Data Warehousing mit ETL Tools

Positionierung
Funktionsbereiche
Architekturkonzepte

Bearbeiter/in:
Regine Stopka
Markus Alig
Laci Hohner

Basel, 17. Dezember 2001

Inhaltsverzeichnis

1. Data Warehousing mit ETL Tools

Für den Datenaquisitionsprozess zur Erstellung eines Data Warehouses werden mittlerweile vielerorts Tools eingesetzt. Die Anbieter dieser Tools versprechen, dass das gewünschte Warehouse innerhalb kürzester Zeit und mit minimalem Aufwand erstellt werden kann. Doch spiegelt dieses Versprechen auch tatsächlich die Realität wieder? Inwieweit kann tatsächlich die Entwicklungszeit für ein Data Warehouse verkürzt werden, und in welchem Masse wird tatsächlich Aufwand reduziert? Fragen wie diese stellt sich mancher, der vor dem Aufbau eines Data Warehouses und dem Einsatz eines ETL Tools steht.

Zu Beginn dieses Papieres wird zunächst kurz beschrieben, welche Überlegungen überhaupt dazu führten, die Idee eines Data Warehouses ins Leben zu rufen. Danach folgt eine Positionierung der ETL Tools innerhalb einer Data Warehouse Architektur. Im Anschluss daran werden die Funktionsbereiche dieser Tools skizziert. Diese Bereiche stellen die Grundlagen für das Folgende dar: Exemplarisch werden zwei Data Warehouse Architekturen beschrieben, wie sie in der Praxis erfolgreich implementiert und betrieben werden. Abschliessend wird auf einige wenige Punkte eingegangen, die bei einer ETL Tool Evaluation zu beachten sind.

An dieser Stelle sei ausdrücklich darauf hingewiesen, dass die vorliegende Dokumentation keineswegs ein Leitfaden zur Erstellung eines Data Warehouses darstellt. Zwar werden viele relevante Aspekte zum Thema Data Warehousing gestreift, da sie Implikationen auf den Einsatz eines ETL Tools haben. Keinesfalls jedoch werden sie erschöpfend behandelt. Insofern wird für dieses Dokument lediglich ein einzelner Aspekt des Themenkomplexes Data Warehousing herausgegriffen und näher beleuchtet.

Eine intensive Diskussion aller Datenmanagementaspekte findet sich z.B. in [Dippold/Meier/Ringgenberg/Schnider/Schwinn 2001].

2. Der Nutzen analytischer Systeme

Nicht selten scheitern grosse Data Warehouse Projekte oder führen zu sehr hohen Kosten, die dann in keinem Verhältnis zum gewonnen Nutzen stehen. In diesen Fällen liegt die wichtigste Ursache oft darin, dass der Aufbau eines Data Warehouses in erster Linie als technisches Projekt betrachtet wird, das nicht die notwendige Unterstützung im Business geniesst. Selbst wenn bei solchen Projekten in aller Regel umfangreiche Investitionen in Hardware und Software getätigt werden müssen, muss man sich stets bewusst sein, dass ein Data Warehouse Projekt mittel- oder langfristig nur erfolgreich sein wird, wenn damit auch ein Nutzen erzielt werden kann.

In einem analytischen System entsteht der entscheidende Nutzen dadurch, dass aus Daten, die in operativen Systemen nicht in geeigneter Form für Auswertungen verfügbar sind, Informationen gewonnen werden können. Informationen aus den verschiedensten Datenquellen entlang der ganzen Wertschöpfungskette im Unternehmen werden also in einer Weise zur Verfügung gestellt, die es ermöglicht, Entscheidungen auf den unterschiedlichsten Ebenen eines Unternehmens schnell und zielgerichtet zu treffen. Dabei werden Informationen über Kunden, Partner, Lieferanten, Lagerbestände, Produkte, Verkäufe, Transaktionen usw. in einem analytischen System in einer integrierten Sicht abgebildet und dies unabhängig von der technischen Plattform, von der die Daten stammen.

Die Integration von Informationen erlaubt es z.B., die folgende Fragen zu stellen und auch zu beantworten:

- Welche Kunden werfen die grössten Erträge ab?
- Welche Kunden haben Potential zu Ertragssteigerungen?
- Welche Produkte werden zukünftig von einer bestimmten Kundengruppe mit hoher Wahrscheinlichkeit in Anspruch genommen?
- Wo liegt das Potential für Kostenreduktionen?

Zusammenfassend kann festgestellt werden, dass mit Hilfe analytischer Systeme eine möglichst breite Informationsbasis zur Verfügung gestellt werden soll, um die Gewinnung von Wettbewerbsvorteilen zu ermöglichen.

3. Positionierung der ETL Tools in der Gesamtarchitektur

Wie gerade eben ausgeführt, ist es für jedes Unternehmen von strategischer Bedeutung, über eine breite Datenbasis mit analytischen Informationen verfügen zu können. Nach deren Erstellung muss diese in der Folge laufend aktualisiert werden, wobei der Datenqualität ein besonderes Augenmerk geschenkt werden sollte. Insbesondere in grösseren Unternehmen ist dies ein Unterfangen, das mit hohen Kosten verbunden ist.

Ein Grundsatz, der sich in diesem Zusammenhang bewährt hat, ist der folgende: ‚Think big, start small'. Zunächst werden die Daten aus einem betriebswirtschaftlichen Teilbereiche wie bspw. dem Marketing in der gewünschten Form zur Verfügung gestellt. Erst wenn dieser Prozess abgeschlossen ist, wird der nächste Teilbereich in Angriff genommen. Die Bereitstellung der Daten erfolgt also im Zuge eines iterativen Prozesses.

Nur auf diese Weise kann gewährleistet werden, dass innerhalb eines überschaubaren Zeitrahmens das gewünschte Ergebnis erreicht wird. Darüber hinaus ist diese Vorgehensweise ein Garant für die Flexibilität und Dynamik, die notwendig ist, um auch auf kurzfristige Bedürfnisse reagieren zu können.

Neben all den wichtigen Komponenten, die es bei der Realisierung des Gesamtprojektes zu beachten gilt, ist die Wahl und der Einsatz eines professionellen ETL Tools anstelle von konventioneller Programmierung von besonderer Bedeutung. Nachfolgende Graphik zeigt auf, wo das ETL Tool im Rahmen des technischen Umfeldes zu positionieren ist.

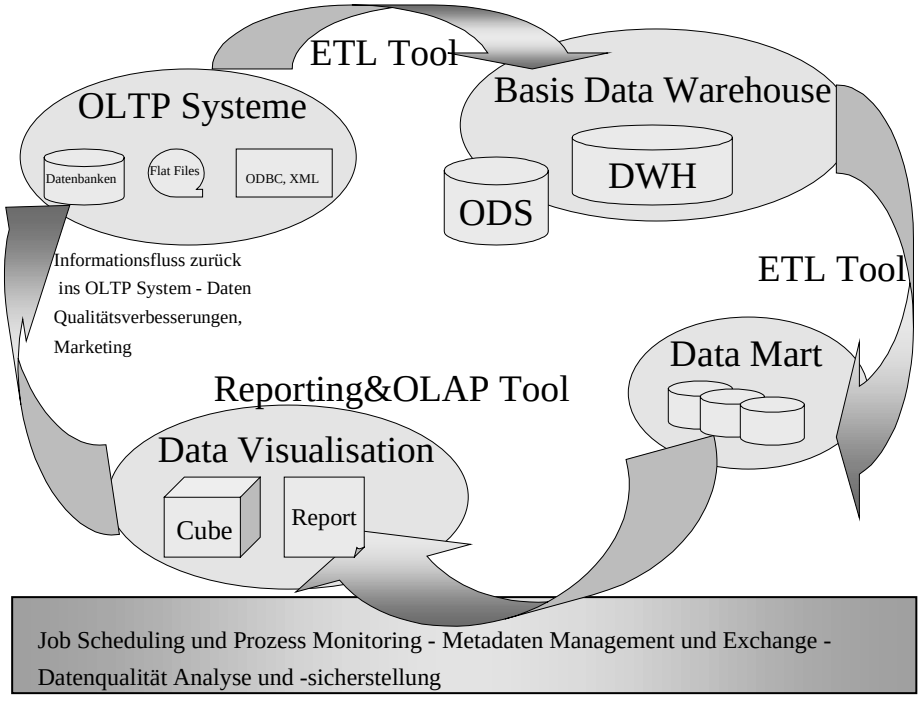

Abbildung 1: Postitionierung ETL Tool innerhalb der Gesamtarchitektur

Die sogenannten OLTP Systeme dienen als Datenbasis für das Data Warehouse. Der Begriff OLTP ist die Abkürzung für Online Transaction Processing und bezeichnet die operativen Systeme der Unternehmung. Mit Unterstützung eines ETL Tools werden die relevanten Daten aus diesen Quellsystemen extrahiert und in ein Data Warehouse oder in einen Operational Data Store (ODS) geladen. Letzterer beinhaltet quasi eine Kopie der Quelldaten und dient in erster Linie dazu, die operativen Systeme von nachgelagerten Verarbeitungsprozessen zu entlasten.

Anschliessend wird wiederum ein ETL Tool eingesetzt, um ein oder mehrere Data Marts mit Daten zu füllen. Bei Data Marts stehen – im Gegensatz zum Data Warehouse – die Auswertebedürfnisse der Benutzer im Vordergrund. Mit anderen Worten, die Daten sind in einer Art und Weise strukturiert, die es erlaubt, Auswertungen effizient und schnell erstellen zu können. Diese Data Marts dienen also als Datenquelle für die sogenannten Visualisierungstools. Mit diesen Auswertewerkzeugen können sowohl statische Reports als auch Datenwürfel erstellt werden. Die Datenwürfel – oder Cubes, wie sie auch genannt werden – enthalten vorausberechnete Kennzahlen zu bestimmten, im voraus festgelegten Kriterien und erlauben dem Endbenutzer das eigenständige ‚Slice and Dice' in den Daten.

4. Funktionsbereiche eines ETL Tools

Untenstehende Graphik verdeutlicht die Funktionsbereiche eines ETL Tools:

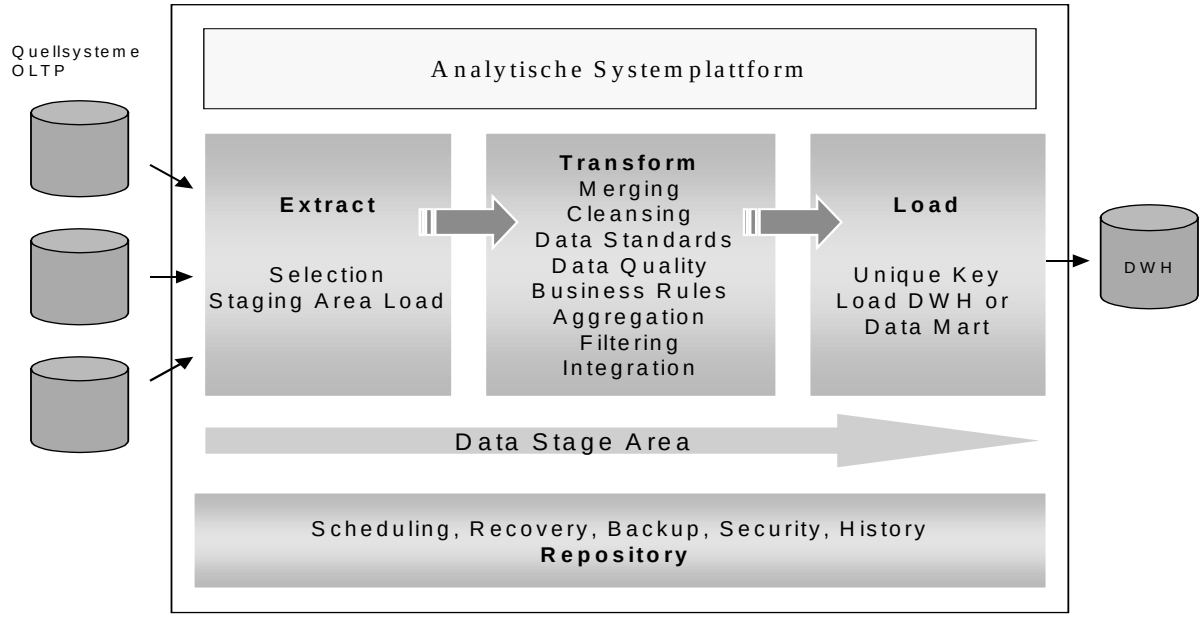

Abbildung 2 : Funktionsbereiche eines ETL Tools

Ganz links sind die sogenannten OLTP – online transaction processing – Systeme schematisch dargestellt, rechts ist das Zielsystem – das Data Warehouse – abgebildet. Die Prozesse, die im Zuge der Erstellung eines Warehouses durchlaufen werden, sind im mittleren Teil des Schaubildes aufgeführt. Zusammenfassend werden diese einzelnen Prozesschritte als ETL Prozess bezeichnet, wobei sich die Abkürzung ETL aus den drei Begriffen Extract, Transform und Load ergibt. Im nachfolgenden Kapitel werden einzelne Teilaspekte des ETL Prozesses herausgegriffen und kurz erläutert.

4.1. Datenextraktion

Das ETL Tool unterstützt die unterschiedlichen Quellstrukturen, Plattformen und Datenbanksysteme des Unternehmens. Bei der Datenselektion und –extraktion werden die Daten zunächst einmal aus den Quellsystemen auf die Zielplattform übertragen. Das kann zunächst einmal eine Data Staging Area sein – also ein near real time data mart – oder aber ein Data Mart resp. das Data Warehouse.

Quelldaten werden in Form von relationalen Datenstrukturen und Flatfiles unterstützt. Aber auch Informationen aus firmenfremden Informationssystemen. wie bspw. von Marketingdaten, Finanzinformationen oder geographischen Informationsdaten können integriert werden.

4.2. Data Cleaning

Im Zuge der Überführung von Daten aus den operativen Systemen in ein Warehouse werden oftmals Datenqualitätsprobleme sichtbar – dazu zählen inhaltliche Fehler, Dateninkonsistenzen oder ungültige Datenformate. Des weiteren liefert die Quelle u.U. Werte, die ausserhalb des zulässigen Wertebereiches liegen.

Im Rahmen des ETL Prozesses wird definiert, welche Dateninhalte und –konstellationen einen Fehlerfall darstellen und wie in einem derartigen Fall zu verfahren ist. So werden – anstelle der teilweise unterschiedlichen Source-spezifischen Datenwerte und -formate – diese in Standard-Default Werte überführt und besitzen damit ein einheitliches Format. Zudem besteht die Möglichkeit, spezielle Tools zur Verbesserung der Datenqualität in die ETL Tools zu integrieren. Diese Tools erkennen Muster und sind in der Lage, die Umwandlung in Zielmuster zu veranlassen. Damit ist bspw. eine Überprüfung von Adressdaten oder die Bereinigung von Dubletten möglich. Alternativ dazu können auch eigene Routinen oder Scripte für die Datenbereinigung integriert werden.

4.3. Datentransformationen

Einem Data Mart resp. Data Warehouse liegt ein Datenmodell zugrunde, das auf die Informationsbedürfnisse seiner Benutzer zugeschnitten ist. Im Rahmen des Datentransformationsprozesses werden die Quelldaten in das Zieldatenmodell überführt. Neben einer einfachen Zuordnung von Datenelementen der Quellsysteme in das entsprechende Feld des Zielmodells werden auch Berechnungen oder Aggregationen durchgeführt. Des weiteren können Filter appliziert werden, damit nur Daten mit bestimmten Werten weiterverarbeitet werden.

4.4. Bildung von Surrogate Keys

In einem Data Warehouse werden in aller Regel nicht die Schlüsselfelder der Quellsysteme verwendet, sondern sogenannte künstliche Schlüssel – Surrogate Keys – gebildet. Mit dieser Vorgehensweise soll vermieden werden, dass Daten zusammengeführt werden, die logisch nicht zusammengehören. Die Surrogate Keys können Timestamps sein, oder einfach eine fortlaufende Zahl. Die Generierung dieser Surrogate Keys wird gleichfalls von den ETL Tools unterstützt.

4.5. Historisierung

Ein wesentliches Kennzeichen eines Data Warehouses ist der Aufbau von Zeitreihen. Damit ist es möglich, Datenanalysen über mehrere Zeitperioden hinweg durchzuführen. Diese sogenannte Historisierung der Daten wird gleichfalls im Rahmen des ETL Prozesses durchgeführt und damit auch – mehr oder weniger – von den ETL Toolanbietern unterstützt. Welche Funktionalitäten hier im Einzelnen eingesetzt werden können,

hängt zum einen natürlich von der Art und dem Umfang der zu historisierenden Daten ab und zum anderen natürlich auch vom gewählten ETL Tool.

4.6. Load Zieltabelle

Abschliessend werden die Daten in die Zieltabellen geladen. Die heutigen ETL Tools unterstützen die meistverbreitetsten Datenbankensysteme wie beispielsweise DB2, Oracle sowie auch XML. Darüber hinaus bieten die ETL Tools die Integration von Loadutilities an, die von dem DBMS unterstützt werden – eine Funktionalität, die insbesondere bei grossen Datenmengen von Vorteil sein kann.

4.7. Metadaten Repository

In der Regel verfügen alle neueren ETL Tools über ein Repository, in dem Metadaten abgelegt werden. Hierbei handelt es sich allerdings lediglich um die technischen Metadaten, d.h. um jene Informationen, die zur Verwaltung und Steuerung der ETL Prozesse notwendig sind sowie um Business Rules, die in Form von Transformationsregeln umgesetzt werden. Die Business Metadaten – die benutzerrelevanten Daten also – werden nicht gespeichert.

Die User- und Gruppen Verwaltung sowie die der Berechtigungsklassen ist jedoch im Repository abgebildet. Ebenfalls wird die Versionierung bei der Entwicklung unterstützt. Die Dokumentation für die analytische Systemumgebung ist meistens in den Tools integriert, sowie als Schnittstelle für andere DWH-Tools vorhanden. Diese Dokumentation ist für die Verwaltung des Informationssystem notwendig.

5. Architekturkonzepte

Der Funktionsumfang heutiger ETL Tools ist im allgemeinen sehr breit. Darüber hinaus sind neben einer Vielzahl von unterstützen Datenbanksystemen auch Schnittstellen zu ERP-Systemen verfügbar. Dank dieses grossen Spektrums gibt es zahlreiche Möglichkeiten und Varianten, wie ein ETL Tool eingesetzt resp. wie es in eine Architektur eingebunden werden kann.

Zu nennen ist in diesem Zusammenhang natürlich die einfachste Variante, nach dem Motto 'Plug and Play': das ETL Tool wird auf einem Server installiert und ohne weitere Ergänzungen genutzt. In einem derartigen Szenario wird neben dem Server für die analytische Plattform lediglich ein geeignetes Datenbanksystem benötigt. So kann sich in kleinen bis mittleren Projekten ETL basiertes Data Sourcing bereits nach einer kurzen Anlaufphase effizienter gestalten als es mit dem Einsatz von herkömmlichen Techniken möglich wäre.

In den nachfolgenden Abschnitten soll an Hand von zwei grundlegend unterschiedlichen Architekturbeispielen gezeigt werden, wie ETL Tools innerhalb einer Gesamtarchitektur eines komplexen Umfeldes sinnvoll eingesetzt werden können.

5.1. Architekturbeispiel 1

5.1.1. Charakteristiken

Im nachfolgenden Beispiel gehen wir von folgenden Annahmen aus:

- Der Grossteil der funktionalen Anforderungen innerhalb des Data Warehouse Prozesses wird durch das ETL Tool abgedeckt.
- Es wird ein Minimum an zusätzlichen Komponenten wie spezialisierte Tools, selbstentwickelte Generatoren, usw. verwendet.
- Die Datenextraktion kann durch quellspezifische Extrakte erfolgen – beispielsweise durch Export-Utilities oder durch die Verarbeitung bestehender Files, welche bereits für andere Zwecke aufbereitet wurden.
- Die Steuerung der Verarbeitungsprozesse erfolgt im wesentlichen über das im ETL Tool enthaltene Job-Scheduling.
- Die Prozesssteuerung erfolgt kontrolliert über Systemgrenzen hinweg. Für die Datenlieferung wird dabei von einer Push-Strategie ausgegangen.

5.1.2. Übersicht

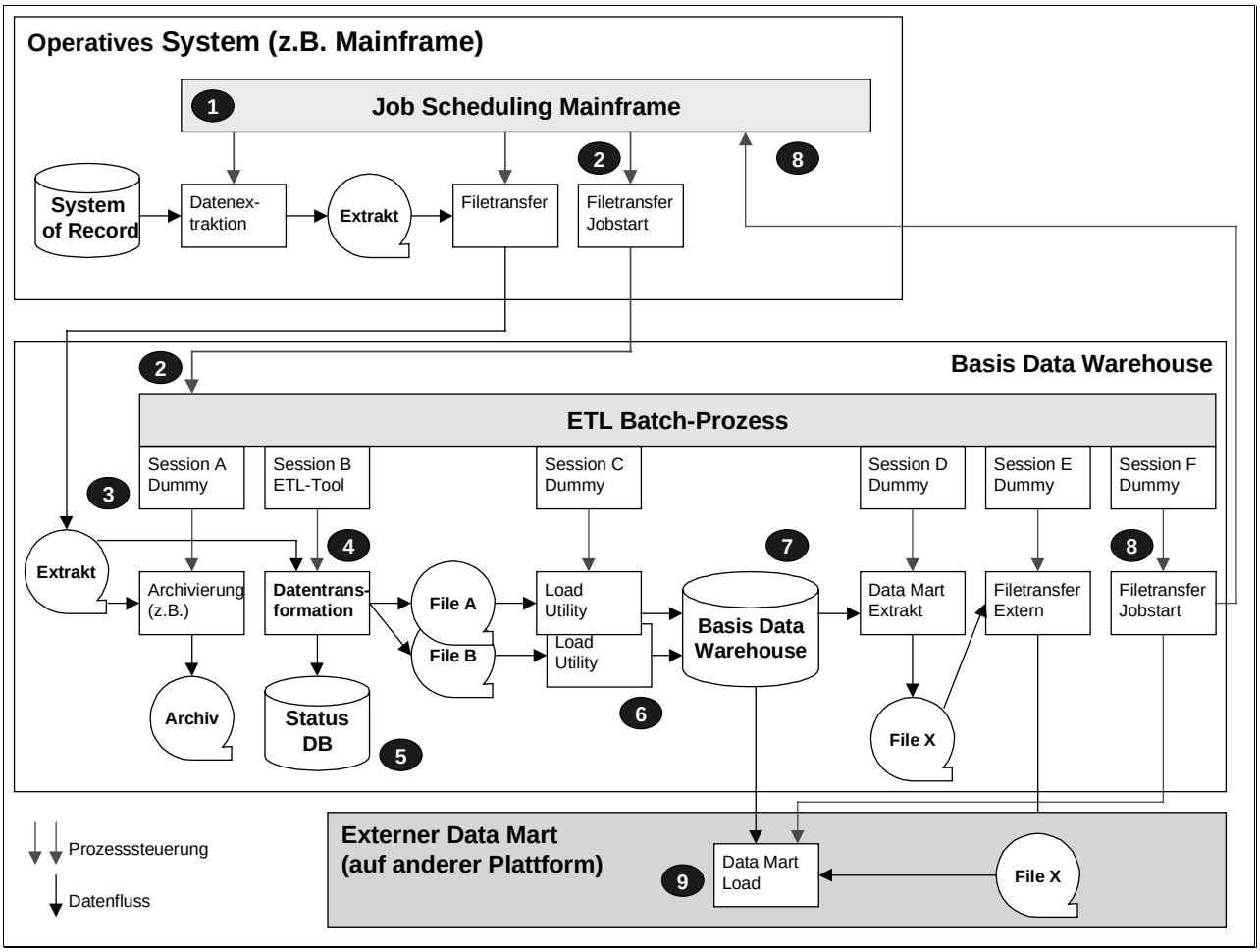

Abbildung 3: Graphische Übersicht Architektur 1

Erläuterungen:

1	Die Extraktion der Daten aus den operativen Systemen ist eingebettet in den fixen Produktionsplan und erfolgt mittels konventionellen Programmen oder Datenbank Utilities. Anschliessend werden die Daten auf den Basis Data Warehouse Server übermittelt.
2	Nach erfolgter Datenübermittlung werden die Parameter zum Start der ETL-Verarbeitung auf den Basis Data Warehouse Server übermittelt. Dadurch wird ein Batch-Prozess angestossen, welcher sich unter der Kontrolle des ETL Tools befindet. Dieser Prozess steuert den Aufruf verschiedener Teilprozesse.

3	Im Rahmen eines Pseudeprozesses werden tool-fremde Verarbeitungsschritte ausgeführt, d.h. Teilprozesse, die nicht mit jener Verarbeitungslogik umgesetzt werden, die durch das ETL Tool erstellt wurde. Dazu gehören der Aufruf von Utilities, Betriebssystembefehlen, Shell Scripts, File-Transfers usw. Im vorliegenden Beispiel wird zuerst die Archivierung der ankommenden Files in die Wege geleitet.
4	Die eigentlichen Datentransformationen, welche die angekommenen Daten in das Zielmodell überführen, werden mit Hilfe des ETL Tools vorgenommen. In engine-basierten Tools werden dabei die in Form von Metadaten erfassten Business Rules zum Ausführungszeitpunkt interpretiert. Neben Datenbereinigungen (Korrekturen), der Vergabe von Default-Werten, können auch Datenaggregate und spezielle Sichten erstellt werden. Neuere ETL Tools erlauben es auch, die für die Bildung der History notwendigen Zusatzinformationen zu generieren.
5	Statusinformationen zur Verarbeitung, z.B. die Anzahl der gelesenen Sätze, werden in speziellen Tabellen direkt nachgeführt.
6	Die bei Warehouse Anwendungen typischerweise grossen Datenmengen werden aus Performance Gründen in der Regel mit speziellen Ladewerkzeugen in die Zieldatenbank eingefügt.
7	Das Basis Data Warehouse umfasst das ganze Zielmodell in weitgehend normalisierter Form. Die Datenbestände werden in der Regel mehrere Monate historisiert (6 – 36 Monate).
8	Nach Abschluss des ETL-Batches wird die Kontrolle wieder an das die Verarbeitung anstossende operative System übergeben. Ausserdem können weitere, Data Mart spezifische Verarbeitungen angestossen werden, sei es auf der selben Plattform oder auf externen Mart Servern.
9	Ein Data Mart, das sich im vorliegenden Architekturbeispiel auf einer anderen Plattform befindet, kann seine Daten entweder direkt aus spezifischen Extrakten beziehen, oder aber der Datenzugriff erfolgt direkt über das ETL Tool. Genügende Übertragungskapazitäten vorausgesetzt, darf mit einem zufriedenstellenden Durchsatz gerechnet werden.

5.1.3. Vorzüge

Marktübliche ETL Tools bieten sowohl bezogen auf Quell- wie auch auf Zielsysteme eine grosse Anzahl von Schnittstellen zu den verschiedensten Systemplattformen und Datenbanksystemen an. Mit einem einzigen Tool stehen dem Entwickler so eine Vielzahl von unterschiedlichen Werkzeugen zur Verfügung. Job-Steuerung und Verarbeitungslogik werden mit einem einzigen Tool realisiert. Diese Aspekte führen dazu, dass die für die Entwicklung der Sourcing Prozesse erforderlichen Softwarekomponenten bereits nach einer kurzer Proof of Concept-Phase mit vergleichsweise bescheidenem Aufwand implementiert werden können. Ein weiterer Vorteil besteht darin, dass sämtliche Toolkomponenten das gleiche ‚Look&Feel' haben, d.h. über eine integrierte Bedienungs- und Entwicklungsoberfläche verfügen.

Werden in einer Entwicklungsumgebung nur wenige Tools eingesetzt, wird die Einarbeitung von neuen Projektmitarbeitern deutlich erleichtert. Der Entwickler kann so als Generalist seine Kenntnisse in der Anwendung der Sourcing-Tools vertiefen und sich effizienter der Lösung der Business-Anforderungen widmen. Zusätzlich verringert sich der Bedarf an technischem Supportpersonal zur Unterstützung der Entwicklungsabteilungen.

5.1.4. Nachteile

Obwohl die ETL Tools in den letzten Jahren im Funktionsumfang wesentlich erweitert wurden, können verschiedene Spezialsituationen nicht in allen Fällen effizient umgesetzt werden. Dies kann zu umständlichen und komplizierten Lösungen führen, welche die Wartbarkeit erschweren.

Wenn der Sourcing Prozess weniger modular aufgebaut ist, können komplizierte Mappings entstehen. Dies führt zur Entstehung von vielen Freiräumen für die Entwickler und erschwert die Durchsetzung von Warehouse weiten Standards.

Die Steuerung der Batch-Prozesse erfolgt durch das ETL Tool selbst und nicht durch gängige auf dem Markt erhältliche Tools. Dies kann bei den Betriebsequipen in Rechenzentren zu einer schlechten Akzeptanz führen, mit der Folge, dass Verarbeitungsprobleme vermehrt durch die Softwareentwickler gelöst werden müssen. In dieser Hinsicht stellt besonders das Monitoring der laufenden Batchprozesse ein Fremdkörper dar und erfordert in der Regel den Einbau zusätzlicher Funktionen zur Information des Operatings.

5.1.5. Eignung

Besonders beim Aufbau von mittelgrossen analytischen Systemen ermöglicht dieser Ansatz – verglichen mit herkömmlicher Programmierung – bereits nach kurzer Zeit eine Effiziensteigerung. Dabei kann in jenen Teilbereichen, in denen der Einsatz eines ETL Tools als ungeeignet erscheint, durchaus auf ergänzende Komponenten zurückgegriffen werden.

Da selbst in komplexen und grossen Warehouse-Projekten bereits im Rahmen der Evaluations- und Pilotphase qualitativ hochwertige Ergebnisse erzielt werden müssen, werden idealerweise zunächst einfache Architekturkonzepte umgesetzt. Damit ist ein guter Grundstein für einen späteren Ausbau des Systems gelegt und somit auch die Voraussetzung für die Generierung von Business Nutzen.

5.2. Architekturbeispiel 2

5.2.1. Charakteristiken

Im nachfolgenden Beispiel gehen wir von folgenden Annahmen aus:

- Das ETL Tool ist Bestandteil eines umfassenden Frameworks, welches aus gekauften oder selbstentwickelten Komponenten besteht. Es wird insbesondere dort eingesetzt, wo Daten auf Grund von Business Rules umgeformt werden müssen.

- Für verschiedene Funktionen mit hohem Standardisierungspotential, wie z.B. Ladevorgänge, Datenformatprüfungen, Daten Cleaning und die Ermittlung von Delta-Records werden Generatoren verwendet. Diese erzeugen aufgrund der erfassten Metadaten Jobs Scripts oder SQL Statements. Alternativ dazu werden Datenbank Utilities oder andere Werkzeuge eingesetzt, mit deren Hilfe Sort, Merge oder Split Funktionen über Parameter gesteuert werden können.

- Es wird davon ausgegangen, dass eine grosse Anzahl von Files aus operativen Systemen angeliefert werden. Der Anstoss der Verarbeitung wird durch die Ankunft der Files auf der Warehouse Plattform ausgelöst. Ergänzend zu dieser Fileverarbeitung können jedoch auch Zugriffe auf operative Datenbanken erfolgen.

- Für die Jobsteuerung werden Komponenten ausserhalb des ETL Tools verwendet. Mittels Standard-Prozeduren werden umfangreiche Daten über die Verarbeitung gewonnen, welche für das Monitoring der Verarbeitung, das Problem Tracking und für die Darstellung des Load-Status des analytischen Systems verwendet werden können.

5.2.2. Übersicht

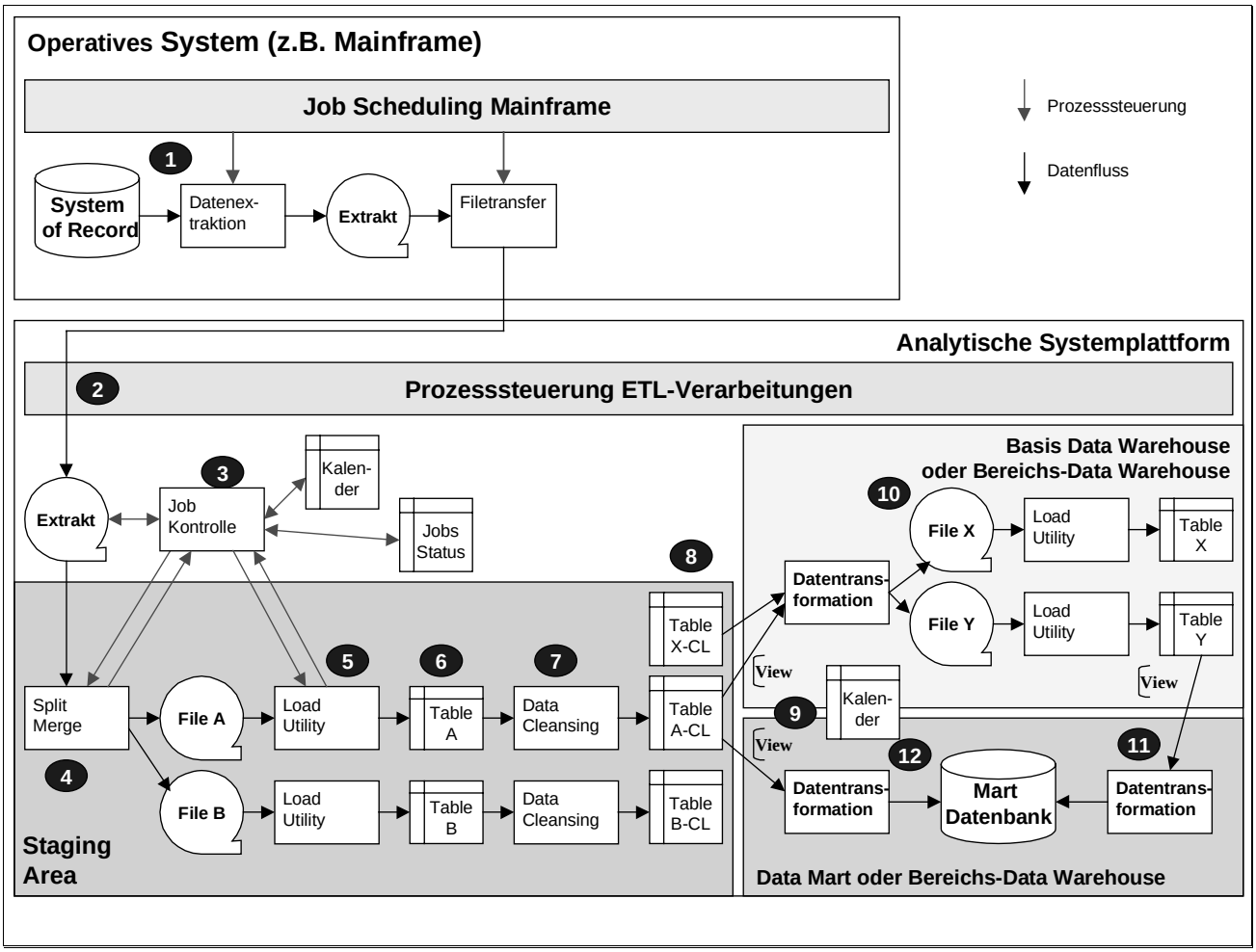

Abbildung 4: Graphische Übersicht Architektur 2

Erläuterungen:

1	Die Extraktion der Daten aus den operativen Systemen ist in einen fixen Produktionsplan eingebettet und erfolgt mittels konventionellen Programmen oder Datenbank Utilities. Anschliessend werden die Daten auf die analytische Plattform übermittelt.

2	Der Load Prozess wird über Trigger angestossen. Im vorliegenden Beispiel wird für die Steuerung der Verarbeitung das bereits existierende Scheduling System eingesetzt, d.h. die entsprechende Funktionalität des ETL Tools wird nicht genutzt. Alternativ zu dieser Vorgehensweise könnten separate Scheduling-Komponenten eingesetzt werden, die stärker auf die Sourcing-Bedürfnisse in komplexeren Data Warehouse Umgebungen ausgerichtet sind. Von Eigenentwicklungen sollte im Bereich der Job-Steuerung auf Grund der hohen Komplexität Abstand genommen werden.
3	Zu Beginn der Verarbeitung können unterschiedliche Prüfungen auf File-Ebene vorgenommen werden. Dazu gehören beispielsweise Vollständigkeits- und Sequenzprüfungen von Files. Alle Jobschritte generieren Status- und Statistikinformationen, welche in den Statustabellen online nachgeführt werden und somit jederzeit Auskunft über den Ladezustand des Warehouses geben können. Die Jobsteuerung erfolgt über standardisierte Scripts, die im Idealfall generiert werden können und auch defaultmässig Restarts ermöglichen.
4	In komplexen Systemen müssen Files oftmals zunächst für die Verarbeitung vorbereitet werden, insbesondere dann, wenn auf bestehende Standard-Feeds zugegriffen wird, welche ursprünglich für analytische Zwecke in Legacy Systemen erstellt wurden. Zu solchen Vorbereitungsarbeiten gehört die Aufteilung von Files nach den unterschiedlichen Record-Typen. Andererseits können auch Files mit identischem Layout konsolidiert werden, die von unterschiedlichen Systemen (z.B. Niederlassungen) stammen. Dies kann durch generische Prozesse unter Verwendung von Standard-Utilities erfolgen und führt zu einer wesentlichen Vereinfachung der nachfolgenden Schritte.
5	Die bei Warehouse Anwendungen typischerweise grossen Datenmengen werden aus Performance Gründen in der Regel mit DBMS unterstützten Ladewerkzeugen geladen.
6	Die Input-Files werden vollständig in eine Tabelle geladen. Diese enthalten neben den ausschliesslich alphanumerische Daten noch zusätzliche Statusfelder.
7	Data Cleaning ist ein Eckpfeiler für den Erfolg eines analytischen Systems. Die Integration von Daten unterschiedlichster Quellen kann Datenprobleme in operativen Systemen akzentuieren. Für Datenformatsprüfungen werden beispielsweise SQL-Scripts eingesetzt, welche auf Metadaten mit Cleaning Regeln basieren. Alternativ dazu bietet sich der Einsatz von Standard-Tools an. Gerade im Bereich der Dublettenerkennung oder der Standardisierung von Adressen empfiehlt sich der Einsatz dieser Standard-Tools. Sie bieten Funktionen, die entweder über ein herkömmliches ETL Tool aufgerufen werden können oder die der parametergesteuerte Bearbeitung der Files dienen, welche von dem ETL Tools erstellt wurden.

8	Nach Abschluss des Data Cleaning werden die bereinigten Daten in Staging-Tabellen geladen. Diese stehen für die Weiterverarbeitung in das Warehouse spezifische Zielmodell zur Verfügung. Je nach Organisation eines Unternehmens können unterschiedliche Bereichs-Warehouses oder Data Marts beliefert werden.
9	Unterschiedliche Datenbezieher können nun über Views, die über Kalendertabellen gesteuert werden, auf die Daten zugreifen. Nachdem die Daten von allen Applikationen übernommen worden sind, können diese wieder von den Staging Tabellen gelöscht werden.
10	Der Aufbau eines Basis Data Warehouse bzw. eines bereichsspezifischen Data Warehouses erfolgt basierend auf den Staging Tabellen. Es handelt sich hier um einen kompletten ETL Prozess, bei dem der Schwerpunkt der Aktivitäten bei der Integration der Daten in das Warehouse Modell liegt. Neben der Bildung von Warehouse weiten Keys wird in den weitgehend normalisierten Zieltabellen auch die Historisierung der Daten vorgenommen. Aus Performance Gründen müssen gezielte Denormalisationen teilweise bereits im Basis Data Warehouse vorgenommen werden. In nachgelagerten Prozessen können zudem bereits erste Aggregate erstellt werden, falls diese von mehreren Data Marts bezogen werden.
11	Der Aufbau eines Data Marts besteht wiederum aus einem kompletten ETL Prozess. Die Daten werden aus einem Basis- oder bereichsspezifischen Data Warehouse bezogen. Das Zielmodell des Data Marts ist dabei auf die Auswertung der Daten optimiert. Den Datenbankmodellen liegt dabei eine Star- oder Snowflake-Struktur zugrunde. Diese dienen dann als Basis für das Reporting mit den entsprechenden Query- und OLAP Tools.
12	Alternativ können die Daten für den Aufbau von Marts direkt aus den Staging Tabellen bezogen werden. Gegenüber dem direkten Bezug aus operativen Systemen bietet sich hier der entscheidende Vorteil, dass die Daten bereits das Data Cleaning durchlaufen haben.

5.2.3. Vorzüge

Der Hauptvorteil dieses Ansatzes besteht darin, dass spezialisierte Komponenten die entsprechenden Aufgaben sehr effizient und meist auch sehr performant erfüllen können. Der Einsatz eines solchen Architekturkonzeptes gibt den Verarbeitungsweg in einem hohen Mass vor. Die Verwendung von einheitlichen Prozessen und Generatoren, welche für wiederkehrende Schritte mit wenig Business Logik eingesetzt werden können, ermöglicht eine leichtere Standardisierung der Verarbeitung und führt zu einer modularen Verarbeitung. Unter der Voraussetzung, dass die eingesetzten Komponenten die erforderliche Stabilität und Robustheit haben, können Aufwand und Entwicklungszeit reduziert werden.

5.2.4. Nachteile

Es versteht sich von selbst, dass dieser Ansatz grosse Ressourcen für Entwicklung und Unterhalt bindet, insbesondere dann, wenn viele Komponenten selbst entwickelt werden. Der dargestellte Ansatz mit einer doch recht grossen Anzahl von einzelnen Prozessen kann zu einer Zersplitterung der Verarbeitung führen, was die Gewinnung bzw. die Nutzung von Metadaten über die Datentransformationen erschwert. Eine tendenziell grössere Anzahl von Job-Steps kann zudem die Wartung erschweren und bedingt auch einen erhöhten Supportbedarf.

Es besteht die Gefahr, dass ein recht komplexes Gebilde entsteht, dessen Steuerung und Beherrschung besonders in Ausnahmesituationen viel Erfahrung und Sachverstand benötigt (Black-Boxes). Diese Gefahr ist umso grösser, je mehr Infrastruktur Komponenten selbst entwickelt wurden. Dadurch erhöht sich die Abhängigkeit von Schlüsselpersonen mit dem entsprechenden Know-How in den technischen Abteilungen.

Ferner besteht die immanente Gefahr, dass der Aufbau einer derartigen Infrastruktur unterschätzt wird, da die Entwicklung meist aufgrund von 'Normal-Situationen' geplant wird. In der Praxis wirken jedoch Spezialfälle erschwerend beim Aufbau einer Infrastruktur. In der Konsequenz bedeutet dies, dass Spezialsituationen meist parallel oder nur mit geringem Vorlauf zur eigentlichen Applikationsentwicklung im Framework integriert werden, mit dem Ergebnis, dass sich entweder die Entwicklung verzögert, oder Work Arounds erstellt werden müssen.

5.2.5. Eignung

Das Architekturbeispiel 2 zeigt in vereinfachter Form, dass der Aufbau analytischer Systeme in komplexen Umgebungen sehr anspruchsvoll sein kann. Wie schon früher erwähnt wurde, sind komplexe, vor allem technologisch orientiere Warehouse Projekte oft zum Scheitern verurteilt. Der Aufbau eines solchen Systems soll daher behutsam und iterativ erfolgen. Dabei werden die bei diesen Schritten gewonnen Erfahrungen im Entwicklungsprozess genutzt. Es ist dabei zu beachten, dass die Entwicklung von modularen Bausteinen und Generatoren zeitaufwändig ist. Der Aufbau und Betrieb einer komplexen Infrastruktur erfordert das notwendige Personal und Know How. Da die Kosten in der Regel auf die einzelnen Business-Applikationen umgelegt werden müssen, können solche Vorhaben nur realisiert werden, wenn der Kreis der Anwender genügend gross ist.

6. Praktischer Einsatz des ETL Tools

An dieser Stelle werden einige wenige Aspekte herausgegriffen, die zwar wichtig sind, jedoch im Rahmen einer Produkte-Evaluation leicht vergessen werden.

6.1. Customizability des Tools

Mit dem Begriff ‚customizability' ist die Anpassungsmöglichkeiten des Tools auf die bestehende Umgebung gemeint. Letztlich gehört auch dieser Punkt in den Bereich der vom Kunden gewünschten Tool-Funktionalität und muss im Rahmen einer Produkte-Evaluation beantwortet werden.

So ist auch hier – wie überhaupt bei dem Evaluationsprozess – die zentrale Frage, was benötigt wird. Es gibt ETL Tools, die in hohen Masse anpassbar sind und diesbezüglich kaum Wünsche offen lassen. Diese Freiheitsgrade bedingen jedoch auch einen vergleichsweise hohen Initialaufwand, der geleistet werden *muss*, bevor das Tool erstmalig eingesetzt werden kann. Der zeitliche Rahmen bewegt sich hier durchaus im Bereich von mehreren Wochen. Dieser Aufwand ist nur dann gerechtfertigt, wenn dieser hohe Grad an Einbindung in die bestehende Infrastruktur zwingend notwendig sind.

Ist dies jedoch nicht der Fall, so ist man mit einem derartigen Tool sehr schlecht bedient. Andere ETL Tools können innerhalb eines halben Tages installiert werden und sind dann funktionsbereit. Logischerweise bieten diese Tools nicht so viele Anpassungsmöglichkeiten, wie die oben genannten.

Dieser Aspekt ist deswegen so wichtig und wird herausgegriffen, weil er eine grosse Implikation auf den Zeitplan hat. Es ist nicht einfach nur eine Funktion, die nicht zur Verfügung steht und für die irgendein Workaroud gefunden werden muss. Wird hier die falsche Entscheidung getroffen, so kann dies dazu führen, dass zu einem späteren Zeitpunkt der Zeitplan, das Projekt und die Akzeptanz massiv in Frage gestellt werden wird.

6.2. Entwicklungszyklus mit einem ETL Tool

Überlegungen hinsichtlich des beabsichtigten Entwicklungszyklus sind auch beim Einsatz eines ETL Tools sehr wichtig. In der Regel ist es ratsam, zumindest die Entwicklungs- und Testumgebung von der Produktionsumgebung zu trennen. Denn nur so kann eine stabile Produktionsumgebung garantiert sowie das Risiko minimiert werden, unfertige ETL Prozesse ‚versehentlich' in die produktive Umgebung zu schreiben.

Besteht Klarheit darüber, in welcher Weise die Entwicklung, Tests und Produktion voneinander getrennt werden sollen, so muss geprüft werden, wie dies mit dem ETL Tool umgesetzt werden kann. Dabei ist nicht nur die zur Verfügung stehende Funktionalität an sich zu prüfen, sondern gleichfalls die Anzahl der erforderlich ETL Komponenten. So kann die Trennung von Entwicklungs- und Testumgebung beispielsweise bedingen, dass zwei ETL-Engines parallel eingesetzt werden müssen. Angesichts der Lizenzkosten der ETL Tools stellt diese Position u.U. einen nicht unerheblicher Kostenfaktor dar.

6.3. Versionierung

Eng verknüpft mit der Vorgehensweise bei der Entwicklung des ETL Prozesses ist die Frage nach der Versionierung, resp. der entsprechenden Funktionalität, die von dem ETL Tool angeboten wird.

Im einfachsten Fall ist das lediglich ein Check-in/Check-out Mechanismus, der eigentlich kaum als Versionierung bezeichnet werden kann. In der Regel hat dieser primär die Aufgabe, die Konsistenz der Objekte sicherzustellen, indem verhindert wird, dass mehrere Benutzer das gleiche Objekt editieren. Das eigentliche Versionenhandling muss ausserhalb des Tools durchgeführt werden.

Andere ETL Tools verfügen über einen recht komfortablen Mechanismus zur Versionenkontrolle. Zu berücksichtigen ist hierbei, auf welchem Level die Versionierung zur Verfügung steht. Das heisst konkret, ob ein einzelnes Objekt, wie bspw. ein Mapping versioniert werden kann, oder ob die Versionierung lediglich auf einem höheren Level – z.B. auf Ebene Folder – möglich ist. Idealerweise werden die verschiedenen Objekte – Source- und Targetdefinitionen sowie Mappings – in einer Weise in der Folderstruktur gruppiert, wie sie auch später versioniert werden sollen.

In diesem Zusammenhang sollte auch überprüft werden, ob eine Fallback-Funktion zur Verfügung steht, d.h. ob die Möglichkeit besteht, frühere Versionen der Objekte wieder zu aktivieren. Das kann notwendig werden, wenn sich herausstellt, dass Änderungen der aktuellsten Version fehlerhaft sind und deshalb schnell die vorhergehende Version aktiviert werden muss.

6.4. Aufbau- und ablauforganisatorische Aspekte

Hat man sich für den Einsatz eines ETL Tools entschlossen, so sind auch organisatorische Aspekte zu bedenken und zu entscheiden. So gibt es bei den meisten Toolherstellern ein gewisses Rollenverständnis. Exemplarisch seien hier die folgenden Rollen genannt:

Administrator

Mapping Designer/Conversion Specialist

ETL Prozesse Verantwortlicher (Operator)

Master Infrastructure.

In einem ersten Schritt müssen die verschiedenen Rollen bestimmt und deren Inhalte definiert werden. Darauf basierend kann entschieden werden, wie diese Rollen zusammenspielen. Im Anschluss daran wird schliesslich bestimmt, wer diese Rollen wahrnimmt. Ein wesentlicher Faktor zur Beantwortung dieser Fragen ist die Projektgrösse.

Gewisse Entscheidungen dürften relativ einfach zu treffen sein – da der Mapping Designer letztlich die Aufgaben wahrnimmt, die bis anhin von einem Programmierer abgedeckt wurden, liegt es nahe, dass diese Personen diesen Aufgabenbereich übernehmen.

Andere Bereiche können sich etwas schwieriger gestalten, wie beispielsweise das Operating. Folgende Möglichkeiten sind hier denkbar:

- Der Mapping Designer, der bereits die inhaltliche Verantwortung für die Loadprozesse übernimmt?
- Das Operating, das den Betrieb aller anderen Prozesse im Rechenzentrum steuert?
- Evtl. auch der Endanwender – beispielsweise wenn er für die Bereitstellung von Inputfiles sorgt und nun auch die nachfolgende Verarbeitung anstossen möchte.

Nebst dieser rein organisatorischen Entscheidung sind damit natürlich auch technische Implikationen verbunden, die gelöst werden müssen. Exemplarisch auch hier einige Fragestellungen:

- Wie wird die Scheduling Komponente des ETL Tools integriert mit dem bereits eingesetzten Tool, das alle übrigen – Nicht-ETL-Prozesse – steuert?
- Wie werden fehlerhafte Loads bereinigt?
- Gibt es eine Möglichkeit einem Endbenutzer eine Funktion zur Verfügung zu stellen, die es ihm ermöglicht, Prozesse zu triggern, ohne sich um technische Belange kümmern zu müssen?
- Wie kann eine ausgewogene Auslastung des Systems erreicht werden?

7. Beurteilung

Wie vorliegendes Dokument verdeutlicht, ist der Einsatz eines ETL Tools in einer Data Warehouse Umgebung keineswegs trivial. Alle Aspekte der möglichen alternativen Architekturansätze müssen bedacht und entsprechend umgesetzt werden.

An dieser Stelle sei nochmals kurz auf die Versprechungen der ETL Toolanbieter eingegangen: ‚Durch den Einsatz eines ETL Tools könne Zeit und Aufwand gespart werden.' Diese Feststellung muss sicherlich differenziert betrachtet werden. Zum einen ist die Alternative zu einem ETL Tooleinsatz in Betracht zu ziehen: Die konventionelle Programmierung. Zum anderen ist darüber hinaus zu unterscheiden zwischen der Phase, in der eine Data Warehouse Architektur aufgebaut wird und der Zeit danach – wo die gesamte Umgebung funktionsfähig für die Entwicklung zur Verfügung steht.

Beim Aufbau einer kompletten Data Warehouse Infrastruktur ist das ETL Tool eine Infrastrukturkomponente unter vielen, die es den Bedürfnissen entsprechend zu integrieren gilt. Abhängig vom Komplexitätsgrad der gesamten Umgebung, verursacht das ETL Tool an dieser Stelle einen mehr oder minder grossen zusätzlichen Aufwand.

Anschliessend kann das ETL Tool jedoch im Rahmen einer funktionsfähigen Infrastruktur sicher zeitsparend eingesetzt werden, insbesondere auch im Vergleich zu konventionellen Programmierung. In diesem Bereich liegen wohl auch die grössten Vorteile eines ETL Tools. Zu nennen sind in diesem Zusammenhang die folgenden Punkte:

- Sicherstellen der notwendigen Flexibilität
- Gute System Dokumentation und Metadaten Unterstützung
- Geringerer Wartungsaufwand
- Personenunabhängigkeit

Last but not least kann das ETL Tool auch als ‚single point of truth' für Semantik, Regeln und Methoden angesehen werden.

Literatur

Dippold/Meier/Ringgenberg/ Schnider/Schwinn 2001 — Dippold, R.; Meier, A.; Ringgenberg, A.; Schnider, W.; Schwinn, K.: Unternehmensweites Datenmanagement – Von der Datenbankadministration bis zum modernen Informationsmanagement. 3. Auflage. Wiesbaden 2001

Autoren

Die Autoren sind erfahrene Spezialisten aus dem Bereich Business Intelligence & Data Warehousing der Systor AG mit den Arbeitsschwerpunkten Projektmanagement, Beratung, ETL, OLAP und Warehousing.
Kontakt: regine.stopka@systor.com