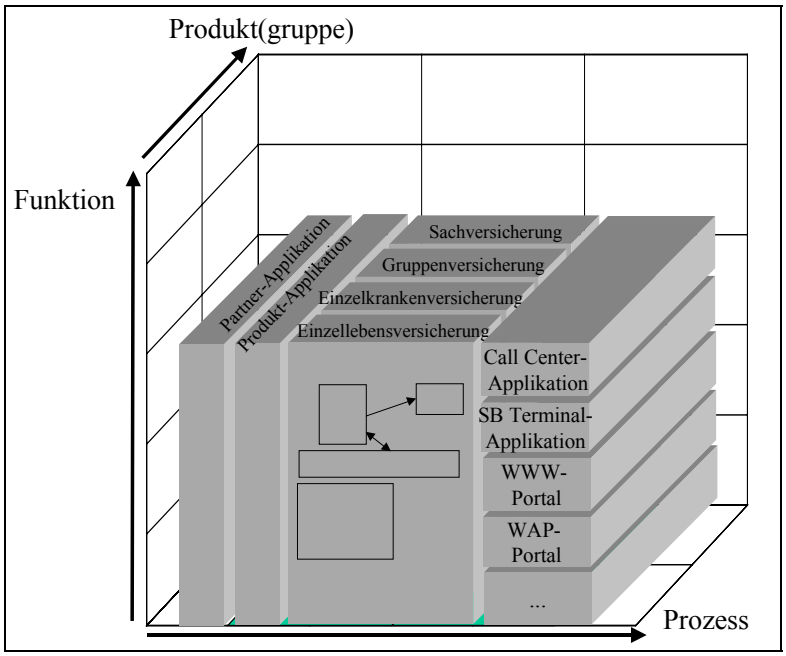


Abbildung 3: Applikationslandschaft mit horizontalen Applikationen [Winter 2000a, S. 135]

Die soeben beschriebenen Applikationstypen für die Abwicklung der Geschäftsprozesse lassen sich unter dem Begriff „operative Applikationen“ zusammenfassen. Im Unterschied dazu unterstützen die sog. analytisch-dispositiven Applikationen in erster Linie die Management-Prozesse. Mit dem Data-Warehouse-Konzept steht mittlerweile eine leistungsfähige Zwischenschicht zur Integration der Daten aus vertikalen operativen Applikationen zur Verfügung. Dadurch wird eine Entkopplung dieser operativen Applikationen von analytisch-dispositiven Applikationen erreicht. Die analytisch-dispositiven Applikationen greifen auf die notwendigen Daten nicht direkt zu, sondern setzen auf dem Data Warehouse auf. Durch die integrierten Daten des Data Warehouse werden einerseits eine Vielzahl von Auswertungen und Analysen überhaupt erst möglich, andererseits würde ein direkter Zugriff auf die operativen Systeme eine erheblich zusätzliche Systembelastung bedeuten. Darüber hinaus können analytisch-dispositiven Systeme auf Basis des Data Warehouse unabhängig von der Verfügbarkeit der operativen Systeme arbeiten. Durch die Entkopplung über das Data Warehouse wird eine erhöhte Verfügbarkeit der Daten erreicht. Um multidimensionale Analysen zu ermöglichen werden die Daten im Data Warehouse bzw. nachgelagerten Data Marts in speziellen Datenmodellen gespeichert, die für diesen Zweck optimiert sind (z. B. Star- oder Snowflake-Schema). Neben dem Datenmodell sind auch die Daten selbst auf die jeweiligen Analysezwecke abgestimmt. Dies erfolgt neben der Auswahl der relevanten Daten durch Bereinigung und Integration der Daten.

Ungelöst ist allerdings noch die Frage, ob dem Data Warehouse bei der Einbeziehung der horizontalen Applikationen eine ähnliche Rolle zukommen kann. Im folgenden Kapitel werden deshalb aktuelle Integrationskonzepte betrachtet. Darauf aufbauend werden in Kapitel 5 Architekturvarianten vorgestellt, in denen horizontale Applikationen – teilweise unter Zuhilfenahme weiterer Integrationskonzepte – mit dem Data Warehouse gekoppelt werden.

4 Integrationskonzepte für horizontale Applikationen

Wird bei der Integration von Daten aus heterogenen Applikationen nicht eine rein analytische Nutzung in den Vordergrund gestellt, stehen neben dem Data-Warehouse-Konzept eine Reihe von Alternativen zur Verfügung. In Abhängigkeit vom verfolgten Integrationszweck ergeben sich aus dem gewählten Konzept sowohl Vor- als auch Nachteile, die es zu berücksichtigen gilt. Bei Projekten zur Integration von horizontalen Applikationen in die betriebliche Applikationslandschaft werden in der Praxis momentan die Konzepte Integration mittels Schnittstellen, Integration mittels ODS oder Integration mittels Messaging favorisiert. Obwohl dies Konzepte zumeist nicht in Reinform sondern in Kombination miteinander implementiert werden, sind sie im folgenden zum besseren Verständnis getrennt beschrieben (vgl. [Schelp 2001]). Inwieweit sich die vorgestellten Konzepte mit dem Data-Warehouse-Ansatz kombinieren lassen, wird anschliessend in Kapitel 5 anhand von drei Fallbeispielen untersucht.

4.1 Integration mittels Schnittstellen

Zur Erfüllung ihrer Aufgabe benötigen horizontale Applikationen Zugriff auf Daten aus unterschiedlichen vertikalen Applikationen entlang der Wertschöpfungskette. Beispielsweise hängt der Erfolg eines WWW-Portals für Endkunden in hohem Masse davon ab, inwieweit der Kunde mit Hilfe des Portals seinen Kundenprozess abwickeln kann. Bereits bei einer simplen Bestellung, die nur eine Aktivität des Kundenprozesses darstellt, werden Adressdaten des Kunden, Stammdaten der Artikel, Bestandsdaten für Artikel, Daten über die Kreditwürdigkeit des Kunden u. ä. benötigt. Hierzu ist eine Anbindung der Portal-Applikation an die vertikalen Systeme nötig. Existieren neben der Portal-Applikation weitere vertikale Applikationen, entsteht an dieser Stelle schnell ein Schnittstellenproblem, wie nachfolgend dargestellt wird:

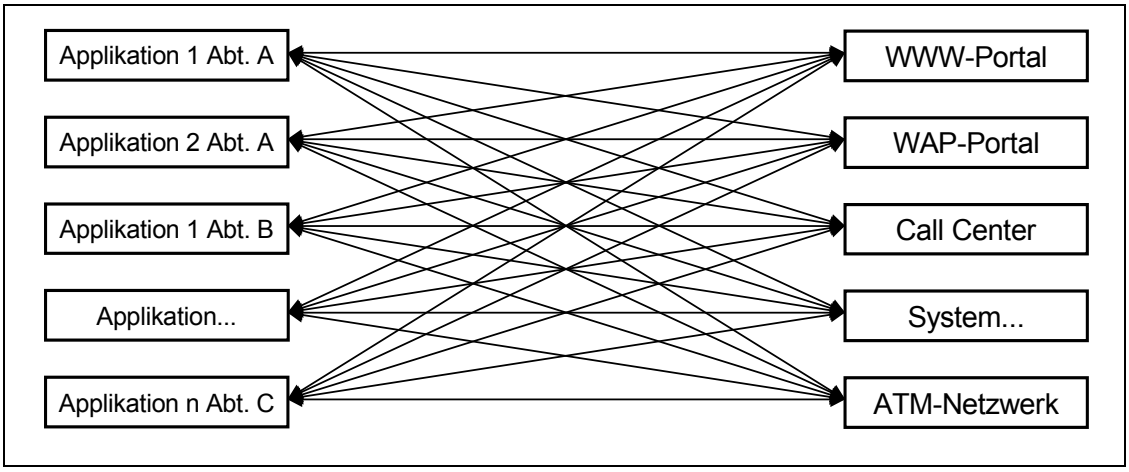


Abbildung 4: Schnittstellen zwischen vertikalen und horizontalen Applikationen [Schelp 2001]

Die hohen Anforderungen, die das massenhafte Abwickeln von Transaktionen über das Internet an das Informationssystem eines Unternehmens stellt, haben zu einer Belebung der Diskussion von Integrationskonzepten unter dem Stichwort Enterprise Application Integration (EAI) geführt. Enterprise Application Integration ist ein von Analysten geprägter Sammelbegriff für Konzepte, Methoden und Software-Werkzeuge zum Modernisieren, Konsolidieren und Koordinieren der Anwendungssysteme eines Unternehmens.

4.2 Integration mittels Operational Data Store

Das Bestreben, die operativen Daten zu einer logischen Gesamtsicht zusammenzuführen, liegt auch, mit einer anderen Zielsetzung, dem Data Warehousing zugrunde. Es erscheint somit sinnvoll zu prüfen, ob die Lösungsansätze aus dem Data-Warehouse-Bereich, auch auf diese Situation übertragen werden können. Mit dem Konzept des Operational Data Stores bietet sich eine Lösung an, die zu einer Architektur wie in Abbildung 5 führt.

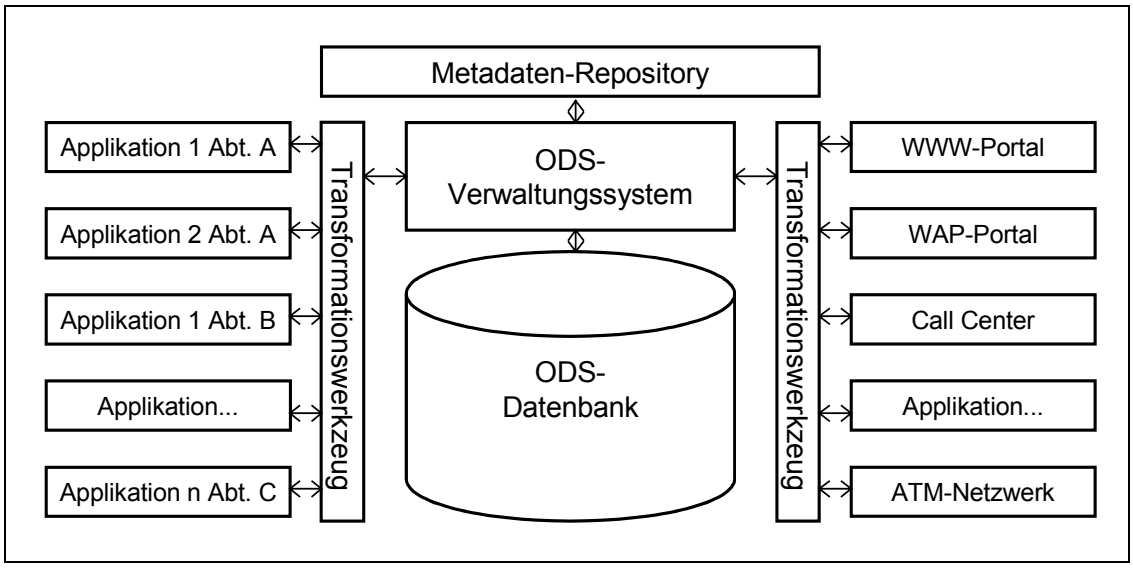


Abbildung 5: Integration horizontaler und vertikaler Applikationen mit Hilfe eines ODS [Schelp 2001]

Diese Architektur folgt prinzipiell der klassischen Data-Warehouse-Architektur, unterscheidet sich jedoch von ihr durch die besonderen Eigenschaften des ODS. Ähnlich wie im Data-Warehouse-System sind Transformationsschichten und ein Metadaten-Management nötig. In der ODS-Datenbank werden die von den horizontalen Applikationen benötigten Daten vorgehalten. Diese Daten, oft über eine Vielzahl vertikaler Applikationen verstreut, werden analog zu den Datenflüssen im Data Warehouse mit Hilfe einer Transformationsschicht aus den Quellsystemen extrahiert, aufbereitet und im ODS zusammengeführt. Die dafür nötigen Metadaten werden in einem Metadaten-Repository abgelegt. Im Unterschied zum Data Warehouse sind die Daten im ODS grundsätzlich nicht historisiert, können geändert werden und die Aktualität der Daten ist sehr hoch. Inmon beschreibt das Beispiel eines Flugreservierungssystems, bei dem Transaktionen im operativen System innerhalb von 1 bis 2 Sekunden im ODS nachgeführt werden. Gleichzeitig räumt er jedoch ein, dass die Entwicklung solcher hochperformanter Systeme sehr schwierig und aufwendig sowie nur für seltene Fälle wirklich nötig ist [Inmon 1999, S. 59 ff.].

Da in einem Data Warehouse definitionsgemäss Daten längerer Zeiträume vorgehalten werden, können in einem ODS analyserelevante, betriebliche Sachverhalte eines viel kürzeren Zeitraums abgebildet werden. Auf diese Weise können kurzfristig entstehende Informationsbedarfe für Analysen und Auswertungen befriedigt werden, die häufig auf mittleren und unteren Führungsebenen entstehen. Insofern kann ein ODS zukünftig als operative, integrierte Datenbasis für analytisch-dispositve Applikationen eingesetzt werden.

Ein möglicher Anwendungsbereich ergibt sich beispielsweise im Rahmen der (operativen) Budgetierung. Bisher sind die Planungsinformationen aus der Budgetierung für die verschiedenen Planungsbereiche dezentral in den dafür vorgesehenen operativen Systemen abgelegt worden. Somit ist die Budgetüberwachung (Vergleich von Soll-Werten der Planung mit den Ist-Werten) nur für den einzelnen Planungsbereich möglich. Die Folge ist, dass für eine das Gesamtunternehmen betreffende Budgetüberwachung aufwendige Auswertungsläufe notwendig sind, um der Unternehmensführung einen aktuellen Überblick über den Planfortschritt zu geben. Bereits realisierte oder zu erwartende Budgetüberschreitungen werden möglicherweise zu spät erkannt. Die Möglichkeit, schreibenden Zugriff auf den ODS vorzunehmen, erlaubt beispielsweise sowohl die Budgetierung als auch die Überwachung aller Planungsbereiche auf einem integrierten Datenbestand auszuführen. Aufwendige Auswertungsläufe können entfallen. Eine regelmässige Aktualisierung der ODS-Datenbasis mit Ist-Werten aus den Planungsbereichen liefert somit beständig – möglicherweise tagesaktuell – einen Überblick zum Planfortschritt. Informationsrückflüsse aufgrund von Budgetanpassungen können direkt im ODS erfasst und die Auswirkungen auf das Unternehmen überprüft werden.

4.3 Integration mittels Messaging

Es wird ersichtlich, dass sich der ODS aufgrund seines Eigenschaftsprofils weder den operativen Applikationen noch dem Data Warehouse eindeutig zuordnen lässt. Damit ergeben sich bei der Positionierung des ODS in der betrieblichen Applikationslandschaft Fragestellungen, die nicht ohne weiteres aus den Erfahrungen beim Data Warehousing beantwortet werden können. Erhebliche Probleme können beispielsweise die hohen Ansprüche an die Aktualität der Daten und die Möglichkeit des schreibenden Zugriffs auf den ODS verursachen.

In der Praxis haben sich daher neben den datenzentrierten ODS-Konzepten sogenannte Messaging-Lösungen als weitere Alternative zur Kopplung der operativen Systeme miteinander etabliert. Messaging bezeichnet den Austausch von Nachrichten (Messages) zwischen Applikationen in Form von spezifisch formatierten Daten, die beispielsweise Ereignisse, Nachfragen oder Antworten beschreiben. Die Koordinierung und korrekte Verteilung der Nachrichten übernimmt ein Messaging-Server, der über Kanäle zu allen beteiligten Applikationen verfügt. Messaging ermöglicht den eingebundenen Applikationen u. a. die Lokalisierung von Ressourcen, das Einhalten von Abhängigkeiten und Verarbeitungsbedingungen sowie die Wiederverwendung des Pro-

grammcodes für das Message-Handling. Die wesentliche Funktion ist jedoch die Kommunikation von unterschiedlichen Applikationen über System- und Plattformgrenzen hinaus. Konzeptionell lässt sich eine solche Architektur wie folgt darstellen:

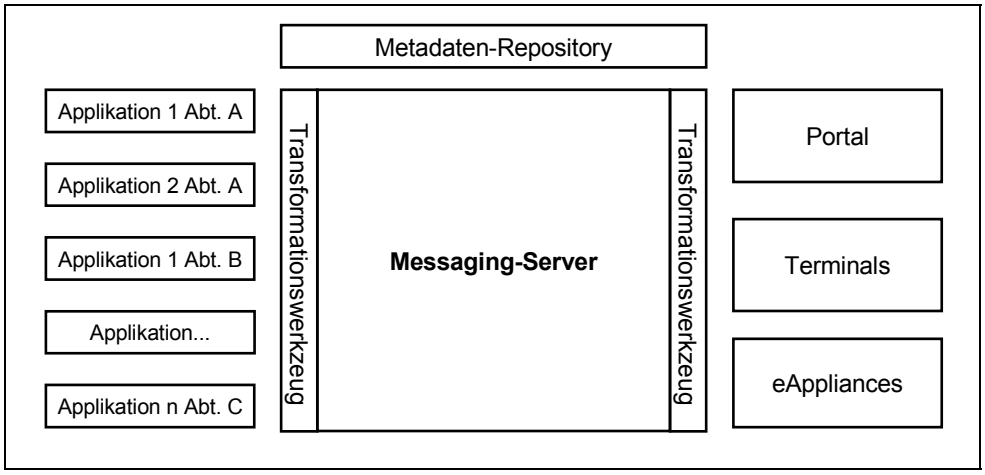


Abbildung 6: Messaging-basierte Kopplung operativer Systeme [Schelp 2001]

Probleme können bei dieser Architekturvariante u. a. durch die asynchrone Verarbeitung der Nachrichten in sogenannten Message-Stacks und bei hoher Kapazitätsauslastung der operativen Applikationen auftreten.

Für bestimmte Anforderungen erscheint daher die Betrachtung einer Kombination aus daten- und nachrichtenorientierten Architekturen sinnvoll. Abbildung 7 verdeutlicht den Aufbau einer solche Lösung auf konzeptioneller Ebene:

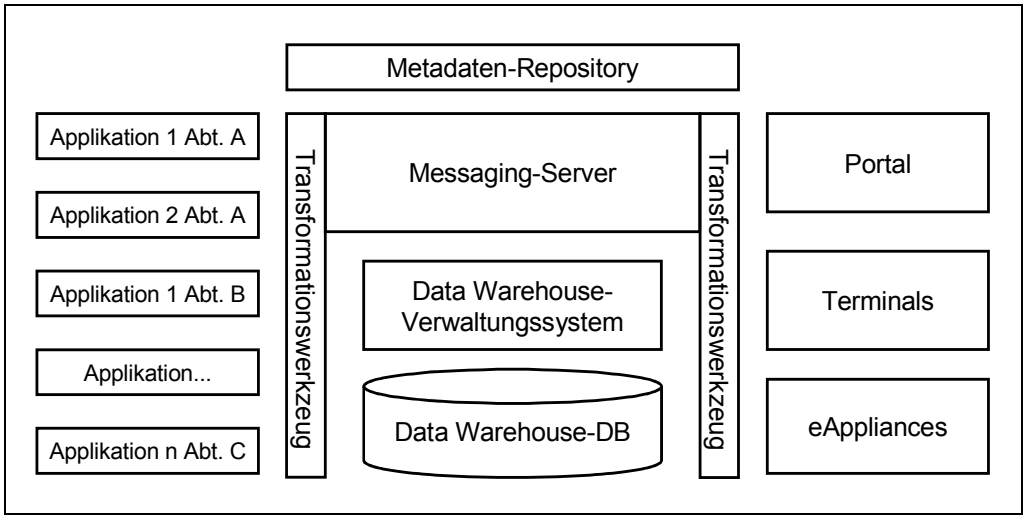


Abbildung 7: Kombination aus ODS- und Messaging-Architektur [Schelp 2001]

So können beispielsweise die sich weniger schnell ändernden Kundenprofile im ODS zwischengespeichert werden, während aktuelle Bestandsdaten für die Auftragsdurchführung aus den ope-

rativen Applikationen extrahiert werden. Da nur noch vergleichsweise wenige Daten in Echtzeit aus den Vorsystemen benötigt werden, kann der Zugriff auf diese auch bei Kapazitätsengpässen leichter performant gestaltet werden.

Im zurückliegenden Kapitel 4 wurden konzeptionelle Lösungsansätze für die Integrationsproblematik beschrieben. Im folgenden Kapitel werden unter Zugrundelegung dieser Konzepte erste Architektur-Szenarien vorgestellt, die auf eine flexible Integration von vertikalen, horizontalen und analytisch-dispositiven Applikationen zielen.

5 Architekturszenarien für die Anbindung von horizontalen Applikationen

Die in diesem Kapitel vorgestellen Fallbeispiele sind im Rahmen des CC DW2 erarbeitet worden. Ziel ist, die im Mittelpunkt dieses Arbeitsberichts stehende Frage der *Kopplung (horizontaler) Applikationen mit dem Data Warehous*e anhand von Anwendungsszenarien unter praxisnahen Bedingungen zu untersuchen. Im Vordergrund steht die Diskussion um den konzeptionellen Entwurf der Informationssystemarchitektur, die in Verbindung mit folgenden Fragen geführt wird:

1. Was spricht für bzw. gegen den Aufbau eines Operational Data Stores zusätzlich zum Data Warehouse?
2. Welche kritischen Anforderungen führen dazu, Daten im Operational Data Store bzw. im Data Warehouse vorzuhalten?
3. Welches sind die Hauptproblemfelder bei der Einbindung eines Operational Data Stores zusätzlich zum Data Warehouse?

5.1 Fallbeispiel 1: Kreditvergabe im Internet

5.1.1 Szenario und Prozessmodell

In diesem Kapitel wird ein Anwendungsszenario aus dem Bereich Finanzdienstleistungen skizziert. Es geht um die Beantragung, Vergabe und Auszahlung eines Kredits. Vorstellbar sind hier Konsumentenkredite, die zwar eine detaillierte Prüfung der Kundensituation erfordern, aber durchaus über das Internet abgewickelt werden können. Voraussetzung für die Vergabe des Kredits ist eine bestehende Kundenbeziehung zwischen dem Kreditinstitut und dem Kunden. Im anderen Fall wird lediglich ein unverbindliches Angebot erstellt. Für das Fallbeispiel wurde ein konkreter, möglichst repräsentativer Geschäftsvorfall entworfen. Abbildung 8 zeigt das Prozessmodell für den Geschäftsvorfall:

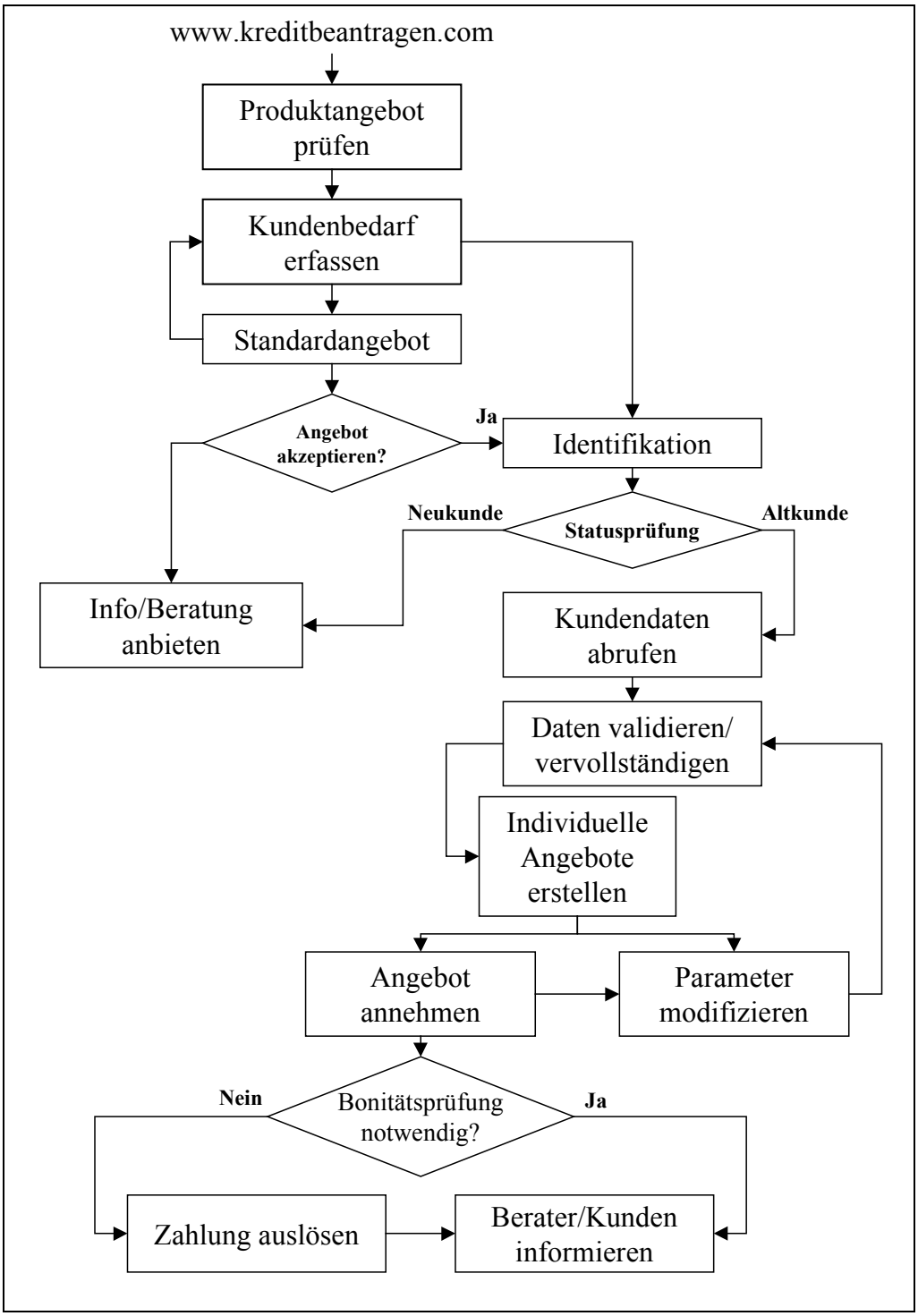


Abbildung 8: Prozessmodell zum Anwendungsszenario „Internet-Applikation“

Der Kunde besucht die Internet-Präsenz des Kreditinstituts, nachdem er durch eine Suche im Internet auf das Angebot aufmerksam geworden ist. Dabei spielt es zurzeit keine Rolle, ob der Besucher dieser Internet-Präsenz bereits Kunde dieses Kreditinstituts ist oder einen potenziellen Neukunden darstellt.

Der Kunde prüft zunächst das Produktangebot und entscheidet sich für die Möglichkeit, ein Kreditangebot erstellen zu lassen. Zu diesem Zweck muss der individuelle Kundenbedarf erfasst werden, indem die vom Kunden gewünschten Parameter des Kredits abgefragt werden, wie zum Beispiel Kreditsumme, Laufzeit sowie persönliche Daten (Geburtsdatum, Wohnort usw.). Anschliessend erfolgt die Berechnung des Kredits, der dem Kunden in Form eines Standardangebots präsentiert wird . Der Kunde kann sich nun durch Variation der Parameter weitere Angebote erstellen lassen oder sich für die Fortsetzung entscheiden. Durch letztere Variante entstehen zwei mögliche Fortsetzungsszenarien:

1. *Der Kunde betrachtet das Angebot als unverbindlich*: In diesem Fall werden detaillierte Informationen über das Unternehmen ausgegeben, wie zum Beispiel die nächstgelegene Niederlassung aufgrund der Eingabe seines Wohnortes bei der Erfassung der persönlichen Daten.

2. *Der Kunde entscheidet sich das Angebot anzunehmen*: In diesem Fall muss sich der Kunde identifizieren. Als Neukunde erfolgt die Weiterverarbeitung seines Antrags analog eines unverbindlichen Angebots mit dem Hinweis, dass er sich an die nächstgelegene Zweigstelle des Kreditinstituts wenden soll. Als Altkunde werden die für die Weiterverarbeitung benötigten Kundendaten abgerufen.

   Neben den bereits erfassten Daten zum Kredit sind für die endgültige Genehmigung des Kredits weitere Daten zu erfassen bzw. vom Kunden zu prüfen, wie zum Beispiel Nettoeinkommen, Dauer der Beschäftigung beim derzeitigen Arbeitgeber, Höhe der gewünschten Rückzahlungsraten oder Auszahlungswunsch. Nach der Prüfung der Bestandsdaten kann es zu einer Überarbeitung des bisherigen Angebots kommen, wenn die Kreditparameter nochmals modifiziert werden.

   Akzeptiert der Kunde das Angebot, so muss entschieden werden, ob eine weitergehende Bonitätsprüfung notwendig ist. Ist diese erforderlich, so ist der zuständige Kundenberater für diesen Kunden zu informieren, der daraufhin Kontakt mit diesem aufnimmt. Gleichzeitig wird der Kunde darüber informiert. Im anderen Fall wird die Zahlung ausgelöst. Auch in diesem Fall werden Kunde und Kundenberater über diesen Vorgang informiert, indem entsprechende Vertragsdokumente beispielsweise per E-Mail versandt werden.

5.1.2 Datenkategorien

Auf der Grundlage des vorgestellten Anwendungszenarios und der Prozessspezifikation werden Datenkategorien identifiziert. Die wichtigsten Datenkategorien sind in Tabelle 1 zusammengefasst. Die einzelnen Datenkategorien sind den Gruppen Besucherverhalten und Kundenverhalten zugeordnet. Diese Unterscheidung in Besucher, zu denen noch keine Geschäftsbeziehung besteht (Besucherverhalten), und solche, die bereits Kunde sind (Kundenverhalten), unterstützt die Analyse des Endbenutzerverhaltens und ermöglicht so eine ständige Überprüfung und Anpassung des Webangebots.

Prozessschritt	Datenkategorien	
Produktangebot prüfen	Produktinformationen	**Besucherverhalten**
Kundenbedarf erfassen	Kundenpräferenzen, Kreditparameter	
Standardangebot	Angebotsdaten, Kreditparameter, Produkt-/Zinsdaten	
Info/Beratung anbieten	Angebotsdaten, Kundenkontaktinformationen, Kundeninstruktionen, Beraterauftrag/-instruktionen	
Identifkation	Kundendaten (z. B. Login-, Security-Informationen)	**Kundenverhalten**
Kundendaten abrufen	Angebotsdaten, Kundenstammdaten	
Daten validieren/vervollständigen	Kundenstammdaten, Kundenzusatzdaten	
Individuelles Angebot erstellen	Angebotsdaten, Kreditparameter, Produkt-/Zinsdaten	
Parameter modifizieren	Kundenpräferenzen, Kreditparameter	
Angebot annehmen	Angebotsdaten, Kreditparameter, Produkt-/Zinsdaten	

Prozessschritt	Datenkategorien	
Zahlung auslösen	Kundendaten Zahlungsauftragsdaten	
Berater/Kunden informieren	Beraterauftrag/-instruktionen Produktdaten, Kontaktinformationen, -instruktionen Angebotsdaten, Produkt-/Zinsdaten	

Tabelle 1: Datenkategorien zum Anwendungsszenario „Internet-Applikation“

5.1.3 Informationssystemarchitektur zum Anwendungsszenario

Nachdem die wesentlichen Datenkategorien identifiziert worden sind, besteht nun die Aufgabe darin, für das gewählte Anwendungsszenario und unter Rückgriff auf den Prozess eine Informationssystemarchitektur zu entwerfen. Als Grundlage für den Entwurf der Informationssystemarchitektur ist exemplarisch die Datenkategorie *Kundenstammdaten* ausgewählt worden. Die Entscheidung für diese Auswahl kann damit begründet werden, dass gerade Kundendaten bei der Einführung neuer Geschäftsmodelle eine zentrale Rolle übernehmen. Es geht um den geschäftswirksamen Einsatz und Erfassung der Kundendaten. Oft wird die Einführung neuer Geschäftsmodelle auch damit begründet, die Potenziale einer Kundengesamtsicht nutzen zu wollen.

Um die Komplexität der Darstellung zu vermindern, werden Datenflüsse und Datenrückflüsse nachfolgend in zwei separaten Sichten auf die Informationssystemarchitektur dargestellt. In der in Abbildung 9 dargestellten Architektur stehen zunächst die Datenflüsse im Mittelpunkt der Betrachtungen. Die Nummerierungen dienen zur Strukturierung der verschiedenen Datenflüsse, um diese besser beschreiben zu können.

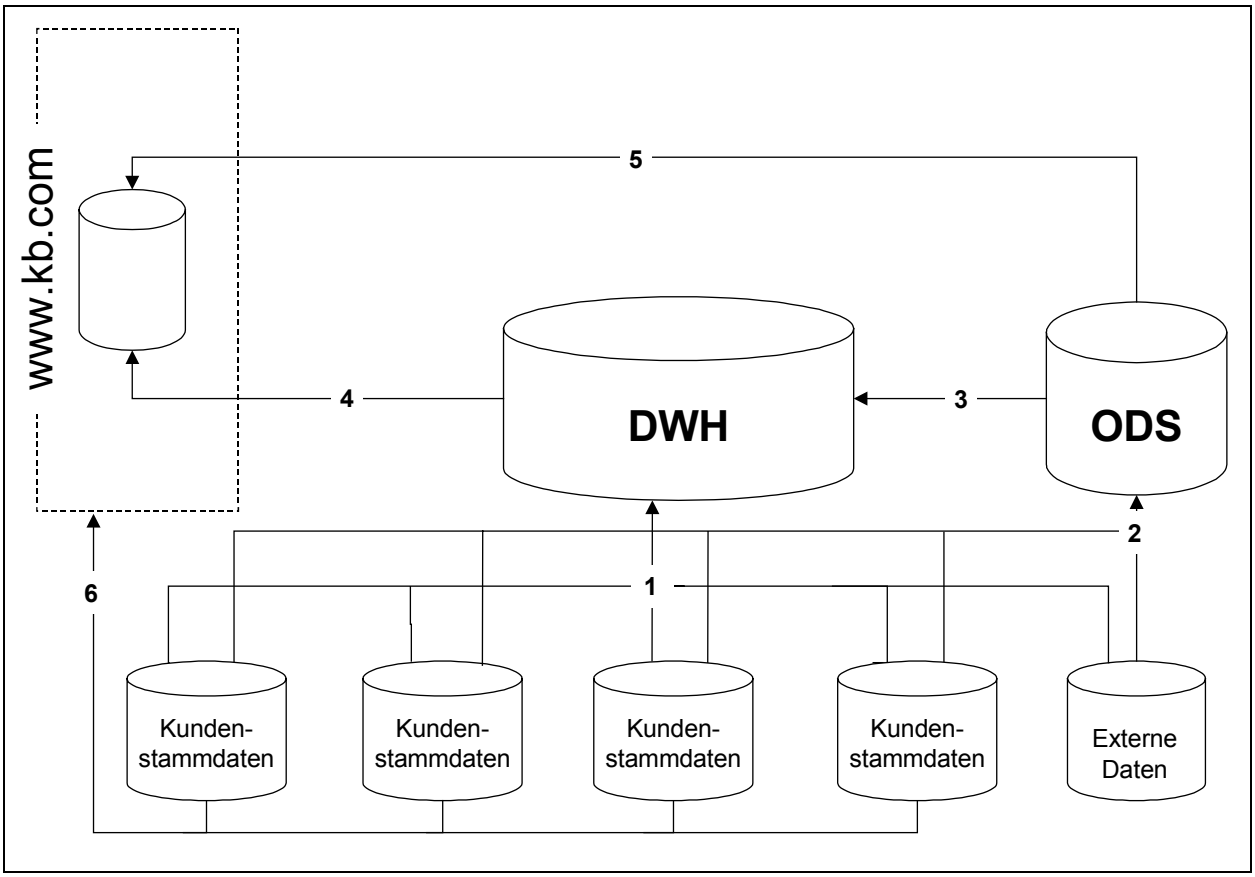


Abbildung 9: Datenflüsse zum Anwendungsszenario „Internet-Applikation“

Im Rahmen der Diskussion über die Konstruktion einer für dieses Anwendungsszenario geeigneten Informationssystemarchitektur ist schnell deutlich geworden, dass für die Abwicklung des Geschäftsvorfalls, die Einführung eines Operational Data Stores neben dem Data Warehouse notwendig ist. Die Gründe für diese Entscheidung werden im Anschluss an die Beschreibung der entwickelten Architektur vorgestellt und beantworten zugleich die zu Beginn dieses Kapitels formulierten Fragen. Vorausgesetzt wird, dass der ODS keine zusätzliche Applikationslogik bereitstellt, sondern der operativen Entkoppelung dient und ausschliesslich im betrachteten Anwendungsbereich Datenbereitstellungsfunktionen übernimmt. Es gilt jedoch zu beachten, dass bei der Erarbeitung des Lösungsansatzes ausschliesslich Datenflüsse bzw. Datenrückflüsse untersucht worden sind, die für den betrachteten Anwendungsfall sinnvoll und relevant sind. Es sind beispielsweise potenzielle zirkuläre Datenflüsse innerhalb des Application Servers oder Datenrückflüsse zum Data Warehouse ausgeschlossen worden. Die Einführung eines Application Servers verfolgt hier das Ziel, anwenderseitig das Antwortzeitverhalten der Internet-Applikation zu verbessern. Zu diesem Zweck nimmt der Application Server eine Datenzwischenspeicherung bei der Abwicklung des Geschäftsvorfalls vor. Darüberhinaus zeichnet er ggf. Protokolldateien auf, die bspw. für das Web-Log-Mining nützlich sind. Datenrückflüsse in das

Data Warehouse wurden ausgeschlossen, da definitionsgemäss für das Data Warehouse ausschliesslich ein lesender Zugriff zulässig ist.

Nachfolgende Tabelle listet die identifizierten Datenflusskategorie unter Angabe von Quell- und Zielsystem auf. Darüber hinaus werden jeder Datenflusskategorie ihre Vor- und Nachteile für die Überführung bzw. Transformation der Daten zugeordnet, die in der Diskussion mit Architektur-Experten der Partnerunternehmen ermittelt wurden. Dadurch werden die getroffenen Entwurfsentscheidungen nachvollziehbar dokumentiert. Die Nummerierung korrespondiert mit derjenigen in Abbildung 9.

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
1	Operative Systeme	Data Warehouse	• Integrierte Daten • Grundlage für analytische Systeme • Historisierung • Hohe Datenqualität	• Zeitverzug • Mangelnde Aktualität
2	Operative Systeme	Operational Data Store	• Detaillierte Daten • Aktuell, zeitnah	• Qualitätsanspruch geringer
3	Operational Data Store	Data Warehouse	• Daten sind bereits im Data Warehouse • Detaillierte Daten	• Abgleich aufwendig • Historisierung nicht aufrechtzuerhalten • Schlechtere Datenqualität
4	Data Warehouse	Internet-Applikation	• Integrierte Daten • Notwendig, wenn Daten nicht im ODS verfügbar sind	• Geringe Aktualitä • Performance nicht ausreichend • Historisierung • Real-Time nicht möglich • Hohe Komplexität

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
5	Operational Data Store	Internet-Applikation	• Detaillierte Daten • Aktuell, Zeitnah • Anwendungsbez. integriert • Grössere Performance als Data Warehouse	• Nicht vollständig integriert, da nur Teildatenbestände anwendungsfallbezogen geladen werden
6	Operative Systeme	Internet-Applikation	• Detaillierte Daten • Aktuell, Zeitnah • Erforderlich, wenn ODS zur Applikationssteuerung eingesetzt wird • Notwendig, wenn Daten weder im ODS noch im Data Warehouse vorhanden sind	• Hohe Schnittstellenkomplexität • Nicht integriert • Zusatzbelastung

Tabelle 2: Bewertung der Datenflüsse

Beispielsweise fliesst Datenfluss 1 von den operativen Systemen zum DWH, das diese Daten in integrierter, historisierter Form und in hoher Qualität zur Verfügung stellt. Das DWH bildet somit die Grundlage für die analytisch-dispositiven Systeme. Nachteilig wirken sich dabei der Zeitverzug bei der Bereitstellung der Daten und die damit verbundene Aktualität der Daten aus.

Die Unterscheidung zwischen Datenflüssen und Datenrückflüssen resultiert aus der unidirektionalen, vertikalen Datenintegration im klassischen Data-Warehouse-Konzept. Dort fliessen die Daten aus den operativen Datenquellen in das DWH und von dort weiter in Data Marts bzw. analytisch-dispositive Applikationen. Ein Rückfluss der Daten in die entgegengesetze Richtung ist nicht vorgesehen. Genau dieser Rückfluss gewinnt aber im Sinne eines geschlossenen Datenfluss-Kreislaufs immer mehr an Bedeutung. Analog zur Bewertung der Datenflüsse wird deshalb in diesem Fallbeispiel eine erweiterte Konzeption entwickelt, die Datenrückflüsse zwischen Architekturkomponenten explizit berücksichtigt. In Abbildung 10 sind die für das Anwendungsszenario identifizierten Datenrückflüsse in einer eigenen Sicht auf die Architektur aufgezeigt.

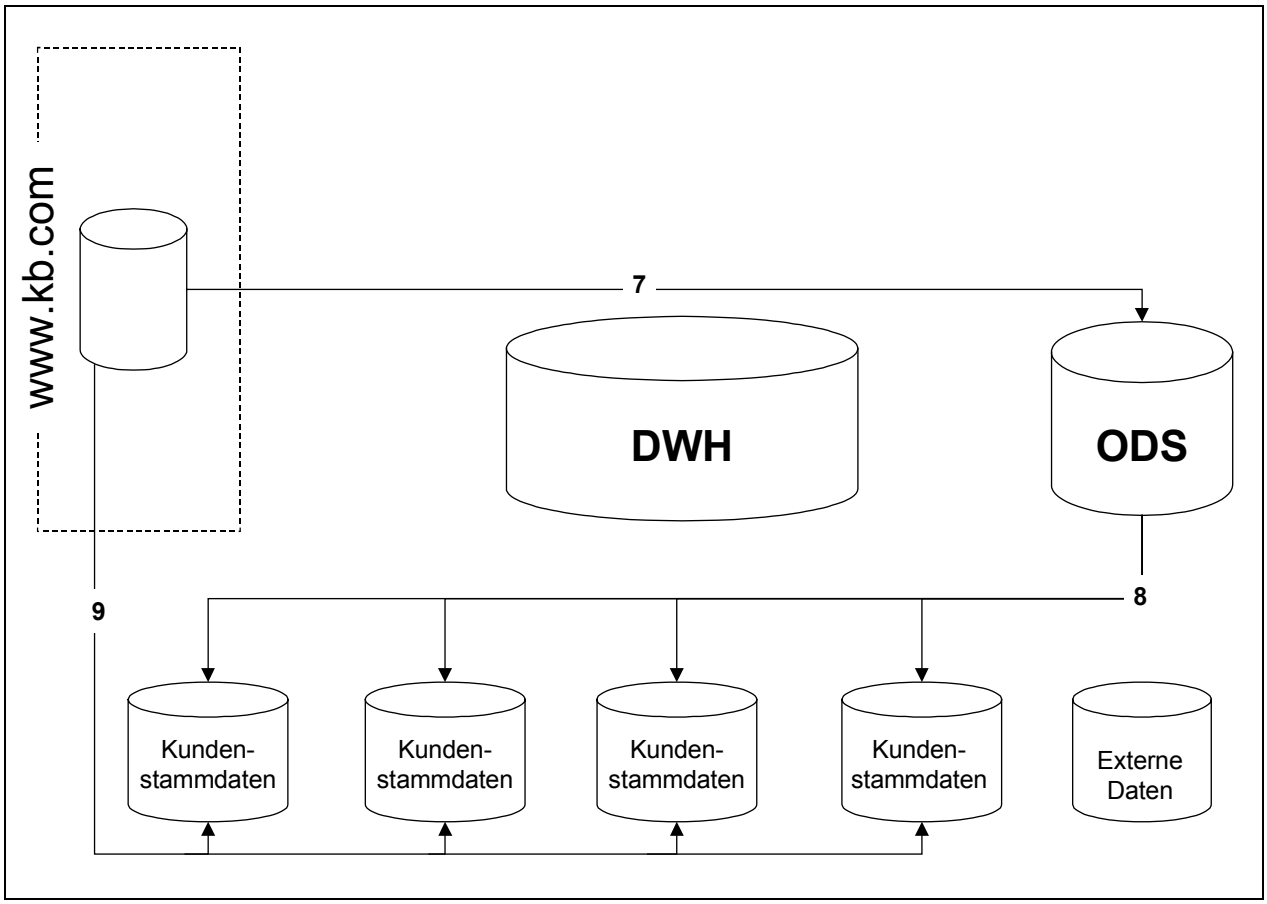


Abbildung 10: Datenrückflüsse zum Anwendungsszenario „Internet-Applikation“

Auch für die Datenrückflüsse werden nachfolgend die Vor- und Nachteile tabellarisch einander gegenüber gestellt. So ist etwa Datenfluss 7 von der Internet-Applikation zum ODS notwendig, um Modifikationen sofort in die Informationslogistik einzuspeisen. Die Entkopplung der Internet-Applikationen von den operativen Systemen über den ODS bewirkt eine Verringerung der Belastung der operativen Systeme. Weiterhin kann so die Anzahl der Schnittstellen verringert werden, da der ODS mehrere horizontale Applikationen bedienen kann. Nachteile dieses Datenflusses konnten in der Diskussion nicht festgestellt werden.

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
7	Internet-Applikation	Operational Data Store	• Modifikationen werden sofort sichtbar • Operative Entkoppelung • Geringere Schnittstellenkomplexität	

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
8	Operational Data Store	Operative Systeme	• Folgt aus Datenrückflüss 7 • Ablösung bilateraler Schnittstellen	• Hohe Schnittstellen-komplexität • ETL-Logik der Rückführung in die operativen Systeme sehr komplex
9	Internet-Applikation	Operative Systeme	• Ständige Datenaktualität • Erforderlich, wenn Daten weder für ODS noch Data Warehouse be-stimmt sind	• Hohe Schnittstellen-komplexität • ETL-Logik der Rückführung in die operativen Systeme sehr komplex • Zusatzbelastung

Tabelle 3: Bewertung der Datenrückflüsse

Die zuvor durchgeführte (qualitative) Bewertung möglicher Datenflüsse und Datenrückflüsse bildet die Grundlage für die Erarbeitung eines (optimalen) Datenfluss-Szenarios für das gewählte Anwendungsbeispiel. Folgender Lösungsansatz, in dem die beiden Sichten auf Datenflüsse und Datenrückflüsse wieder zusammengeführt wurden, ist unter Berücksichtigung der vorangegangen Argumentation denkbar:

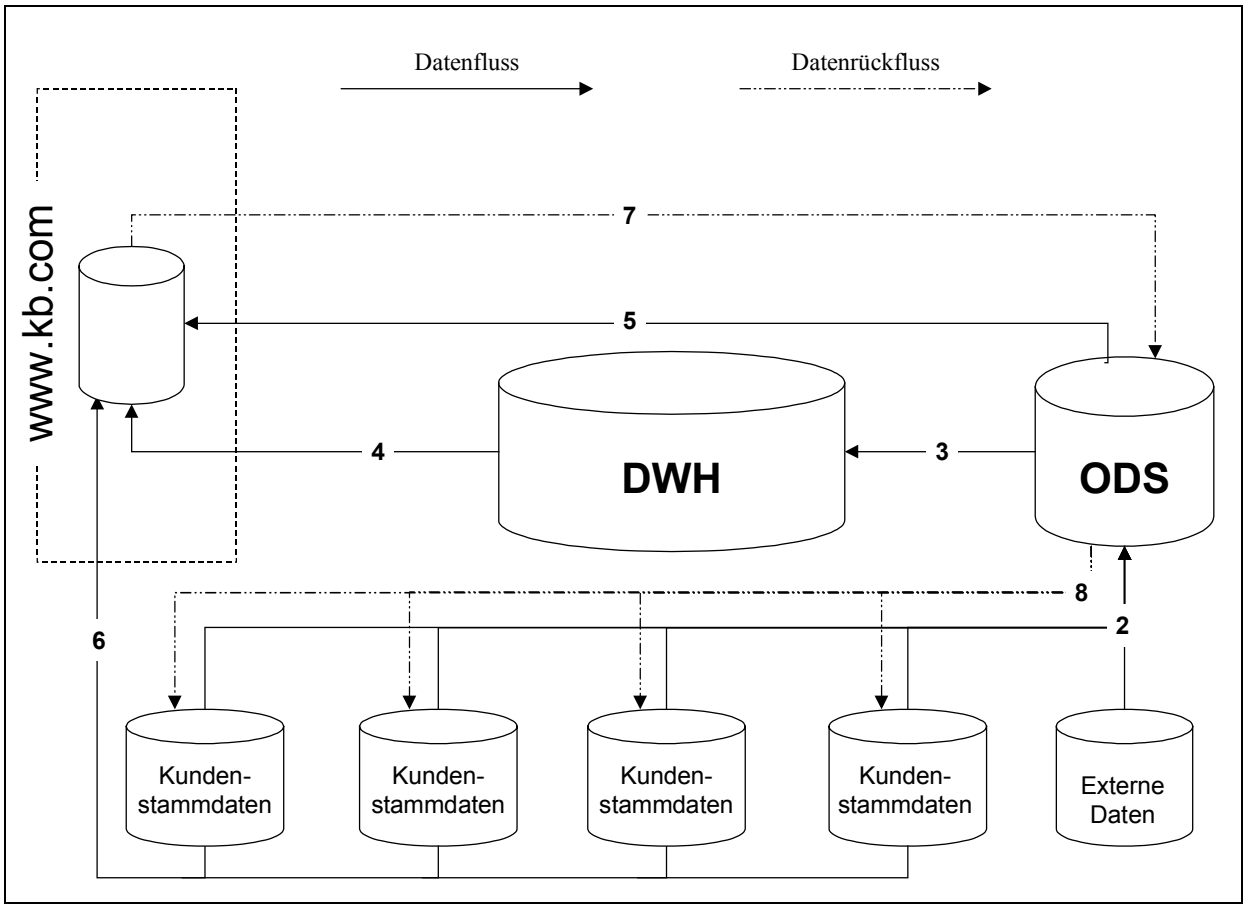


Abbildung 11: Informationssystemarchitektur zum Anwendungszenario „Internet-Applikation“

Das in Abbildung 11 gezeigte Datenfluss-Szenario ist unter der Prämisse zu betrachten, dass es sich hier um eine anwendungsfallspezifische Architektur handelt, bei der es darum geht, die mit der Einführung des neuen Geschäftsmodells erforderliche horizontale Internet-Applikation mit den anwendungsspezifischen Daten zu versorgen. Die in dieser Abbildung nicht angezeigte Verbindung zwischen operativen Systemen und dem Data Warehouse wird implizit unterstellt. Diese Verbindung existiert, jedoch mit der Zielsetzung, dass mit dem Data Warehouse vor allem analytisch-dispositive Applikationen unterstützt werden.

Datenfluss 2 aus Abbildung 11 repräsentiert entsprechende ETL-Prozesse, die der Bereitstellung von anwendungsspezifischen Daten im ODS für die horizontale Applikation dienen. Die mit dem ODS verfolgten Ziele, eine aktuelle, integrierte Datenbasis und insbesonders zeitnahe Datenbereitstellung für die operative Nutzung zu ermöglichen, kann somit sichergestellt werden. Durch den ODS werden sonst notwendige separate Schnittstellen zu den operativen Systemen reduziert. Es existiert eine einzige Schnittstelle zur Applikation (Datenfluss 5). Der ODS entkoppelt die operativen Systeme von horizontalen Applikationen. In Verbindung mit dem Application Server ist eine Trennung von Applikationslogik und Datenhaltung möglich.

Der Datenfluss 3 übernimmt eine Doppelfunktion. Er dient sowohl der Überführung von nicht im Data Warehouse enthaltenen Daten aus dem ODS als auch der Rückführung, um möglicherweise analytisch-dispositiven Applikationen Daten aus dem ODS verfügbar zu machen. Die Datenflüsse 4 und 6 sind notwendig, wenn es um die Bereitstellung von Daten geht, die nicht im ODS verfügbar sind.

Die durch den Datenrückfluss 7 angezeigte Verbindung zum ODS ist notwendig, um Datenmodifikationen registrieren und erfassen zu können. Der Datenrückfluss 8 ist eine Folge des Datenrückflusses 7, da es schliesslich darum geht, die modifizierten Datenbestände des ODS mit den operativen System abzugleichen. Zugleich stellt jedoch dieser Datenfluss eine kritische Komponente während des Betriebs dar. Die Komplexität der erforderlichen Logik für die Rückführung der Daten in die operativen Systeme ist nicht zu unterschätzen.

Entwurfsentscheidung zur Kombination Data Warehouse und ODS	
Vorteile	**Nachteile**
1) Einfachere und somit „schlankere“ (horizontale) Applikation durch Schnittstellenminimierung (z. B. Ablösung von n:m File-Transfers)	1) Erhöhter Aufwand der Datenpflege durch ein weiteres Datenhaltungssystem
2) Transparenz und Trennung von Daten und Applikationslogik	2) Gefahr von Redundanzen und Inkonsistenzen
3) Optimierung für operative Nutzung	3) Komplexität der Gesamtarchitektur steigt und somit wird der produktive Betrieb von Data Warehouse und Operational Data Store erschwert
4) Hohe Aktualität der Daten in Verbindung mit den Vorteilen eines integrierten, konsolidierten Datenbestandes	4) Zusätzliche Kosten für die Entwicklung und Betrieb des ODS
5) Lastverteilung durch Entkoppelung	
6) Schnellere Realisierbarkeit durch Verringerung des Aufwands für Schnittstellen	

Tabelle 4: Entwurfsentscheidung Data Warehouse/ODS

Zusammenfassend lässt sich feststellen, dass sich für die Bereitstellung des Operational Data Stores neben dem Data Warehouse sowohl Vor- als auch Nachteile anführen lassen. Im betrach-

teten Fallbeispiel überwiegen die Vorteile einer kombinierten Architektur, wie der Gegenüberstellung in Tabelle 4 entnommen werden kann. Insbesondere die Erreichung einer hohen Aktualität der Daten in Verbindung mit den Vorteilen eines integrierten, konsolidierten Datenbestandes war in diesem Fallbeispiel ausschlaggebend. Hierfür wurden die Nachteile in Form von zusätzlichem Aufwand und steigender Komplexität in Kauf genommen, obwohl damit das Risiko für Redundanz und Inkonsistenz steigt. Für abweichende Anwendungsbereiche müssen beim Entwurf der Architektur die jeweiligen Anforderungen berücksichtigt werden, was zu anderen Ergebnissen führen kann.

5.2 Fallbeispiel 2: Adressänderung über ein Call Center

5.2.1 Szenario und Prozessmodell

Das zweite Anwendungsszenario beschreibt die Abwicklung von Geschäftsvorfällen über ein Call Center aus der Sicht eines grösseren Versicherungsunternehmens. Sogenannte Call Center werden mittlerweile von vielen Firmen für die telefonische Betreuung von Kunden eingesetzt. Sie ermöglichen die massenweise Abfertigung von Kundenanrufen durch eine spezialisierte Aufbau- und Ablauforganisation sowie ein hohes Mass an Unterstützung durch Informationssysteme. Dabei können neben Geschäftsvorfällen, bei denen ein Produkt oder eine Leistung verkauft werden, auch solche mit rein informatorischem Charakter anfallen, wie bspw. eine Adressänderung, die vom Kunden bekannt gegeben wird.

Von Mitarbeitern des IWI-HSG und Mitarbeitern mehrerer Finanzdienstleistungsgesellschaften wurde gemeinsam ein exemplarischer Prozess entwickelt, der einen typischen Geschäftsvorfall abbildet, für dessen Abwicklung Daten aus unterschiedlichen Applikationen benötigt werden. Der Prozess wird dabei aus Sicht des Mitarbeiters im Call Center beschrieben, der eingehende Kundenanrufe entgegennimmt. Den genauen Ablauf zeigt folgendes Prozessmodell:

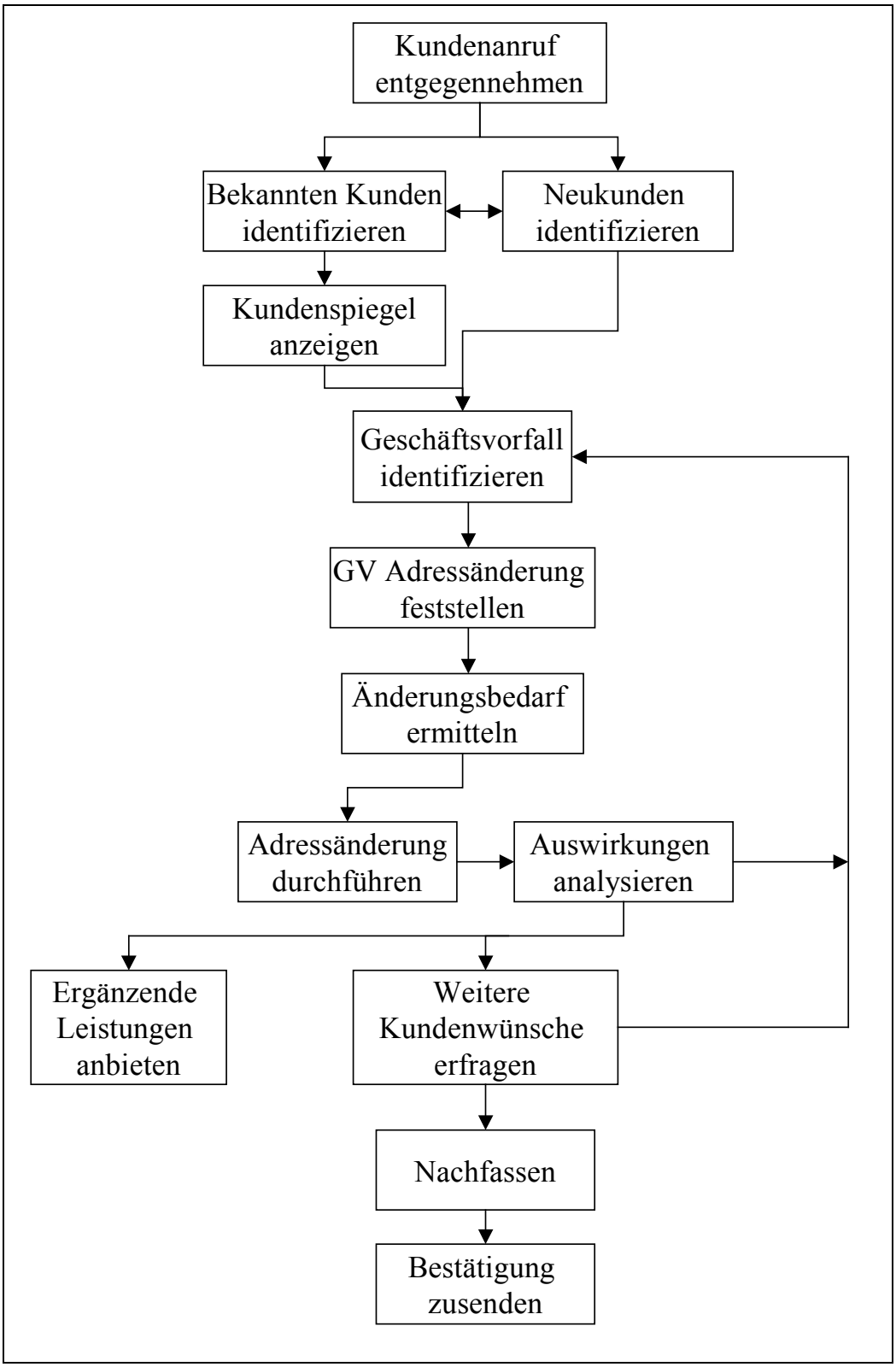


Abbildung 12: Prozessmodell zum Anwendungsszenario „Call Center“

Der Prozess startet somit mit dem Entgegennehmen des Kundenanrufs. Nun wird der Kunde als bekannt oder als Neukunde identifiziert. Handelt es sich um einen bekannten Kunden, ist ein sog. Kundenspiegel für den Kunden verfügbar, der über abgeschlossene Verträge, gemeldete Schä-

den, bisherige Kontakte etc. Auskunft gibt und es dem Call Center Mitarbeiter ermöglicht diesbezügliche Fragen des Kunden zu beantworten. Im Beispielprozess ist der Kunde bereits bekannt und möchte eine Änderung seiner Adresse bekanntgeben. Der Call Center Mitarbeiter nimmt die Änderung vom Kunden entgegen und ermittelt welcher Änderungsbedarf sich daraus in welchen Systemen ergibt. Nachdem die Adressänderung durchgeführt ist, sind noch die daraus resultierenden Auswirkungen zu analysieren. Auf Basis dieser Analyse können dann dem Kunden weitere ergänzende Leistungen angeboten werden. Lässt sich beispielsweise ein Ehepaar scheiden und gibt die gemeinsame Wohnung auf, können daraus unterschiedliche versicherungstechnische Konsequenzen entstehen. Hat der Kunde keine weiteren Wünsche mehr, fragt der Mitarbeiter den Kunden nach dessen Zufriedenheit mit dem gebotenen Service und nimmt gegebenenfalls Kritik und Kommentare entgegen. Nach dem Ende des Telefongesprächs erhält der Kunde per Post eine schriftliche Bestätigung der Abwicklung des Geschäftsvorfalls.

5.2.2 Datenkategorien

Um den Kunden bei der Bearbeitung seines Geschäftsvorfalls kompetent betreuen zu können, benötigt der Call-Center-Mitarbeiter verzögerungsfreien Zugriff auf aktuelle Daten des Kunden und seiner Verträge. Beispielsweise kann das Vertrauen des Kunden zum Unternehmen erheblich gestört werden, wenn der Gesprächspartner nicht über Informationen zu anderen, parallel laufenden Geschäftsvorfällen verfügt. Tabelle 11 zeigt die für die Ausführung des Prozesses benötigten Datenkategorien im Überblick und ordnet sie den Prozessschritten zu, bei denen sie betroffen sind.

Prozessschritt	Datenkategorien
Kundenanruf entgegennehmen	Kundenstammdaten (Telefonnummer), Kontakthistorie
Bekannten Kunden identifizieren	Kundenstammdaten
Kundenspiegel anzeigen	Vertragsdaten, Schadensdaten, Kontakthistorie, laufende Geschäftsvorfälle
Geschäftsvorfall identifizieren	Kundenwunsch, Geschäftsvorfallsverzeichnis, Workflow-Status, Kontakthistorie

Prozessschritt	Datenkategorien
Änderungsbedarf ermitteln	Kundenwunsch, Kundenstammdaten, Geschäftsvorfall-Leitfaden
Adressänderung durchführen	Kundenstammdaten, Kundenzusatzdaten
Auswirkungen analysieren	Geschäftsvorfall-Leitfaden, Kundenstammdaten, Kundenspiegel, Verkaufsdaten
Weitere Kundenwünsche erfragen	Kundenwunsch
Nachfassen	Kundenzufriedenheit
Bestätigung zusenden	Kundenstammdaten

Tabelle 5: Datenkategorien zum Anwendungsszenario „Call Center“

Ausgehend von den genannten Datenkategorien können die für die Verarbeitung dieser Daten vorhandenen Applikationen identifiziert werden. Dieser Applikationen bilden zusammen mit dem Data-Warehouse-System die Komponenten, aus denen im nächsten Schritt eine passende Architektur entwickelt wird.

5.2.3 Informationssystemarchitektur zum Anwendungsszenario

Für den Entwurf der Informationsarchitektur zum Call Center Szenario wurde von der Arbeitsgruppe eine Vorgehensweise gewählt, die sich von der des Internet-Szenarios unterscheidet. Ausgangspunkt waren neben den identifizierten Datenkategorien die zentralen Fragestellungen:

1. Welche kritischen Anforderungen führen zur gewählten Lösung?

2. Welches sind die Hauptproblemfelder der gewählten Lösung?

Zur Beantwortung dieser Fragen wurden zunächst die Anforderungen der Call-Center-Applikation definiert, sowie Charakteristika erhoben, die den Informationssystemen zueigen sind, welche die Call Center Anwendung mit Daten versorgen. Hierbei wurde die Annahme ge-

troffen, dass ein Data Warehouse existiert, dass die Daten der operativen Systeme integriert. Folgende Tabelle zeigt dies im Überblick:

Anforderungen	**Charakteristika**	
Call Center Applikation	**Data Warehouse**	**Operative Systeme**
• Verfügbarkeit 7 x 24 h • Änderungen möglichst sofort sichtbar • Feedback ans Marketing	• wöchentliche Aktualität • Einzelsatzdaten • 16h/Tag verfügbar • kein Schreibzugriff • nicht alle Daten verfügbar	• 7-21 Uhr verfügbar • mehrere Partner-Systeme • Mehrere Vertragssysteme • Heterogene Umgebung

Tabelle 6: Anforderungen und Charakteristika der Informationssystemarchitektur

Aus der Tabelle wird ersichtlich, dass besonders die Aktualität und die Verfügbarkeit der Daten als kritische Faktoren für ein reibungsloses Zusammenspiel der beteiligten Applikationen aufgedeckt wurden. Um dem Kunden über das Call-Center einen echten Mehrwert in Form eines Rund-um-die-Uhr-Services zur Verfügung stellen zu können, muss die Call-Center-Applikation 7 x 24h verfügbar sein und dabei Zugriff auf die aktuellsten Daten haben. Allerdings wurde sowohl für das Data Warehouse als auch für die operativen Systemen von geringerer Verfügbarkeit ausgegangen, da typischerweise in der Nacht Sicherungs- und Batch-Prozesse ausgeführt werden, wodurch die betroffenen Applikationen während dieser Zeit nicht zur Verfügung stehen. Um die Anforderungen an Aktualität und Verfügbarkeit besser zu verstehen, wurden im nächsten Schritt Datenflüsse innerhalb der Architektur identifiziert, die von der Call Center Applikation benötigt werden, um die gestellten Anforderungen erfüllen zu können. Dabei wurde unterschieden zwischen lesenden Datenflüssen, die von den versorgenden Systemen zur Call-Center-Applikation verlaufen, und schreibenden Datenrückflüssen, die von der Call-Center-Anwendung zurück in die Herkunftssysteme ablaufen. Tabelle 7 zeigt die von den Datenflüssen betroffenen Datenkategorien, die Zugriffsart sowie die erforderliche Aktualität der Daten:

Datenkategorie	**read**	**write**	**Aktualität**
Kundenstammdaten und Adressen	x	x	realtime oder tagesaktuell mit Änderungskennzeichen

Datenkategorie	read	write	Aktualität
Kontakthistorie	x	x	realtime
Vertragsgrunddaten	x		realtime
Vertragsdetaildaten	x		realtime
Geschäftsvorfälle	x	x	realtime
Schadensgrund	x		realtime
Kundenwunsch	x	x	realtime

Tabelle 7: Datenkategorien, Zugriffsarten und Aktualitätsanforderungen

In der Diskussion erwies sich die geforderte 7 x 24 h-Verfügbarkeit in Kombination mit der Notwendigkeit, dem Kunden jederzeit am Telefon aktuelle Daten liefern zu können, als die am schwierigsten zu erfüllende Anforderung. Um diese Anforderungen erfüllen zu können, wurde eine kombinierte Architektur entworfen, die einerseits integrierte Daten aus dem Data Warehouse bezieht und andererseits durch die Kombination eines Pufferspeichers mit asynchronen Messaging-Mechanismen die Down-Zeiten von operativen Systemen und Data Warehouse überbrückt:

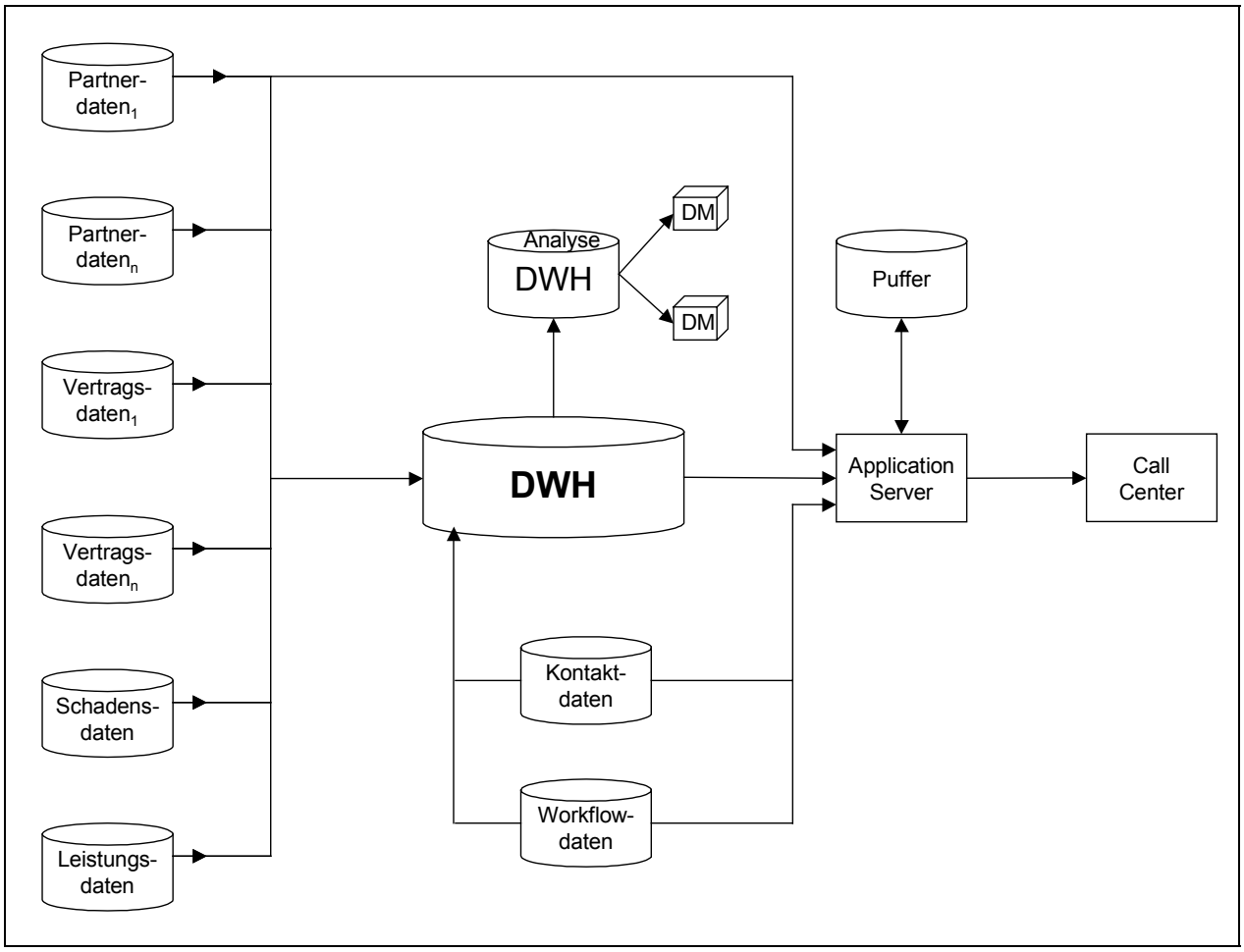


Abbildung 13: Datenflüsse zum Anwendungsszenario „Call Center“

Diese Architektur bezieht ihre Stärken aus dem Zusammenspiel von Datenintegration über das Data Warehouse und Funktionsintegration über einen zentralen Application Server. Der Application Server ist die Software-Komponente, welche die Einhaltung der Geschäftslogik über die heterogenen operativen Systeme hinweg sicherstellt und damit zentrale Bedeutung für die Integration hat. Die Komponenten und Datenflüsse der Gesamtarchitektur sind in Abbildung 13 dargestellt. Um die Verfügbarkeit des Data Warehouse zu maximieren, werden die Daten im Data Warehouse in ein zweites DWH gespiegelt, dass als Datenquelle für die analytischen Anwendungen dient. Die Aktualität der Daten im zweiten DWH muss darüberhinaus nicht so hoch sein, wie im Basis-DWH. Je nach Anforderungen der analytischen Applikationen kann es ausreichend sein, die Daten im Analyse-DWH nur wöchentlich zu aktualisieren.

Die Überlegungen zu den Datenrückflüssen innerhalb der Architektur sind geprägt von der Prämisse, dass die Daten im DWH nicht-volatil sind und damit der Weg über das DWH zurück in die operativen Systeme versperrt ist. Abbildung 14 zeigt die in der Architektur verlaufenden Datenrückflüsse.

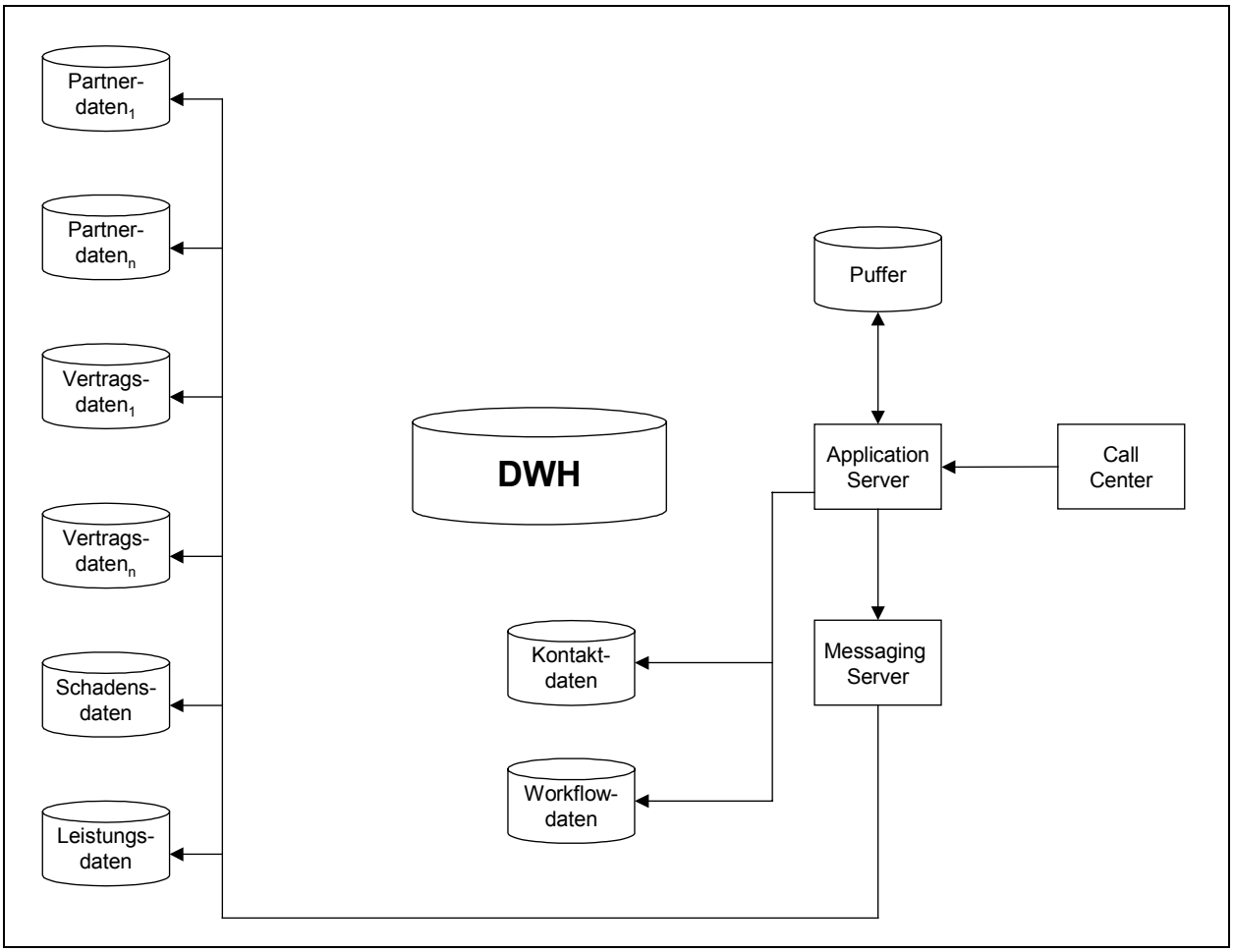


Abbildung 14: Datenrückflüsse zum Anwendungsszenario „Call Center“

Folgerichtig führen in dieser Architektursicht keine Datenrückflüsse zurück zum Data Warehouse. Es musste also eine Alternative gefunden werden, die letztlich über den asynchronen Nachrichtenversand, der von einem Messaging-Server koordiniert wird, realisiert wurde. Diese Lösung orientiert sich am Konzept der Messaging-basierten Kopplung von operativen Systemen aus Kapitel 4. Die Auflösung der semantischen Beziehungen der Daten erfolgt über die im Application Server abgelegte Logik. Die eigentliche Verteilung der Daten in die sie verarbeitenden operativen Systeme übernimmt dann der Messaging-Server. Mit dieser kombinierten Lösung gelingt es einen geschlossenen Datenfluss-Kreislauf zu gestalten, der Voraussetzung für die globale Konsistenz des Systems ist.

Bei einer Implementierung des erarbeiteten Lösungsansatzes gilt es, besonderes Augenmerk auf einige Hauptproblemfelder zu richten. Folgende Probleme wurden als erfolgskritisch identifiziert:

1. Bewältigung der Komplexität durch eine Vielzahl von Datenflüssen und Systemkomponenten
2. Aufwand für Aufbau und Betrieb des Application Servers
3. Implementierung und Pflege der Geschäftslogik
4. Integriertes Metadatenmanagement
5. Unterschiedliche Verfügbarkeit der Zuliefer-Systeme
6. Potenzielle Inkonsistenzen durch Ausfall einzelner Komponenten
7. Wartbarkeit

Bei näherer Betrachtung wird deutlich, dass es sich bei den genannten Punkten weniger um spezifische Probleme der hier vorgestellten Architektur handelt, sondern vielmehr um schwierige Aufgabenstellungen, die in jeder komplexen Informationsarchitektur bewältigt werden müssen. Unterstützung bieten hier Methoden und Modelle der Software- und Systementwicklung.

Im Gegensatz zu Fallbeispiel 1 kommt die hier entworfene Architektur ohne einen expliziten ODS aus. Die am Entwurf beteiligten Experten waren einstimmig der Meinung, dass die zusätzliche Funktionalität, die ein ODS bietet, durch die Kombination von Data Warehouse und Application Server problemlos bereitgestellt werden kann. Ausschlaggebend für diese Entscheidung, war das Vermeiden möglicher Nachteile durch die Einführung eines ODS zusätzlich zum Data Warehouse wie sie in Fallbeispiel 1 beschrieben werden (vgl. Tabelle 4).

5.3 Fallbeispiel 3: Wertpapierhandel im Internet

5.3.1 Szenario und Prozessmodell

Im Mittelpunkt dieses Anwendungsszenarios steht der Wertpapierhandel über das Internet. Nachdem im Zuge des Börsenbooms in den letzten Jahren immer mehr Kleinanleger in Aktien investieren, sind viele Finanzinstitute dazu übergegangen, den Handel mit Wertpapieren auch über den Kanal Internet anzubieten. Anhand der Spezifikation eines typischen Prozesses lassen sich die benötigten Daten für die einzelnen Prozess-Schritte beim Abwickeln von Transaktionen

über ein Online-Wertpapierhandelssystem ermitteln. Die einzelnen Prozess-Schritte lassen sich wie folgt grafisch darstellen:

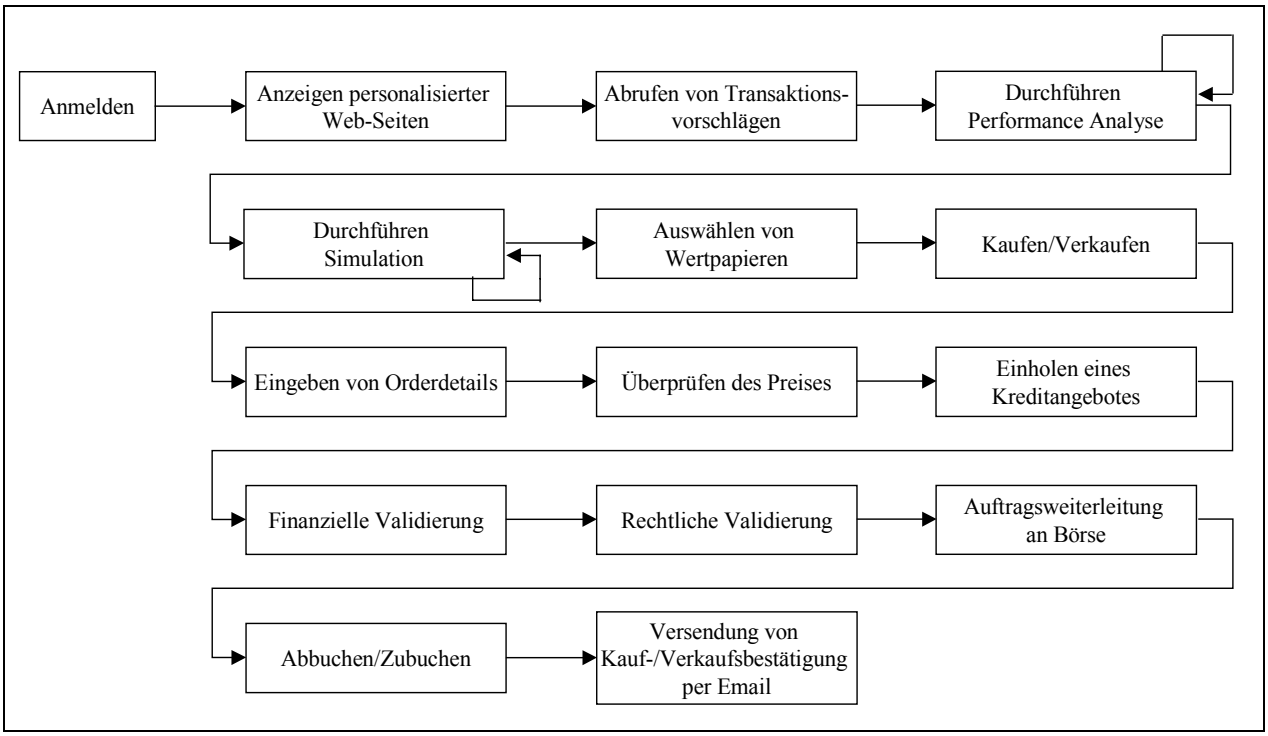


Abbildung 15: Prozessmodell zum Anwendungsszenario „Wertpapierhandel“

Betrachtet wird die Interaktion eines Kunden mit dem System, in der Absicht, Wertpapiere zu kaufen oder zu verkaufen. Dazu muss sich der Kunde zunächst beim System anmelden. Verläuft die Anmeldung erfolgreich, wird dem Kunden ein personalisierter Bereich angezeigt. Hier findet er Informationen, die auf seine persönlichen Interessen abgestimmt sind sowie Angaben zu seinem Depot. Darüber hinaus werden Funktionen zur Verwaltung, Analyse und Kontrolle des Depots angeboten, Kaufempfehlungen stehen zum Abruf bereit und es können Nachrichten abgerufen und versandt werden. Im hier dargestellten Prozess informiert sich der Kunde zunächst über vorliegende Kauf- und Verkaufempfehlungen. Anhand von Simulation und Analyse wählt der Kunde Wertpapiere zum Kauf aus. Um den Kauf abzuwickeln, muss der Benutzer nun Orderdetails wie die Kennung und Menge der gewünschten Wertpapiere eingeben. Zusätzlich werden vom System Adress-Angaben aus den Kundenstammdaten ergänzt. Stellt der Kunde im Verlauf der Order fest, dass seine Liquidität für den Kauf nicht ausreicht, hat er die Möglichkeit direkt aus der Online-Anwendung ein Kreditangebot des Finanzinstituts einzuholen. Ist die Finanzierung seitens des Kunden geklärt, kann er schliesslich seine Order abschicken. Vom Handelssystem wird im Anschluss erneut die finanzielle und rechtliche Validität der Transaktion überprüft. Fällt das Ergebnis positiv aus, wird die Order elektronisch an die Börse weitergeleitet.

5.3.2 Datenkategorien

Analog zum Aufbau der beiden vorangegangenen Fallbeispiele zeigt Tabelle 8 die vom jeweiligen Prozesschritt betroffenen Datenkategorien. Im Fallbeispiel „Call Center“ kommuniziert der Kunde mit einem Mitarbeiter des Unternehmens, der wiederum mit der Call-Center-Applikation interagiert. Dadurch wird die Komplexität und das Verhalten des Systems vor dem Kunden verborgen. Dagegen wird der Kunde im vorliegenden Fallbeispiel direkt mit der Applikation konfrontiert. Sowohl Benutzerfreundlichkeit als auch Verhalten der Applikation in Hinblick auf Antwortgeschwindigkeit und Qualität der Daten haben entscheidenden Einfluss auf die Akzeptanz des Angebots eines direkten Wertpapierhandels über das Internet beim Kunden. Wird das Vertrauen des Kunden in die Applikation durch Systemfehler, fehlerhafte Daten, Überlastung o. ä. gestört, kann dies eine Abwanderung zu einem konkurrierenden Anbieter zur Folge haben.

Prozessschritt	**Datenkategorien**
Anmelden	Kundenzugangsdaten
Anzeigen personalisierter Web-Seiten	Kundenstammdaten, Kundenprofildaten, Analysedaten, Depot-/Kontodaten, Kursdaten, Auftragsdaten, Produktdaten
Abrufen von Transaktionsvorschlägen	Kundenprofildaten, Analysedaten, Produktdaten
Durchführen Performance Analyse	Kundenprofildaten, Analysedaten, Produktdaten
Durchführen Simulation	Simulationsparameter, Auftragsdaten, Produktdaten, Prognosen
Auswählen von Wertpapieren	Kundenzugangsdaten, Produktdaten, Auswahl
Kaufen/Verkaufen	Kundenzugangsdaten, Kundenstammdaten, TAN
Eingeben von Orderdetails	Produktdaten, Kursdaten

Prozessschritt	Datenkategorien
Überprüfen des Preises	Produktdaten, Preisdaten
Einholen eines Kreditangebotes	Kundenzugangsdaten, Kundenstammdaten, Kreditlimit, Kreditrisiko-Profil, Kreditbedarf, Kreditdetails
Finanzielle Validierung	Bestelldaten, Preisdaten, Kontostand
Rechtliche Validierung	Vertragsdaten, Bestelldaten
Auftragsweiterleitung an Börse	Bestelldaten
Abbuchen/Zubuchen	Bestelldaten, Preisdaten
Bestätigung zusenden	Bestelldaten, Kundenstammdaten

Tabelle 8: Datenkategorien zum Anwendungsszenario „Wertpapierhandel“

Aus der Tabelle wird ersichtlich, dass neben den obligatorischen Kundenstammdaten nahezu alle verfügbaren Daten gebraucht werden, die über den Kunden im Unternehmen existieren. Um dem Kunden die Möglichkeit zu geben, ohne Unterstützung eines Mitarbeiters Transaktionen über das Internet abwickeln zu können, müssen bspw. Daten über Produkte, Verträge, Preise, Konto und Depotstände verfügbar gemacht werden. Dabei genügt es jedoch nicht, die Daten aus den entsprechenden Applikationen zu lesen, sondern bei der Durchführung von Transaktionen fallen auch Schreibzugriffe an. Die Möglichkeit Simulationen und Analysen selbstständig über das Internet durchzuführen, ist eine weitere kritische Anforderung. Im nächsten Abschnitt wird unter Einbeziehung der spezifischen Anforderungen und der existierenden Applikationslandschaft eine Informationsarchitektur zur Umsetzung des Prozessmodels diskutiert.

5.3.3 Informationssystemarchitektur zum Anwendungsszenario

Von den drei hier betrachteten Anwendungsszenarien ist der Wertpapierhandel über das Internet bei weitem das komplexeste Szenario, da es dem Kunden eine weitreichende und komplexe Funktionalität zur Verfügung stellt, ohne dass Mitarbeiter des Unternehmens direkt am Prozess beteiligt sind. Wegen der Anforderung, Transaktionen direkt über das Internet abwickeln zu können und dabei zur Entscheidungsunterstützung Simulations- und Analysefunktionalitäten, auch unter Nutzung externer Daten, zur Verfügung zu stellen, erfordert das Wertpapierhandel-Beispiel die Einbeziehung und Integration einer grossen Zahl von Applikationen und Datenquellen. Durch die vielfältigen Integrationsbeziehungen und die hohe Anzahl von Schnittstellen ergibt sich somit eine besonders komplexe Informationslogistik. Deshalb ist die Darstellung und Diskussion der Architektur zu Zwecken der Komplexitätsreduzierung wiederum in zwei Sichten aufgeteilt. Wie in Fallbeispiel 1 zeigt eine Sicht die Datenflüsse, bei denen lesend auf Datenquellen unterschiedlichster Art zugegriffen wird, während die zweite Sicht diejenigen Datenflüsse behandelt, durch die Daten von der Wertpapierhandel-Applikation in die Quellsysteme zurückgeschrieben werden. Abbildung 16 zeigt zunächst die Informationsarchitektur aus Sichtweise der Datenflüsse hin zur Wertpapierhandel-Applikation:

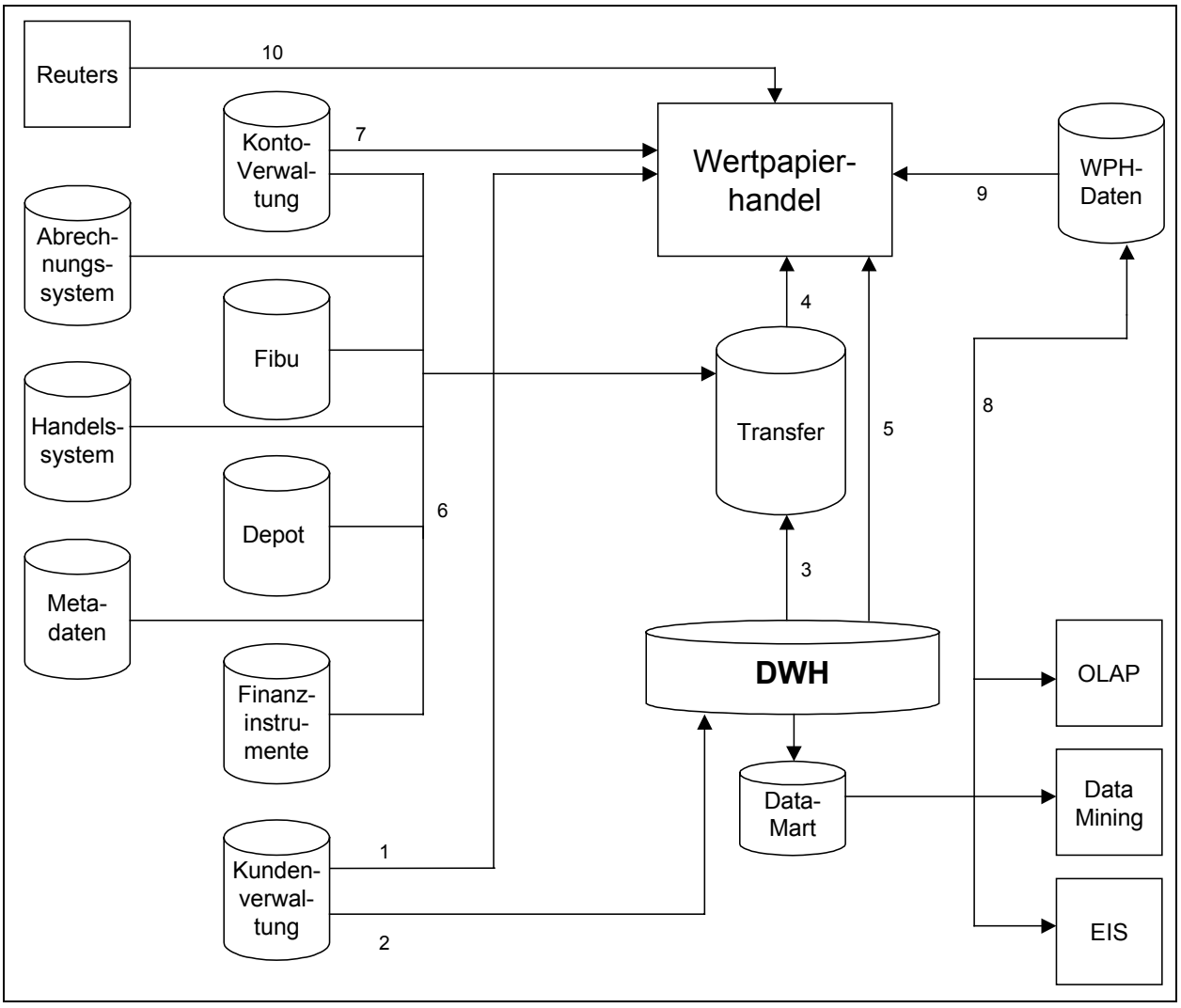


Abbildung 16: Datenflüsse zum Anwendungsszenario „Wertpapierhandel“

Wesentliche Komponente in diesem Architekturentwurf ist die zentrale Transfer-Komponente, die sowohl ODS- als auch Data-Warehouse-Charakteristika aufweist. Bei der Erarbeitung der vorliegenden Lösung wurde ganz bewusst versucht, sich von konzeptionellen Einschränkungen und Begrenzungen zu lösen, die durch starre Definitionen von Integrationskonzepten in der Literatur erwachsen. Bestes Beispiel ist das dogmatische Verbot des Schreibzugriffs auf das Data Warehouse. Obwohl dieses Verbot in vielen Anwendungsszenarien wohlbegründet werden kann, erscheint es aufgrund der spezifischen Anforderungen des Wertpapierhandel-Szenarios sinnvoll, sich zunächst von vorgegebenen Definitionen zu lösen, um die Voraussetzungen für das Entwickeln innovativer Lösungen zu schaffen.

Die Transfer-Komponente integriert Daten aus operativen Systemen ausgerichtet an den spezifischen Anforderungen der Wertpapierhandel-Applikation. Während das Data Warehouse Daten zu Analysezwecken integriert, sind die Anforderungen der Internet-Applikation eher mit denen

einer herkömmlichen transaktionsorientierten operativen Applikation vergleichbar. Neben der Forderung nach integrierten Daten sind die Geschwindigkeit des Zugriffs auf diese Daten sowie die Qualität und Aktualität der Daten von entscheidender Bedeutung. Die Redundanz der Datenhaltung wird dabei bewusst in Kauf genommen. Es handelt sich um eine kontrollierte Redundanz, bei der das Auftreten von Inkonsistenzen weitgehend ausgeschlossen ist. Im Unterschied zum Data Warehouse wird auf die Transfer-Komponente auch schreibend zugegriffen. Da der Wertpapierhandel über das Internet nur einen Teil des gesamten Wertpapierhandels darstellt, müssen die Daten ständig mit den Daten in den klassischen Handelssystemen synchronisiert werden. Um dem Kunden analytische Funktionalitäten wie Simulationen und Analyse zu ermöglichen, werden über die Transfer-Komponente auch Daten aus dem Data Warehouse für die Wertpapierhandel-Applikation nutzbar gemacht. Gleichzeitig wird die Wertpapierhandel-Applikation vom Data Warehouse entkoppelt. Die Nutzung analytischer Daten ist somit nicht an die Verfügbarkeit des Data Warehouses gebunden und das Data Warehouse wird nicht durch Massenzugriffe aus dem Internet zusätzlich belastet. Tabelle 9 zeigt die Bewertung der Datenflüsse anhand ihrer Vor- und Nachteile im Einzelnen:

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
1	Kundenverwaltung	Internet-Applikation (WPH)	• Kundendaten werden benötigt	
2	Kundenverwaltung	Data Warehouse	• DWH als Analysebasis • Lastverteilung • Integration • Historisierung • Schnittstellen-minimierung	
3	Data Warehouse	Transfer-Komponente	• nicht doppelt	• kompliziert • an DWH-Update-Zyklen gekoppelt
4	Transfer-Komponente	Internet-Applikation (WPH)	• notwendig	
5	Data Warehouse	Internet-Applikation (WPH)	• Performance • Detaillierungsgrad	• Aktualität • nicht alle Daten vorhanden • Performance

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
6	Operative Systeme	Transfer-Komponente	• Aktualität • Resultate werden in op. Systeme berechnet	• Synergien aus DWH nicht ausgenutzt
7	Kontoverwaltung	Internet-Applikation (WPH)	• Aktualität • Performance	
8	Analytische Applikationen	Internet-Applikation (WPH)	• notwendig für Simulation/Analyse	
9	WPH-Datenbasis	Internet-Applikation (WPH)	• notwendig	• Desintegration
10	Externe Daten	Internet-Applikation (WPH)	• notwendig	• Desintegration • neue Schnittstellen

Tabelle 9: Bewertung der Datenflüsse

Die Tabelle macht vor allem auch mögliche Nachteile des Entwurfs deutlich. Durch das Einführen der zusätzlichen Transfer-Komponente werden eine Reihe von Schnittstellen benötigt, die zum Teil sehr komplexe Transformationen durchführen müssen. Schwierigkeiten sind vor allem bei der Aktualität und Qualität zu erwarten. Verursacht werden diese Schwierigkeiten bspw. durch begrenzte Verfügbarkeit der beteiligten Komponenten, unterschiedliche Datenmodelle und die Synchronisierung beim Schreibzugriff. Aus der Perspektive des schreibenden Zugriffs sind in erster Linie die in Abbildung 17 dargestellten Datenrückflüsse von Bedeutung:

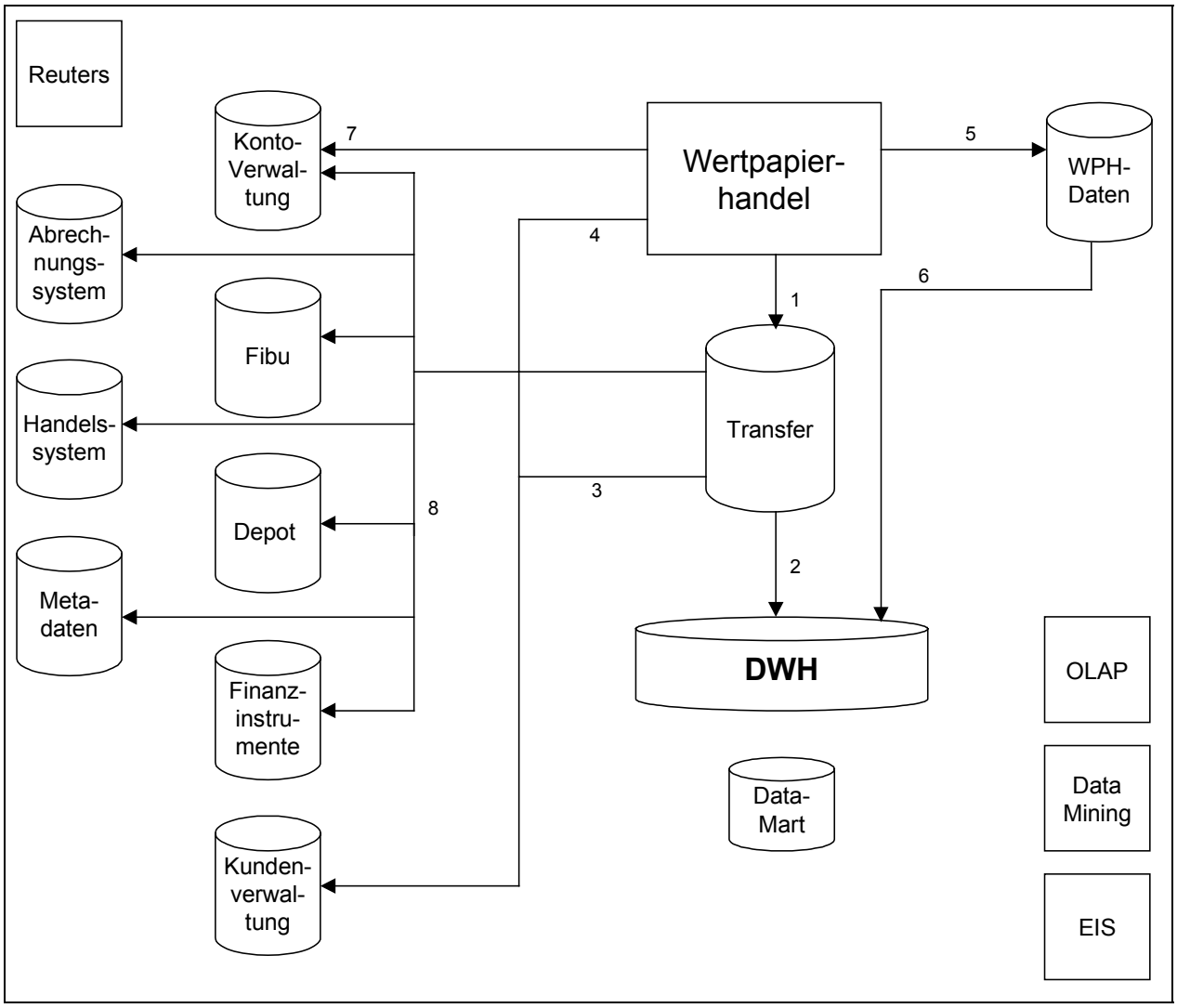


Abbildung 17: Datenrückflüsse zum Anwendungsszenario „Wertpapierhandel“

Auch bei der Ausführung der Datenrückflüsse kommt der Transfer-Komponente eine zentrale Rolle zu. Mit Ausnahme der Direkt-Durchgriffe auf die Kundenverwaltung und die Kontoverwaltung, die aus Konsistenz- und Aktualitätsgründen geschehen, werden sämtliche Schreibzugriffe auf die operativen Systeme über die Transfer-Komponente abgewickelt. Ziel ist hierbei vor allem eine Schnittstellenreduzierung, wie sie auch durch das ursprüngliche Data-Warehouse-Konzept verfolgt wird. In der Bewertung der Datenrückflüsse wurde jedoch dieser Lösung eine so hohe Komplexität angelastet, dass sie momentan in der Praxis nicht realisierbar scheint. Die Vor- und Nachteile der einzelnen Datenrückflüsse sind wiederum in einer Tabelle übersichtlich zusammengefasst:

Nr.	Komponenten		Vorteile	Nachteile
	Quelle	Ziel		
1	Internet-Applikation (WPH)	Transfer-Komponente	• Unabhängigkeit • Operative Entkoppelung	
2	Transfer-Komponente	Data Warehouse	• Nutzung der WPH-Daten für Analyse	
3	Transfer-Komponente	Kundenverwaltung	• notwendig	• zur Zeit unrealistisch
4	Internet-Applikation (WPH)	Kundenverwaltung	• keine Redundanzen	• keine Vermischung von Kundendaten mit WPH-Daten • zur Zeit unrealistisch
5	Internet-Applikation (WPH)	WPH-Datenbasis	• notwendig	
6	WPH-Datenbasis	Data Warehouse	• Nutzung der WPH-Daten für Analyse	
7	Internet-Applikation (WPH)	Kontoverwaltung	• notwendig • Aktualität	
8	Transfer Komponente	Operative Systeme	• notwendig • Aktualität	• zur Zeit unrealistisch

Tabelle 10: Bewertung der Datenrückflüsse

Die Bewertung der Datenrückflüsse lässt den eher visionären Charakter der entworfenen Lösung erkennen. Obwohl konzeptionell schlüssig und sinnvoll, gibt es bei der Umsetzung von Konzepten, die einen geschlossenen Datenfluss-Kreislauf zum Ziel haben, in der Praxis noch so grosse Probleme, dass sie von Architektur-Experten als zur Zeit unrealistisch bewertet werden.

Nichtsdestotrotz geben aktuelle Entwicklungen in den Bereichen Data Warehousing, Knowledge Management und Enterprise Application Integration berechtigten Anlass zur Hoffnung, dass in naher Zukunft Konzepte und Software-Produkte verfügbar sind, die eine Umsetzung dieses Architekturentwurfs ermöglichen.

Abschliessend werden die beiden getrennten Sichten auf Datenflüsse und Datenrückflüsse wieder zusammengeführt, um den Architekturbewurf ganzheitlich darzustellen:

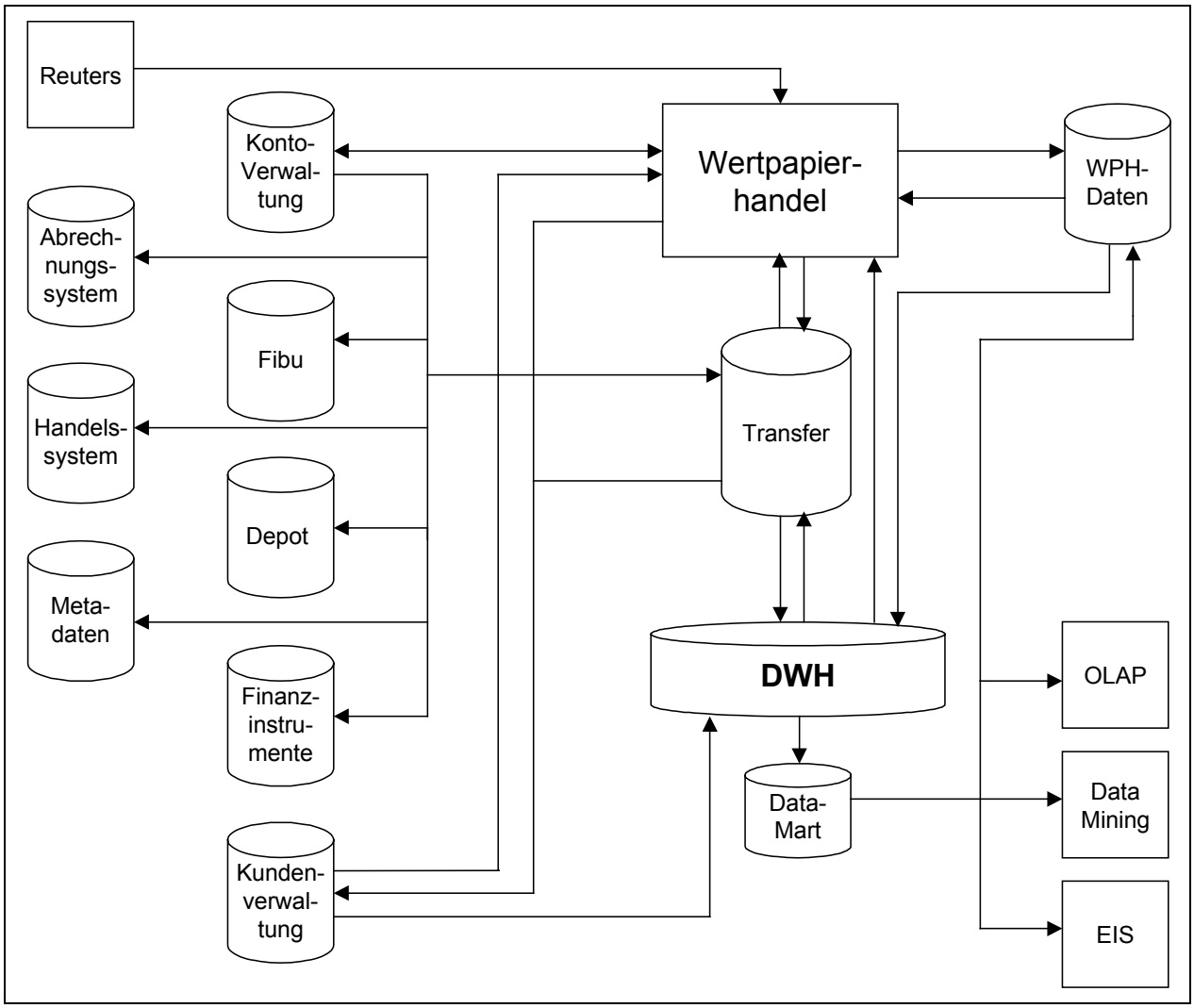


Abbildung 18: Informationssystemarchitektur zum Anwendungsszenario „Wertpapierhandel“

Besonders deutlich wird die grundlegende Idee eines geschlossenen Datenfluss-Kreislaufes unter Einbeziehung von horizontalen und vertikalen operativen Applikationen, analytisch-dispositiven Applikationen sowie externer Datenquellen. Dieses Konzept bildet letztlich die Voraussetzung für eine unternehmensweite Informationslogistik, die sämtliche Vertriebskanäle mit den benötigten Daten versorgt.

6 Zusammenfassung und Ausblick

Ausgangspunkt des vorliegenden Arbeitsberichts ist die zunehmende Bedeutung der Integration von Informationssystemen zur Schaffung einer unternehmensweiten Informationslogistik. Diese Integrationsbestrebungen sind getrieben von Anforderungen, die aus neuen Geschäftsmodellen (z. B. E-Business), Strategien zur Ausrichtung auf den Kunden (CRM) sowie Ansätzen für die Optimierung von Geschäftsabläufen (z. B. SCM) erwachsen. Die Informationssystemarchitektur ist ein wesentlicher Bestimmungsfaktor für die erforderlichen Anstrengungen, um heterogene Applikationen zu integrieren. Entsprechend standen zur Einordnung und Bestimmung von Systemkomponenten Architekturfragen im Mittelpunkt der Betrachtungen. Das vorgestellte Modell der betrieblichen Applikationslandschaft bietet hierzu einen strukturierten Rahmen für Begriffsbestimmungen und unterstützt so die organisatorische Einordnung horizontaler Applikationen.

Im Zusammenhang mit dem Data Warehouse sind vor allem die Integration über Schnittstellen, mittels Operational Data Store sowie Messaging von Interesse. Abhängig von den jeweiligen Anforderungen und Bedingungen kann die Integration mit dem Data Warehouse mit besonderem Schwerpunkt auf einer dieser Integrationsformen gelingen. Je komplexer sich allerdings die vorherrschende Applikationslandschaft gestaltet, desto sinnvoller erscheint eine flexible Kombination der drei dargestellten Konzepte.

Kapitel 5 zeigt anhand von drei Fallbeispielen praxisnah verschiedene Möglichkeiten für die Gestaltung integrierter Architekturen zur Unterstützung unterschiedlicher Geschäftsmodelle. Trotz der zahlreichen Unterschiede in den entworfenen Architekturen lassen sich auch einige grundlegende strukturelle Gemeinsamkeiten erkennen. In allen drei Ansätzen bleibt das Data Warehouse die wesentliche Grundlage für die analytisch-dispositive Applikationen wie z. B. OLAP-Tools, Reporting-Tools und Data Mining. Der steigenden Bedarf nach Analysemöglichkeiten in Applikationen, die zur direkten Abwicklung von Geschäftsvorfällen oder anderen operativen Aufgaben dienen (z. B. CRM), kann ebenfalls teilweise durch das Data Warehouse abgedeckt werden.

Operative Applikationen, die Prozesse über die Grenzen klassischer funktionsorientierter Applikationen hinweg unterstützen, benötigen hierfür Daten aus sämtlichen betroffenen Applikationen. Vielfach sind diese Daten bereits im Data Warehouse integriert. Das Data Warehouse integriert die Daten jedoch zu Analyse-Zwecken, während horizontale Applikationen wie CRM-

Systeme oder E-Business-Anwendungen hauptsächlich transaktionsorientiert arbeiten und somit stark abweichende Anforderungsprofile abweichen, die eine direkte Nutzung eines vorhandenen Data Warehouses als integrierte Datenbasis für horizontale Applikationen verhindern. In allen drei Fallbeispielen macht dieser Umstand die Einführung einer weiteren integrierten Datenbasis erforderlich. Diese Komponente integriert die Daten nach operativen Gesichtspunkten und weist starke Ähnlichkeit mit dem in der Literatur beschriebenen Konzept des ODS auf. Wichtiger als eine buchstäbliche Orientierung am ODS-Konzept ist das Vermeiden konzeptineller Fehler auf der einen und die Einbeziehung technischer Machbarkeitsbedingungen auf der anderen Seite.

Für das Vorgehen beim Entwurf einer Informationssystemarchitektur zur Kopplung von Data Warehouse mit operativen (horizontalen) Applikationen lassen sich aus den betrachteten Fallbeispiele einige grundlegenden Richtlinien ableiten:

- Die Ausgestaltung einer spezifischen Architektur ist in hohem Masse abhängig von der existierenden Systemlandschaft. In der Praxis werden Informationssystemarchitekturen nur in seltenen Fällen ohne Einschränkungen durch existierende Applikationen entworfen. Eine gründliche Analyse der existierenden Systeme und ihrer Beziehungen zueinander unter Zugrundelegung des in Kapitel 3 beschriebenen Modells der betrieblichen Applikationslandschaft ist daher unbedingt erforderlich.

- Ein weiterer erfolgskritischer Faktor ist die enge Abstimmung der Architektur an den Anforderungen des Geschäftsmodells. Um diese Anforderungen berücksichtigen zu können, ist zunächst eine Analyse und darauf basierend eine Spezifikation notwendig. Als hilfreich hat sich dabei die Orientierierung an den zugrundeliegenden Prozessen erwiesen. Vor allem ist zu ermitteln, welche Daten zu welchem Zweck integriert werden müssen. Dies hat entscheidenden Einfluss auf die Auswahl bzw. Kombination der zur Verfügung stehenden Integrationskonzepte.

- Anhand der Spezifikation der Applikationslandschaft und der Anforderungen können nun Datenflüsse zwischen Komponenten der Architektur betrachtet und bewertet werden. Zur Komplexitätsreduzierung hat sich dabei eine Unterscheidung in zwei Sichten bewährt. Dabei werden Datenflüsse, die konsumierende Applikationen mit Daten versorgen, und somit für die Quellsysteme den Charakter von Lesezugriffen haben, von Datenrückflüssen unterschieden, die aus Sicht der versorgenden System den Charakter eines Schreibzugriffes aufweisen. Für die Beurteilung des Gesamtentwurfs ist es jedoch erforderlich, die zwei Sichten ab-

schliessend wieder zusammenzuführen, um das Zusammenspiel der Datenflüsse beurteilen zu können.

Sowohl bei der Aufarbeitung der theoretischen Grundlagen als auch bei der praktischen Erprobung und Umsetzung der Konzepte in den Fallbeispielen wird das Potenzial einer gezielten Nutzung von Metadaten für eine integrierte Informationslogistik immer deutlicher. Die Verwaltung und Bereitstellung zusätzlicher semantischer Informationen durch Metadaten ist bereits heute wesentliche Grundlage für Datenqualitätsmanagement, Applikationsintegration sowie Datenschutz- und Sicherheit. Gelingt eine Integration dieser Bereiche über ein zentrales Metadatenmanagement, könnten neben einer strukturierten und redundanzfreien Dokumentation von Daten und Systemen durch Metadaten in Zukunft wesentliche Fortschritte in Bereichen wie Auswirkungsanalyse für Wartung und Weiterentwicklung, flexible Anpassung von Systemen, zentrale Versorgung verschiedener Systeme mit Änderungsinformationen sowie Automatisierung von Abläufen durch Metadatensteuerung erzielt werden.

Literaturverzeichnis

[Bach, Österle 2000]

Bach, Volker; Österle, Hubert (Hrsg.): Customer Relationship Management in der Praxis. Springer Verlag, Berlin, Heidelberg, New York et. al. 2000.

[Ferstl, Sinz 1993]

Ferstl, Otto K.; Sinz, Elmar J.: Grundlagen der Wirtschaftsinformatik. Band 1. Oldenbourg Verlag, München, Wien 1993.

[Fleisch, Österle 2001]

Fleisch, Elgar; Österle, Hubert: Das Tor zur IT-Welt. Thesen zum erfogreichen Portaleinsatz. Computerwoche extra, 2/2001, S. 28-31; 2001.

[Forrester 2000]

Forrester Research, Inc.: Customer Context Servers. Forrester Research Report, Oktober 2000.

[Heine 1999]

Heine, Peter: Unternehmensweite Datenintegration. Modular-integrierte Datenlogistik in betrieblichen Informationssystemen. B. G. Teubner Stuttgart, Leipzig 1999.

[Hofmann 2001]

Hofmann, Roman E.: Customer Relationship Management bei Schweizer Erstversicherungen. Diplomarbeit Master of Business Engineering MBE 3, St. Gallen 2001.

[Inmon 1999]

Inmon, William H.: Building the Operational Data Store. 2nd Ed. John Wiley & Sons, Inc., New York 1999.

[Jung, Winter 2000]

Jung, Reinhard; Winter, Robert (Hrsg.): Data Warehousing Strategie. Erfahrungen, Methoden, Visionen. Springer Verlag, Berlin, Heidelberg, New York et. al. 2000.

[Mertens 2000]

Mertens, Peter: Integrierte Informationsverarbeitung 1. Administrations- und Dispositionssysteme in der Industrie. 12. Auflage. Gabler, Wiesbaden 2000.

[Riehm 1997]

Riehm, Rainer: Integration von heterogenen Apllikationen. Dissertation der Universität St. Gallen Nr. 2006, St. Gallen 1997.

[Ruh, Maginnis, Brown 2001]

Ruh, William A.; Maginnis, Francis X.; Brown, William J.: Enterprise Application Integration. John Wiley & Sons, Inc., New York 2001.

[Scheer 1998]

Scheer, August-Wilhelm: ARIS - Vom Geschäftsprozeß zum Anwendungssystem. 3. Auflage. Springer Verlag, Berlin, Heidelberg, New York et. al. 1998.

[Schelp 2001]

Schelp, Joachim: DV-Architekturen für die Enterprise Application Integration, 1st Draft, Arbeitspapier, Kompetenzzentrum Application Integration Management, CC-AIM-01-01, Institut für Wirtschaftsinformatik IWI1, Universität St. Gallen, St. Gallen 2001.

[Schmidt 2000]

Schmidt, Herrad (Hrsg.): Modellierung betrieblicher Informationssysteme. Proceedings der MobIS-Fachtagung 2000. Siegen 2000.

[Winter 2000a]

Winter, Robert: Zur Positionierung und Weiterentwicklung des Data Warehousing in der betrieblichen Applikationsarchitektur. In: [Jung, Winter 2000].

[Winter 2000b]

Winter, Robert: Zur Positionierung und Weiterentwicklung des Data Warehousing in der betrieblichen Applikationsarchitektur. In: [Schmidt 2000].