Performance Industrialisierung mit ODI 12c

Irina Gotlibovych
MT AG
Ratingen

Schlüsselworte
Data Warehouse, Industrialisierung, Automatisierung, Standardisierung, Performance, Architektur, ODI, Best Practice

Der Performance auf der Spur
Ein Data Warehouse ist kein statisches Konstrukt und ist dem Wandel der Zeit unterworfen. Es kommen immer neue Anforderungen, die Umgebung verändert sich mit der Zeit, Umstrukturierungen im Unternehmen finden statt. Heutzutage müssen Daten in einem Data Warehouse nahezu permanent zur Verfügung stehen, die Zeitfenster für die Beladung werden immer kleiner. Mit immer stärker wachsendem Datenvolumen muss wiederholt nach neuen Wegen gesucht werden, um ausreichende Performance zu gewährleisten. Idealerweise sollten Logik und Implementierung der Prozesse so gestaltet sein, dass sie im Hinblick auf zukünftige Veränderungen ausreichend flexibel sind. Bei einem bestehenden Data Warehouse kann man aber nicht alles neu bauen, die angestrebten Änderungen sollen möglichst viel Gewinn bei relativ wenig Zeitaufwand bringen – das Preis-Leistungs-Verhältnis muss stimmen.

Beispiel aus der Praxis
Dieser Problematik begegneten wir in einem Kundenprojekt, wo ein Data Warehouse mithilfe von Oracle Data Integrator betrieben wird. Die Ausgangssituation sah wie folgt aus:

- Gemeinsame Verarbeitung von unabhängigen Gesellschaften
- „Alles oder Nichts“ Prinzip bei ETL-Strecken
- Schlechte Performance

Folgende Ziele wurden als Aufgabenstellung definiert:

- Zeitlich und fachlich unabhängiges Laden von verschiedenen Gesellschaften
- Performanceverbesserung

Wie geht man vor? Wie erreicht man die Ziele mit möglichst wenig Änderungsaufwand? Und vor allem, wie garantiert man Flexibilität und Skalierbarkeit für die Zukunft? Wir haben uns für den Industrialisierungsansatz entschieden. Data Warehouse Industrialisierung ist ein bewährtes Prinzip, um Aufwände und Kosten in allen Phasen eines Projekts zu reduzieren und auch zukünftig flexibel gegenüber Änderungen zu bleiben. Durch das standardisierte Vorgehen und das Einhalten von Konventionen ermöglicht man Generierbarkeit und damit Automatisierbarkeit des Systems.

Modularisierung
Im ersten Schritt wurde die Verarbeitung einzelner Gesellschaften unabhängig gemacht. Sollte also eine der Quellen Lieferprobleme haben, können Daten anderer Quellen entgegengenommen und verarbeitet werden. Um eine performante Verarbeitung zu garantieren wurde eine durchgehende Partitionierung nach der Quellgesellschaft durchgeführt. Damit ODI jetzt nicht mehr mit allen Daten gleichzeitig, sondern nur mit jeweiligen Partitionen arbeiten kann, können Objekte und Prozesse entsprechend konfiguriert werden. Dank eingeführter Standardisierung konnten parametrisierte Projektvariablen eingeführt werden, die im neuen einheitlichen SCD Knowledge Modul verwendet wurden. Mithilfe dieses Knowledge Moduls haben wir eine partitionsweise Verarbeitung ermöglicht. Der Vorteil dieser Lösung ist, dass quasi keine Änderungen an den vorhandenen ODI Mappings nötig waren. Es gibt

weiterhin nur einen Ladeplan für alle Gesellschaften, dieser ist jetzt aber parametrisiert und über einzelne Schedulers unabhängig ausführbar.

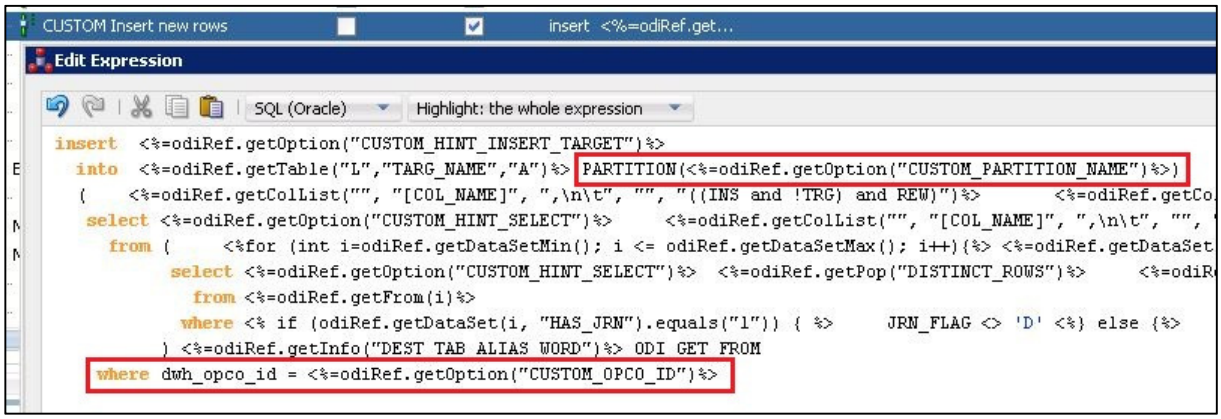

Abb. 1: Parametrisiertes Knowledge Modul

Diese Lösung ist flexibel und durch einfache Konfigurationen weiter skalierbar. Mit wenig Änderungsaufwand konnten wir durch geschickte Ausnutzung der Partitionierung und dank generischer Parametrisierung von Knowledge Modulen eine bemerkenswerte Laufzeitverbesserung erreichen:

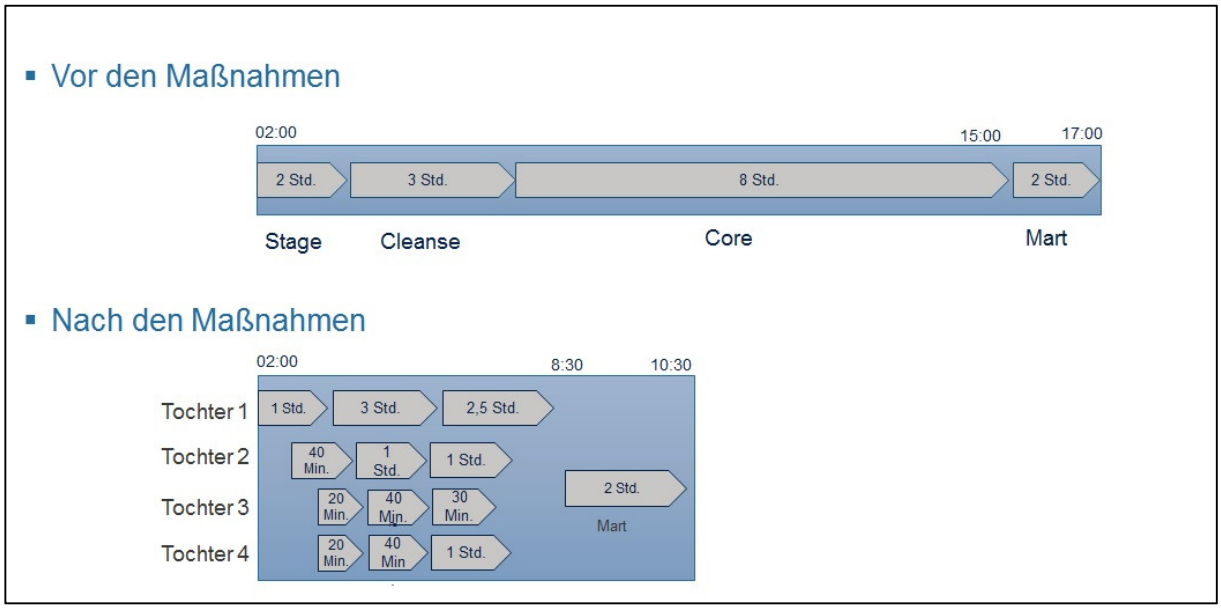

Abb. 2: Performance vor und nach der Modularisierung

Deltaverarbeitung

Es war abzusehen, dass Datenmengen sich zukünftig weiter erhöhen und neue laufzeitintensive Anforderungen hinzukommen werden. Da von diesem Standpunkt gesehen die erreichte Performance zum Teil immer noch nicht ausreichend war, mussten wir nach weiteren Wegen suchen, um Ladezeiten zu verbessern. Die Vorsysteme lieferten die Daten täglich als Fullload an, da sahen wir Potenzial: automatische Deltaberechnung. Direkt nach der Datenlieferung im Stage soll automatisiert Deltaberechnung stattfinden. Anschließend wird nicht mehr der komplette Bestand verarbeitet, sondern nur die geänderten Daten – bei Tabellen mit geringem Änderungsanteil reduziert sich die Datenmenge dadurch enorm. Um dem Prinzip der Industrialisierung treu zu bleiben, sollte die neue ETL-Logik nicht für jedes ODI Mapping einzeln umgesetzt werden. Stattdessen wurde einmalig ein dynamisches „Deltaframework“ entwickelt, welches für jede Tabelle bzw. ODI Mapping angewendet werden kann. Die eigentliche Umstellung auf Deltaverarbeitung kann jetzt mittels Generierung und Konfiguration geschehen, anstatt durch wiederholte und zeitaufwändige Entwicklung.

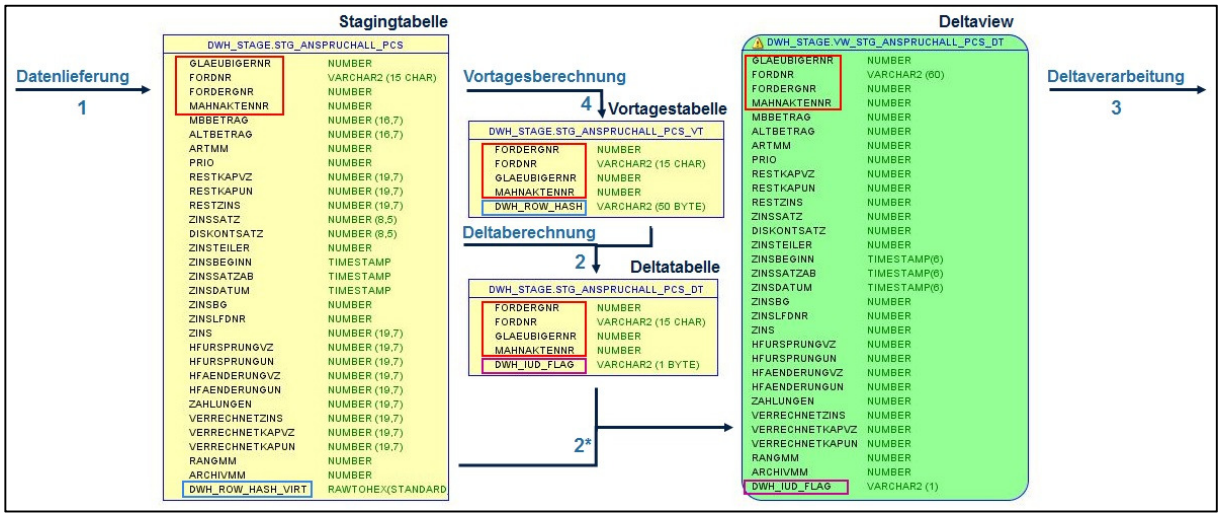

Abb. 3: Datenfluss bei Deltaverarbeitung

Die eigentliche Deltaberechnung musste einigen Anforderungen genügen. Sie sollte performant sein und auch für sehr große Tabellen zuverlässig funktionieren. Der zusätzliche Speicherverbrauch zum Vorhalten der benötigten Delta- und Vortagesdaten sollte minimal bleiben. Nach eingehender Evaluierungsphase haben wir uns für folgendes Vorgehen entschieden:

- Deltabildung über Hashwert der Zeile statt über alle Daten
- Verwendung der Oracle Funktion STANDARD_HASH
  - Parameter 1: Verkettung aller Spalten der Tabelle
  - Parameter 2: Hashmethode SHA1

Die resultierende Gesamtlaufzeit der einzelnen ETL-Strecken war bemerkenswert. Das im Stage berechnete Delta kann sowohl bei der Verarbeitung im Cleanse als auch Core verwendet werden. Abhängig von der Menge der gelieferten Änderungen haben wir eine Laufzeitreduktion von bis zu 90% vermessen können! Die gewählte Methode erlaubt sukzessive Umstellung der Langläufer auf Deltalogik ohne weitere Entwicklungsaufwände. Alle verwendeten Funktionalitäten sind dynamisch und können durch entsprechende Parametrisierung bzw. Konfiguration automatisiert installiert werden.

```
exec STAGE_DELTA.P_CREATE_TAB_vortag('ANSPRUCHALL');
exec STAGE_DELTA.P_CREATE_TAB_delta('ANSPRUCHALL');
exec STAGE_DELTA.P_add_hash_col('ANSPRUCHALL');
exec STAGE_DELTA.P_add_iud_col('ANSPRUCHALL');
exec stage_delta.P_CREATE_VIEW_DELTA('ANSPRUCHALL');
exec stage_delta.P_CREATE_VIEW_IN_CLN('ANSPRUCHALL');
```

Abb. 4: Generierung neuer Strukturen

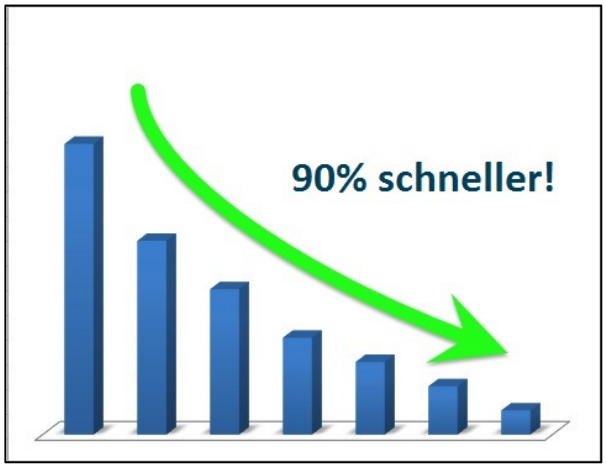

Abb. 5:Laufzeitreduktion durch Deltaverarbeitung

Fazit

Die umgesetzten Maßnahmen bringen nicht nur beeindruckende Laufzeitverbesserungen mit sich, sondern bieten auch dem Kunden zukünftig die Möglichkeit, durch Konfigurierung anstelle von Implementierung flexibel auf Neuanforderungen zu reagieren. Die Lösung ist zudem einfach skalierbar und wartbar. In diesem Vortrag werden zusätzlich zu kreativen Performancemaßnahmen auch wenig bekannte Möglichkeiten vom ODI vorgestellt, die so in anderen Werkzeugen nicht zu finden sind.

Kontaktadresse:
Irina Gotlibovych
MT AG
Balcke-Dürr-Allee, 9
D-40882 Ratingen

Telefon: +49 (0) 2102 309 61-0
Fax: +49 (0) 2102 309 61-10
E-Mail irina.gotlibovych@mt-ag.com
Internet: www.mt-ag.com