Edition TDWI

Modellierung von Business-Intelligence-Systemen

Leitfaden für erfolgreiche Projekte auf Basis flexibler Data-Warehouse-Architekturen

von
Michael Hahne

1. Auflage

Thematische Gliederung:

Wirtschaftsinformatik

dpunkt.verlag 2014

Verlag C.H. Beck im Internet:
www.beck.de
ISBN 978 3 89864 827 1

Inhaltsverzeichnis: Modellierung von Business-Intelligence-Systemen – Hahne

1 Business-Intelligence-Architektur

Unter dem Sammelbegriff *Business Intelligence* werden Konzepte des Data Warehouse, OLAP und Data Mining diskutiert. Durch die zunehmend strategische Ausrichtung der Informationsverarbeitung erhalten diese Konzepte einen neuen Stellenwert in der Praxis. Unter dem Begriff *Analyseorientiertes Informationssystem* werden Systemlösungen im Bereich Business Intelligence mit der Ausrichtung an der Analyseanforderung zusammengefasst.

Der Fokus analyseorientierter Informationssysteme liegt in der zeitnahen Versorgung betrieblicher Entscheidungsträger mit relevanten Informationen zu Analysezwecken. Diese Systeme zielen somit auf die Unterstützung der dispositiven und strategischen Prozesse in einem Unternehmen ab und bilden damit ein logisches Pendant zu den operativen Systemen, die zumeist in Form einer integrierten betriebswirtschaftlichen Standardsoftware wie z. B. SAP ECC (ERP Central Component) eingesetzt werden.

1.1 Data Warehouse

Allen analyseorientierten Informationssystemen gemeinsam ist eine geeignete zugrunde liegende Datenbasis. Diese bildet damit eine wesentliche Komponente, auf deren Grundlage die verschiedenen Auswertungssysteme aufsetzen. Dem Aufbau dieser zentralen Datenbasis widmet sich die Diskussion seit einigen Jahren unter dem Stichwort Data Warehouse. Hierunter soll im Folgenden ein unternehmensweites Konzept verstanden werden, dessen Ziel es ist, eine logisch zentrale, einheitliche und konsistente Datenbasis für die vielfältigen Anwendungen zur Unterstützung der analytischen Aufgaben von Führungskräften aufzubauen, die losgelöst von den operativen Datenbanken betrieben wird.

Der Begriff *Data Warehouse* geht auf Inmon zurück. Inmon beschreibt ihn mit der Aufgabe, Daten zur Unterstützung von Managemententscheidungen bereitzustellen, die die folgenden vier wesentlichen Eigenschaften aufweisen [Inmon 1996, S. 33]:

- Themenorientierung
- Vereinheitlichung
- Zeitorientierung
- Beständigkeit

Die in einem Data Warehouse abzulegenden Daten orientieren sich an dem Informationsbedarf von Entscheidungsträgern und beziehen sich demnach auf Sachverhalte, die das Handeln und den Erfolg eines Unternehmens bestimmen. Die Daten fokussieren sich daher auf die Kernbereiche der Organisation. Diese datenorientierte Vorgehensweise unterscheidet sich deutlich von den prozessorientierten Konzepten der operativen Anwendungen.

Eine wesentliche Eigenschaft eines Data Warehouse ist ein konsistenter Datenbestand, der durch eine Vereinheitlichung der Daten vor der Übernahme entsteht. Diese Vereinheitlichung bezieht sich sowohl auf die Struktur wie auch auf die Formate, häufig müssen die verwendeten Begriffe, Codierungen und Maßeinheiten zusammengeführt werden.

Für die Managementunterstützung werden Daten benötigt, die die Entwicklung des Unternehmens über einen bestimmten Zeitraum repräsentieren und zur Erkennung und Untersuchung von Trends herangezogen werden. Dazu wird der Data-Warehouse-Datenbestand periodisch aktualisiert und der Zeitpunkt der letzten Aktualisierung definiert damit einen Schnappschuss des Unternehmensgeschehens, der je nach Ladezyklus Minuten, Stunden, Tage, Wochen oder Monate zurückliegen kann.

Der Begriff *Data Warehouse* beschreibt ein unternehmensweites Konzept, dessen Ziel die Bereitstellung einer einheitlichen konsistenten Datenbasis für die vielfältigen Anwendungen zur Unterstützung der analytischen Aufgaben von Fach- und Führungskräften ist. Diese Datenbasis ist losgelöst von operativen Datenbanken zu betreiben.

Das vierte wesentliche Charakteristikum bezieht sich auf die Beständigkeit der Daten in einem Data Warehouse. Da diese in der Regel nur einmal geladen und danach nicht mehr geändert werden, erfolgt ein Datenzugriff im Allgemeinen nur lesend. Einmal erstellte Berichte auf Basis dieses Datenbestands sind daher reproduzierbar, da auch in späteren Perioden die Datenbasis die gleiche ist. Diese Eigenschaft wird mit dem Begriff der Nicht-Volatilität umschrieben.[1] Die Beständigkeit bezieht sich aber auch auf ein verlässliches annähernd gleichbleibendes Antwortzeitverhalten.

Die Einordnung eines Data Warehouse in die IT-Struktur eines Unternehmens ergibt sich aus der in Abbildung 1–1 dargestellten Referenzarchitektur. Ausgangsbasis dieser Architektur sind die operativen Vorsysteme, aus denen periodisch Datenextrakte generiert werden. Im Rahmen des ETL-Prozesses (*extract transform load*, ETL) erfolgen die Bereinigung und Transformation der Daten aus den verschiedenen Vorsystemen sowie externen Datenquellen zu einem konsistenten einheitlichen Datenbestand und der Transport in das Data Warehouse. Hierbei sind die beiden Phasen des erstmaligen Befüllens sowie der regelmäßigen periodischen Aktualisierungen zu unterscheiden.

1. Inmon beschreibt diese vierte Eigenschaft mit dem Begriff *non-volatile*, der sich auf die Änderungshäufigkeit bezieht [Inmon 1996, S. 35 ff.]

Dieser ETL-Komponente kommt beim Aufbau eines Data Warehouse eine zentrale Bedeutung zu, denn ein hoher Anteil des Aufwands beim Aufbau eines Data Warehouse resultiert aus der Implementierung von Zugriffsstrategien auf die operativen Datenhaltungseinrichtungen.[2]

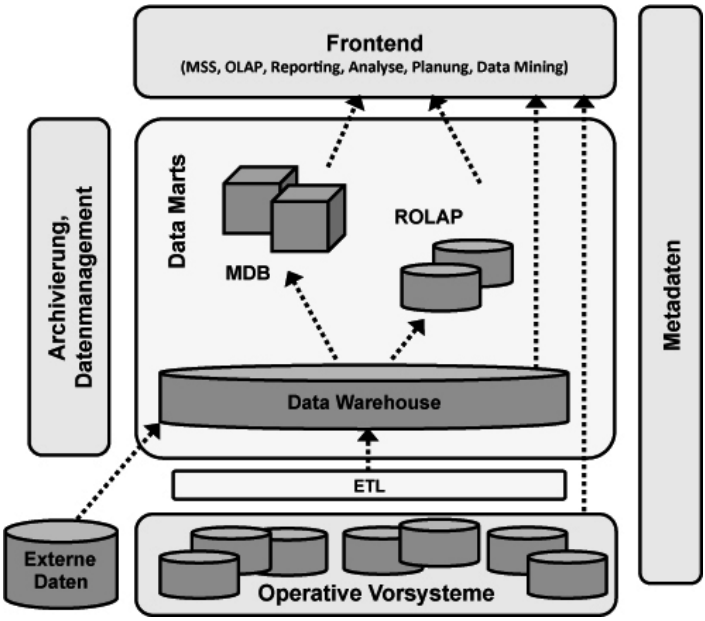

Abb. 1–1 *Data-Warehouse-Referenzarchitektur*

Aus diesem Datenbestand können des Weiteren kleinere funktions- oder bereichsbezogene Teilsichten in sogenannten *Data Marts* extrahiert werden. Diese müssen wiederum periodisch aus dem Data-Warehouse-Datenbestand aktualisiert werden. Für diese Teildatenbestände kommen im Allgemeinen sogenannte *OLAP-Datenbanken* zum Einsatz, deren Diskussion Gegenstand des nächsten Abschnittes ist.

Die Auswertung über die Frontend-Applikationen kann sowohl direkt auf dem zentralen Data Warehouse erfolgen als auch auf den einzelnen Data Marts aufsetzen. In Data-Warehouse-Konzepten können auch Applikationen wie beispielsweise Management-Support-Systeme auf diesen Datenbeständen basieren, d. h., die Datenbasis für diese Systeme kann auch in einem Data Warehouse liegen. Hier verbinden sich also bekannte Konzepte des Managementsupports mit dem neuen Konzept des Data Warehouse zu einer neuen Systemkategorie. Eine weitere wesentliche Erweiterung ergibt sich aus dem Ansatz des Online Analytical Processing (OLAP), der im folgenden Abschnitt dargestellt wird.

2. In [Jiang 2011] wird unter Berücksichtigung des Aufwands der Datenintegrationsprozesse ein Ansatz der metadatengesteuerten Generierung vorgeschlagen.

1.2 OLAP und mehrdimensionale Datenbanken

Der Begriff OLAP beschreibt ein Leitbild für eine endanwenderorientierte Analysetechnik und wird häufig konträr zum sogenannten Online Transaction Processing (OLTP) gesehen. Online Analytical Processing (OLAP) ist ein mittlerweile anerkannter Bestandteil für eine angemessene DV-Unterstützung betrieblicher Fach- und Führungskräfte und bietet einen endanwenderorientierten Gestaltungsrahmen für den Aufbau von Systemen zur Unterstützung dispositiver bzw. analytischer Aufgaben [Gluchowski/Chamoni 2009, S. 197 ff.].

Als zentrales Charakteristikum gewährleisten multidimensionale Sichtweisen auf unternehmensinterne und -externe Datenbestände brauchbare Näherungen an das mentale Unternehmensbild des Managers. Betriebswirtschaftliche Variablen bzw. Kennzahlen (wie z. B. Umsatz oder Kostengrößen) werden entlang unterschiedlicher Dimensionen (wie z. B. Kunden, Artikel, Regionen) angeordnet, und diese Strukturierung gilt als geeignete entscheidungsorientierte Sichtweise auf betriebswirtschaftliches Zahlenmaterial. Bildlich gesprochen werden die quantitativen Kenngrößen in mehrdimensionalen Würfeln gespeichert, deren Kanten durch die einzelnen Dimensionen definiert und beschriftet sind.

OLAP soll es Benutzern ermöglichen, flexible komplexe betriebswirtschaftliche Analysen wie auch Ad-hoc-Auswertungen mit geringem Aufwand eigenständig durchführen zu können. Um dieses Ziel zu erreichen, wurden von Codd, Codd und Sally 12 Regeln als Anforderung an OLAP-Lösungen definiert:[3]

1. Die *mehrdimensionale konzeptionelle Sicht* auf die Daten wird als elementarstes Wesensmerkmal für OLAP postuliert. Diese Darstellungsform ermöglicht eine Navigation in den Datenwürfeln mit beliebigen Projektionen und Verdichtungs- und Detaildarstellungen.
2. *Transparenz* beschreibt die nahtlose Integration in Benutzerumgebungen.
3. Eine offene Architektur gewährleistet *Zugriffsmöglichkeiten* auf heterogene Datenbasen, eingebunden in eine logische Gesamtsicht.
4. Ein *gleichbleibendes Antwortzeitverhalten*, selbst bei vielen Dimensionen und sehr großen Datenvolumina, ist ein wesentlicher Aspekt.
5. Postuliert wird auf Basis einer *Client-Server-Architektur* die Möglichkeit verteilter Datenhaltung sowie der verteilten Programmausführung.
6. Aufgrund der *generischen Dimensionalität* stimmen alle Dimensionen in ihren Verwendungsmöglichkeiten überein.
7. Betriebswirtschaftliche mehrdimensionale Modelle sind oft sehr gering besetzt. Das *dynamische Handling »dünnbesetzter Würfel«* ist elementar für eine optimale physikalische Datenspeicherung.

3. Die zwölf Regeln wurden von Codd, Codd und Sally 1993 postuliert [Codd et al. 1993].

8. Unter *Mehrbenutzerfähigkeit* in OLAP-Systemen wird der gleichzeitige Zugriff verschiedener Benutzer auf die Analysedatenbestände, verbunden mit einem Sicherheits- und Berechtigungskonzept, verstanden.
9. Der Kennzahlenberechnung und Konsolidierung dienen *unbeschränkte dimensionsübergreifende Operationen* innerhalb einer vollständigen integrierten Datenmanipulationssprache.
10. Eine ergonomische Benutzerführung soll *intuitive Datenmanipulation* und Navigation im Datenraum ermöglichen.
11. Auf Basis des mehrdimensionalen Modells soll ein leichtes und *flexibles Berichtswesen* generiert werden können.
12. Die Forderung nach einer *unbegrenzten Anzahl an Dimensionen und Aggregationsebenen* ist in der Praxis schwer realisierbar.

Dieses Regelwerk ist nicht unumstritten und erfuhr verschiedene Erweiterungsvorschläge u. a. von der Gartner Group [Gartner 1995].[4] Eine etwas pragmatischere und technologiefreie Variante zur Definition der konstituierenden Charakteristika von OLAP stammt von Pendse und Creeth, die ihren Ansatz mit FASMI benennen [Pendse/Creeth 1995]:

1. *Fast*: Ganz konkret wird für das Antwortzeitverhalten ein Grenzwert von zwei Sekunden für Standardabfragen und 20 Sekunden für komplexe Analysen festgelegt.
2. *Analysis*: Benutzern muss es ohne detaillierte Programmierkenntnis möglich sein, analytische Berechnungen und Strukturuntersuchungen auf Basis definierter Verfahren und Techniken ad hoc zu formulieren.
3. *Shared*: Für den Mehrbenutzerbetrieb werden Berechtigungsmöglichkeiten bis auf Datenelementebene sowie Sperrmechanismen bei konkurrierenden Schreibzugriffen gefordert.
4. *Multidimensional*: Die mehrdimensionale Sichtweise ist ein elementares Wesensmerkmal analytischer Systeme.
5. *Information*: Für OLAP-Systeme ist die verwaltbare Informationsmenge bei stabilem Antwortzeitverhalten ein kritischer Bewertungsfaktor.

Verschiedene Ansätze zur Definition dessen, was OLAP ausmacht, resultieren in der Anforderung nach Vereinheitlichung und dem Setzen von Standards. Dieses hat sich der OLAP-Council zum Ziel gesetzt.[5] Diese Diskussion ist losgelöst von Implementierungsaspekten, für die es jedoch gleichermaßen verschiedenste Architekturansätze gibt.

---

4. Zur Kritik an den OLAP-Regeln vgl. u. a. [Holthuis 2001]. Zu der Diskussion der Regeln und der Erweiterungen siehe [Gluchowski/Chamoni 2009, S. 202 ff.].
5. Der OLAP-Council wurde 1995 als Informationsforum und Interessenvertretung für OLAP-Anwender gegründet. Eine Zusammenstellung der Definitionen gängiger verwendeter OLAP-Begriffe findet sich auf der Homepage des OLAP-Council.

Online Analytical Processing (OLAP) als Grundprinzip für den Aufbau von Systemen zur Unterstützung von Fach- und Führungskräften in ihren analytisch geprägten Aufgaben basiert im Kern auf einer mehrdimensionalen konzeptionellen Sicht auf die Daten mit Möglichkeiten der Navigation in den Würfeln mit beliebigen Projektionen und auf verschiedenen Verdichtungsstufen.

1.3 Architekturvarianten

Die Architektur von BI-Systemen dient der Beschreibung der wesentlichen Komponenten mit ihren Eigenschaften und Funktionen sowie deren Beziehung untereinander. Dabei sind in der Praxis sehr unterschiedliche Formen und Ausgestaltungen anzutreffen, die nicht immer aufgrund proaktiver Entscheidungen etwa aus einer BI-Organisation heraus entstehen, sondern oft das Ergebnis historisch gewachsener Landschaften sind. Jedoch zeigt sich, dass eine saubere Architektur als Grundlage für die BI-Systeme in Unternehmen Vorteile für Entwicklung und Betrieb mit sich bringt. Dies drückt sich auch in der zunehmend bedeutenden Rolle des BI-Architekten aus.[6]

Bekannte Architekturvarianten unterscheiden sich deutlich hinsichtlich der Anzahl der Komponenten, der gesamten Komplexität, des Aufwands für Entwicklung und Betrieb sowie der Performance und Skalierbarkeit. Aber auch die Fähigkeit, effizient und agil mit neuen Anforderungen umzugehen und den Wandel zu unterstützen, ist ein wesentliches Erfolgskriterium für verschiedene Ansätze.[7]

1.3.1 Stove-Pipe-Ansatz

In den Fällen, in denen im Unternehmen keine übergreifenden Auswertungen erforderlich sind, kann eine dezentrale Architektur mit unabhängigen Data Marts wie in Abbildung 1–2 dargestellt durchaus sinnvoll sein. Bei diesem Ansatz, auch unter dem Namen »Stove Pipe« (Ofenrohr) bekannt [Kimball et al. 2008, S. 249], entstehen einzelne Silos bzw. Inseln, da die Daten für jeden Anwendungsbereich isoliert aus den Quellsystemen extrahiert und aufbereitet werden [Kemper et al. 2010, S. 22 f.]. Dabei erfolgt die Transformation der Daten redundant, was auch zu entsprechenden Aufwänden und potenziellen Inkonsistenzen im Falle einer Änderung führt.

Diese Datensilos sind nur eingeschränkt in einem anderen Kontext nutzbar und bieten damit eine sehr schlechte Unterstützung für bereichsübergreifende Auswertungen. Für autonome Organisationseinheiten kann dieser Ansatz aufgrund der leichteren Berücksichtigung fachlicher Anforderungen jedoch durchaus geeignet sein [Sinz/Ulbrich-vom-Ende 2009, S. 189 f.]. Oftmals handelt es sich aber um eine rein

6. Für eine Übersicht der verschiedenen Architekturvarianten siehe auch [Kemper et al. 2010, S. 21 ff.]. Zu Aspekten der Agilität siehe auch [Göhl/Hahne 2011, S. 12 ff.].
7. Die Übertragung der Methoden agiler Softwareentwicklung auf die Domäne Business Intelligence wird eingehend in [Hughes 2008] diskutiert.

historisch gewachsene Struktur, die den im Zeitablauf gestiegenen Anforderungen nicht mehr gerecht wird.

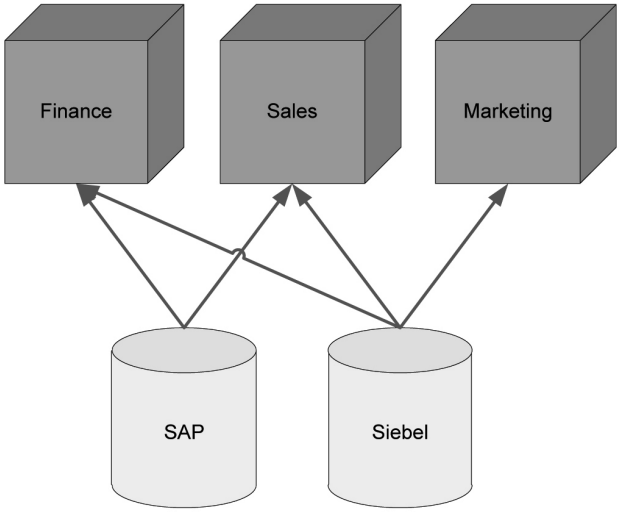

Abb. 1–2 *Unabhängige Data Marts*

Der Übergang von einer derartigen Struktur mit unabhängigen Data Marts zu einer besser geeigneten Architektur mit logischer Integration der Daten gestaltet sich im Allgemeinen recht schwierig, da es auch keine standardisierten bewährten Migrationskonzepte gibt. Dies wird auch durch empirische Untersuchungen untermauert.[8]

Der Stove-Pipe-Ansatz ist oftmals historisch bedingt und liefert keine Integration, sondern isolierte Data Marts.

1.3.2 Data Marts mit abgestimmten Datenmodellen

Eine erste Möglichkeit zur Entschärfung der Probleme des Stove-Pipe-Ansatzes besteht in der Abstimmung der Data-Mart-Datenmodelle bzw. Dimensionsstruktur (*conformed dimensions*, siehe Abb. 1–3). Diese erleichtern die Gewährleistung von Konsistenz und Integrität der dispositiven Daten. Neben den Dimensionen sind auch die Kennzahlen abgestimmt, man spricht hier von *conformed facts*.

Die Abstimmung und der Aufbau eines konsolidierten Datenbestands erfolgt dabei virtuell durch die Abstimmung und Koordination zwischen den Unternehmensbereichen ohne den Aufwand für dessen Entwicklung. Andererseits geht dies einher

8. In einer Studie aus 2006 wurde die Bedeutung und Verbreitung einzelner Architekturformen untersucht. Demzufolge ist die Hub-and-Spoke Architektur mit knapp 40% am weitesten verbreitet, unabhängige Data Marts kamen nur bei gut 10% der befragten Unternehmen zum Einsatz (Ariyachandra/Watson 2006).

mit einem erhöhten Aufwand für die Abstimmung [Sinz/Ulbrich-vom-Ende 2009, S. 191].

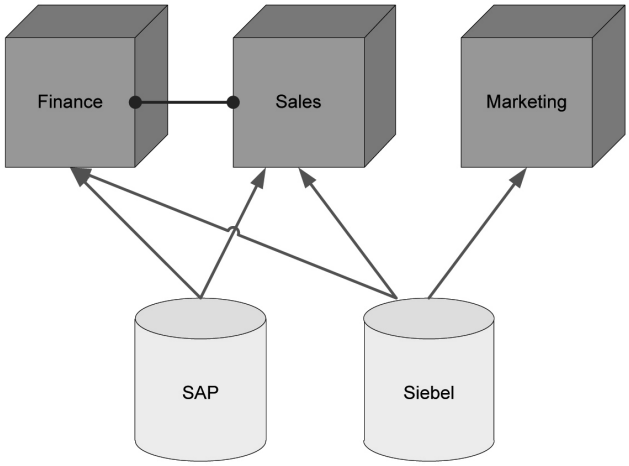

Abb. 1–3 *Abgestimmte Data-Mart-Modelle (Conformed)*

Auch bei dieser Gestaltungsalternative erfolgen die Transformationen redundant, sodass die Risiken möglicher Inkonsistenzen sowie erhöhte Aufwände im Fall einer Änderung bestehen bleiben.

Die Ausprägung der Data Marts geschieht typischerweise kontextbezogen, sodass sich diese hinsichtlich der Granularität in allen Dimensionen unterscheiden. Des Weiteren berücksichtigen diese Data Marts ggf. unterschiedliche betriebswirtschaftliche Anreicherungen und basieren auf verschiedenen Aggregationsniveaus in den Dimensionshierarchien. Somit bleiben übergreifende Auswertungen oftmals mit Informationsverlusten verbunden.

1.3.3 Core Data Warehouse

Sind für die analytischen Anwendungen nur Daten aus einer Anwendungsdomäne zu berücksichtigen, kann der Verzicht auf Data Marts eine Alternative sein. Stattdessen ist ein zentrales Core Data Warehouse aufzubauen, auf dem die Analysen direkt erfolgen. Neben dem Aspekt der Analyse hat ein Core Data Warehouse auch eine Sammel- und Integrationsfunktion. Es gewährleistet dadurch die Qualitätssicherung und hat eine Distributionsfunktion.

Dieser in Abbildung 1–4 visualisierte Architekturansatz stößt hinsichtlich der Anzahl der Benutzer schnell an seine Grenzen und ist kritisch bei einem größeren Datenvolumen, auf dem direkt Auswertungen stattfinden [Sinz/Ulbrich-vom-Ende 2009, S. 188]. Aufgrund der fehlenden anwendungsspezifischen Aufbereitung dispositiver Daten ist dieser Ansatz nicht für verschiedene ggf. zu integrierende Anwen-

dungsdomänen geeignet, da es an der kontextbezogenen Aggregation betriebswirtschaftlicher Daten mangelt.

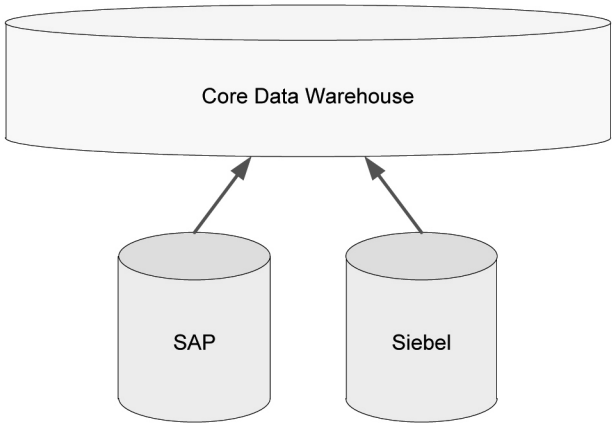

Abb. 1–4 *Zentrales Core Data Warehouse*

Auf Basis der zentralen Architektur eines Core Data Warehouse sind Betrieb und Pflege des Systems zunächst leichter zu realisieren als bei einer Silo-Architektur. Aufgrund der wenigen Komponenten sind auch Änderungen in den Datenstrukturen direkt für alle Anwendungen sichtbar. Dadurch stehen Änderungen im Rahmen eines Change-Prozesses relativ schnell zur Verfügung. Komplexere Lösungen stoßen aber aus Performance- und Administrationsgründen schnell an ihre Grenzen [Kemper et al. 2010, S. 23].

In einem Core Data Warehouse erfolgt die Speicherung dispositiver Daten nach ersten Transformationsschritten der Bereinigung und Harmonisierung für unterschiedlichste Auswertungszwecke für eine Vielzahl von Benutzern und weist daher einen hohen Grad von Mehrfachverwendbarkeit der Daten zusammen mit einer starken Detaillierung auf.

Gerade die Integrationsfunktion eines Core Data Warehouse ermöglicht Analysen auf abgestimmten harmonisierten Datenbeständen. Jedoch stößt dies bei mehreren Geschäftsfeldern mit stark divergierenden Geschäftsprozessen an seine Grenzen, da eine Integration auf allen Ebenen für alle Bereiche oftmals nicht sinnvoll mit vertretbarem Kostenaufwand realisierbar ist.

In diesen Fällen, in denen einzelne Geschäftseinheiten durch unterschiedliche Produkt- oder Marktstrukturen gekennzeichnet sind, ist der Einsatz mehrerer autarker Core Data Warehouses sinnvoll, die jeweils auf die Anforderungen einer strategischen Einheit fokussieren. Dies ist exemplarisch in Abbildung 1–5 dargestellt.

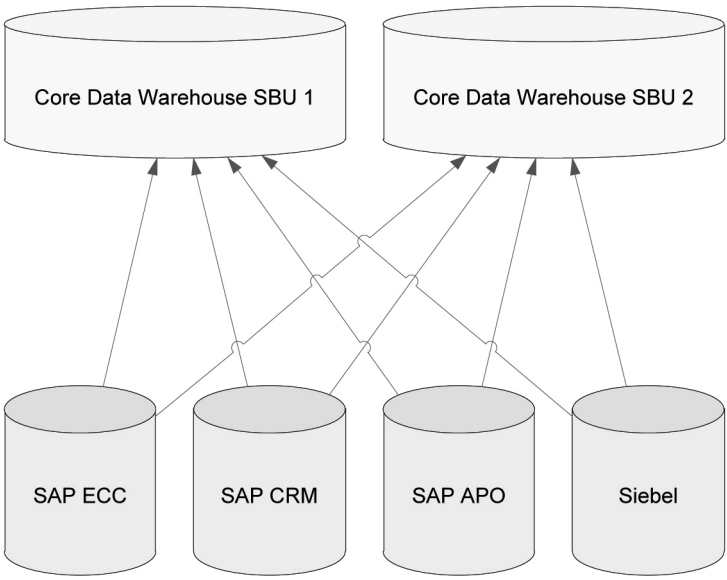

Abb. 1–5 *Mehrere Core Data Warehouses*

Eine solche Architektur mit mehreren Core Data Warehouses findet sich typischerweise bei Konzernen und spartenorientierten Unternehmen.

In Konzernen und Unternehmen mit sehr unterschiedlichen Geschäftsprozessen sind oftmals einzelne Core Data Warehouses für jede strategische Geschäftseinheit vorzufinden.

1.3.4 Hub-and-Spoke-Architektur

Ein Core Data Warehouse ist eine zentrale Architekturkomponente in vielen Varianten von BI-Architekturen, da es sehr gut geeignet ist, um die folgenden Funktionen zu erfüllen [Kemper et al. 2010, S. 39 f.]:

- Sammlung und Integration von dispositiven Daten
- Distribution, also Verteilung der abgestimmten Daten an nachgelagerte Komponenten wie etwa Data Marts
- Qualitätssicherung, da die syntaktische und semantische Stimmigkeit der dispositiven Daten durch die Integration und Harmonisierung gesichert ist

Die Distributionsfunktion kommt insbesondere in der um Data Marts erweiterten Architektur zum Tragen, denn aus dem Core Data Warehouse erfolgt die Bewirtschaftung der Data Marts auf Basis geeigneter Aggregations- und Transformationsprozesse. Oftmals wird diese Form daher auch als Hub-and-Spoke-Architektur bezeichnet.

Die Data Marts sind dabei im Regelfall immer noch unterschiedlich hinsichtlich ihrer Granularität[9] in allen Dimensionen, der Verwendung unterschiedlicher Formen der Aggregation und Nutzung verschiedener Dimensionshierarchien sowie auch bezogen auf die betriebswirtschaftliche Anreicherung. Da sich die Data Marts aus dem Core Data Warehouse ableiten, wird auch von abhängigen Data Marts gesprochen (vgl. [Gluchowski et al. 2008, S. 129f.]).

Die Vorteile einer solchen Architektur können direkt aus Abbildung 1–6 abgeleitet werden, denn es treten viel weniger Schnittstellen auf, sodass die Logik zur Transformation nicht redundant vorzufinden ist. Ein weiterer Vorteil der Architektur liegt in der zentralen Datenintegration und Aufbereitung. Im Wesentlichen ist dies auch die Grundlage des Architekturverständnisses nach Inmon mit der Corporate Information Factory (CIF) (siehe Abschnitt 1.3.6 sowie [Inmon et al. 2001]).

In einer Hub-and-Spoke-Architektur dient das Core Data Warehouse als Hub und erfüllt die Aufgabe der Integration, Qualitätssicherung und Datenverteilung an die Data Marts als Spokes, die einen hohen Grad an Anwendungsorientierung und vordefinierte betriebswirtschaftliche Anreicherungen und Aggregationen aufweisen.

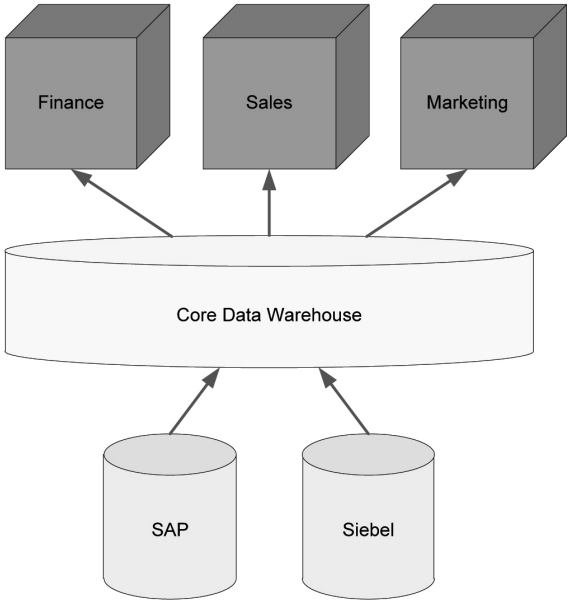

Abb. 1–6 *Core Data Warehouse mit abhängigen Data Marts*

9. Zum Begriff der Granularität siehe auch [Hahne 2005, S. 23].

In der Praxis findet sich jedoch häufig ein Mix verschiedener Architekturansätze wie in Abbildung 1–7 exemplarisch dargestellt, der teilweise historisch entstanden oder aber das Ergebnis bewusster Gestaltung sein kann.

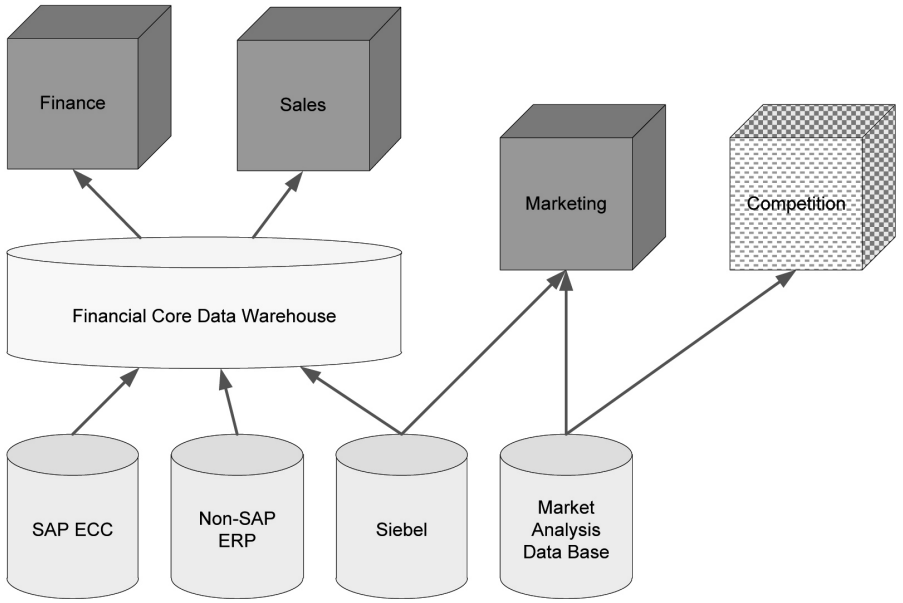

Abb. 1–7 *Gemischte Architektur*

In dem Beispiel in Abbildung 1–7 ist erkennbar, dass teilweise auch virtuelle Data Warehouses eingesetzt werden. Diese ermöglichen einen direkten Zugriff auf die Daten des Core Data Warehouse. Dieser Aspekt wird zunehmend kontrovers diskutiert, und die Frage, ob das Core Data Warehouse direkt abgefragt werden darf, ist nicht eindeutig zu beantworten. Aufgrund der negativen Erfahrungen der Vergangenheit findet dieser Ansatz aber immer weniger Zuspruch.

Grenzen bei der Analyse direkt auf dem zentralen Datenbestand ergeben sich unter anderem durch die zunehmend großen Datenvolumina und die Gefahr von Abfragen, die sehr langsam sind und damit das System sehr stark beanspruchen.

1.3.5 Data-Mart-Busarchitektur nach Kimball

In der Sichtweise nach Kimball sollte ein Core Data Warehouse dimensional modelliert sein [Kimball/Ross 2002, S. 10 ff.]. Er nennt es Dimensional Data Warehouse. Es handelt sich hierbei um ein Repository, das sehr wohl für Auswertungen genutzt werden soll bzw. kann. Die einzelnen Data Marts dieser sogenannten Data-Mart-Busarchitektur werden auch Subject Areas genannt.

Bei dieser Architektur gibt es demzufolge ein zentrales Repository, das wie in Abbildung 1–8 verdeutlicht, zwar im Sinne eines Hubs die Data Marts bedient, jedoch selbst schon ein mehrdimensionales Modell der integrierten atomaren Daten darstellt. Daher auch der Name Dimensional Data Warehouse für das Repository, das ebenfalls atomare Granularität aufweist.

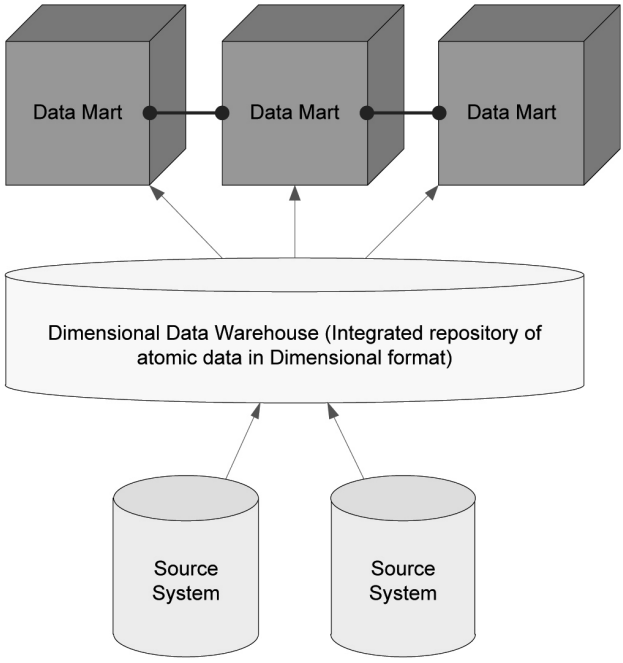

Abb. 1–8 *Data-Warehouse-Architektur nach Kimball [Kimball 2002]*

Wichtig ist in dieser Sichtweise, dass alle Modelle ausschließlich in dimensionaler Form vorliegen und die Auswertungen auf diesem zentralen Repository stattfinden. Zusätzliche Data Marts müssen nicht notwendigerweise persistiert vorliegen, eine Speicherung ist also optional, die Modelle sind aber alle untereinander abgestimmt und nutzen das Prinzip der »conformed dimensions and facts« im Sinne einer *Enterprise Bus Architecture* (siehe [Kimball et al. 2008, S. 114 ff.]).

In einer Architektur nach Kimball [Kimball 2002] sind alle Modelle dimensional strukturiert, insbesondere auch das Core Data Warehouse als zentrales granulares Repository.

1.3.6 Corporate Information Factory nach Inmon

Während nach Kimball die dimensionale Modellierung für das gesamte Data Warehouse verpflichtend ist, sieht Inmon deren Nutzen nur auf der Ebene der Data Marts, für die er auch immer den dimensionalen Ansatz empfohlen hat (»… if I had

to design a data mart tomorrow, I would not consider using any other approach« [Inmon 2000].).

Es sind aber nur die Departmental Data Marts, für die der Ansatz der dimensionalen Modellierung zum Tragen kommt. Diese implementieren ja gerade die abteilungs- oder funktionsbezogene Sichtweise für die Endbenutzerzugriffe. Dabei basieren die zwingend physisch gespeicherten Data Marts auf einheitlichen Strukturen zur Aggregation und Anreicherung (vgl. [Inmon et al. 2001, S. 110 ff.]).

Eine zentrale Komponente der Corporate Information Factory (CIF) ist das Enterprise Data Warehouse, ein auf einem normalisierten Modell basierendes zentrales Repository von granularen Daten, das als Basis im Sinne eines Hubs innerhalb einer Hub-and-Spoke-Architektur fungiert und auch nicht für direkte Analysen des Endanwenders genutzt werden soll (siehe Abb. 1–9). Die dimensionale Modellierung ist nach Inmon für diesen wie auch für die meisten anderen Bereiche in der Architektur ungeeignet (vgl. [Inmon et al. 2008, S. 18 ff.]).

In der Architektur nach Inmon [Inmon 2008] kommt die dimensionale Modellierung nur für die Data Marts zum Einsatz. Das Core Data Warehouse ist normalisiert modelliert und soll eine möglichst flexible granulare Basis darstellen.

Das Architekturverständnis von Inmon erfuhr im Zeitablauf eine Erweiterung u. a. um Aspekte des Umgangs mit unstrukturierten bzw. semistrukturierten Daten. Diese Architektur wurde als DW 2.0 eingeführt [Inmon et al. 2008].

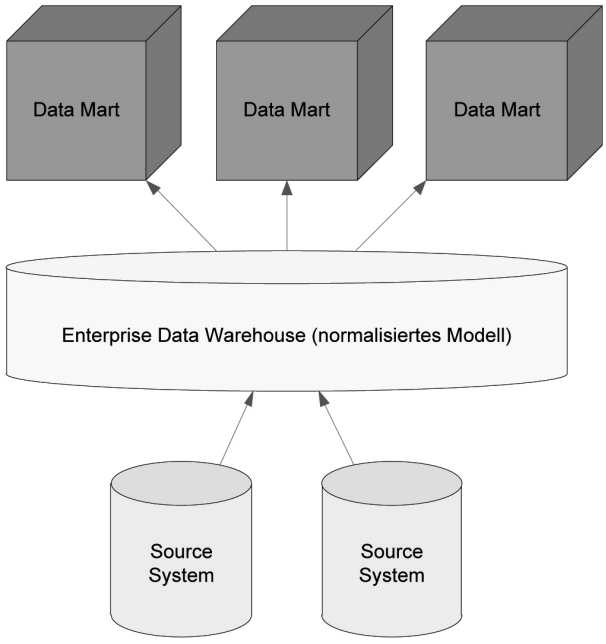

Abb. 1–9 *Corporate Information Factory (CIF) nach Inmon bzw. DW-2.0-Architektur*

Aspekte des sogenannten *Cross Media Storage Manager* und des *Nearline Storage* wurden zu einem umfassenden Konzept verschiedener Sektoren zur Speicherung ausgebaut, die auf einem Konzept des Information Lifecycle Management (ILM) basieren (siehe hierzu auch [Hahne 2007] sowie [Hahne 2011]). Durch die Einführung verschiedener Sektoren kann den unterschiedlichen Anforderungen an die Service Levels der Daten im Laufe des Informationslebenszyklus besser Rechnung getragen werden. Neben dem Onlinesektor treten der Nearlinesektor und der Archivsektor hinzu. Aspekte des Information Lifecycle Management im BI-Kontext gehen auf die Inmon-Architektur zurück [Inmon 2008, S. 55 ff.].

1.3.7 Architekturvergleich Kimball und Inmon

Obwohl die öffentliche Diskussion, sicherlich auch durch die beiden Protagonisten selbst bewusst verursacht, sehr kontrovers und polarisierend geführt wird, haben beide Vorstellungen von einer guten Data-Warehouse-Architektur auch Gemeinsamkeiten. Die beiden Architekturvarianten mit ihren zentralen Eigenschaften sind in der Tabelle 1–1 gegenübergestellt (vgl. [Adamson 2010, S. 24 ff.]).

Architektur	Begriffe	Charakteristika	Mehrdimensionalität
Inmon	Corporate Information Factory (CIF), Enterprise Data Warehouse (EDW)	EDW ist integriert und atomar. EDW ist nicht im direkten Zugriff. Data Marts basieren auf EDW und sind physisch getrennt.	Nur für Data Marts.
Kimball	Dimensional Data Warehouse, Data-Mart-Bus-Architektur	Dimensional Data Warehouse: integriert auf Basis von »conformed dimensions« im Star-Schema. Subject Areas im Dimensional Data Warehouse heißen Data Marts. Data Marts müssen nicht separat sein.	Alle Daten sind mehrdimensional organisiert.

Tab. 1–1 *Gegenüberstellung von Kimball- und Inmon-Architektur*

Zu den Gemeinsamkeiten gehört auch die Einsicht, dass eine logische Integrationsschicht sinnvoll und notwendig ist. In beiden Sichtweisen ist diese Anforderung über ein zentrales Repository abgedeckt. Des Weiteren soll dieses in beiden Fällen atomare Granularität aufweisen. Unterschiedlich ist hingegen die Speicherform dieses zentralen Datenpools. Während in der Kimball-Architektur die dimensionale Modellierung ein zwingendes Kriterium ist, sieht Inmon hier im Wesentlichen normalisierte Modelle vor. Auch der Zugriff, der auf das Dimensional Data Warehouse nach Kim-

ball der Normalfall ist, wird im Fall des Core Data Warehouse nach Inmon im Regelfall ausgeschlossen, da dies durch die dahinterliegenden Data Marts abgedeckt ist.

Ein weiteres Unterscheidungsmerkmal ist die Persistenz der Data Marts. Denn in der Kimball-Sichtweise ist deren physische Speicherung optional. Bei Inmon ist die Persistierung der Data Marts immer vorgesehen, um schlechte Performance bei Zugriffen auf das Core Data Warehouse per se auszuschließen.

1.4 Schichtenmodell der BI-Architektur

Heutige Data-Warehouse-Architekturen sind in mehrfacher Hinsicht mit neuen Anforderungen konfrontiert, die eine klare Architektur sowie adäquate Methoden des Entwurfs fordern. Neben den klassischen Erwartungen an die Kostenstrukturen für den Aufbau und den Betrieb von BI-Systemen sind zunehmend gerade die geschäftlichen Anforderungen, sogenannte Business Requirements, die aufgrund der stetig zunehmenden Dynamik des Geschäftslebens einer starken Volatilität unterworfen sind, wesentliche Treiber für neue BI-Projekte. Gefordert sind Architekturen und Konzepte, die ein schnelles Reagieren auf neue Herausforderungen ermöglichen und somit die Time-to-Market von BI-Applikationen drastisch senken. Die konzeptionelle Antwort hierauf sind mehrschichtige Data-Warehouse-Architekturen, die als logische Klammer zu verstehen sind. Diese implizieren in der Konsequenz unterschiedliche Modellierungsansätze für die differenzierten Anforderungen der diversen Ebenen innerhalb solcher Strukturen. Eine zentrale Rolle für die Gewährleistung der notwendigen Flexibilität, um auf neue geschäftliche Anforderungen schnell reagieren zu können, spielt dabei ein zentrales unternehmensweites Data Warehouse, ein *Enterprise Data Warehouse* (EDW), das einerseits die vollständige Historie abbildet und andererseits auch die Mechanismen zur Historisierung implementiert. Dies gewährleistet, auf Basis eines abgestimmten harmonisierten Data-Warehouse-Datenbestands, neue Anforderungen in sogenannten Data Marts on Demand schnell erfüllen zu können.

Das heutige Wirtschaftsleben ist gekennzeichnet durch eine hohe Komplexität und rasche Veränderungen der allgemeinen Rahmenbedingungen, die die Marktteilnehmer mit neuen Herausforderungen des Informationsmanagements konfrontieren. Dabei rücken dispositive und strategische Aspekte zunehmend in die Betrachtung. Business Intelligence und Data Warehousing sind dabei die Konzepte, die eine unternehmensweite Informationsversorgung für diese Zwecke realisieren sollen.

Damit sind aber auch Probleme hinsichtlich der Konsistenz und Flexibilität verbunden, die sich in unkontrollierten Datenflüssen, wiederholten und redundanten Extraktionen der gleichen Daten, vielfältigen inkonsistenten Datenmodellen, hohen Entwicklungskosten, Einschränkungen bei der Erfüllung von Informationsbedürfnissen, Informationsinseln und einer insgesamt eher unzuverlässigen unternehmensweiten Datenbasis niederschlagen. Zu allem Übel stehen die IT-Abteilungen vor einem Flickenteppich aus Werkzeugen, der hohe Pflege- und Integrationsanstrengungen sowie ständige Nacharbeit fordert. Dies treibt nicht nur die Gesamtkosten in die

Höhe, sondern beeinflusst auch deutlich die Flexibilität im Unternehmen – mithin entscheidend für die Wettbewerbsfähigkeit.

Eine zukunftsorientierte Architektur für die dispositive Informationsversorgung ist nach heutigem allgemeinem Verständnis im Wesentlichen durch eine mehrschichtige unternehmensweit ausgerichtete Data-Warehouse-Struktur gegeben (vgl. [Haupt/Hahne 2007] und [Hahne/Böttiger 2006]). Demzufolge sind die in Abbildung 1–10 dargestellten Schichten zu differenzieren. Im Acquisition Layer erfolgt die Aufnahme der untransformierten Rohdaten aus den Quellsystemen und den externen Datenquellen. Ziel der Transformations- und Harmonisierungsprozesse ist die Ablage im Integration Layer, in dem die aufbereiteten Daten in ihrer vollständigen Historie vorliegen. Die Aggregation hinsichtlich der betriebswirtschaftlichen Anforderungen erfolgt auf dem Weg in den Reporting Layer (vgl. [Hahne 2011, S. 60 ff.]).

In der Praxis haben sich verschiedenste Varianten mehrstufiger Architekturen ausgebildet. Je nach Art und Komplexität der Aufgaben der Harmonisierung und Transformation kommen durchaus sehr viele unterschiedliche Stufen der Datenveredelung zum Einsatz. Insofern kann bei Multi-Layer-Architekturen von einem logischen Konzept gesprochen werden, das jeweils unternehmensindividuell auszuprägen ist.

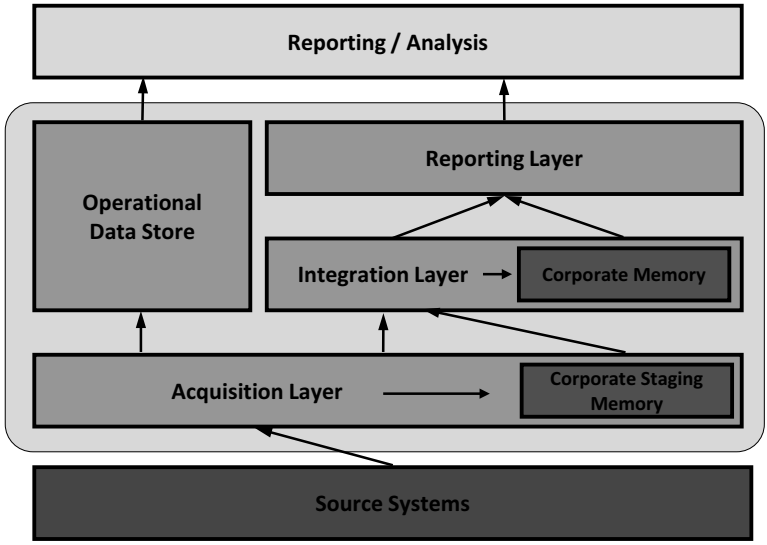

Abb. 1–10 *Konzeptionelle Multi-Layer-Architektur*

Neben dem Bereich des Acquisition Layer als erste Schicht im Data Warehouse werden die Schritte der Datenaufbereitung im Integration Layer auch gerne in eigene Schichten mit dedizierten Aufgaben unterteilt. Hierzu gehören die unterschiedlichen Stufen der Harmonisierung. Die Anwendung von Businesslogik erfolgt dann zumeist auf dem Weg Richtung Reporting Layer auch wieder über ggf. mehrere Schichten.

1.4.1 Acquisition Layer

Im Allgemeinen wird die Staging Area als Ort der temporären Ablage für extrahierte Daten verstanden, die nach erfolgter weiterer Verarbeitung und deren Qualitätssicherung zu löschen sind. Dies trägt aber nicht dem Umstand Rechnung, dass oftmals Daten aus Quellsystemen erneut zu extrahieren sind, wenn sich an der weiteren Logik der Aufbereitung etwas geändert hat oder gar Fehler auftraten. Zwar bieten Quellsysteme die Option erneuter Extraktion auch schon gelieferter Daten, jedoch impliziert dies zwei grundsätzliche Probleme. Einerseits halten operative Systeme keine langfristige Historie vor, sodass nur auf relativ aktuelle Daten zurückgegriffen werden kann. Andererseits ist gerade die erneute Extraktion großer Datenvolumina ein Problem der Laufzeit, sodass diese Option eher theoretischer Natur ist.

Eine zentrale Aufgabe der Staging Area in einem Data Warehouse ist daher, die komplette Extraktionshistorie in Form des Corporate Staging Memorys zur Verfügung zu stellen. Prinzipien des Datenmanagements ermöglichen dabei die effiziente Ablage dieser Daten »auf Vorrat« in günstigeren Umgebungen (z. B. in einem Nearline Storage[10]), sodass das Corporate Staging Memory nicht unbedingt zu den gleichen Kosten wie die aktuellen Data-Warehouse-Datenbereiche betrieben wird. Die Datenrepräsentationsform ist für diesen Bereich im Allgemeinen relationaler Art und die Modellierung erfolgt in normalisierter Form auf der konzeptionellen Ebene z. B. auf Basis eines Entity-Relationship-Modells (ERM). Endbenutzer haben auf die Staging-Daten im Regelfall keinen Zugriff, denn dieser liegt allein in der Hoheit der ETL-Prozesse [Kimball/Caserta 2004, S. 31 f.].

In Abbildung 1–11 ist der Datenfluss in Richtung Acquisition Layer in der Übersicht dargestellt. Für jede Datenquelle und jede Lieferstruktur gibt es ein Pendant im Staging-Bereich, das den angelieferten Strukturen entspricht. Der gesamte Prozess wird begleitet durch entsprechende Metadaten, die Informationen über Lieferungen beinhalten und beschreiben, was wann geladen wurde. Dieser Layer in der Gesamtarchitektur kann auch außerhalb des eigentlichen Data Warehouse in vorgelagerten Systemen abgebildet sein.

10. Zum Begriff Nearline Storage siehe auch [Hahne 2007] und [Haupt/Hahne 2007].

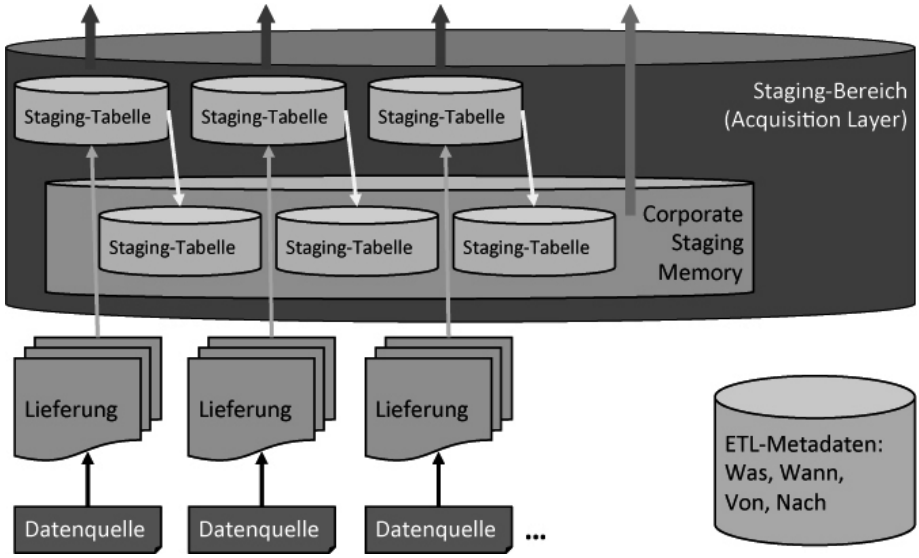

Abb. 1–11 *Datenfluss im Staging-Prozess*

Lieferstrukturen sind im Allgemeinen durch Feldlisten und deren Formate spezifiziert. Jede Lieferung wird mit einem eindeutig identifizierbaren Schlüssel versehen und im Staging-Bereich in der korrespondierenden Tabelle gespeichert. Die langfristige Ablage erfolgt dabei im Corporate Staging Memory, das als logisches Konzept zu verstehen ist und durchaus auch auf der gleichen Tabelle implementiert sein kann. Die Formen der Implementierung variieren dabei je nach eingesetzter Technologie. Dieser Zusammenhang wird in Abbildung 1–12 dargestellt.

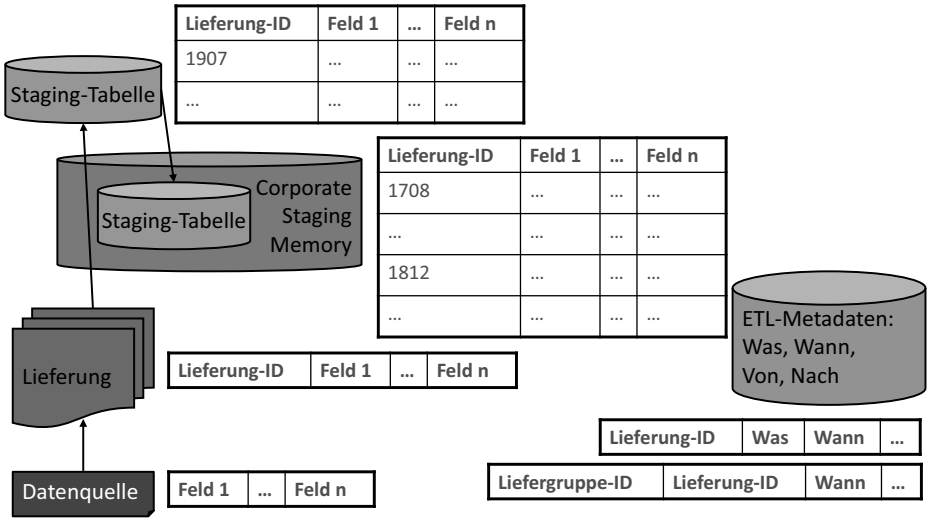

Abb. 1–12 *Staging-Bereich im relationalen DBMS*

1.4.2 Integration Layer

Erwähnenswert ist ebenfalls der Umstand, dass die Aufgaben der Transformation und Harmonisierung erst nach dem Staging-Bereich stattfinden, auf dessen Basis über Transformationsprozesse Lieferungen für Stammdaten und Bewegungsdaten erzeugt werden, die in die nächste Ebene weitergereicht werden (siehe Abb. 1–13). Hierbei erfolgt die Verknüpfung diverser Extraktstrukturen, die auf Basis entsprechender Logiken zusammenzuführen sind.

Eine zentrale Aufgabe der Transformation von Lieferungen besteht in der Historisierung von Stammdaten, um damit alle Szenarien des Umgangs mit strukturellen Veränderungen zu ermöglichen.[11] Die vollständig historisierten Daten liegen im Integration Layer im Allgemeinen in themenbezogener übergreifender Form vor, deren Gestaltung auf Basis eines ERM erfolgen kann. Die hierbei auftretenden Modelle sind normalisiert und haben einen unternehmensweiten Fokus. Ein weiterer für diese Ebene relevanter Aspekt ist das Delta-Handling, also der Umgang mit neuen Daten und deren Weiterverarbeitung.

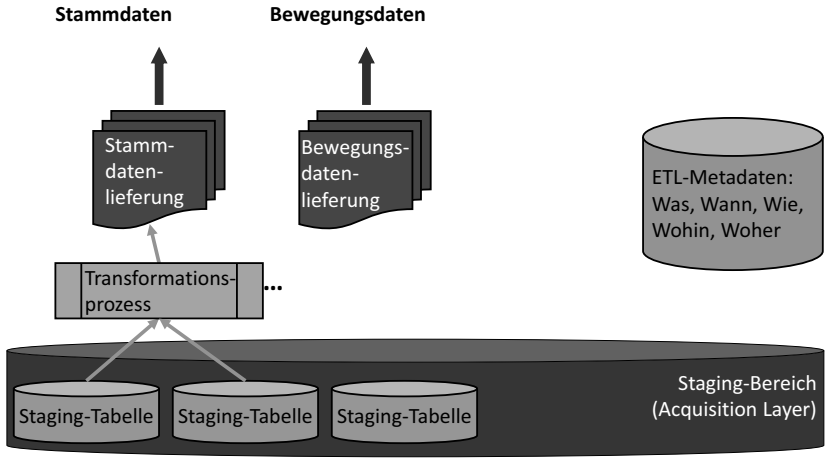

Abb. 1–13 *Transformation und Integration im EDW*

Ziel der Integrations- und Harmonisierungsschritte ist die konsistente Ablage im zentralen Integration Layer. Konzeptionell entspricht diese Ebene dem Begriff des Core Warehouse und auch dem des Enterprise Data Warehouse. [12]

Wie in Abbildung 1–14 verdeutlicht, hat auch der Integration Layer einen Teilbereich, in dem die langfristige Ablage der integrierten Daten erfolgt. Der Teilbereich der langfristigen Speicherung wird oftmals auch als Corporate Memory bezeichnet

11. Zu den Aspekten der Zeitabhängigkeit und Historisierung vgl. [Chamoni/Stock 1998], [Hahne 2003a] sowie [Hahne 2005].
12. Teilweise wird diese Ebene auch als Basisdatenbank bezeichnet, vgl. dazu [Bauer/Günzel 2013, S. 51 ff.].

und stellt den Pool der kompletten Historie aufbereiteter Geschäftsdaten dar. Um dem Aspekt des Information Lifecycle Rechnung zu tragen, kann dieses sehr gut in einem Nearline Storage abgelegt werden.[13] Form und Ausprägung dieser hinsichtlich Kosten und Effizienz optimierten Ablageform variieren dabei sehr stark je nach eingesetzter Technologie, auf deren Basis das Core Warehouse implementiert wird.

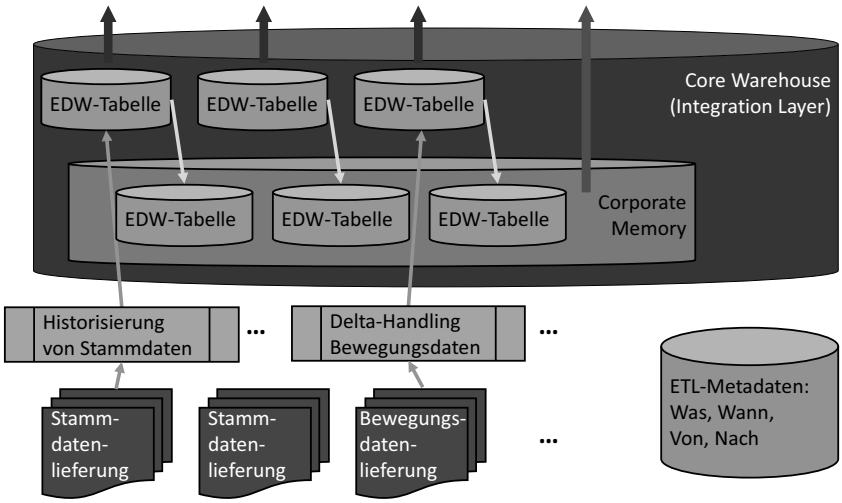

Abb. 1–14 *Historisierung und Delta-Handling für das EDW*

1.4.3 Reporting Layer

Ein wesentlicher Vorteil des Aufbaus mehrschichtiger Architekturen und eines Enterprise Data Warehouse liegt in der dadurch entstandenen Flexibilität, mit der auf neue geschäftliche Anforderungen an BI-Anwendungen reagiert werden kann. Auf Basis der im Integration Layer vorgehaltenen Daten können neue Data Marts recht flexibel aufgebaut werden, um neue Anforderungen umzusetzen. Sind andere Data Marts mit der Zeit obsolet, können diese auch problemlos einfach gelöscht werden. Die komplette Historie der Daten findet sich ja bereits im Integration Layer, sodass keinerlei Datenverluste damit verbunden sind. Wie in Abbildung 1–15 dargestellt erfolgt der Aufbau der Data Marts durch Prozesse der Aggregation und betriebswirtschaftlichen Aufbereitung der im Integration Layer vorhandenen Daten. Dabei spielt es keine Rolle, ob diese im Bereich der aktuellen Daten liegen oder aus dem Pool des Corporate Memory zu beziehen sind.

Die Modellierung der üblicherweise mehrdimensional strukturierten Data Marts erfolgt sehr nah am Fachanwender, sodass hierbei die Methoden der semantischen

13. Der Begriff Nearline Storage geht zurück auf die Corporate Information Factory (CIF) nach Inmon [Inmon 2001].

mehrdimensionalen Modellierung, wie beispielsweise ADAPT[14], anzuwenden sind. ADAPT wird in Kapitel 3 detailliert diskutiert.

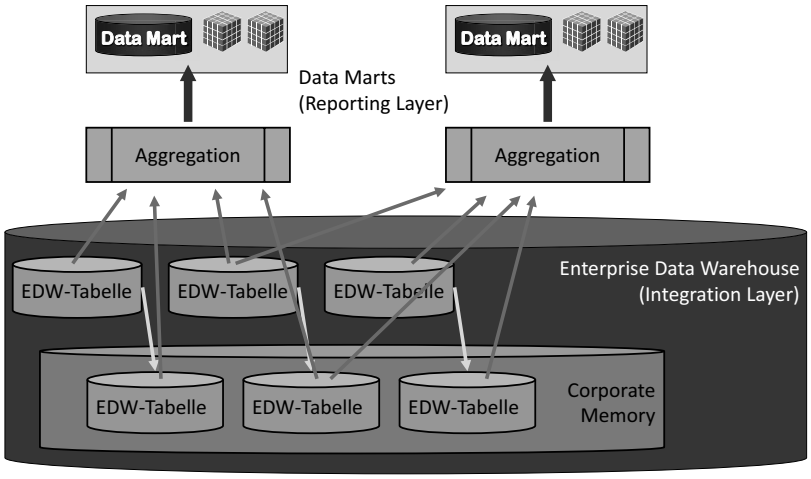

Abb. 1–15 *Nutzung des EDW für Data Marts on Demand*

1.4.4 Modellierung im Schichtenmodell

Zusammenfassend können die folgenden Methoden der Modellierung den unterschiedlichen Ebenen im Kontext eines Data Warehouse zugeordnet werden [Hahne 2011, S. 63]:

- Quellsysteme:
  - Normalisierte Modellierung z. B. in Form eines ERM
  - Strukturierung nach Modulen, Funktionen oder Prozessen
  - Transaktionsorientiert
- Acquisition Layer:
  - Normalisierte Modellierung etwa auf Basis des ERM-Ansatzes
  - Strukturierung entlang der Extrakt-Strukturen
  - Langfristige Speicherung der Extraktionshistorie

14. ADAPT steht für Application Design for Analytical Processing Technologies und wurde 1996 von Dan Bulos als konzeptionell ausgerichtete Modelliermethode für mehrdimensionale Datenstrukturen vorgeschlagen (vgl. hierzu [Hahne 2010] sowie Kapitel 3).

- Integration Layer (Core Warehouse, EDW):
  - Im Allgemeinen normalisierte Modellierung auf Basis eines ERM
  - Unternehmensweite Ausrichtung
  - Themenbezogene übergreifende Strukturierung
  - Langfristige Speicherung der kompletten Historie
  - Vollständige Historisierung
- Reporting Layer (Data Marts):
  - Mehrdimensionale Modellierung
  - Auswertungsbezogene Strukturierung

Somit stehen für jede Ebene entlang der Aufbereitung der Daten von der Quelle bis zur Auswertungsebene in den Data Marts dedizierte Methoden der Modellierung zur Verfügung, die den unterschiedlichen Anforderungen der einzelnen Layer in der Architektur gerecht werden.

Systeme, die nach dem Grundprinzip des mehrschichtigen Aufbaus gestaltet sind, haben zum einen den Vorteil, dass sie den Aspekt der Flexibilität voll berücksichtigen und damit das, was als »anticipating the unknown« beschrieben wird, leichter ermöglichen. Den neuen geschäftlichen Anforderungen kann dadurch schnell und flexibel begegnet werden. Zum anderen kommt aber noch ein weiterer Vorteil hinzu, der erst bei Betrachtung des Betriebs eines Data Warehouse und den damit zusammenhängenden Change-Prozessen deutlich wird. Die Komplexität im Betrieb eines Data Warehouse steigt überproportional mit der Anzahl genutzter Applikationen und den abzubildenden Datensträngen. Eine saubere Architektur reduziert den durch neue Applikationen oder Datenbereiche hinzukommenden Aufwand nicht nur bei der Erstellung neuer Applikationen oder Wartung bestehender Anwendungen, sondern auch bei deren Betrieb. Oftmals sind SLAs (Service Level Agreements) erst durch derartige klare Strukturen überhaupt zu gewährleisten.