

High-End DB-Applikationen für alle Kunden

Möglichkeiten zur Lizenzkostenreduzierung

Dr. Kurt Franke

Cellent Finance Solutions AG

Kurt.Franke@cellent-fs.de, Kurt-Franke@web.de

Agenda

- cfs Kurzvorstellung
- High-End-DB-Applikation
 - Characteristica
 - Anforderungen an Datenbanksoftware
 - Kundenbezug
- 2 Möglichkeiten zur Erreichbarkeit aller Kunden
- Möglichkeit A: abgestuftes Downsizing
- Möglichkeit B: Embeded Oracle Software Licence
- Gegenüberstellung Downsizing ↔ ESL
- Zusammenfassung

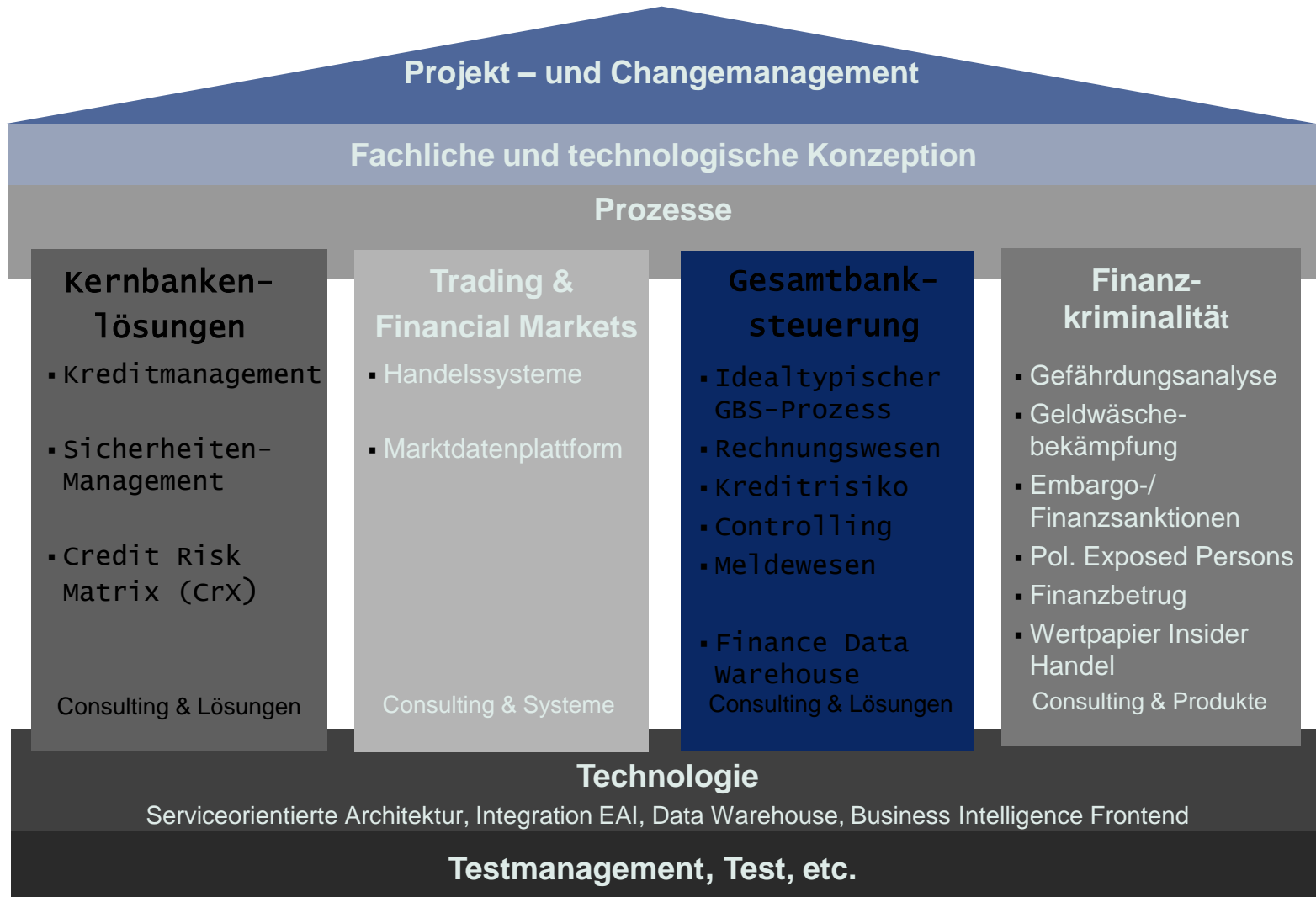
Cellent Finance Solutions auf einen Blick



Wissen – Können - Machen

Firmensitz:	Stuttgart
Marktspräsenz:	seit 1985
Aktionärsstruktur:	100% Landesbank Baden-Württemberg (LBBW)
Aufsichtsrat:	Rudolf Zipf (Finanz-Vorstand der LBBW)
Anzahl Mitarbeiter:	200
Standorte:	Stuttgart und München
Umsatz:	€ 27 Mio.

Das Angebot der Cellent Finance Solutions



Bekämpfung von Finanzkriminalität

Kundenstamm von mehr als 800 Instituten in 50 Ländern

- Geldwäschebekämpfung
- Gefährdungsanalyse
- Bekämpfung der Terrorismusfinanzierung
- Embargoüberwachung
- Kundenscreening
- PEP-Identifikation
- Bekämpfung von Finanzbetrug
- Wertpapier-Insiderhandel



Referenzen SMARAGD Anti-Finance Crime Suite (Auszug)



Characteristica einer High-End-DB-Applikation

- Vorhaltung großer gespeicherter Gesamtdatenmenge
- hoher Datendurchsatz
 - große Lademengen pro Tag
 - Integration mit großem Anteil bereits früher geladener Daten
 - große Mengen selectiv zu bereinigender Daten
 - teilweise Subsets aus allen bisherigen Ladezeiträumen
 - ➔ hochperformante Massenverarbeitung
 - findet im Batchbetrieb statt
- hochselective Zugriffe über Tool
 - zur Bearbeitung von Inhalten
 - nur geringe Wartezeiten im Sekundenbereich tolerierbar

Anforderungen an die Datenbank

- performante Massenverarbeitung hoher Datenmengen
und
- schnelle selective Tool-Zugriffe

erreichbar durch

- Parallelisierung von SQL und Verarbeitungslogik
und
- Reduzierung der IO-Mengen für beide Fälle

Anforderungsliste an Datenbank-Software

- SQL-Parallelisierung
 - → Enterprise Edition
- Reduzierung der IO-Mengen
 - → Datenkomprimierung
 - Enterprise Edition ausreichend bei Direct Load Kompression (Batch)
 - sonst zusätzlich kostenpflichtige Option
 - → Indexkomprimierung (Key Compression)
 - auch mit Standard-Edition möglich
 - → Bitmap-Indexe wenn sinnvoll
 - → Enterprise Edition
 - → Partitionierung von Tabellen und Indexen
 - → Enterprise Edition plus kostenpflichtige Option
 - → spezieller Datenlayer optimiert für Tool-Zugriff
 - in jeder Edition implementierbar, Teil der Applikation

Applikation und Kunden

- Applikation benötigt Enterprise Edition + Partitioning Option
 - für Kunden mit höheren Datenmengen
 - für die sie entwickelt wurde
 - liegt bei diesen mit Oracle-Lizenzkosten im üblichen Rahmen
- aber:
 - bei Kunden mit kleineren Datenmengen
 - mit Standard-Edition als passende Variante
 - Oracle-Lizenzkosten liegen deutlich über dem üblichen Rahmen
 - eventuell Hinderungsgrund für Applikationseinsatz
 - auch wenn fachlich optimal passend
 - großer Anteil der potentiellen Kunden fällt weg

Wie können diese Kunden erreicht werden ?

- 2 mögliche Vorgehensweisen
- A) abgestuftes Downsizing der Applikation
 - zusätzlich zur High-End-Variante
 - bis auf Ebene Standard-Edition
 - für jeden Kunden das passende Angebot
- B) Einsatz der Embedded Oracle Software Licence
 - bei allen Kunden, die keine High-End-Variante benötigen
 - Applikation ist identisch zur High-End-Variante
 - Einschränkung:
 - keine volle Oracle Lizenz für EE + Partitioning
 - deutliche Kostenreduzierung
 - Kunde darf Applikationsdatenbank nicht anderweitig nutzen
 - kein direkter SQL-Zugriff

Möglichkeit A: abgestuftes Downsizing

- keine Unterschiede für Applikations-Zugriff auf Datenbank
- aber:
 - unterschiedliche Varianten der Datenbank-Applikation
 - durch Verwendung bzw. Nichtverwendung von DB-Features
 - Mehraufwände durch die verschiedenen Varianten
 - eine Kernversion der Applikation
 - mit unterschiedlichen Zusatzkomponenten für DB-Feature-Verwendung
 - zur Vermeidung des Auseinanderlaufens bei Fortentwicklung
 - Switch auf andere Variante
 - Oracle Edition kann nicht einfach geändert werden
 - Erstellung DB in neuer Edition und Einspielen eines Backups
 - bei Änderung bestimmter Features nicht einfach möglich
 - z. B. partitionierte auf nicht-partitionierte Tabellen und vice versa
 - Einsatz der DBMS-Re-Definition Methodik
 - unabhängig von der Switch-Richtung immer in EE

Möglichkeit A: DB Applikation

- PL/SQL Applikation in der Datenbank
 - hauptsächlich für Batch-Prozessing
 - nur eine einzige Implementierung
 - mit unterschiedlichen Komponenten in variantenabhängigen Teilen
 - wo nicht dynamischer Code aus anderen Gründen notwendig

- Einsatz der Conditional Compilation

- viele Probleme werden bereits beim Compilieren entdeckt

- Steuerungspackage analog zu dbms_db_version

```
CREATE OR REPLACE PACKAGE my_app_variant AS BEGIN
    use_compression CONSTANT BOOLEAN := TRUE;
    use_partitioning CONSTANT BOOLEAN := FALSE;
END my_app_variant;
```

- Verwendung:

```
IF my_app_variant.use_partitioning THEN
    INSERT into TEMP-Table
    EXCHANGE PARTITION with target table
ELSE INSERT into target table directly $END
```

Möglichkeit A: SQL Installationsscripte

- mit Datamodeller Tool generierte SQL-Scripte
 - → es sind verschiedene Varianten notwendig (kein IF möglich)
- nur ein Datenmodell für komplexeste Variante
 - nur das komplexeste Script wird direct generiert
 - weniger komplexe durch Herausschneiden von SQL-Klauseln
 - mit Tools wie sed, awk, perl etc.
 - ggf. auch mit Ersetzungen
 - idealerweise im Modell bereits alle Information auch auf Tabellenebene
 - z. B. Tablespace-Zuordnung
 - damit lassen sich komplexe Ersetzungen vermeiden
 - Script-Aufbereitung idealerweise in Build-Prozess integriert
 - im kundenspezifischen Teil wird benötigte Variante definiert
 - damit wird auch PL/SQL-Steuerungspackage zusammengestellt

Möglichkeit B: Embedded Oracle Software Licence

- nur eine einheitliche Applikation, keine Varianten
 - kein zusätzlicher Entwicklungsaufwand für Applikation
- Anforderungen von Oracle:
 - keine sichtbare Oracle-Installation
 - Oracle Universal Installer im Silent-Mode mit Responsefile
 - als Teil der Applikations-Installation
 - Kunde darf keinen directen Datenbankzugriff haben
 - nur über Applikation
 - Datenbank-Administration
 - mit Applikation
 - oder mit separatem Tool
 - in mehreren Applikationen einsetzbar
 - für Flexibilität muss SQL-Script-Ausführung möglich sein
 - nur vom Applikationshersteller bereitgestelltes SQL
 - signiert zur Vermeidung von Kunden-Änderungen

Möglichkeit B: Embedded Oracle Installation

- Gesamt-Installation
 - Applikation und oracle-Software und Datenbank
 - Anlegen der erforderlichen User und Groups auf OS-Ebene
 - Aufruf als root (Unix Systeme) bzw. Administrator (Windows)
- Installation der Oracle Software
 - passende Variante für Zielplattform
 - mit Oracle Universal Installer im Silent Mode
 - als oracle User
 - Verwendung eines Responsefiles
 - Modifizierung des Default-Platform-Responsefiles
 - Anpassung Pfade an Zielsystem
 - automatisches Ausführen der root-Scripte unter Unix
 - Einrichten Autostart beim Boot der Maschine
 - rc-Scripte unter Unix passend zur Variante
 - Services unter Windows

Möglichkeit B: Embedded Oracle DB Installation

- 2 Möglichkeiten
- Methode 1
 - Installation einer leeren Datenbank
 - mit anschließender Installation der Datenbank-Applikation
- Methode 2
 - komplette Vorinstallation der Datenbank-Applikation im Hause
 - ggf. Konvertierung mit „rman convert“ auf Zielplattform
 - hierbei keine Endianess-Konvertierung möglich
 - je eine Quellplattform für Big-Endian und Little-Endian
 - Auslieferung der Datenfiles und init.ora mit geeigneten Parametern
 - Bereitstellung Script mit „CREATE CONTROLFILE“
 - Anpassung Pfade an Zielsystem

Möglichkeit B: Embedded Oracle DB Installation

- Methode 2 Fortsetzung
 - Erzeugung Datenbank aus gelieferten Datenfiles
 - mit bereitgestelltem „CREATE CONTROLFILE“ Script
 - inclusive neuer DB-Id – Aufruf `dbms_backup_restore.zerodbid(0)`
 - Anlegen der Tempfiles
 - Neu-Compilieren aller Datenbank-Code-Objecte (`utlirp.sql / utlrp.sql`)
 - ermöglicht Wechsel 64-Bit auf 32-Bit und vice versa
 - Anpassen der Directory-Pfade an Zielsystem
 - sowie Herausschreiben der neuen Pfade in Dateien
 - und Erteilung notwendiger Java-Rechte auf die Pfade
 - Einrichten Disk-Online-Backupjob in SYS-Schema
 - mit bereitgestelltem `rman`-Script als Datafilecopy
 - inclusive Controlfile Autobackup
 - und Block Change Tracking für optimale Performance
 - Erstellung `spfile` und Restart mit `spfile`
 - unter Unix Eintrag in `/etc/oratab`
 - erster Backup asynchron starten

Möglichkeit B: Embedded Oracle DB Installation

- Tool für DB-Administration
 - Applikations-Upgrades
 - operative Applikations-Aufgaben
 - Wartungsaufgaben
 - allgemeine DB-Administration
- Tool führt nur signierte SQL-Files aus
 - alle bereitgestellten SQL-Files sind signiert
 - automatisch im Rahmen des Build-Prozesses
 - Datenbank-Passwörter sind verschlüsselt in Konfigurationsdateien
 - werden dem Kunden nicht mitgeteilt
- Tool muss für Zielsystem konfiguriert werden
 - JAVA_HOME etc.

Gegenüberstellung Downsizing ↔ ESL

	abgestuftes Downsizing	Embedded Licence
Initiale Zusatzaufwände	SQL-Klausel – Entfernungsscripte, PL/SQL-Codierung alle Varianten	Oracle Installation, Admin-Tool
Fortentwicklung Zusatzaufwände	PL/SQL-Codierung alle Varianten, separate Upgradeteile je Variante	keine
Fehlerbehebung Zusatzaufwände	wegen zusätzlichem Code für Varianten	keine

Zusammenfassung

- Komplette unterschiedliche Lösungen zur Lizenzkostenreduzierung
- ESL hat alle Aufwände bei Einführung
- Downsizing hat einen Teil der Aufwände während Fortentwicklung
- Downsizing hat darüber hinaus eine höhere zu erwartende Fehlerquote
- Auszuwählende Lösung hängt ab von:
 - zu erwartender Lebensdauer der Applikation
 - Verfügbarkeit von Know-How für die Erstellung der Oracle-Installation und eines Admin-Tools
 - Oracle- und OS-Knowhow über verschiedene Plattformen

Fragen & Antworten

Vielen Dank für Ihre Aufmerksamkeit

Kontakt: **Dr. Kurt Franke**

Cellent Finance Solutions AG, Calwer Str. 33, D-70173 Stuttgart

Email: Kurt.Franke@cellent-fs.de , Kurt-Franke@web.de

Phone: +49-(0)711-222992676 , Mobil: +49-(0)171-7963089