**ZUR POSITIONIERUNG UND WEITERENTWICKLUNG DES DATA WAREHOUSING IN DER
BETRIEBLICHEN APPLIKATIONSARCHITEKTUR**

Robert Winter

Institut für Wirtschaftsinformatik, Universität St. Gallen

Müller-Friedberg-Strasse 8, CH-9000 St. Gallen

E-mail: Robert.Winter@unisg.ch

Tel.: +41 71 224 2935 Fax: +41 71 224 2936

ZUR POSITIONIERUNG UND WEITERENTWICKLUNG DES DATA WAREHOUSING IN DER BETRIEBLICHEN APPLIKATIONSARCHITEKTUR

ABSTRACT

Data Warehouse-Systeme haben sich als neue Schicht betrieblicher Applikationen etabliert, die operative, transaktionsorientierte Applikationen von entscheidungsunterstützenden Applikationen entkoppelt. Die zunehmende Bedeutung kanalorientierter, sog. horizontaler Applikationen wirft die Frage auf, ob der Data Warehousing-Ansatz auf die Entkopplung konventioneller operativer Applikationen von horizontalen Applikationen ausgedehnt werden kann oder ob alternative Integrationsansätze dazu besser geeignet sind. Als Grundlage einer entsprechenden Analyse wird ein Modell der Applikationslandschaft eingeführt, das Applikationen in einem durch die Dimensionen „Prozess“, „Produkt“ und „Funktion“ aufgespannten Raum abbildet. Traditionelle, sog. vertikale Applikationen, entscheidungsunterstützende Applikationen und das Data Warehouse-System sowie in der weiteren Folge Querschnittapplikationen und horizontale Applikationen werden in diesem Modell positioniert. Als bedeutende Erweiterung des Data Warehouse-Ansatzes werden auch sog. Operational Data Stores in das Modell integriert. Auf Grundlage der erweiterten Applikationslandschaft wird die neue Rolle des Data Warehouse-System diskutiert. Den Abschluss des Beitrags bildet die Ableitung neuer Forschungsfragen, die sich aus dem erweiterten Verständnis von Data Warehousing in der betrieblichen Applikationsarchitektur ergeben.

1. EINFÜHRUNG

Data Warehouse-Systeme¹ haben sich als neue Schicht betrieblicher Applikationen etabliert. Der Grund liegt darin, dass sich die direkte Basierung entscheidungsunterstützender, „dispositiver“ Applikationen auf geschäftsvorfallabwickelnden, „operativen“ Applikationen als technisch oder wirtschaftlich nicht machbar erwiesen hat: Häufig ist es aufgrund der Schnittstellenkomplexität und der mangelnden Datenqualität (Inkonsistenzen zwischen verschiedenen operativen Applikationen erfordern komplexe Integrationsmechanismen) technisch nicht möglich, dispositive Applikationen zeitnah mit konsistenten, integrierten Daten aus operativen Applikationen zu versorgen. Selbst technisch mögliche „Direktanschlüsse“ scheitern in komplexen Applikationslandschaften fast

¹ In Analogie zum Begriffspaar „Datenbank“ – „Datenbanksystem“ wird im folgenden als Data Warehouse-System die Gesamtheit der Applikationen und Datenbanken bezeichnet, die das Data Warehouse i.e.S. (d.h. die Datenbank, die entscheidungsrelevante, integrierte, historisierte, u.U. verdichtete Daten enthält) nutzbar

immer an einer Wirtschaftlichkeitsanalyse: Es ist nicht wirtschaftlich möglich, eine Vielzahl dispositiver Applikationen direkt mit einer Vielzahl operativer Applikationen zu verknüpfen und diese Verknüpfungen über einen längeren Zeitraum an Veränderungen der Applikationslandschaft anzupassen. Das Data Warehouse-System entkoppelt als Zwischenschicht dispositive und operative Applikationen und ist damit in der Lage, Integrationsmechanismen sowie die daraus gewonnenen Daten für dispositive Applikationen wiederverwendbar zu realisieren und die Anpassung an Änderungen von Applikationen auf jeweils eine einzige Schnittstelle zu begrenzen. Die Wirkung des Data Warehouse-Systems als Zwischenschicht wird in Abbildung 1 illustriert.

Abbildung 1: Data Warehouse-System als Zwischenschicht

Die Basis eines Data Warehouse bilden die operativen Applikationen des Unternehmens bzw. die dazugehörigen Datenhaltungssysteme; hier reicht das Spektrum von Dateien über hierarchische Datenbanken bis hin zu relationalen und objektorientierten Datenbanken. Darüber hinaus werden vielfach auch externe Daten, z.B. demoskopische Daten und Marktdaten, in das Data Warehouse integriert [9]. Darauf bauen Schnittstellensysteme oder Extraktionsprozeduren auf, die die Daten der operativen Applikationen übernehmen, in das Format bzw. logische Datenmodell des Data Warehouse transformieren. Die Daten des Data Warehouse werden durch Aufbereitungssysteme oder Bereitstellungsprozeduren gefiltert und/oder aggregiert und stehen danach den betreffenden dispositiven Applikationen zur Auswertung zur Verfügung bzw. werden in die Datenhaltungssysteme

macht [1, S.17]. Data Warehousing bezeichnet die Gesamtheit aller Aktivitäten, die mit der Entwicklung und dem Betrieb des Data Warehouse-Systems verbunden sind.

der dispositiven Applikationen geladen. Die Architektur des Data Warehouse-Systems in Abbildung 1 ist stark idealisiert; eine ausführliche Beschreibung findet sich in Abschnitt 4).

In diesem Beitrag wird die Weiterentwicklung des Data Warehousing als Komponente der betrieblichen Applikationsarchitektur untersucht. Damit wird versucht, Entwicklungstendenzen der betrieblichen Data Warehouse-Nutzung und damit auch zukünftige Themen anwendungsorientierter Wirtschaftsinformatik-Forschung abzuleiten. Zu diesem Zweck wird in Abschnitt 2 ein Modell der betrieblichen Applikationslandschaft entwickelt und das Data Warehouse-System darin positioniert. In Abschnitt 3 wird die aktuelle Weiterentwicklung der betrieblichen Applikationslandschaft durch Einführung horizontaler Applikationen beschrieben. Ein anderes aktuelles Phänomen im Data Warehousing ist die Einführung von Operational Data Stores als neue Architekturschicht, mit der sich Abschnitt 4 beschäftigt. Eine sich aus den beiden letztgenannten Entwicklungen ergebende, veränderte Rolle von Data Warehouse-Systemen und die daraus resultierende Informationslogistik-Architektur werden in Abschnitt 5 beschrieben. Abschnitt 6 fasst die auf diese Architektur bezogenen neuen Forschungsfragen zusammen.

2. MODELL DER BETRIEBLICHEN APPLIKATIONSLANDSCHAFT

Obwohl in der Wirtschaftsinformatik-Literatur weitgehende Einigkeit über die wettbewerbsstrategische Bedeutung der Informationssystemarchitektur sowie über die Zweckmässigkeit der Verwendung des Architekturbegriffs besteht, [7, S.59], findet sich dafür keine eindeutige, generelle Definition [13, S.13]. Beispielsweise bezeichnet Scheer die Komponenten eines Informationssystems und ihre Beziehungen als Architektur [12, S.3]. Österle definiert die Informationssystem-Architektur als Grobstruktur der Organisation, der Geschäftsfunktionen, der Daten, der Applikationen und der Datenbanken [10, S.26], die den konzeptionellen Rahmen für die Entwicklung von Organisation, Applikationen und Datenbanken bildet [10, S.69].

In Analogie zur Architekturlehre im Bauwesen [7, S.58] [15, S.2] sollte die Informationssystemarchitektur im Sinne eines Bauplans des Informationssystems die Spezifikation und Dokumentation der Applikationen und ihrer Beziehungen unter allen relevanten Blickwinkeln sowie die Konstruktionsregeln für die Erstellung des Bauplans umfassen.

Die Applikationslandschaft ist eine wichtige Komponente der Applikationsarchitektur. Sie dokumentiert auf aggregierter Ebene die wichtigsten Applikationen des Unternehmens und ihren Zusammenhang (d.h. die wichtigsten Informations- und Steuerflüsse zwischen Applikationen). In vielen Unternehmen ist die Applikationslandschaft historisch, d.h. über einen längeren Zeitraum gewachsen. Im schlimmsten Fall wurden Applikationen

planlos neben- und hintereinander eingeführt. Erfolgte eine zumindest rudimentäre Architekturplanung, orientierte sich die Strukturierung von Applikationen an Funktionalbereichen (z.B. Auftragsabwicklung, Materialwirtschaft, Rechnungswesen) oder an produkt(gruppen)orientierten Sparten. In einem Versicherungsunternehmen finden sich beispielsweise in der Regel für jede Produktgruppe (Sparte, z.B. Einzellebens-, Einzelkranken-, Gruppen- oder Sachversicherung) eigene Applikationen mit eigener Datenhaltung. Im Extremfall werden von jeder Sparte separate Kundenstammdaten und ein separates Produktmodell verwaltet. Als Konsequenz wird ein Versicherungsnehmer, der mit einem Unternehmen mehrere Versicherungsverträge abgeschlossen hat, auch in mehreren Applikationen redundant als Kunde geführt. Eine für den Geschäftserfolg und die Kundenbeziehung negative Konsequenz ist, dass der Kunde von Unternehmensseite nicht als *ein* Kunde identifizierbar und ansprechbar ist. Der Kunde wird mehrere Rechnungen erhalten, Cross-Selling-Potenziale bleiben mangels umfassender Kundeninformationen ungenutzt und der Kunde muss sich u.U. über Marketingaktionen ärgern, durch die ihm Versicherungen angeboten werden, die er bereits mit der betreffenden Gesellschaft abgeschlossen hat.

Ein erstes, einfaches Modell der Applikationslandschaft stellt Applikationen in einem Raum dar, der durch die Dimensionen „Funktion“, „Produkt(gruppe)“ und „Prozess“ aufgespannt wird. In der Dimension „Funktion“ werden die verschiedenen Funktionalbereiche (z.B. Antragsbearbeitung, Schadenbearbeitung) abgetragen. In der Dimension „Produkt(gruppe)“ werden die verschiedenen Produkte bzw. Produktgruppen abgetragen. In der Dimension „Prozess“ wird der Ablauf der operativen Geschäftsprozesse abgetragen.

Abbildung 2: Einfaches Modell mit vertikalen Applikationen

Abbildung 2 zeigt beispielhaft die Positionierung traditioneller operativer Applikationen (am Beispiel eines Versicherungsunternehmens) in einem entsprechenden dreidimensionalen Raum. Traditionelle Applikationen umfassen verschiedene Komponenten, die in ihrer Gesamtheit alle funktionalen Aspekte und die vollständigen Geschäftsprozesse für eine bestimmte Produktgruppe abdecken [3, S.2-3]. Die Applikationslandschaft setzt sich aus einer relativ kleinen Zahl solcher Applikationen(saggregate) zusammen, die aufgrund ihres optischen Erscheinungsbilds auch in diesem Modell als „vertikale“ Applikationen bezeichnet werden können.

Eine erste grundlegende Erweiterung „vertikal“ geprägter Applikationslandschaften besteht in der Zusammenführung bestimmter Querschnittsfunktionen in separate Applikationen. So lassen sich z.B. durch Auslagerung der Kundenverwaltung aus allen vertikalen Applikationen in eine einzige, produkt(gruppen)übergreifende Partner-Applikation die weiter oben beschriebenen Redundanzprobleme weitgehend vermeiden. Ähnliche Vorteile werden durch die Auslagerung aller konfigurierungs-orientierten Funktionalitäten aus den vertikalen Applikationen in eine einzige, produkt(gruppen)übergreifende Produkt-Applikation oder die Auslagerung aller Abrechnungs- und Berichts-orientierten Funktionalitäten aus den vertikalen Applikationen in eine einzige, produkt(gruppen)übergreifende Abrechnungs- und Berichtsapplikation erzielt [6] [8] [14]. Obwohl die von derartigen Querschnittsapplikationen verwalteten Daten von allen anderen Applikationen gelesen werden und damit den Charakter von „reference data“ haben, sind sie als operative Daten zu betrachten [2, S.141f]. Die sich ergebende, modifizierte Applikationslandschaft wird durch Abbildung 3 illustriert.

Abbildung 3: Vertikale Applikationen und Querschnittsapplikationen

Applikationen auslagern, bereits Anfang der 90er Jahre einsetzte, wurden aufgrund der stark zunehmenden Bedeutung elektronischer und telefoniebasierter Zugangskanäle erst in letzter Zeit auch zugangskanalspezifische Funktionalitäten in dedizierte Applikationen ausgelagert. Grundlage ist auch in diesem Fall die Erkenntnis, dass bestimmte Zugangsfunktionen nicht für jedes Produkt bzw. jede Produktgruppe in den jeweiligen Abwicklungsapplikationen repliziert werden sollten. Nur wenn produktspezifisches und zugangskanalspezifisches Applikationswissen unabhängig voneinander implementiert wird, ist ein effizientes Multikanalmanagement möglich, d.h. können Entscheidungen über den Vertrieb bestimmter Produkte an bestimmte Kundensegmente unter Nutzung bestimmter Vertriebs- bzw. Zugangskanäle gezielt geplant und schnell umgesetzt werden.

Die Abgrenzung zugangskanalspezifischer Applikationen wird im wesentlichen durch das jeweils im Vordergrund stehende Zugangsmedium impliziert: Kunden fordern immer nachdrücklicher den Zugang zu Produkten bzw. Dienstleistungen nach ihrer Wahl sprachgesteuert über das Telefonnetz, PC-basiert über das Internet, Handy/WAP-basiert über das Mobiltelefonnetz, schriftstückbasiert über den Briefverkehr, über SB-Terminals oder über Aussendienst- bzw. Innendienstmitarbeiter. Zugangsfunktionen sollten deshalb produkt(gruppen)übergreifend in speziell auf einen bestimmten Zugangskanal zugeschnittenen Applikationen erfolgen, z.B. Call Center-Applikation, WWW-Portal, WAP-Portal, Letter Center/Document Management-Applikation, SB Terminal-Applikation oder Innendienst-Applikation. Neben der Unterstützung eines jeweils anderen Zugangskanals können sich diese Applikationen auch durch unterschiedlich ausgelegte Sicherheitsprüfungen und abweichende Funktionsumfänge (z.B. Innendienst-Applikation vs. WWW-Portal) unterscheiden.

Abbildung 5: Applikationslandschaft mit horizontalen Applikationen

Im weiter oben eingeführten Modell der Applikationslandschaft stellen sich zugangskanalspezifische Applikationen als „horizontale“ Applikationen dar: Bestimmte Funktionalitäten werden über alle Produkte bzw. Produktgruppen hinweg für einen bestimmten Abschnitt des Prozesses in dedizierten Applikationen zusammengefasst. Die Positionierung horizontaler Applikationen wird in Abbildung 5 illustriert.

Im Prinzip ändert die Einführung horizontaler Applikationen an der Positionierung von Data Warehousing-Systemen nichts: Auch horizontale Applikationen dienen als operative Applikationen der Abwicklung von Geschäftsfällen und erzeugen bzw. verwalten damit operative Daten. Die gleiche Zwischenschicht, die zur Transformation operativer Daten aus vertikalen Applikationen und Querschnittapplikationen in entscheidungsunterstützende Informationen dient, kann auch zur Transformation operativer Daten aus horizontalen Applikationen in entscheidungsunterstützende Informationen genutzt werden.

4. EINFÜHRUNG VON OPERATIONAL DATA STORES

Die Transformation operativer Daten in entscheidungsunterstützende Informationen durch ein Data Warehouse-System nimmt einen gewissen Zeitraum in Anspruch und erzeugt Informationen, die im Normalfall nicht verändert werden dürfen und die im Normalfall aggregiert sind. Oft besteht jedoch ein dringender Bedarf, sehr schnell auf integrierte, detaillierte Daten aus verschiedenen operativen Applikationen zuzugreifen. Da ein real-time-Zugriff auf integrierte, detaillierte Daten aus operativen Applikationen durch das Data Warehouse-System nicht abgedeckt werden kann, wurde das Konzept des Operational Data Store eingeführt [2, S.143f] [5, S.20]. In einem Operational Data Store werden operative Daten sehr zeitnah zusammengeführt, so dass eine neue Architekturschicht entsteht, deren Daten an Informationsobjekten orientiert, aktuell, änderbar, detailliert, integriert und vor allem real-time zugänglich sind [3, S.4] [4, S.1-2]. Die Positionierung von Operational Data Stores „zwischen“ operativen Applikationen und dem Data Warehouse-System wird aus der Gegenüberstellung der Eigenschaften der jeweils verwalteten Daten in Tabelle 1 deutlich.

	Operative Applikation	Operational Data Store	Data Warehouse
Orientierung	Geschäftsvorfall	Informationsobjekt	Entscheidungsrelevanter Sachverhalt
Zeitbezug	aktuell	aktuell	Historisiert
Änderbarkeit	read-write	read-write	read-only
Aggregationsgrad	detailliert	detailliert	u.U. aggregiert
Integrationsgrad	isoliert	integriert	integriert
Zugänglichkeit	real-time	real-time	abgeleitet / verzögert

Tabelle 1: Operative Applikation, Operational Data Store und Data Warehouse

Ob die Einführung von Operational Data Stores als Zwischenschicht wirklich notwendig ist, sollte im Einzelfall entschieden werden: Solange es nur um die Bereitstellung von Daten für Auswertungen geht, können die durch Operational Data Stores abgedeckten Informationsbedarfe oft auch durch Auslagerung von Funktionalitäten in Querschnitt-Applikationen oder durch häufigere bzw. schnellere Aktualisierungen des Data Warehouse erfüllt werden. Derartige Data Stores sind Komponenten des Data Warehouse-Systems und bilden keine eigene Architekturschicht. Sobald jedoch operative Applikationen Daten mit Hilfe von Operational Data Stores in real-time austauschen, d.h. sobald auch ein schreibender Zugriff auf integrierte, aktuelle Daten erfolgt, ist die Einführung einer zusätzlichen Architekturschicht notwendig [5, S.20].

5. DIE VERÄNDERTE ROLLE DES DATA WAREHOUSING-SYSTEMS

Eine ähnliche Rolle, wie sie das Data Warehouse-System für die Entkopplung von operativen Applikationen und entscheidungsunterstützenden Applikationen spielt, kommt Operational Data Stores für die Entkopplung „vertikaler“ und „horizontaler“ operativer Applikationen zu [3, S.5]. Auch im Fall der Operational Data Stores ist es im Normalfall sinnvoller, wenige Data Stores zur Integration der verschiedenen vertikalen und horizontalen operativen Applikationen zu benutzen als zwischen Applikationen, die Daten austauschen, jeweils paarweise individuelle Schnittstellen zu implementieren.

Es mag an der Ähnlichkeit der Entkopplungsfunktion liegen, dass auch zur Applikationsintegration genutzte Operational Data Stores oft im Data Warehouse platziert werden oder dass vorgeschlagen wird, das Data Warehouse auch unmittelbar für operative, kundenorientierte Dienstleistungen wie z.B. das Kundenbeziehungsmanagement oder e-Commerce zu nutzen (z.B. [11]). Im vorhergehenden Abschnitt wurde jedoch bereits ausgeführt,

dass Operational Data Stores dann, wenn sie definitionsgemäss (d.h. mit Schreibzugriff auf ihre Daten) eingesetzt werden, nicht Teil des Data Warehouse sein können. Die sich ergebende Applikationslandschaft wird durch Abbildung 6 illustriert. Zwischen vertikalen und horizontalen operativen Applikationen lässt sich durch sorgfältig modellierte Operational Data Stores ein effizientes „Closed Loop“-Modell realisieren. Zwischen operativen Applikationen und Business Intelligence-Applikationen kann zwar durch sorgfältige Modellierung des Data Warehouse-Systems eine effiziente Bereitstellung entscheidungsunterstützender Informationen realisiert werden; es muss jedoch bei einem „Open Loop“-Konzept bleiben, weil Informationsrückflüsse definitionsgemäss ihren Weg nicht durch das Data Warehouse nehmen, sondern über Aktionen der Entscheider mehr oder weniger mittelbar in operative Applikationen eingehen.

Abbildung 6: Erweiterte Applikationslandschaft

Obwohl Operational Data Stores und das Data Warehouse-System offensichtlich unterschiedliche Rollen wahrnehmen, können bei der Entwicklung und beim Betrieb dieser Systeme zwei bedeutsame Synergien genutzt werden:

- Daten in Operational Data Stores sind bereits integriert und in real-time verfügbar. Sie stellen damit eine hervorragende Datenquelle für das Data Warehouse-System dar. Zwar wird es weder möglich noch sinnvoll sein, das Data Warehouse ausschliesslich aus Operational Data Stores zu speisen [4, S.4]. Es sollte jedoch jede Möglichkeit genutzt werden, integrierte Daten aus Operational Data Stores zu beziehen und nicht für die Einspeisung ins Data Warehouse redundant integrieren zu müssen.

- Die Metadaten-Infrastruktur, die für den sinnvollen Betrieb eines Data Warehouse-Systems aufgebaut und aufrechterhalten werden muss, sollte prinzipiell auch für die Modellierung und den Betrieb von Operational Data Stores wiederverwendet werden können. Auf Metaebene unterscheiden sich nämlich real-time-Applikationsintegration und Datenaufbereitung für entscheidungsunterstützende Zwecke nicht signifikant.

6. ZUSAMMENFASSUNG UND AUSBLICK

Mit diesem Beitrag wurde der Versuch unternommen, die verschiedenen Arten operativer Applikationen, dispositive Applikationen, das Data Warehouse-System und Operational Data Stores als sich sinnvoll ergänzende Komponenten einer generalisierten betrieblichen Applikationsarchitektur darzustellen. Es wurde gezeigt, dass Data Warehouse-Systeme und Operational Data Stores zwar aufgrund fundamentaler Unterschiede als unterschiedliche Architekturschichten realisiert werden sollten, aber dass sich gleichzeitig durch aufeinander abgestimmte Modellierung dieser Architekturschichten bedeutende Synergien realisieren lassen müssten.

In diesem Zusammenhang müssen jedoch noch viele Fragen konzeptionell oder empirisch geklärt werden:

- Wie gross sind die Gemeinsamkeiten des Metadatenmanagements zwischen Data Warehouse-System und Operational Data Stores genau? Ist es sinnvoller, auf Grundlage erprobter Konzepte aus dem Data Warehousing-Bereich durch Anpassungen Lösungen für die Integration operativer Applikationen abzuleiten, oder ist sogar die Erweiterung zu einem generellen Metadatenkonzept für die Applikationsintegration denkbar?
- Lassen sich Metamodelle, Methoden und Vorgehensmodelle für die Data Warehouse-Entwicklung auch auf die Entwicklung von Operational Data Stores übertragen? Ist auf dieser Grundlage gar die Ableitung genereller Metamodelle, Methoden und Vorgehensmodelle für die Applikationsintegration denkbar?
- Im Gegensatz zu den meisten vertikalen operativen Applikationen und den meisten dispositiven Applikationen kommen für horizontale operative Applikationen häufig Standardsoftwarelösungen zum Einsatz (z.B. CRM-Applikationen, E-commerce-Applikationen, Call Center-Applikationen). Daraus könnte sich die Chance ergeben, zumindest für wichtige Teilbereiche der Informationslogistik Referenzmodelle zu entwickeln.
- Sind Organisationskonzepte aus dem Data Warehousing-Bereich auf den Bereich der Integration operativer Applikationen übertragbar?

LITERATUR

- [1] Böhnlein, M., Ulbrich-vom Ende, A.: Grundlagen des Data Warehousing – Modellierung und Architektur.
- [2] Devlin, B.: Data Warehouse – from Architecture to Implementation. Addison-Wesley: Reading u.a. 1997.
- [3] Imhoff, C.: The Corporate Information Factory, DM Review, December 1999, <http://www.dmreview.com/editorial/dmreview>, 29-03-2000.
- [4] Imhoff, C.: The Operational Data Store: Hammering Away, DM Review, July 1998, <http://www.dmreview.com/editorial/dmreview>, 29-03-2000.
- [5] Kimball, R., Reeves, L., Ross, M., Thornthwaite, W.: The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses. John Wiley & Sons: New York u.a. 1998.
- [6] Koch, G.: Ein Datenmodell als Schlüssel einer flexiblen Gestaltung von Versicherungsprodukten, in: Versicherungswirtschaft, 16/1993, 1052-1055, 1993.
- [7] Lehner, F., Maier, R., Hildebrand, K.: Wirtschaftsinformatik: theoretische Grundlagen; Hanser: Wien 1995.
- [8] Leist, S., Winter R.: Nutzung generischer Produktmodelle im Finanzdienstleistungsbereich am Beispiel des Ergebniscontrolling, Wirtschaftsinformatik, 40, 4, 281-289, 1998.
- [9] Möller, F.: Data Warehouse als Warnsignal an die Datenschutzbeauftragten. In: Datenschutz und Datensicherheit (DuD), 22, 10, 555-560, 1998.
- [10] Österle, H., Brenner, W., Hilbers, K.: Unternehmensführung und Informationssystem: der Ansatz des St.Galler Informationssystem-Managements, 2. Aufl.; Teubner: Stuttgart 1992.
- [11] OVUM Evaluates: CRM Strategies: Technology Choices for the Customer-focussed Business; OVUM Ltd., London 1999.
- [12] Scheer, A.W.: Wirtschaftsinformatik, 6. Aufl.; Springer-Verlag: Berlin u.a. 1995.
- [13] Schmalzl, J.: Architekturmodelle zur Planung der Informationsverarbeitung von Kreditinstituten; Physica: Heidelberg 1995.
- [14] Schönsleben, P., Leuzinger, R.: Innovative Gestaltung von Versicherungsprodukten: flexible Industriekonzepte in der Assekuranz, Gabler: Wiesbaden 1996.
- [15] Sinz, E.J.: Architektur betrieblicher Informationssysteme, in: Rechenberg, P., Pomberger, G. (Hrsg.): Handbuch der Informatik; Hanser: München 1997.