

Betagt, aber zuverlässig und schnell

Uwe Goldhammer, pdv Technische Automation + Systeme GmbH

Ein großes europäisches Industrieunternehmen stand vor der Herausforderung, den Hochlauf der Serienfertigung seines neuen, komplexen Produkts zu bewältigen. Um dem damit einhergehenden anspruchsvollen Terminplan gerecht zu werden, entstand an den vielzähligen Montagearbeitsplätzen im Werk ein erhöhter Informationsbedarf über die auszuführenden Fertigungsaufträge, der nicht in allen Phasen durch das vorliegende ERP-System abgedeckt werden konnte.

Die bereits existierende Infrastruktur von Oracle-Datenbanken und Oracle-Applikations-Servern bildeten somit eine gute Plattform, um die sich schnell ändernden Anforderungen an ein Auftragsinformationssystem (kurz: AIS) umsetzen zu können. Der Faktor „Zeit“ spielte bei der Bereitstellung dieser Lösung eine entscheidende Rolle.

Abbildung 1 zeigt einen Überblick über die Gesamt-Architektur und die aktuelle Ausbaustufe der Lösung. Herzstück bildet das Data Warehouse (DWH), bestehend aus einem Oracle Real Application Server (RAC) 10g R2 mit fünf Knoten auf Basis von SuSE Linux Enterprise Server 10 (SLES, 64 bit).

Das DWH wird mittels Oracle Warehouse Builder 10g R2 über ETL-Prozesse mit Daten aus den unterschiedlichen operativen Systemen des Herstellers versorgt, insbesondere aus dem nationalen ERP-System SAP R/3. Die Datenaktualisierung erfolgt je nach Art der Datenquelle in Intervallen von fünf bis zehn Minuten, manchmal auch nur einmal täglich. Allerdings enthält das DWH nicht nur Auftragsdaten für das neue Produkt, sondern auch Daten für die unterschiedlichen Varianten der bereits vorhandenen Produktlinien. Allein die Auftragsdaten einschließlich der Peripheriedaten (Arbeitsvorgänge, Stücklisten, Materialstamm etc.) umfassen 50 Tabellen mit durchschnittlich 72 Spalten (Max = 270 Spalten) und 12 Mil-

lionen Datensätze (Max = 210 Millionen Datensätze). Insgesamt belegt das DWH auf dem Storage System einen Plattenplatz von 900 GB.

Als Applikations-Server wurden sechs Oracle Forms & Reports Server 10g R2 in der Standalone-Variante unter Microsoft Windows Server 2003 aufgesetzt, da diese Konstellation in der Konfiguration und Wartung erheblich einfacher zu pflegen ist als mit dem zusätzlichen Oracle Infrastructure Server. Die Lastverteilung übernahm daher ein Apache HTTP-Server als einfacher „Redirector“ im Round-Robin-Verfahren.

Anforderungen

Das mit Oracle Forms & Reports 10g (10.1.2.2.0) entwickelte Auftragsinformationssystem (AIS) wird ausschließlich für das neue Produkt von 1340 registrierten

Benutzern verwendet, die an sechs Tagen im Zwei-Schicht-Betrieb arbeiten. Die Antwortzeiten der zum Teil komplexen Recherchen in den Auftragsdaten sollen maximal zwischen fünf und zehn Sekunden liegen. Durchschnittlich arbeiten über 210 Anwender pro Tag mit dieser Applikation.

Realisierung

Aus diesen Anwenderzahlen und der Anforderung, dass ein operatives System wie das AIS keine direkten Datenänderungen an dem Auftragsdatenbestand im DWH vornehmen darf, resultierte die Maßnahme, nicht nur die Datenbank-Instanz DWH auf einem Oracle RAC 10g aufzusetzen, sondern auch die Datenbank-Instanz AIS selbst, ebenfalls unter SLES 10. Die drei Knoten dieses zweiten RACs werden allerdings wie beim DWH auch durch andere

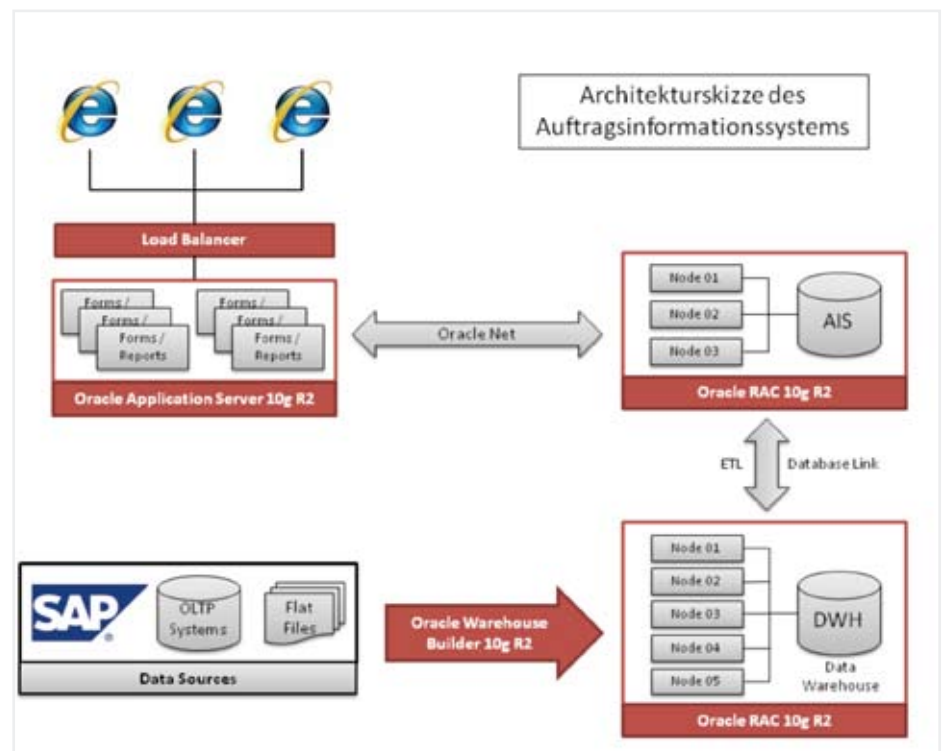


Abbildung 1: Architektur des Auftragsinformationssystems (AIS)

Applikationen ausgelastet, nicht nur durch das AIS allein.

Alle Datenänderungen (beispielsweise zur Erfassung des Fertigungsfortschritts oder von Störungsmeldungen) in den Auftragsdaten müssen somit über eine Schnittstellentabelle abgewickelt werden, die in Intervallen von fünf bis zehn Minuten durch die vom Oracle Warehouse Builder 10g R2 (OWB) kontrollierten ETL-Prozesse verarbeitet und in das DWH übernommen werden.

Dies hatte zwei Konsequenzen: Zum einen konnte nicht mit dem Standard-Record-Locking-Verfahren von Forms gearbeitet werden, zum anderen mussten sämtliche Recherchen in den Auftragsdaten auf die (noch nicht vom ETL-Prozess verarbeiteten) Datenänderungen ausgeweitet werden. Die Übergabe der Datenänderungen aus dem AIS an das DWH erfolgt durch INSERT-Operationen in eine Schnittstellentabelle, die zyklisch alle fünf bis zehn Minuten von einem vom OWB kontrollierten ETL-Prozess verarbeitet werden. Wenn nun im Auftragsinformationssystem dem Anwender aktuelle Daten angezeigt werden sollen, muss im Hintergrund eine MERGE-Operation (genau genommen ein Join) zwischen den Auftragsdaten im DWH und allen (offenen) Datenänderungen im AIS durchgeführt werden, sodass ein Record-Locking durch Forms auf einen einzelnen Schnittstellensatz die Schreib-Lese-Problematik nicht löst. Die Lösung wurde durch eine separate Sperrtabelle (Locking Table) implementiert, in die der Fertigungsauftrag und der Benutzer eingetragen werden, der aktuell eine (Schreib-)Sperrung gesetzt hat. Die Sperrung wird so lange gehalten, wie der Benutzer den Dialog zur Anzeige und Pflege der Auftragsdaten geöffnet hat und somit den Zeitstempel der Sperrung zyklisch im Hintergrund permanent aktualisiert. Ein Timeout von zwei Minuten sorgt zusätzlich dafür, dass diese Sperren regelmäßig (etwa nach einem Sitzungsabbruch) aufgeräumt und entfernt werden.

Der zentrale Dialog zum Starten von Recherchen in den Auftragsdaten stellt dem Anwender etwa fünfzig Suchkriterien bereit, die großteils frei kombiniert werden können und unter Verwendung dynamischer PL/SQL-Abfragen auf rund hundert Tabellen und Views angewendet werden.

Da die Master-Detail-Relationships zwischen diesen Objekten teilweise eine Schachtelungstiefe von sechs bis sieben Ebenen hatten, stellte die Performance-Analyse während der gesamten Entwicklungsphase eine Hauptaufgabe dar, um die geforderten Antwortzeiten von maximal fünf bis zehn Sekunden einhalten zu können.

Die Ergebnisse solcher Recherchen bilden die sogenannten „Trefferlisten“, über die der Anwender dann weiter in die Detail-Informationen zu einem Fertigungsauftrag verzweigen kann. Mit der Aufgabe, die vielfältigen Details zu einem Auftrag aus unterschiedlichen Datenquellen anzuzeigen, wurden weitere Grenzen der Applikationsentwicklung mit Forms ausgelotet. Um der Vielzahl der konzernweit etwa 68.000 Bildschirm-Arbeitsplätze gerecht zu werden, durften die Forms-Dialoge (bei Projektstart) eine Größe von 1024 x 768 Pixel nicht überschreiten. Außerdem sollte der Anwender nicht mehr als zwei bis drei Klicks benötigen, um die gewünschte Detail-Information angezeigt zu bekommen. So wurde unter anderem ein Forms-Dialog entwickelt, in dem etwa 450 Datenfelder (Items) aus sechzig Datenblöcken (Data Blocks) auf dreißig Leinwänden (Canvases) abgebildet sind.

Eine weitere Komplexität stellte die Steuerung der Zugriffsrechte auf diese einzelnen Datenfelder dar. Die erste Ebene in der Rechtevergabe bildet der Dialog selber. Danach folgte als zweite Ebene eine Zusammenfassung (Clusterung) von Lese-/Schreibrechten auf Basis des Registers (Tabs) und der einzelnen Felder (Items), was zusätzlich durch die Zugehörigkeit eines Benutzers zu einer Benutzergruppe (Role) und des angezeigten Fertigungsauftrags zu einem Produkt (mit einer eindeutigen Herstellererienummer) verfeinert werden musste. Die Implementierung dieser Berechtigungen erfolgte auf Basis eines vorhandenen sogenannten „Dialograhmens“, eines selbst entwickelten Frameworks für die Benutzerberechtigungen in Forms-Applikationen, der durch einzelne Steuerungstabellen abgebildet wird.

Berichtswesen

Eine weitere Anforderung bildete der Export der Ergebnisse einer Recherche (die Treffer-

listen) nach Microsoft Excel. Bei der komplexesten Trefferliste müssen 120 Datenfelder ausgegeben werden, die der Anwender gemäß seinen Benutzereinstellungen (User Profile) dynamisch ein- und ausblenden kann. Die Implementierung dieser Aufgabe für die unterschiedlichen Trefferlisten mit Oracle Reports 10g scheiterte aufgrund von Laufzeitfehlern auf dem Oracle Reports Server 10g R2 (Fehlermeldung „REP-69: Internal Error: java.lang.OutOfMemory Error“) bei mehr als 2.500 Ergebnisdatensätzen. Da dieser Oracle-Bug #6327081 gemäß Oracle-Support erst mit dem Forms & Reports Server 11g korrigiert werden sollte, wurde stattdessen eine universelle Funktion auf Basis von Forms WebUtil entwickelt, die dynamisch aus der Kombination der Suchoptionen eines Benutzers und der Liste der eingeblendeten Datenfelder gemäß seinen Benutzereinstellungen eine vorformatierte CSV-Datei erzeugt (CSV = Comma Separated Variables), die anschließend problemlos mit Excel geöffnet werden kann. Die Laufzeitergebnisse unter Verwendung der Funktionen „TEXT_IO“ und „FILE_TRANSFER“ sind mehr als zufriedenstellend.

Soweit die vordefinierten Berichte im PDF- bzw. Excel-Format die Anforderungen nicht abdecken, gibt es zusätzlich die Möglichkeit, dass der Fachbereich unter Verwendung des Oracle Discoverer 10g R2 weitere Berichte und Auswertungen selber zentral bereitstellen kann.

Fazit

Der Erfahrungsbericht aus einem mehrjährigen, erfolgreichen Forms & Reports-Projekt unterstreicht wieder einmal den Wahrheitsgehalt des bekannten Sprichworts „alte Besen kehren gut“, denn auch in Zeiten einer immer stärkeren Marktdurchdringung von J2EE-Applikationen bildet Forms & Reports immer noch eine gute Plattform, um zuverlässige und schnelle Applikationen mit komplexen Aufgabenstellungen und kurzen Entwicklungszeiten zu erstellen.

Kontakt:

*Uwe Goldhammer
goldhammer@pdv-tas.de*