



4 foils

Sandvik Systems Development

Roger Eriksson

Tine Björk

Miljö och verktyg

- 6 DB2 systems (3 test + 2 prod + 1 teknik)
 - ✓ version 7
 - ✓ 650 GB DB2 compressat data
- z/OS V1R4
- DB2 OLAP version 7
- IMS version 6
- Platinum CA tools
 - ✓ Catalog Facility
 - ✓ Log Analyzer
 - ✓ Database Analyzer
 - ✓ Detector
 - ✓ Plan Analyzer
 - ✓ *Fast Unload --> new*

Miljö och verktyg

- ***Boole & Babbage***
- ***DB2EEE version 7***
- ***Abend Aid (Fault Analyzer)***
- **Cobol v 3.2**
 - ✓ Librarian
 - ✓ WebSphere Studio 5.0
- **Natural 3.1.5**
- **Stored Procedures**
 - ✓ Cobol
 - ✓ SQL
 - ✓ Debugging
- **PC utveckning - Visual Studio / Visual Studio .Net**

Miljö och verktyg

- **DB2 Connect EE**
- **Control Center**
- **Visual Explain**
- **Data Warehouse Manager**
- **Power Designer**
- **Business Objects**

Erfarenhetsutbyte

- **Stored Procedures status**

- ✓ **Development Center version 8**

- För att kunna använda secondary authid måste man ha fixpack 3 på klienten och version 1.16 DSNTPSMP

- ✓ **OLEDB**

- För att få Å,Ä,Ö att fungera behövs fixpack 3 på klienten

- **CA Platinum status**

- ✓ **Detector**

- ✓ **Plan Analyzer**

- ✓ **Fast Unload**

- Kopiera data från prod till test mha Migrator och Fast Unload

Erfarenhetsutbyte forts.

● Konvertering från DL/1 till DB2

✓ Mål

- Byta ut databashanterare till DB2
- Komma ifrån begränsningar som finns med hierarkisk databas

✓ Krav

- Systemet får ej vara nere mer än 15 minuter per dygn
- Kunna fortsätta underhålla systemet under konverteringstiden
- Kunna backa tillbaka enkelt och fort
- Kort utvecklingstid, ej förändra affärslogik

✓ Tillvägagångssätt

- Tar bort alla databas anrop från huvudprogrammen och istället anropar en generell subrutin
- Man skapar 2 subrutiner per segment - en läsande, en uppdaterande

Erfarenhetsutbyte forts.

✓ Genomförande

- Sker i 3 olika faser. DL/1 , DL/1 + DB2, och enbart DB2
- Initialt körs DL/1 vägen. Styrning sker via generell subrutin
- Program flyttas löpande. Ingen programförändring har skett förutom att databasanropen ligger i subrutiner
- Spegling av data från DL/1 till DB2 för att kunna styra om till fas 2
- Verifiering av data, dvs att DB2 uppdateras korrekt
- Allt OK, går till fas 3, dvs bara DB2
- När man är klar med fas 3, så bygger man successivt om problem program till att gå direkt till DB2 via SQL

Erfarenhetsutbyte forts.

✓ Fördelar

- Kortare utvecklingstid (samma ändring, ej test av affärslogik, flexibelt dvs pgm flytt löpande, vanliga underhåll + akutändr. lätt att få in)
- Går lätt att lägga ut på konsulter
- Behöver ej stoppa online systemet
- Man kan gå över till DB2 segment för segment
- Övergång styrs via ett program, lätt backa från DL/1- DB2 till enbart DL/1

✓ Nackdelar

- Ett DL/1 segment kan bli 12 DB2 tabeller pga normaliseringsprincipen
- Prestandaproblem pga att man läser in mycket onödiga kolumner. Man måste ge tillbaka hela segmentet ifyllt pga att man inte vet vad anropande program vill ha
- Man läser DB2 postorienterat

Erfarenhetsutbyte forts.

● Övrigt

- ✓ **UPDATE...SET(col1,col2)=(SELECT col1,col2 FROM join)**
 - om resultatet av selecten blir null (dvs ingen träff) och update cols ej tillåter nulls, finns det nån teknik att lösa detta?
- ✓ **SELECT...WHERE col1 = (SELECT MAX(col2)...**
 - col1 tillåter ej nulls, resultatet från subselect blir nulls, behövs ej längre COALESCE för att få matchning på col1
- ✓ **Compressat data.**
 - om man har en tabell med många decimal kolumner som initialt är nollställda och gör en massupdate, blir dataraderna längre, så datat blir oreogat?
- ✓ **Batchpgm med COMMIT och RELEASE DEALLOCATE.**
 - Ser ingen tidsvinst? Är det även nån ZPARM?