

# Hochleistungs-Real-time-Reporting eines Billing Systems mit Oracle GoldenGate

**Karsten Stöhr**  
**ORACLE Deutschland B.V. & Co. KG**  
**Hamburg**

## Schlüsselworte:

Operational Reporting, Query Off-Loading, Real-time Replication, Datenintegration, Oracle Applications, GoldenGate

## Einleitung

Ein europäischer Mobiltelefonie-Netzanbieter suchte eine Real-Time-Reporting-Lösung, welche in einer Hochleistungsumgebung ein permanentes Datenänderungsvolumen von 1 GB/min in Echtzeit replizieren kann. Eine Besonderheit von Telco-Abrechnungssystemen ist, daß 90% der Änderungsdaten in einer einzigen Tabelle erzeugt werden. Dieser Vortrag stellt vor, wie die Aufgabe mit Oracle GoldenGate auf einer durchschnittlichen Hardware umgesetzt wurde und mit welcher Konfiguration von GoldenGate die notwendige Skalierung erreicht wurde. Oracle GoldenGate unterstützt Operational Reporting für nahezu alle Applikationen, auch besonders transaktionsintensive wie z.B. Billing Systeme, Payment Transaction Processing Systeme, Finanztransaktionssysteme, Online Shops, CRM- und ERP-Systeme. Es ist zertifiziert für Oracle Applications, inklusive Siebel, Peoplesoft, E-Business Suite und JD Edwards.

Oracle GoldenGate unterstützt Operational Reporting durch Replizierung der Geschäftsdaten in ein zweites System, welches dediziert für die Ausführung abfrageintensiver Operationen wie z.B. Reporting zur Verfügung steht. Die Daten werden permanent in nahezu Echtzeit synchronisiert, ohne dabei die tägliche Arbeit mit den Applikationen merkbar zu beeinträchtigen. Diese zertifizierte Lösung bietet Vorteile, die mit einer Instanz allein nicht möglich sind:

- **Resource Off-Loading:** Statt alle Abfragen und Reports auf einem einzigen System auszuführen, können Operationen separaten Systemen zugewiesen werden, die für diese Aufgaben optimiert sind.
- **Optimierung:** OLTP- und Reporting-Systeme arbeiten effizienter, wenn sie ihrer jeweiligen Aufgabe entsprechend optimiert (getuned) sind.
- **Vollständige Nutzungsmöglichkeit der Reporting-Instanz:** Oracle GoldenGate setzt den Operationen auf der Reporting-Instanz keine Grenzen. Reports, welche die Erzeugung von temporären Tabellen oder Prozessen erfordern, können auf der Reporting-Instanz laufen, ohne geschäftskritische Systeme zu beeinträchtigen.
- **Unterstützung heterogener Umgebungen:** Wenn die Applikationen auf anderen relationalen Datenbanken laufen sollten, z.B. Microsoft's SQL Server oder IBM DB2, können die heterogenen Fähigkeiten Oracle GoldenGates genutzt werden, um z.B. für die operationale Reportinginstanz eine Oracle Datenbank zu verwenden.

## **Reports zur rechten Zeit**

Viele Anwender würden gerne von den Vorteilen von Echtzeitdaten aus geschäftskritischen Anwendungen in ihren Berichten profitieren, schrecken aber vor der möglichen Beeinträchtigung der täglichen Arbeit mit denselben Anwendungen durch die zusätzlich generierte Last zurück. Zum Beispiel können sehr viele Aktivitäten zum Ende jedes Quartals oder Jahres anstehen. Die Generierung von Berichten über abgeschlossene und offene Aufträge - während dieselben Aufträge aktualisiert werden – birgt das Risiko die Bearbeitung zu verlangsamen.

Eine Möglichkeit, die Belastung der Systeme zu vermeiden, besteht darin, die Reportgenerierung vom primären System auf eine separate dedizierte Reportinginstanz zu verschieben. Dies erfordert jedoch eine regelmäßige Synchronisation der dedizierten Instanz mit dem primären System, welche in der traditionellen Ausführung, Daten in Stapeln vom primären System zu extrahieren, ihrerseits wiederum soviel Last auf dem primären System erzeugen kann, daß die Bearbeitung beeinträchtigt wird. Daher wird die Datensynchronisation zumeist auf nächtliche Stunden außerhalb der üblichen Arbeitszeit verschoben, woraus resultiert, daß die Berichte nie aktuelle Daten enthalten.

Die Umsetzung von Operational Reporting erfährt nun Unterstützung durch Oracle GoldenGate, welches Zugriff auf Daten aus diversen Applikationen in nahezu Echtzeit ermöglicht, ohne dabei die tägliche Arbeit mit den Applikationen merkbar zu beeinträchtigen.

## **Datensynchronisation mit Oracle GoldenGate**

Oracle GoldenGate gestattet die Erfassung, Weiterleitung, Transformation und Übergabe von Datenänderungen zwischen diversen Applikationen und Systemumgebungen. Die Software nutzt eine lose gekoppelte Architektur, um hohe Volumina von Datenänderungen zwischen heterogenen Datenbanken in Sekundenbruchteilen zu bewegen, unter Bewahrung der Transaktionsintegrität.

Wie im nachfolgenden Diagramm gezeigt, besteht die Oracle GoldenGate Architektur aus den zwei eigenständigen Komponenten Capture und Deliver, die durch die Trail Files lose verbunden sind, so daß beide ihre Aufgaben unabhängig von der anderen ausführen können, um eine rasante, nahtlose Datenreplikation zu erreichen, welche wichtig ist für die Unterstützung der Koexistenz von mehreren Versionen oder Instanzen einer Applikation.

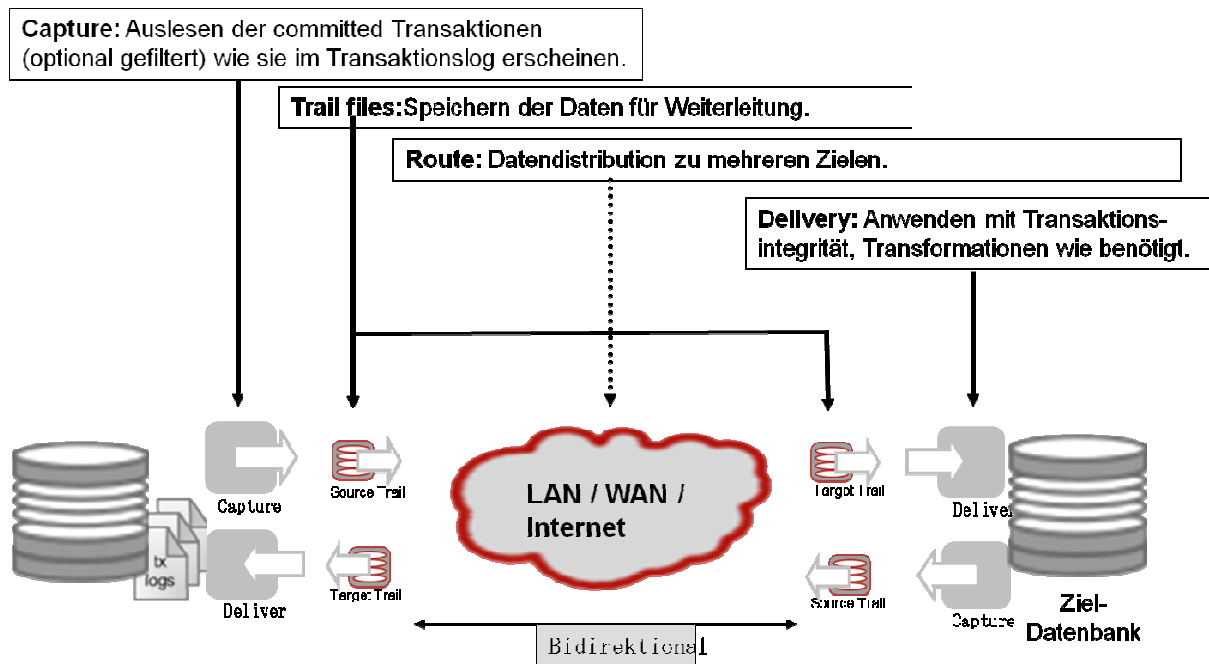


Abb. 1: Oracle GoldenGate Architektur

## Oracle GoldenGate Capture

Die Oracle GoldenGate Capture Komponente befindet sich eng neben der Quelldatenbank und überwacht deren Transaktionslog (Oracle: Redo log) auf neue insert, update und delete Operationen. Jede neue Operation wird umgehend erfasst und gegebenenfalls mit einem konfigurierbaren Filter verglichen. Capture ist verfügbar für eine Vielfalt diverser Datenbanken, darunter Oracle Database, Microsoft SQL Server, IBM DB2 (auch Mainframe), MySQL, Sybase, Enscribe, SQL/MP, SQL/MX und Teradata.

Die Oracle GoldenGate Capture-Komponente erfasst ausschließlich abgeschlossene (committed) Transaktionen, keine Zwischenaktivitäten oder zurückgerollte Transaktionen. Auf diese Weise wird nicht nur die zu transportierende Datenmenge reduziert sondern auch potenziellen Dateninkonsistenzen vorgebeugt.

## Oracle GoldenGate Trail Files

Trail Files, ein integraler Bestandteil des Oracle GoldenGate eigenen Warteschlangenmechanismus, speichern die Datenänderungen in einem transportablen, plattformunabhängigen, universalen Datenformat. Trail Files werden idealerweise auf Quell- und Zielsystem angelegt, existieren aber außerhalb der Datenbanken, um Heterogenität zu gewährleisten, die Zuverlässigkeit zu erhöhen und den Datenverlust zu minimieren. Diese Architektur reduziert eine etwaige Beeinträchtigung des Quellsystems, da keine zusätzlichen Tabellen oder viele Abfragen der Datenbank benötigt werden. Die Capture-Komponente liest und schreibt die erfassten Änderungen immer nur einmal, unabhängig davon, ob die Daten aus den Trail Files zu einem oder mehreren Zielen weitergeleitet werden sollen.

## **Oracle GoldenGate Deliver**

Die Oracle GoldenGate Deliver-Komponente liest die Daten aus dem jüngsten Trail File und übergibt sie der Zieldatenbank unter Verwendung des nativen SQL für das gewünschte RDBMS. Jede Transaktion wird in derselben Reihenfolge und innerhalb derselben Transaktionsgrenzen, wie sie auf der Quelle entstanden, an das Ziel abgeliefert. Dabei verwendet die Deliver-Komponente gewisse Optimierungstechniken für die Datenübergabe, z.B. kann durch Gruppierung der Transaktionen die Commit Rate reduziert werden. Auch die Deliver-Komponente ist verfügbar für eine Vielzahl diverser Datenbanken. Zusätzlich können die Datenänderungen als Flat Files geschrieben werden oder in verschiedenen Formaten, z.B. XML, zu Enterprise Messaging Systemen publiziert werden.

## **Vorteile von Operational Reporting Lösungen mit Oracle GoldenGate**

Reporting-Lösungen unter Verwendung von Oracle GoldenGate bieten die folgenden Vorzüge gegenüber dem Reporting auf dem Produktionssystem:

### **Lastverteilung und Ressourcenschonung**

Anstatt alle Reports und andere ressourcenintensive Operationen auf demselben Primärsystem auszuführen, können die Operationen auf andere Systeme verteilt werden, um zusätzliche Last auf der Produktionsumgebung weitgehend zu vermeiden.

### **Optimierung**

Reportingsysteme arbeiten effizienter wenn sie speziell für diesen Anwendungsfall optimiert sind. Große Sortierbereiche, Partitionen und verschiedene Arten der Indexierung sind besser geeignet für Reportingsysteme als für transaktionale Systeme.

### **Vollständige Verwendbarkeit der Reportinginstanz**

Oracle GoldenGate setzt den Operationen, die auf der Reportinginstanz ausgeführt werden können, keine Grenzen. Alle erforderlichen Operationen, inklusive der Anlage von temporären Tabellen oder sonstiger Prozesse, die zur Aufbereitung der Daten nötig sind, können auf der Reportinginstanz ausgeführt werden, ohne das primäre Anwendungssystem zu beeinträchtigen.

### **Erweiterungsoptionen**

Operationale Reporting-Lösungen sind zumeist für einen rein lesenden Zugriff ausgelegt und werden unidirektional eingesetzt, d.h. Daten fließen nur in einer Richtung vom transaktionalen System zur Reportinginstanz, auf welcher die Abfragen ausgeführt werden. Bei der Verwendung von Oracle GoldenGate steht zusätzlich die Option zur Verfügung, auch bidirektionale Synchronisation von einzelnen oder ganzen Gruppen von Tabellen einzurichten.

### **Freie Wahl der Reporting-Lösung**

Operationales Reporting unter Verwendung von Oracle GoldenGate läßt dem Anwender die freie Wahl, ob er separate Reportingwerkzeuge wie z.B. Oracle BI-Publisher oder die in den Applikationen enthaltenen Werkzeuge einsetzen möchte.

### **Von Oracle zertifiziert**

Oracle GoldenGate ist getestet und zertifiziert für den Einsatz mit führenden Oracle Applications inklusive E-Business Suite, JD Edwards, PeopleSoft und Siebel CRM. Die Entwicklerteams dieser Applikationen haben überprüft, daß Oracle GoldenGate die Arbeit mit Oracle Applications weder negativ beeinflusst noch gefährdet.

## Unterstützung heterogener Umgebungen

Wenn die Applikationen auf anderen relationalen Datenbanken laufen sollten, z.B. Microsoft's SQL Server oder IBM DB2, können die heterogenen Fähigkeiten Oracle GoldenGate genutzt werden, um z.B. für die operationale Reportinginstanz eine Oracle Datenbank zu verwenden.

## Operational Reporting mit Oracle GoldenGate

Oracle GoldenGate unterstützt operationales Reporting durch Replikation der Geschäftsdaten zu einem sekundären System, welches dediziert für leseintensive Aufgaben verwendet wird. Dabei liest Oracle GoldenGate die Datenänderungen aus dem Transaktionslog der Datenbank und nicht aus den Datentabellen selbst. Diese Lösung bietet eine kompakte, non-intrusive und leicht zu konfigurierende Methode für den Zugriff auf die Anwendungsdaten in nahezu Echtzeit und gestattet den Aufbau hochoptimierter Reportingapplikationen.

Wenn die Applikationen auf anderen relationalen Datenbanken laufen sollten, z.B. Microsoft's SQL Server oder IBM DB2, können die heterogenen Fähigkeiten Oracle GoldenGate genutzt werden, um z.B. für die operationale Reportinginstanz eine Oracle Datenbank zu verwenden.

Die nachfolgende Grafik illustriert das Konzept einer typischen Operational Reporting-Lösung für eine beliebige Applikation, in unserem Fall ein Billing-System. Die Lösung repliziert transaktionale Daten aus der Quell-Datenbank in Sekundenbruchteilen in eine separate Reportinginstanz, welche dediziert für abfrageintensive Aufgaben zur Verfügung steht.

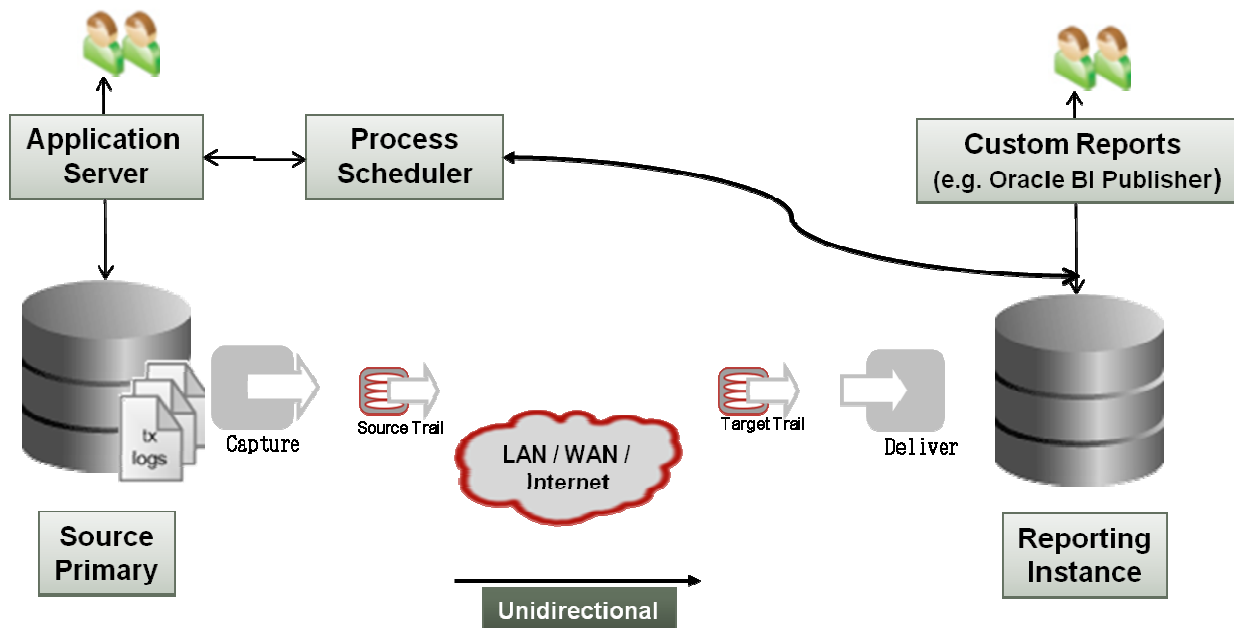


Abb. 2: Operational Reporting mit Oracle GoldenGate

Das operationale Reporting für diesen Anwendungsfall wurde so entworfen, daß alle Tabellen in den Applikationsschemata repliziert werden, um so die Reportingfunktionen auf eine sekundäre non-OLTP-Instanz verlagern zu können. Reports können wahlweise direkt auf dem Zielsystem erstellt und ausgeführt werden oder über einen Process Scheduler der Applikation, welcher so konfiguriert wird, daß er auf die Reportinginstanz zeigt.

## Skalierung für Hochleistungssysteme

Bei steigender Anzahl von Änderungsdaten pro Sekunde, muß die Replizierung entsprechend skalieren können. Oracle GoldenGate bietet dazu die Möglichkeit mehrere der Capture- und Deliver-Komponenten einschließlich der verbindenden Trails parallel auszuführen. Dabei zeigt die Erfahrung, daß in der Regel der Engpaß auf der Zielseite der Replikation entsteht. Das Lesen der Änderungsdaten aus dem Log der Quell-DB erfordert kaum Ressourcen. Auch der Transport der Daten über das Netzwerk geht, genügend Bandbreite vorausgesetzt, schnell genug. Auf der Ziel-DB müssen dagegen alle Transaktionen ausgeführt werden, was Zeit und Ressourcen kostet. Dabei wird die Datenbank selbst beansprucht, die GoldenGate Deliver-Komponente hat nicht viel zu tun, sie übergibt nur die Operationen an die Datenbank.

Die Ziel-Datenbank ist jedoch gut dafür eingerichtet, mehrere Transaktionen parallel auszuführen. Diese Fähigkeit muß nur durch entsprechende Konfiguration von Oracle GoldenGate genutzt werden, d.h. es müssen mehrere Deliver-Komponenten gestartet werden, um parallele Sessions zur Ziel-DB zu nutzen.

Die folgende Grafik veranschaulicht die Skalierungsmöglichkeiten von Oracle GoldenGate in typischen Konfigurationen. Auf diese Weise kann GoldenGate auch nachträglich mit steigenden Anforderungen mitwachsen.

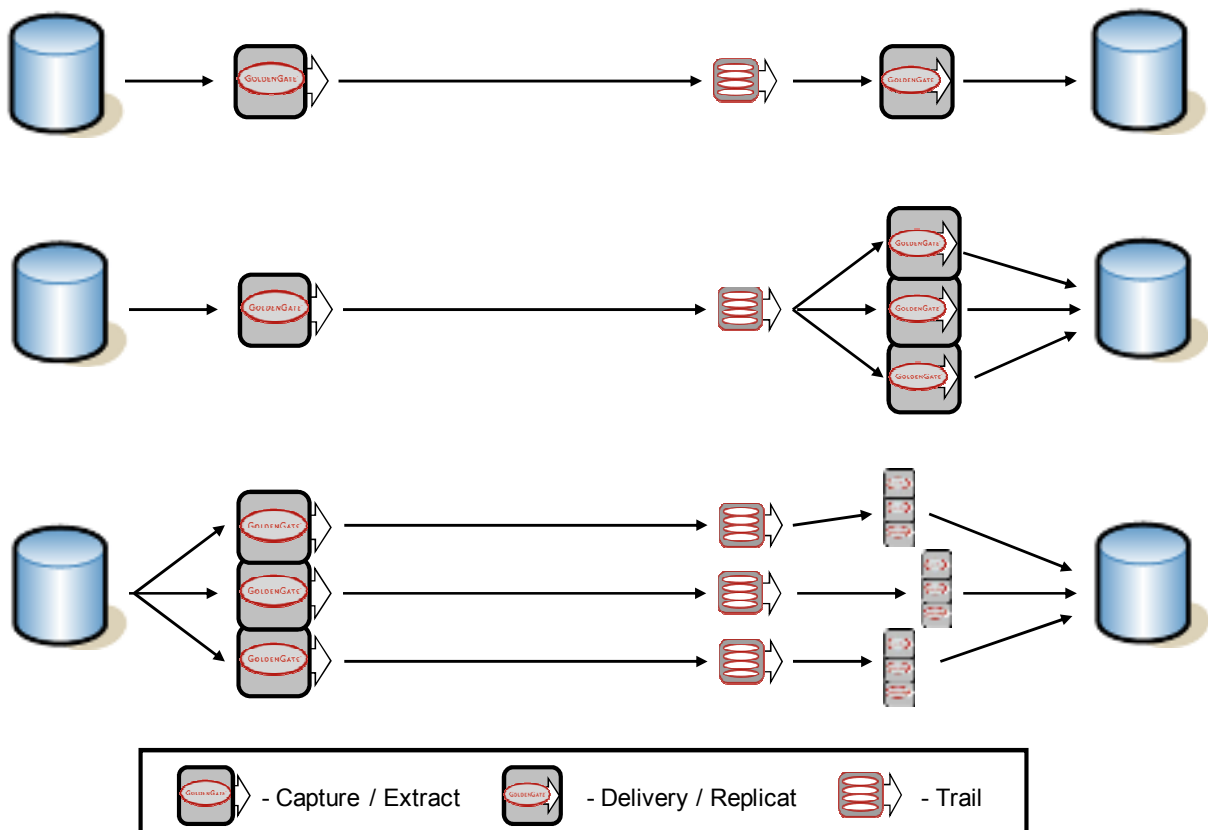


Abb.3: Typische Skalierung einer Oracle GoldenGate Replikation

## Skalierung auf einer einzelnen Tabelle

Bei der Verwendung paralleler Komponenten wird zunächst die Last nach Schemata oder Tabellen aufgeteilt. Doch auch die Änderungen einer einzelnen großen Tabelle können auf parallele Prozesse aufgeteilt werden. Neben der starren Aufteilung über geeignete Filter, z.B. alle Nachnamen von A-K auf Prozess 1 und von L-Z auf Prozess 2, bietet Oracle GoldenGate auch eine sehr leistungsfähige Funktion namens @range.

Die @range-Funktion ordnet automatisch auf Basis der Werte eines konfigurierbaren Satzes von Feldern der betrachteten Tabelle die Operationen auf dieser Tabelle einer gewünschten Anzahl von parallelen Prozessen zu. Die @range-Funktion erleichtert ihren Einsatz noch weiter mit automatischen Voreinstellungen. So gilt der Primärschlüssel der Tabelle als automatisch vorausgewähltes Kriterium, wenn nicht explizit die gewünschten Felder konfiguriert werden, die als Kriterium für die Aufteilung dienen sollen.

Auf Basis der Werte der gewählten Felder berechnet die @range-Funktion einen Hashcode, welcher zwischen 1 und der gewählten Anzahl paralleler Prozesse liegt. Da dieselben Werte der Felder immer zu demselben Hashcode führen, ist sichergestellt, daß Updates auf denselben Record immer vom selben Prozess bearbeitet werden, so daß die Reihenfolge der Operationen auf demselben Record gewahrt bleibt.

Hier ist ein Beispiel für die Verwendung der @range-Funktion in einer Zeile der GoldenGate Konfiguration:

```
TABLE BILLING.CHARGE, FILTER ( @RANGE ( 1, 3, CHARGE_ID ) );
```

Die Replizierung der Änderungsdaten für die betrachtete Tabelle BILLING.CHARGE werden auf drei Prozesse aufgeteilt. Die oben dargestellte Zeile wird im ersten der drei Prozesse verwendet, daher die ,1‘ als erster Parameter. Der zweite Parameter ,3‘ bezeichnet die Anzahl aller parallelen Prozesse. Das Feld CHARGE\_ID ist die Basis für die Generierung des Hashcodes, also für die Aufteilung auf die drei Prozesse. Die Konfigurationen der beiden anderen parallelen Prozesse enthalten entsprechende Zeilen, wobei nur der erste Parameter dann auf ,2‘ und ,3‘ wechselt.

## Operational Reporting für ein Hochleistungs-Billing-System

Bei vielen in der Telekommunikationsbranche verwendeten Billing-Systemen entstehen über 90% der Last und der Änderungsdaten in einer einzigen sehr großen Tabelle. Eine Parallelisierung durch Aufteilung nach Tabellen bringt daher keinen Geschwindigkeitszuwachs. Andererseits sind die Buchungen der Telefoneinheiten eines Kunden unabhängig von den gebuchten Einheiten der anderen Kunden. Daher eignet sich diese Tabelle sehr gut für eine Aufteilung nach Schlüsselwerten wie im vorhergehenden Abschnitt beschrieben.

Die Anforderung des Kunden war die Änderungsdaten aus einem permanent auf der Quelle erzeugten Logvolumen von bis zu 1 GByte pro Minute replizieren zu können. Dies entsprach einem künftig zu erwartenden Volumen, nicht dem aktuellen Logvolumen des Produktsystems des Kunden. Daher wurde für den Test mittels Swingbench ein Load Generator erstellt, welcher durch wiederholte Ausführung von inserts und updates entsprechende Last auf der Quelle erzeugte.

Die verwendete Systemumgebung sah wie folgt aus:

### Quellsystem

- Oracle RDBMS 9.2.0.8 RAC mit 2 Knoten
- Sun M5000 Server, hard partitioniert in zwei Maschinen mit je 4 Dual-CPU's und 32 GB Hauptspeicher
- Alle Daten auf einem Piller SAN Storage-System, welches über eine 2\*4 Gbit Netzwerkverbindung angeschlossen war

### Zielsystem

- Oracle RDBMS 11g RAC mit 4 Knoten
- Pro Knoten 2 Quad-core CPU's und 32 GB RAM
- Storage: 12 \* 300 GB SAS Festplatten

### **Testdurchlauf und –ergebnisse**

- Der Lastgenerator lief kontinuierlich über 75 Minuten
- Erzeugte Archived Logs auf der Quelle:
  - Node1: 188 – 206 = 19 \* 2 GB = 38 GB
  - Node2: 241 – 260 = 20 \* 2 GB = 40 GB
- Dies entspricht einer Rate von 1 GB / min = 60 GB / hr
- Insgesamt mehr als 40 Mio. Operationen:
  - 19 Mio. Inserts in Tabelle CHARGE
  - 19 Mio. Updates in Tabelle CHARGE
  - 1 Mio. Inserts in andere Tabellen
  - 1 Mio. Updates in andere Tabellen

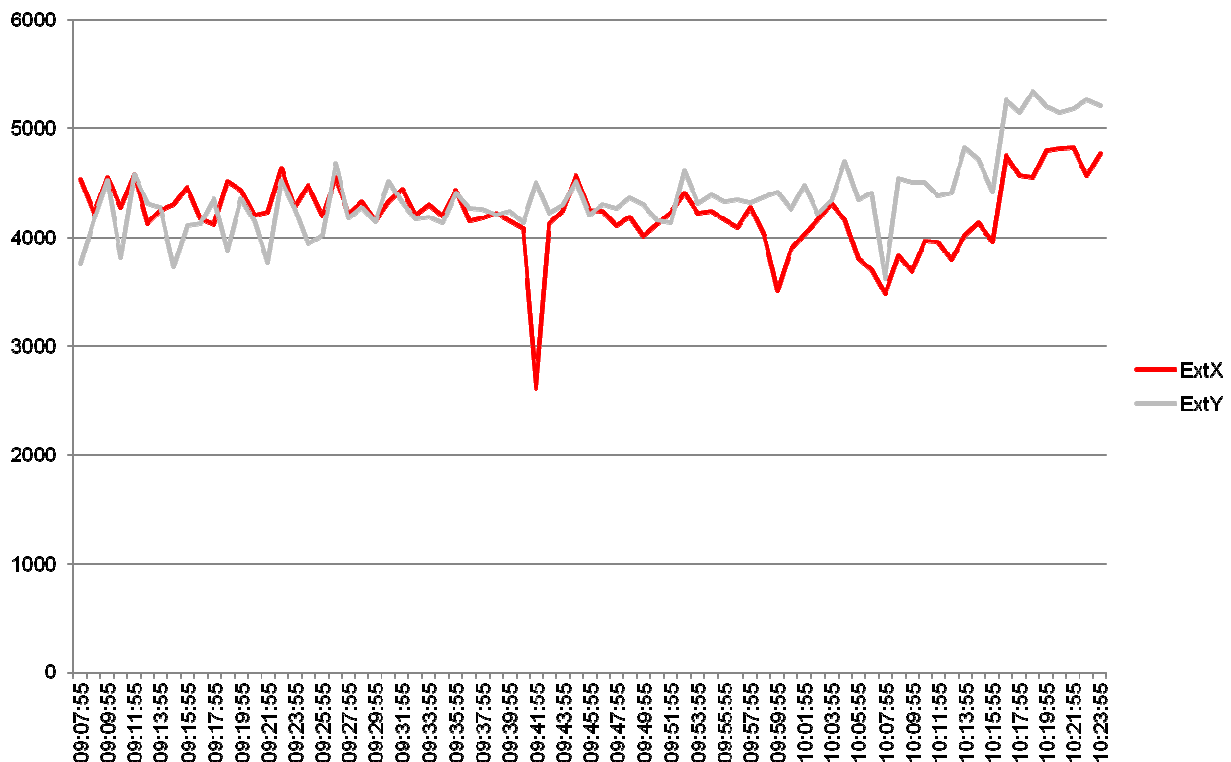


Abb.4: Performanz des GoldenGate Capture auf beiden Knoten des Quellsystems.



Der Graph der Capture-Prozesse für die beiden Knoten des Quellsystems zeigt eine relativ gleichmäßige Last, welche aus der Erzeugung der Daten durch den Load Generator folgt. Zu einem Zeitpunkt brach die Performanz des Capture kurzzeitig auf einem der beiden Knoten ein. Im Rahmen des Tests konnte nicht geklärt werden, ob der Datenbankknoten selbst oder der Capture-Prozess hier kurz gestört wurde.

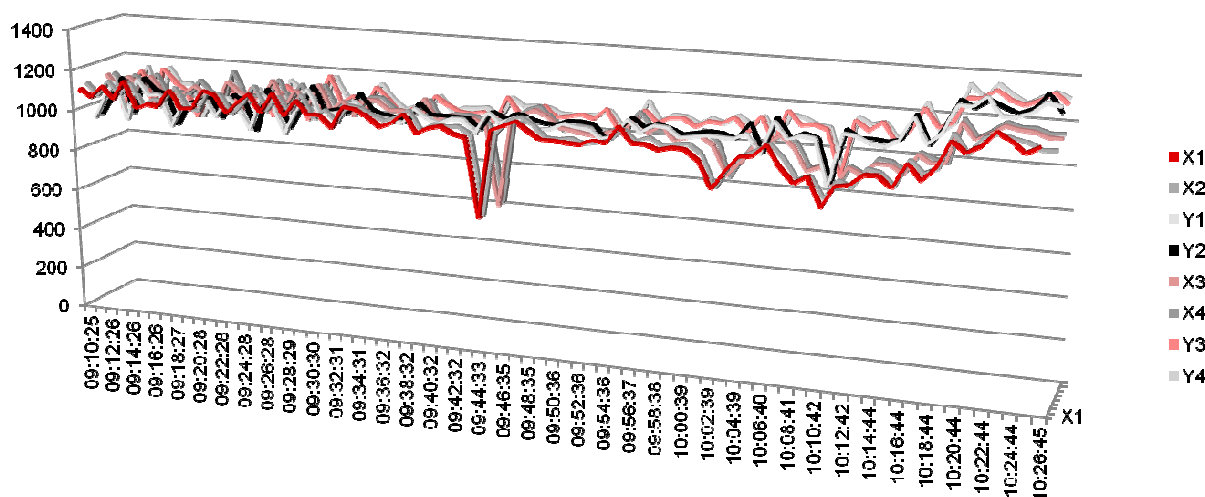


Abb.5: Performanz der GoldenGate Deliver-Prozesse auf dem Zielsystem

Die extrahierten Änderungsdaten wurden mittels insgesamt 8 GoldenGate Deliver-Prozessen an die Ziel-Datenbank übergeben, also mit 8 parallelen Sessions. Wir sehen bei vier der Prozesse wiederum den gleichen kurzen Abfall in der Performanz, was belegt, daß die Deliver-Prozesse tatsächlich in nahezu Echtzeit die angebotenen Daten verarbeiten konnten. In dem Moment, als einer der beiden Capture-Prozesse kurz langsamer lief, mußten auch die Deliver-Prozesse kürzer treten, da einfach nicht mehr Daten von der Quelle geliefert wurden.

Die Tests wurden noch mit der vorhergehenden Version 10.4 von Oracle GoldenGate durchgeführt. Mit dieser Version haben diverse Unternehmen aus verschiedenen Branchen bereits permanente Logvolumen von 1 – 1,5 GByte pro Minute repliziert. Referenzen dafür finden sich auf Oracle’s Webseiten.

Billingsysteme der Telekommunikationsindustrie stellen besondere Anforderungen an die Skalierbarkeit dadurch, daß die weitaus größte Menge der Operationen – oftmals mehr als 90 Prozent – auf einer einzigen Tabelle stattfinden. Doch auch unter diesen besonderen Umständen kann Oracle GoldenGate soweit skalieren, auch diese Daten in Hochleistungsumgebungen weiterhin in nahezu Echtzeit zu replizieren.

Die aktuelle Version GoldenGate 11g bringt weitere Verbesserungen der Performanz. So beherrscht die Capture-Komponente nun ein Read-ahead des Transaktionslogs und die Deliver-Komponente kann die Ausführung der Transaktionen mittels „commit nowait“ veranlassen. Verbesserte Servertechnologien, z.B. der Oracle Exadata-Maschinen, bringen ebenfalls einen Leistungsschub.

Aktuelle Tests von Oracle GoldenGate 11g auf einer Oracle Exadata weisen darauf hin, daß auch die Replizierung von 2 GB pro Minute und mehr realistisch erreicht werden kann.

### **Zusammenfassung**

Mit Hilfe der flexiblen Architektur von Oracle GoldenGate können zertifizierte Lösungen implementiert werden, welche es gestatten, Daten aus Oracle Applications und beliebigen anderen Applikationen in nahezu Echtzeit mit äußerst geringer Belastung der Quellen in dedizierte Systeme zu replizieren, welche speziell für das Reporting optimiert wurden. Zusätzlich können Daten zu weiteren Systemen gesendet werden, welche Zugriff auf die Applikationsdaten benötigen. Dabei wird die tägliche Arbeit mit den Applikationen nicht durch Abfragen beeinträchtigt.

Die Skalierbarkeit von Oracle GoldenGate ermöglicht es, operationales Reporting auch von Quellsystemen mit sehr großen Datenänderungsvolumen – auch unter besonderen Bedingungen wie bei den Abrechnungssystemen – mit Datenreplizierung in nahezu Echtzeit zu implementieren.

Kontaktadresse:

Karsten Stöhr  
ORACLE Deutschland B.V. & Co. KG  
Kühnehöfe 5  
D-22761 Hamburg

Telefon: +49 (0) 40-89091 117  
Fax: +49 (0) 40-89091 250  
E-Mail: karsten.stoehr@oracle.com  
Internet: [www.oracle.com/goto/goldengate/](http://www.oracle.com/goto/goldengate/)