

Bemærk: Dette tidsskrift må kun distribueres i virksomheder med OracleEkspert medlemskab

OracleEkspert

#25

NYHEDER 22

Oracle og Macromedia partnerskab om E-Learning

Oracle får ny finansdirektør

Perregaard og Skelbæk stævner Oracle

Oracle bekræfter ønske om at opkøbe PeopleSoft

10g kommer til Apple-plattformen

Vinnere af ThinkQuest er udpeget

4. kvartals regnskab

LIVE 23

DesWeb SIG

Dato: 2004-08-18

OracleEksperts
Databasekonference 2004

Dato: 2004-08-25

Database Forum 2004

Dato: 2004-10-28

Oracle Open World 2004

Dato: 2004-12-05

Miracle Master Class med Tim Gorman

Dato: 2005-01-17

STRUKTURERET UDVIKLING 2

af Marc de Oliveira

Erfarne udviklere er bekymrede over, at den objektorienterede udvikling og extreme programming er ved at kaste udviklingsprocessen mange år tilbage, og at solide erfaringer om hvordan systemudvikling bør foregå er ved at gå tabt.

DET HEMMELIGE WEB-VÆRKTØJ: ORACLE DESIGNER 4

af Sten Vesterli

Oracle Designers Web PL/SQL Generator er et stærkt værktøj til hurtigt at bygge robuste web-applikationer. Denne artikel forklarer, hvordan man genererer web-moduler fra Designer og forklarer de skærbilleder, som Oracle Designer fremstiller.

I JUST MIGHT TELL YOU THE TRUTH 12

med Morten Egan - af Mogens Nørgaard

ODTUG-KONFERENCEN 2004 16

af Marc de Oliveira

Fra den 20.-24. juni afholdt Oracle Development Tools User Group sin årlige konference. Denne gang i Scottsdale, Arizona.

GROANS FRA MOGENS 20

af Mogens Nørgaard



MIRACLE

DBVision Aps

OracleEkspert
2004

25. august kl 9-16

15 indlæg i tre spor

Normalpris: 2550.-

Abonnenter: 2050.-

5. tilmelding er gratis

STRUKTURERET UDVIKLING



OracleEkspert fylder med dette nummer fire år. Dette fejres med endnu 50 spørgsmål til OECP-spillet og en stor fødselsdagskage, lavet af chefkonditor Rainer Wurst, til konferencedeltagerne ude på Trekroner den 25. august. Husk at tilmelde dig, hvis du ikke allerede er det!

Et emne, der for tiden diskuteres en del, er struktureret udvikling.

Erfarne udviklere er bekymrede over, at den objektorienterede udvikling og extreme programming er ved at kaste udviklingsprocessen mange år tilbage, og at solide erfaringer om hvordan systemudvikling bør foregå er ved at gå tabt.

En ophedet del af denne debat er foregået på ODTUGs mailliste, hvor meget kompetente folk som David Hay (forfatter til bøgerne Data Model Patterns og Requirement Analysis), Dai Clegg (medforfatter til bogen CASE Method og nuværende strateg i udviklingen af JDeveloper hos Oracle), Sue Harper (Product Manager for Designer og JDeveloper), Dr. Paul Dorsey (forfatter til bøgerne Oracle9i JDeveloper Handbook, Oracle Designer Handbook og Oracle Database Design Using UML Object Modeling), Ulka Rodgers (forfatter til bogen Oracle: A Database Developer's Guide) og mange andre deltog aktivt.

Eet af resultaterne af denne debat var, at der ikke findes nogen anvendelige modeller for hvordan man skal bygge komplekse applikationer vha objektorienterede metoder eller extreme programming. Jeffrey Jacobs fremhævede bogen "Extreme Programming Refactored: The Case Against XP by Matt Stephens, Doug Rosenberg", hvori det fremgår at det berømte "Chrysler" projekt, som XP er blevet så berømt på, viste sig at være en katastrofe. Projektet kunne ikke skaleres, nåede aldrig deadlines, blev meget dyrere end planlagt og endte med at blive opgivet!

Også Paul Dorsey hævder, at der ikke findes anvendelige metoder i nogen UML eller OO-bøger. Ja, selv Dai Clegg, som nok er den førende OO-metode-profil hos Oracle, indrømmer at OO-litteraturen er umoden med hensyn til udviklingsmetoder.

Den objektorienterede verden mangler input fra traditionelle metodefolk og fra folk med viden om datamodellering. Det er en af grundene til at JDeveloper slet ikke er udviklet med disse ting for øje. Metoder og processer har ikke været i fokus hos traditionelle Java-udviklere.

Sammenligner man Designer med JDeveloper i denne kontekst, er der et fundamentalt problem med JDeveloper, som gør værktøjet uanvendeligt til struktureret udvikling: Det er nødvendigt, at have et Repository til metadata. Så længe JDeveloper er baseret på flade filer, vil det ikke være muligt at manipulere metadata på en måde der tillader håndtering af abstrakte elementer med afhængigheder til programkoden. Ordentlig integration og genbrug kræver feks en stærkt struktureret mekanisme til håndteringen af generelle ændringer, der går på tværs af een eller flere applikationer mv.

Dai Clegg løftede sløret for at Oracle faktisk er i gang med at undersøge mulighederne for at implementere et Repository i JDeveloper. Han lovede dog ikke noget om hvornår et sådan Repository eventuelt ville blive implementeret.

Jeg mener også, at JDeveloper har et stort problem med at deres modeller ikke er adskilt fra koden. Koden er modellen, og modellen er koden. Dette gør det meget vanskeligt at skifte platform, hvis man får behov for at flytte sine applikationer væk fra Java-stakken. Designers struktur med generatorer, der genererer den færdige kode ud fra modellen virker væsentlig sundere, da man derved kan genbruge sine modeller på tværs af mange platforme.

Alt i alt er der enighed om: 1) at Designer er det mest avancerede værktøj til analyse og metodestyret udvikling, 2) at JDeveloper er det Oracle-værktøj der vil være mest fremdrift i i fremtiden, og 3) at Oracle arbejder på at gøre JDeveloper mere egnet til struktureret udvikling.

Jeg vil derfor mene, at hvis man ønsker at udføre struktureret og veldokumenteret udvikling, bør man pt vælge Designer som overordnet værktøj til understøttelse af hele udviklingsprocessen. Hvis man har brug for Java-programmer, kan man bruge JHeadstart som generator til JDeveloper (se omtalen af JHeadstart på side 17), og først når JDeveloper har fået et repository og funktionalitet til at understøtte hele udviklingsprocessen bør man overveje at migrere sine modeller til JDeveloper.



Oplag:150 kopier

Udgives af:

.....PYTHIA Information
.....Kongensvej 3
.....2000 Frederiksberg
.....Danmark

Telefon:2683 9991

Email:Info@OracleEkspert.dk

Web:www.OracleEkspert.dk

Ansvarshavende redaktør:

.....Marc de Oliveira
.....Marc@OracleEkspert.dk

Betalinger:

.....Lasse Christensen
.....Faktura@OracleEkspert.dk

Groans fra Mogens:

.....Mogens Nørgaard
.....Mogens@OracleEkspert.dk

Bentes Bøger:

.....Bente Rosenkrantz-Theil
.....Bente@OracleEkspert.dk

Rettigheder:

PYTHIA Information ejer alle rettigheder til indholdet af **OracleEkspert**.

Kopiering af bladet i dele eller helhed må kun ske efter skriftligt samtykke fra PYTHIA Information.

PYTHIA Information forbeholder sig rettigheder til at offentliggøre og genudgive de trykte artikler, tips mv, samt at tillade bladets læsere at anvende indholdet til såvel personlige som kommercielle formål.

PYTHIA Information kan ikke drages til ansvar for eventuelle fejl og mangler i Indholdet af **OracleEkspert**. Artikler mv stilles tilrådighed uden garanti af nogen art.

Pris:

EnkeltnummerDKK 300,00

1 års abonnement (6 numre):

1. abonnementDKK 1800,00

Pr ekstra bladDKK 600,00

Annoncer:

Annoncer til **OracleEkspert** skal være PYTHIA Information i hænde senest den 15. i måneden inden bladets udgivelse.

Annoncepriser kan findes på:

www.OracleEkspert.dk

Password: oekptia

Dato 25. august
Tid Kl. 9 - 16
Sted Trekroner
Pris 2550.-

Pris for abonnenter: 2050.-

Ved tilmelding af fem deltagere fra samme firma, kommer den ene gratis med.

ORACLE
ekspert
2 0 0 4

Tilmelding:

www.OracleEkspert.dk
salg@OracleEkspert.dk
tlf: 2683 9991

Program

Mandskabsbygningen 1. sal	Kommandantbygningen 1. sal	Kommandantbygningen stueetagen
Transport fra Langelinie Registrering og morgenmad		
Mit liv med Groans fra Mogens Mogens Nørgaard	Vælg det rette værktøj - gennemgang af Oracles værktøjer Sten Vesterli	Portlet baseret på MVC, Apache Struts og Oracle-værktøjer Flemming G. Jensen
Hold op med at samle statistikker Mogens Nørgaard & Jakob Hammer-Jakobsen	Anbefalinger til at komme i gang med SCM Marc de Oliveira	Vigtige tips til egenudvikling af rapporter mod eBusiness Suite Rune Mørk & Martin Zangger
3-retters frokostmenu i mandskabsbygningens stueetage		
Hvordan sorteres Æ, Ø og Å rigtigt? Martin Jensen	Rå trace filer: Oracle9i og 10g new features Mogens Nørgaard & Johannes Djernæs	Rapportering med OLAP Arnar Dagsson & Bente Rosenkrantz-Theil
Kan man ikke gøre det lidt bedre? - om overvågning, tuning, backup mv BMC - Philip Prægel	Det hemmelige web-værktøj: Oracle Designer Sten Vesterli	Application Performance Management på Oracle AS og Oracle DB Quest - Dennis Kroer
Fødselsdagslagkage		
UML og datamodellering Anders Groth & Bente Rosenkrantz-Theil	Hvorfor IT-projekter fejler - og hvordan man undgår det Sten Vesterli	Datamining i praksis Jens Hejlesen

Der kan komme enkelte ændringer i det endelige program

DET HEMMELIGE WEB-VÆRKTØJ: ORACLE DESIGNER



Sten Vesterli er partner i scott/tiger og arbejder til daglig som Oracle-konsulent. Sten har især beskæftiget sig med Oracle's application server – lige siden den hed Oracle Web Server 1.0 tilbage i 1996. I løbet af årene har han præsenteret på talrige internationale Oracle-konferencer og skrevet en lang række artikler samt bogen "Oracle Web Applications 101" (Oracle Press). sten.vesterli@scott-tiger.dk

Indledning

Oracle Designers Web PL/SQL Generator er et stærkt værktøj til hurtigt at bygge robuste web-applikationer. Denne artikel forklarer, hvordan man genererer web-moduler fra Designer og forklarer de skærm billeder, som Oracle Designer fremstiller.

Desuden gennemgår den de vigtigste indstillinger til styring af den genererede applikation.

Hvordan Oracle Designer gør tingene

Oracle Designer er et Computer Aided Software Engineering (CASE) værktøj. Det bruger en *model* af applikationen, der ligger gemt i et *repository*. Selve applikationen bliver *genereret* "med et tryk på en knap".

Man kan bruge Designer på mange måder: Fra klassisk Information Engineering (IE) til moderne Rapid Application Development (RAD).

Ved IE bruger man de fleste af Designers værktøjer, mens udviklingen flyder ned ad det velkendte vandfald fra analyse over design til generering. Dette er en omhyggelig, formel udviklingsmetode velegnet til store, komplekse og/eller kritiske systemer. Oracle har udviklet en metode ved navn Oracle CDM Advantage med veldefinerede deliverables og støtteværktøjer.

Ved RAD bruger man mest Design Editor værktøjet til iterativ udvikling, hvor man hurtigt kan bygge og videreudvikle prototyper. Denne metode egner sig bedst til mindre applikationer, når kravene er uklare eller ændrer sig hurtigt.

Desuden kan man indlæse eksisterende databaser og Oracle Forms applikationer i Designer til dokumentation eller videreudvikling.

Web-applikationer med Designer

Denne artikel beskriver RAD-metoden, som er velegnet til små prototype-drevne projekter. Andre anvendelsesformer er omfattende beskrevet i Designers fremragende hjælpeetekster.

Konstruktion af databasen

Når man bygger databasen, er slutresultatet det samme: Tabeller i databasen. Men med Designer kan man komme til målet på to måder: Man kan starte med en Entitets-Relationsmodel i E-R Diagrammer værktøjet eller gå direkte til Design Editoren. Begge måder har fordele og ulemper som vist i tabel 1.

	Fordele	Ulemper
E-R Diagrammer	Fokus på det logiske design	Man skal bruge flere forskellige værktøjer
Design Editor	Hele designet i ét trin	Man skal selv kode kolonner til primærnøgler og fremmednøgler

Tabel 1: Sammenligning af E-R Diagrammer med Design Editor

Hvis man beslutter sig for at bruge E-R Diagrammer værktøjet, kan man koncentrere sig om at modellere de "ægte" data, som applikationen har brug for, og man kan lade Database Design Transformeren tilføje de nødvendige kolonner til primær- og fremmednøgler.

Hvis man kun bruger Design Editoren, behøver man ikke at arbejde med flere forskellige værktøjer og har direkte adgang til server-modellen med alle detaljer fra starten.

Design Capture

Designer indeholder mange gode værktøjer til at bygge nye databaser, men de fleste web-applikationer bliver ikke bygget fra bunden. De baseres i stedet på tabeller, der allerede anvendes i eksisterende systemer, eventuelt med mindre tilføjelser.

For at understøtte dette giver Designer mulighed for at indlæse den eksisterende database. Denne proces kaldes "Design Capture". Man kan herefter om nødvendigt ændre tabeldesignet med Design Editor og generere scripts til at indføre ændringerne i databasen.

Det er en forudsætning for fremstilling af web-applikationer med Designer, at Designer kender tabellerne. Man skal altså lave Design Capture for at kunne bygge en ny web-applikation på eksisterende tabeller.

Konstruktion af applikationen

En Designer-genereret PL/SQL web-applikation fungerer på samme måde som en håndskrevet PL/SQL web-applikation: Programkoden er PL/SQL packages, og man har brug for mindst en Apache webserver og mod_plsql modulet for at etablere kommunikation mellem en webbrowser og PL/SQL-koden i databasen. Dette får man sammen med databasen (siden 8.1.7) og sammen med Oracle Application Server (alle versioner).

Forskellen mellem en håndskrevet applikation og en Designer-genereret er, at man i stedet for selv at programmere bygger *moduler* i Design Editoren og derefter *genererer* PL/SQL-koden ved et tryk på en knap. Moduldefinitionerne er gemt i Designers database (repository'et) og den genererede PL/SQL-kode gemmes

som en fil (og kan automatisk installeres i databasen).

Oracle-skemaerne i en Designer web-applikation

En Designer-genereret web-applikation bruger normalt PL/SQL-kode i to eller tre skemaer:

- Applikations-skemaet indeholder den applikations-specifikke kode.
- Toolkit-skemaet indeholder Designer PL/SQL Web Library'et. Dette er en samling PL/SQL-pakker, som Oracle leverer. De kaldes af den Designer-genererede web-applikation i applikations-skemaet. I databaser før version 8.1.7, lå "PL/SQL Web Developers Toolkit" også i dette skema.
- SYS-skemaet indeholder "PL/SQL Web Developers Toolkit" (fra database version 8.1.7). Dette er en samling PL/SQL-pakker, som bruges af alle typer PL/SQL web-applikationer (både håndskrevne og Designer-genererede). Oracle leverer disse pakker sammen med databasen (fra 8.1.7 og frem) og sammen med applikations-serveren.

Opsætning af en Database Access Descriptor

For at kunne køre en Designer-genereret web-applikation (eller for at kunne køre enhver anden type PL/SQL web-applikation), skal man fortælle Oracle HTTP Serveren, hvordan en URL omsættes til en databaseforbindelse.

Dette gøres ved at tilføje en Database Access Descriptor (DAD) til Oracle Web/Application Server konfigurationen. I den Oracle HTTP Server, der følger med databasen version 8.1.7 og senere, starter man på velkomstkærm-billedet og klikker først på "mod_plsql" linket og derefter på "Gateway Database Access Descriptor Settings" og så "Add Default". Man angiver et navn på DAD'en, Oracle brugernavn, password og connect string.

Dette gør det muligt at køre Designer-genererede applikationer gennem URL'er som denne:

```
http://<yourserver>/pls/<dad>/<module>$.startup
```

Hello World

Den første eksempel-applikation er som altid det klassiske "Hello World"-program. Dette er trivielt i Oracle Designer, men denne lille applikation illustrerer Design Editorens grundlæggende virkemåde udmærket. Den viser, hvordan man bruger *User Text* i Designer og hvordan man indstiller nogle af de grundlæggende *generator preferences* for at styre den genererede applikation.

Fremstilling af modulet

For at fremstille et modul skal man vælge fanebladet "Modules" i Design Editoren, vælge "Modules" grenen i træet og klikke på det grønne "plus"-tegn for at tilføje et modul.

Man skal angive modulets *short name*, *name* og *purpose* og vælge "Web PL/SQL" som *language*. Derefter skal man klikke **Next** for at gå til trinnet User Text.

User Text

I denne dialog viser "Text location" listen til venstre alle de steder, hvor der kan tilføjes tekst til modulet, og feltet "User text" til højre viser det indtastede. Til "Hello

World" eksemplet kan man indtaste følgende tekst:

```
<h1>Hello World</h1>
<p>The time is
htf.nobr(to_char(sysdate,
'yyyy-mm-dd hh24:mi:ss'))
```

Bemærk kaldet til NOBR funktionen i HTF-pakken. Designer genkender automatisk kald af HTF-pakken og *udfører* funktionen i stedet for at vise funktionens navn. Denne funktion omgiver teksten med <NOBR> </NOBR> tags, der fortæller browseren, at linien ikke skal ombrydes. Dette er ikke nødvendigt for selve websiden, men med mindre man kalder en pakke som Designer genkender (som HTF-pakken i dette eksempel), vil Designer ikke *udføre* TO_CHAR funktionen. I stedet ville Designer simpelthen placere PL/SQL-koden på websiden.

Klik **Finish** for at afslutte definitionen af modulet.

Generering af modulet

For at generere modulet skal man højreklikke på det og vælge Generate på pop-up menuen.

Indstilling af Generator Options

Før man genererer første gang, bør man klikke på **Options** knappen for at indstille Generator Options:

I feltet "Install Generated PL/SQL" skal man angive et brugernavn (man kan også vælge at skrive brugernavnet, en skråstreg og passwordet) og en connect string for at få Designer til automatisk at installere den genererede kode i databasen. Hvis man ikke angiver et password, vil Designer spørge efter det, hver gang man forsøger at installere den genererede kode.

I afsnittet "Run Generated Applications" kan man angive stien til webbrowseren og den første del af en URL, der svarer til den Oracle-bruger, man angiver under "Install Generated PL/SQL". Hvis man udfylder disse felter, kan man klikke på en knap i Designer for automatisk at køre det genererede applikationsmodul gennem webbrowseren.

Generering

Når generatoren er indstillet, klikkes **OK** for at lukke Options-dialogen og derefter **Start** for at starte generatoren. Hvis genereringen kører korrekt, vises nogle meddelelser i Message-vinduet, og derefter ser man et kommandoprompt, hvor en SQL*Plus session automatisk installerer de genererede pakker i det angivne database-skema. Til sidst kommer meddelelsen "Generation is complete".

Resultatet

For at se det nye applikationsmodul køre, skal man vælge Messages | Build Actions, vælge "Run generated applikation..." linien og klikke **Run** for at starte webbrowseren og se applikationen. Applikationen vil



Figur 1: Den første genererede applikation

nu se ud omtrent som vist i figur 1.

Designer har tydeligvis ikke helt forstået opgaven – applikationen returnerer den aktuelle tid og dato korrekt, men der er to problemer med dette modul:

- Det viser HTML-tags i stedet for at lade browseren fortolke dem
- Det har et ekstra "About"-link, som vi ikke har bedt om.

HTML-koderne bliver vist, fordi Designer i bedste mening forsøger at beskytte os mod at skulle kende HTML, og About-linket bliver som standard altid tilføjet til alle moduler.

Forbedringer med Generator Preferences

For at styre, hvordan Designer genererer applikationen, bruger man *Generator Preferences*. Disse kan sættes på forskellige niveauer – for at ændre på modulniveau, skal man højreklikke på modulet og vælge "Edit Generator Preferences" på popup-menuen.

Man bliver nødt til at kende nogle få, vigtige indstillinger for at kunne styre Designer. Designers udemærkede hjælp forklarer dem alle grundigt – man kan blot vælge en preference og trykke F1 for at slå den op i hjælpen. Alle preferences er grupperet under overskrifter, der kan udfoldes og lukkes sammen, og alle har både et navn (som MODSUB) og en beskrivelse (som "Substitute HTML Reserved Characters").

For at løse de to problemer med Hello World applikationen, skal man ændre to indstillinger:

1. For at få Designer til *ikke* at konvertere HTML-tags til tekst, skal MODSUB ("Substitute HTML Reserved Characters") sættes til "No". Denne indstilling findes i gruppen "Text Handling".
2. For at få Designer til ikke at fremstille "About"-siden, skal man sætte MODALR ("Startup Page: About Page Hyperlink") til "No". Denne indstilling findes i gruppen "Startup Page".

Når man har ændret disse to indstillinger, kan man gemme indstillingerne og generere igen. Nu burde applikationen se nogenlunde ud som i figur 2.



Figur 2: Hello World med de rigtige indstillinger

Liste over ansatte

Det næste eksempel-program er en liste over alle ansatte i EMP-tabellen. Dette eksempel vil vise, hvordan man bruger *module components* og *table usages* til at hente data fra databasen og hvordan man tilpasser applikationens udseende med flere preferences og en applikationsspecifik formateringspakke.

Dette og de følgende eksempler arbejder med EMP tabellen, så man bliver nødt til at indlæse definitionen af denne tabel i Designer's Repository ved hjælp af

design capture som beskrevet ovenfor.

Fremstilling af modulet

For at fremstille et modul, der bruger data fra databasen, skal man stadig angive et navn og vælge sproget "PL/SQL Web" på det første skærmbillede. Derefter kan man klikke **Next** hele vejen gennem Module-guiden, indtil "Goodbye"-skærmbilledet. Her skal man vælge "Create modulet and then invoke a Wizard to create a Module Component" før man vælger **Finish**.

Fremstilling af en Module Component

Når Module-guiden er færdig, starter Module Component-guiden.

Et modul i sig selv gør i virkeligheden ikke noget – det fungerer hovedsagelig som beholder for *module components*. Disse er de byggesten, der bruges i applikationen, og de kan igen indeholde *table usages*.

Først skal man indtaste et navn på modulkomponenten og klikke **Next**. Det er ikke nødvendigt at indtaste en titel her – den kommer fra en sidestart-pakke senere.

I trinnet "Table Usage", kan man vælge den *base table*, som modulkomponenten skal baseres på, og man kan yderligere vælge *lookup* tabeller. Lookup-tabeller bruges, når der skal hentes supplerende oplysninger i en anden tabel, der er forbundet til basistabellen gennem en fremmednøgle. Hvis man for eksempel ønsker at vise ansatte (fra EMP tabellen) med afdelingsnavn (der kun findes i DEPT tabellen), kan man bruge EMP tabellen som basistabel og DEPT tabellen som lookup-tabel.

Til et simpelt modul som dette, skal man blot vælge EMP tabellen som basistabel og ingen lookup. Derefter kan man klikke **Next**, indtil man kommer til trinnet "Operations" i Module Component guiden.

Dette trin er meget vigtigt: Det definerer hvilke operationer, der er mulige på basistabellen. Standardindstillingen er en ren forespørgselsapplikation uden mulighed for at ændre data, så det er ikke nødvendigt at ændre noget til dette eksempel. Klik **Next** for at gå til trinnet "Select Items".

Her vælger man de elementer, som man ønsker at arbejde med i modulkomponenten. Designer fremstiller et *bound item* for hver kolonne i basistabellen, man vælger her. Det er ikke nødvendigt at vælge alle kolonner i tabellen – for at holde modulet simpelt kan man nøjes med at vælge de relevante.

Afkrydsningsfelterne ud for hvert element bestemmer, om de vises for brugeren eller ej – dette kan bruges til f.eks. at medtage en tabs primærnøgle uden at vise den.

Når man har valgt kolonner, skal man klikke **Next**, indtil man kommer til trinnet "Item Operations".

Her kan man detaljeret angive, præcis på hvilke bound items, hver operation er tilladt. Man kan for eksempel beslutte, at en modulkomponent tillader Update, men kun af bestemte kolonner. For at fortælle dette til Designer, ville man sætte modulkomponentens action til "Update", men kun sætte enkelte bound items til "Update".

I dette simple eksempel skal alle elementer sættes til Query = No. Når der ikke er noget element med Query, vil Designer gå direkte til resultatet, og det er det ønskede i dette eksempel. Hvis nogen elementer har Query = Yes, vil Designer automatisk bygge et

søgeskærm-billede – det gøres i næste eksempel. Når man er færdig, skal man klikke **Next** flere gange for at gå til guidens sidste trin.

På dette skærm-billede skal man vælge "Create Module Component and then invoke the Display Wizard" og derefter klikke **Finish**.

Fastlæggelse af modulkomponentens udseende

Efter Module Component guiden fremkommer Module Component Display guiden. Denne guide gør det let at definere, hvordan elementerne i modulkomponenten skal vises i webapplikationen.

Indstillingen "Layout style" bestemmer hvilke skærm-billeder, webapplikationen kommer til at bestå af. I dette simple eksempel kan man lade standardindstillingen <Null> være og klikke **Next** for at gå videre til fanebladet "Controls".

På dette skærm-billede kan man definere de labels, som Designer skal placere på web-skærm-billederne for dataelementerne. Når man har indtastet nogle værdier, kan man klikke **Next** for at videre til fanen "Context".

Context Items er de elementer (items), som bruges i applikationen til at identificere en række. Dette har intet at gøre med definitionen af den underliggende tabel – det betyder blot, at Designer vil vise disse elementer overalt, hvor der vises en liste over rækker fra en tabel.

Der er tre mulige situationer:

1. **Ingen context items defineret.** I dette tilfælde viser Designer en oversigtsliste, der kun består af det første dataelement i rækken. Dette dataelement er et hyperlink, som brugeren skal klikke på for at se et detalje-skærm-billede med de resterende dataelementer.
2. **Nogle, men ikke alle elementer er defineret som context items.** I dette tilfælde vises en oversigtsliste med alle context items, og det første element er et hyperlink til et detalje-skærm-billede.
3. **Alle elementer er defineret som context items.** I dette tilfælde viser Designer alle elementer på oversigtsskærm-billedet, og der findes intet detalje-skærm-billede.

I dette simple eksempel er formålet at fremstille en liste over alle ansatte i EMP-tabellen uden detalje-skærm-billede. Derfor skal man klikke på den lille pil for at flytte alle elementer fra venstre side ("Available items") til højre side ("Context items"). Herefter skal man klikke **Finish** for at lukke guiden.

Generering af modulet

For at generere modulet, skal man højreklikke på det i Designer Navigator og vælge "Generate" på popup-menuen. Designer husker generator-indstillingerne, så man kan blot klikke **Start** i dialogen "Generate Web PL/SQL".

Når genereringen er afsluttet, skal man vælge Messages | Build Actions, vælge linien "Run generated applikation..." og klikke **Run** for at starte webbrowseren og se det nye modul.

Bemærk, at der nu igen er et "About" hyperlink, fordi generator preferences ikke blev ændret. Herudover viser Designer kun 10 resultater ad gangen og bygger

automatisk funktionalitet til at bladere frem og tilbage i resultatsættet.

Forbedring af modulet

Dette modul har den ønskede funktionalitet, men det kan stadig forbedres lidt. Ved at indføre en standard-pakke til formatering og fremstille et *Preference Set* kommer den genererede applikation til at se bedre ud.

En særlig formateringspakke

Designer har preferences, som man kan bruge til at define standard HTML, som altid skal vises øverst og nederst på alle websider. Denne preference har dog stærkt begrænset længde, og hvis sidestarten eller –slutningen skal ændres, ville man blive nødt til at regenerere alle moduler.

En bedre løsning er at fremstille en pakke (man kan for eksempel kalde den *FORMAT*), der indeholder funktioner, der returnerer den nødvendige HTML til at start og afslutte en side. Derefter kan man blot ændre denne pakke for at ændre applikationens udseende uden at skulle generere forfra. Specifikationen kunne f.eks. se således ud:

```
create or replace package format
as
  function pageStart (title in varchar2)
    return varchar2;
  function pageEnd return varchar2;
end;
```

Man skal selvfølgelig også fremstille pakkens body, så den returnerer nyttig HTML (man kan downloade et eksempel på en *FORMAT*-pakke fra www.scott-tiger.dk/webstart).

PAGESTART funktionen tager en parameter, så man kan tilpasse sidehovedet med sidens titel osv.

Fremstilling af et Preference Set

For at gøre det lettere at håndtere alle de nødvendige generator preferences, som skal indstilles for en applikation, tilbyder Designer muligheden for at fremstille *Preference Sets*.

For at oprette et preference set, skal man finde og åbne punktet "Preference Sets" i Navigatoren. Der findes et underpunkt, der hedder "Web PL/SQL Generator". Man kan højreklikke på dette og vælge "Create Preference Set". Derefter skal man give preferencesættet et navn og gemme det. Nu optræder det nye preferencesæt i Navigatoren.

Herefter skal man højreklikke på det, vælge "Edit Generator Preferences" på popup-menuen og sætte preferences som anført i Tabel 2.

Hvis man ønsker at vise mere end 10 rækker på en side, kan man også sætte ZONLRS (Record List: Record Set Size) til en større værdi.

Læg særligt mærke til PKGLST preferencen: Denne fortæller Designer, at hver gang teksten "format" optræder et sted hvor Designer forventer brugerdefineret tekst, skal Designer forsøge at kalde en pakke ved navn *FORMAT* (i stedet for at vise teksten "format" på skærmen).

For at bruge dette preference set i modulet, skal man højreklikke på modulet og vælge "Edit Generator Preferences" på popup-menuen. I "Preference Set" listen øverst i vinduet Generator Preferences skal man herefter vælge det preference set, man netop har oprettet.

Name	Description	Value	Kommentar
MODSUB	Substitute HTML Reserved Characters	No	For at gøre det muligt at bruge HTML-tags i User Test
MODALR	Startup Page: About Page Hyperlink	No	For at fjerne "About" hyperlinket
PKGLST	PL/SQL Package List	format	De pakker til fremstilling af HTML, som Designer genkender.
MODSFA	Standard Footer On All Pages	Yes	Instruktion om at bruge en footer
MODSFT	Standard Footer	format.pageEnd	Et kald til Format-pakken
MODSHA	Standard Header On All Pages	Yes	Instruktion om at bruge en header
MODSHD	Standard Header	format.pageStart ('Employee List')	Et kald til Format-pakken med en sidetitel

Tabel 2: Et simpelt Preference Set

Generering af modulet

Når man har genereret modulet igen og kører det, bør man se noget i stil med figur 3.



Figur 3: Liste over ansatte

Søg efter ansatte

Den næste applikation bygger videre på den liste over ansatte, som lige blev bygget. Det nye er tilføjelse af en webside, hvor brugeren kan indtaste søgekriterier.

Kopiering af modulet

Da selv meget små ændringer i properties eller preferences i et Designer-modul kan få Designer til at generere helt anderledes kode, er det en god idé at opbygge applikationens funktionalitet trin for trin.

For at gøre dette, bør man arbejde videre på en kopi af de tidligere modul: Højreklik på modulet og vælg "Copy Object..." på popup-menuen.

Guiden "Extended Copy" fremkommer. På grund af de mange afhængigheder mellem objekter, kan det være lidt omstændigt at kopiere moduler – det er bedst blot at klikke **Next** hele vejen gennem denne guide uden at ændre standardindstillingerne.

Når kopieringsguiden afslutter, vises det nye modul ikke med det samme – man skal vælge Edit | Requery All for at opdatere skærbilledet.

Redigering af modulet

Kopieringsguiden fremstiller et nyt modul, normalt med samme navn som det oprindelige modul, efterfulgt af et tal. Man bør give modulet både et nyt "name" og et nyt "short name".

Redigering af modulkomponenten

Det næste trin har til formål at få Designer til at fremstille en webside, hvor brugeren kan indtaste en restriktion på navnet på den ansatte. Selvom det involverer en ny webside, gør Designer det overordentlig enkelt: Man behøver blot at markere et eller flere items som søgbare (Query = Yes).

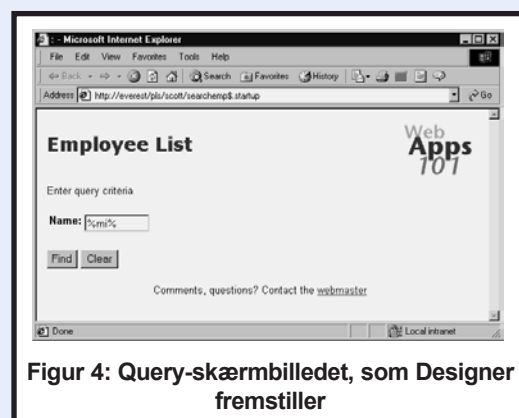
Man skal først åbne det nye modul, derefter modulkomponenten og derefter den tilhørende table usage.

Her skal man dobbeltklikke på hver kolonne, som brugeren skal kunne søge på, gå til fanen Operations og sætte "Query" til "Yes". Når man er færdig, skal man blot klikke **Finish**.

Når der er mindst ét element i et modul, der er søgbart, fremstiller Designer automatisk en webside til at indtaste søgebetingelser.

Generering af modulet

Når man har genereret modulet og kører det, starter det nu med et query-skærbillede som vist i figur 4.



Figur 4: Query-skærbilledet, som Designer fremstiller

Bemærk teksten "Enter query criteria". Dette tilføjes automatisk af Designer, og der er ingen let måde at slippe af med det. Dette er et af de steder, hvor man bliver nødt til at acceptere Designers måde at bygge applikationen.

Vedligehold ansatte

Det næste eksempel viser, hvordan man kan foretage alle normale databaseoperationer på EMP-tabellen: Insert, Update og Delete.

Kopiering og redigering af modulet

Først skal man fremstille en kopi af modulet fra det foregående eksempel og ændre både "name" og "short name".

Det næste trin er at tillade insert, update og delete operationer på modulkomponenten. For at gøre dette, skal man åbne modulet, navigere til modulkomponenten og dobbeltklikke på modulkomponentens ikon for at fremkalde guiden "Edit Module Component". Her skal man åbne fanebladet Operations og afkrydse feltene "Insert", "Update" og "Delete". Derefter skal man klikke **Finish**.

Det sidste trin er at definere præcis hvilke elementer, der kan indsættes og opdateres. For at gøre dette skal man navigere længere ned til modulkomponentens table usage for EMP og dobbeltklikke på table usage ikonen. Dialogboksen "Edit Base Table Usage" fremkommer. Her skal man vælge fanen "Operations" og tillade insert og update for de ønskede elementer. Det er normalt en god idé ikke at gøre tabellens primærnøgle opdaterbar. Når man er færdig, skal man klikke **Finish**.

Generering af tabel-API'et

Designer-genererede PL/SQL web-applikationer tilgår ikke tabellen direkte, når der skal ændres data (insert, update eller delete). I stedet kalder applikationsmodulets PL/SQL-pakke et *Table API*. Dette er en anden PL/SQL-pakke, som håndterer alle disse operationer og samtidig tilføjer standardværdier osv. Man kan ikke generere et modul, der ændrer data i databasen uden først at generere et Table API.

Det er heldigvis ret let med Designer:

1. Øverst i Design Editor Navigatoren skal man vælge Server Model fanebladet.
2. Vælg den eller de ønskede tabeller.
3. Vælg Generate | Generate Table API.
4. Vælg databasen som "Target for Generation" og giv forbindelsesoplysninger.
5. Klik **Start**.

Designer fremstiller nu tabeladgangspakken og et antal triggere for at fange alle insert, update og delete sætninger.

Valg af Layout

Designer understøtter tre forskellige layouts til PL/SQL webapplikationer:

1. List (standard)
2. Form
3. List/Form

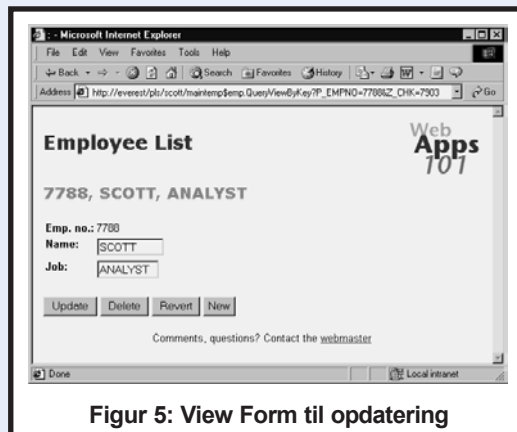
Man kan vælge layout for en modulkomponent på fanebladet "Display".

Kun Form, List og List/Form er gyldige indstillinger for

PL/SQL web-applikationer (de øvrige er ikke aktive og gælder kun for Forms og/eller Reports applikationer).

List Layout

Som standard bruger Designer *List* layoutet. Dette layout har fire separate skærbilleder: En *Query Form*, en *Record List*, en *View Form* (se figur 5 nedenfor) til at vise og/eller opdatere en record og en *New Form* til at indtaste nye records.



Figur 5: View Form til opdatering

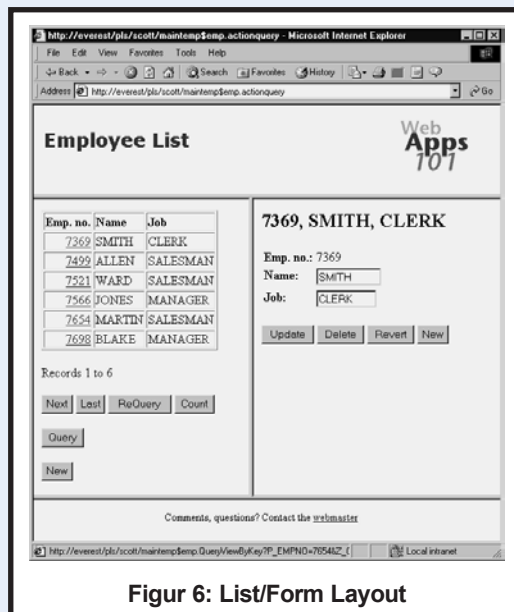
Form Layout

Der findes også et *Form* layout. Dette layout bruger ikke nogen oversigtsliste (Record List) og bruger kun *Query Form*, *View Form* og *Insert Form*.

Query Form og Insert Form er som for List layoutet, men View Form er lidt anderledes: Den har nu Next/Previous knapper til at bladere gennem alle records én ad gangen, og for at slette en record kan man nu afkrydse et "Delete" afkrydsningsfelt.

List/Form Layout

Det sidste layout er *List/Form*, hvor *Record List*, *View Form* og *Insert Form* placeres i forskellige frames på samme side som vist på figur 6. Oversigtslisten placeres normalt til venstre og View eller Insert Form til højre.



Figur 6: List/Form Layout

Som standard vises Query-skærbilledet på en separat side. Man kan dog sætte en generator preference LFQOSP (Query Form On Separate Page) til "No" for også at få Query Form'en med på siden (som standard vises den i øverste frame).

Opdelingen af skærbilledet i separate frames kan styres med generator preferences i grupperne "Frames", "Frames – custom" og "Frames – Default". For at lære at bruge dette layout, bliver man nødt til at læse hjælpen og eksperimentere lidt med disse preferences.

Da alle moderne browsere understøtter frames, kan man godt bruge dette layout. Det viser alle informationer på en mere kompakt måde, men fordi man faktisk viser flere forskellige websider samtidig, kan informationen i de enkelt frames komme ud af takt, hvad der kan forvirre brugeren. Man bliver nødt til at afprøve det og vise det til brugerne for at kunne bestemme, om List/Form layout er nyttigt i en bestemt applikation.

Brug af Lookups

Den næste funktionalitet, vi vil tilføje, er muligheden for at vælge værdier svarende til fremmednøgler. Man kan gøre dette ved at tilføje en table usage for lookup-tabellen, der indeholder værdierne, men dette kan være lidt kompliceret og er ikke altid nødvendigt.

En simpel lookup

I det simpleste tilfælde udgør en enkelt kolonne i basistabellen alene en fremmednøgle til en unik nøgle i en anden tabel.

Da Designer allerede kender denne relation, kan man blot ændre "display type" for et bound item fra Text til Poplist. Designer vil så automagisk fremstille en drop-down liste med alle nøgleværdier i den fremmede tabel.

I EMP eksemplet kan man for eksempel medtage DEPTNO og sætte display type til Poplist for at se en drop-down liste over alle afdelingsnumre.

Værdilister (List of Values)

Hvis man ønsker at vise et andet felt fra den fremmede tabel, skal man oprette en værdiliste (List of Values – LOV). Disse lister kan oprettes enten for hele applikationen som en *reusable* LOV, der kan anvendes af alle applikationens moduler, eller i et bestemt modul som en almindelig LOV, som kun kan anvendes af modulet.

Hvis man ønsker at bruge værdilisten i et separat vindue, kan man vælge at vise flere værdier fra den underliggende tabel, men hvis værdilisten skal bruges som poplist, bør man kun vælge én kolonne.

Når man har oprettet en LOV, kan man tilknytte den til et bound item. Fra Navigatoren kan man højreklikke på punktet Associated List of Values, vælge menupunktet Include List of Values og vælge den ønskede LOV.

Brug af en LOV i et separate vindue

Hvis man tilknytter en LOV, men ikke ændrer display type for et bound item, vil Designer fremstille et hyperlink med teksten "LOV" ved siden af feltet for det pågældende bound item. Når brugeren klikker på dette link, vises værdilisten i et separat browservindue. Brugeren kan derefter klikke på et hyperlink for den ønskede værdi i LOV-vinduet for at lukke LOV-vinduet og returnere værdien til feltet.

Brug af en LOV som Poplist

Hvis man ønsker at bruge værdilisten som drop-down liste, skal man ændre display type for det tilknyttede bound item til Poplist. Når man gør dette, vises den

første LOV-kolonne i værdilisten i drop-down listen, og der vises intet separat LOV-vindue.

Multi-Row skærbilleder

I alle de foregående applikationer har View og Insert skærbillederne kun vist en enkelt række. Siden Designer version 6i er det imidlertid også muligt at generere multi-row skærbilleder. Med disse kan man indsætte, opdatere eller slette flere records i samme operation. Bemærk at multi-row skærbilleder kun kan defineres for layout-typerne Form og List/Form.

Multi-Row View-skærbilleder

For at generere et multi-row View-skærbillede, skal man sætte modulkomponentens egenskab "Rows Displayed" til en værdi større end 1. Dette er tilstrækkeligt til at fremstille View-skærbilleder, der viser flere records.

Imidlertid ønsker man oftest også at vise hver record på én linie på skærbilledet. For at opnå dette, sætter man normalt også generator preference ZONVTC (View Form: Number of Items per Line) til en værdi svarende til antallet af bound items pr. record. Hvis for eksempel et View-skærbillede viser EMPNO, ENAME og JOB, sætter man ZONVTC til værdien 3, så de tre bound items for hver record vises som én linie.

Multi-Row Insert-skærbilleder

For at give brugeren mulighed for at indsætte flere nye records på samme skærbillede, skal generator preference ZONIBR (Insert Form: Number of Blank Rows Displayed) sættes til en værdi større end 1.

For at få hver record til at stå på én linie sætter man som normalt også generator preference ZONITC (Insert Form: Number of Items per Line) til en værdi svarende til antallet af bound items pr. record. Hvis for eksempel et Insert-skærbillede viser ENAME, JOB, SAL og COMM, sætter man ZONITC til værdien 4, så de fire bound items for hver record kommer til at stå på samme linie.

Master-Detail moduler

Når man har en master-detail relation mellem to tabeller, kan man oprette moduler, der bruger denne relation. For at gøre dette, opretter man en modulkomponent for hver tabel og tilføjer et *Key Based Link* mellem de to table usages.

Som standard fremstiller Designer hyperlinks til alle detail-rækker fra det skærbillede, der viser masteren. Man kan dog også vise detail-rækker på maste-rens View-skærbillede. For at opnå dette, skal man:

- Sætte Layout Style for både master og detail modulkomponent til 'List'
- Sætte Query? for alle items i detail modulkomponenten til No. Man kan ikke vise en master-detail relation på én webside, hvis detail-modulkomponenten åbner et Query-skærbillede.
- Åbne moduldiagrammet, skifte til Display View og trække den hvide Window-ramme til den dækker begge modulkomponenter.

Det er *ikke* muligt at opdatere *både* master og detail på samme webside. Hvis man vil opdatere begge, skal man ændre masteren på View-skærbilledet for masteren og derefter klikke på det relevante detail-

hyperlink for at gå til View-skærbilledet for detailrækken.

Plads til programmering

Alle disse eksempler har vist, hvad man kan opnå ved blot at sætte properties og generator preferences. Men der er selvfølgelig en grænse for, hvor langt man kan komme med blot at skrive værdier i dialogbokse. For at give mulighed for at komme ud over standard-funktionaliteten, giver Designer mulighed for, at udvikleren kan tilføje sin egen kode på foruddefinerede forbindelsespunkter.

Dette giver mulighed for at definere særlig behandling af bestemte begivenheder – svarende til de triggere, vi kender fra Oracle Forms. Man er stadig begrænset af de skærbilleder og den navigation, som Designer bruger, men med PL/SQL kan man reagere på server-events og med JavaScript kan man reagere på events i webbrowseren.

Egen PL/SQL-kode

Den PL/SQL-kode, man tilføjer, bliver naturligvis udført på serveren - webbrowseren forstår jo ikke PL/SQL.

Man kan tilføje sin egen PL/SQL-kode på modulniveau eller på modulkomponent-niveau.

PL/SQL-kode på modulniveau

På modulniveau kan man kun tilføje erklæringer (private eller public). Disse erklæringer bliver medtaget i den PL/SQL-pakke, som Designer bygger for modulet.

PL/SQL-kode på modulkomponentniveau

På modulkomponent-niveau kan man tilføje både erklæringer og event-triggere. Erklæringerne bliver medtaget i den PL/SQL-pakke, som Designer bygger for modulkomponenten, og trigger-koden bliver udført som svar på bestemte events. Følgende triggere står til rådighed:

- PreQuery
- PostQuery
- PreInsert
- PostInsert
- PreUpdate
- PostUpdate
- PreDelete
- PostDelete
- ServerValidate

Egen JavaScript-kode

Med kode på serveren har man adgang til alle muligheder, som PL/SQL giver, og adgang til alle databa-

sens data. Men hver gang man har brug for at udføre kode på serveren, skal brugeren trykke på en knap eller klikke på et hyperlink og derefter vente på, at serveren udfører sine beregninger og reagerer med en ny webside.

I nogle tilfælde giver dette unødvendig ventetid – hvis man blot ønsker at kontrollere, at brugeren har indtastet en gyldig e-mail adresse, behøver man ikke at spørge serveren for at finde ud af, om brugeren medtog et "@" tegn. Denne form for validering kan klares med JavaScript, der udføres af browseren.

I Designer kan man tilføje JavaScript kode på modulkomponent-niveau eller for et enkelt bound item.

JavaScript på modulkomponentniveau

På modulkomponent-niveau kan man tilføje en Validate-trigger. Denne trigger udløses hver gang brugeren er færdig med et skærbillede, så den kan bruges til validering, der er afhængig af flere felter.

JavaScript for et bound item

For et enkelt bound item, kan man tilføje nogle få typer triggere:

- OnBlur
- OnChange
- OnClick
- OnFocus
- OnSelect

I hver af disse triggere kan man skrive sin egen JavaScript-kode, der bliver medtaget i den genererede applikation og udført, når den tilsvarende JavaScript-event udløses.

Konklusion

Denne artikel har demonstreret nogle af fordelene ved at bruge Oracle Designer til at fremstille PL/SQL web-applikationer:

- Høj produktivitet – fuldt funktionsduelige applikationer som disse kan udvikles på under en time.
- Høj kvalitet – Oracle Designer bygger al koden, så der er mindre mulighed for programmørfejl.
- Gode vedligeholdelsesmuligheder – applikationen er gemt i Designers Repository, så en ny programmør kan let arbejde videre med den.

Referencer: [Oracle Web Applikations 101](#) af Sten Vesterli, ISBN 0-07-213221-3.

Udgivet maj 2001 af Oracle Press (udsolgt fra forlaget, men kan af og til fås på eBay)



www.PythiaInformation.com

Designer-eftersyn

Udnytter I Designer optimalt?

Jeres Designer-installation efterses for bla:

- Udviklingsmetode
- Konsistens
- Genbrug
- Opsætning af præferencer
- Maksimal automatisering
- Dokumentation
- Sikkerhed
- Brug af Workarea

1 dag inkl statusrapport: DKK 9.500



DON'T-ASK-ME-NOTHIN'-ABOUT-NOTHIN'

I - JUST - MIGHT - TELL - YOU - THE - TRUTH

MED-MORTEN-EGAN

This is a difficult interview for me to conduct. Morten Egan is a very, very bright young man with a hell of a lot of knowledge about Oracle products, and especially around iAS and OracleAS.

But recently he has shown symptoms of light to medium-strength depression. After a few beers and a lot of talk, it is my conclusion that the iAS and AS products are to blame.

Claiming this without hard proof is not fair to Oracle at all, so this is a first attempt at explaining why these products cause real damage to people charged with deploying them in anything but standard and simple ways.

During the interview it dawned on Morten and me that there is enough material for at least three interviews. So this is the first of a minimum of three A Man and his AS interviews.

The second interview will be about High Availability (HA) options in iAS/AS, and the third interview will focus on the documentation. If – contrary to what we are used to – anyone should react to this interview with comments, critique or even tips & tricks, I'll be happy to include these in the following interviews.

I think you'll have fun reading the interview, since humour, irony and sarcasm are tools widely used by both Morten and myself, but you should also gain some hardcore insights into the challenges you might face when you need to deploy iAS and AS from Oracle.

So, Morten, what have your career been like so far? Where have you worked and what have you (mainly) been occupied with?

Well, my career with Oracle started just after college. I was fortunate enough to get a trainee position in Support in Oracle Denmark. That trainee period was spent mostly in the database support group, for the database on VMS/*nix/MVS platforms. After a year as a trainee, I was hired as a regular supporter. After close to two years in support, I was called to headquarters to learn about this new fantastic product that Larry had announced, called Raw Iron. The idea was that I learned about the product from the development group, and then I would teach this to the other supporters in Europe. As it turned out, when I got there, there really was no product yet to learn about, so I spent my three months developing on the product, more than learning about it. I apparently did something right, because 6 months later I was offered a full time position in the group. After development, I had a short stint back in Oracle Denmark in the consulting group. Today I work in Miracle A/S.

What did you do in Oracle Development?

Well, the product I was working on, was called Raw Iron. It was supposed to be an easy to install, black box database. So the target customers were small mom-and-pop shops that could not afford to hire a full time dba, but still had to run an Oracle database. So the goal of the product was to create a self-sustaining database, that on a low level tried to continuously

clean/tune/administer itself. Not to a level of superb performance, but to a level where the only human intervention would be to change the backup tape.

So my tasks with this product involved two main areas. One was to create and maintain patches and product updates. The other was to try and define the actions the database should take, when running into the different kinds of administration issues you would normally encounter as a normal dba.

What really happened to project Raw Iron/Appliance? Was it sold to many customers? Has it been stable at the customers that bought it?

So what happened to this obscure product that not many people know existed? Well as soon as it hit a maturity level where it was actually working, Oracle decided to change strategy (big surprise), and Raw Iron/DB Appliance was scrapped, and most of the group were moved to creating/defining/testing what Oracle calls "Oracle Certified Configurations for Linux" (It should be noted that a lot of the features we created for this product later moved into the main database. Things like drop of a tablespace including the datafiles etc.).

It was sold to a lot of customers that didn't really need it, which I think was the biggest failure in getting this product out. Since it was a black box (you were only allowed to have sqlnet communication with the machine, no telnet or ftp) it does not sell well to a company that already has a dba department. Imagine that you as a dba were told that you couldn't change anything to the database at all! Not many dbas would accept a product like that.

The few customers who bought it (and used it successfully) though, ran really stable. Since I was from Denmark, the product got a bit more attention in Denmark, and we had about 5 customers here, two of them have had problems. One was because the clean-up jobs didn't get installed, so after running for



~2 years without any cleanup in log files etc. the machine simply ran out of disk space. The other customer had some hardware problems, but that is about it. We have never heard from the three others, so it either runs great or has been replaced by something else.

When did you first come across iAS/OracleAS? And how did it go?

Well, my very first introduction to Oracle web products was with OWS version 2.1. A very simple web-server with the OWA (Oracle Web Agent, PL/SQL cartridge), and not much more. It worked OK, and then I happily forgot everything about the webserver until it was suddenly called iAS R1. At this point I was beginning to get a lot of consulting jobs, where people were moving from Forms in client/server mode, to web based Forms. And this is where the story begins, and my mostly trouble free life ended.

What is your impression of iAS/OracleAS?

Phew, so how can I answer this in a sort of - diplomatic/I still like Oracle - way? Not possible! The releases from 1.0 to 9.0.2 were complete failures, in my opinion. It was the mother of all integration products, trying to get so many different Oracle products to work under the same umbrella. I like to compare it to what Larry hammered into our brains for a couple of years: Best of Breed vs. Complete Suite. He said that if you wanted to ease administration, make sure that everything worked together etc. going with the Oracle OneSuite solution would be far better. I think he used the car analogy to describe the difference. Where if you went best of breed, you would buy the steering mechanism from Ford, the body from Toyota and other parts from whoever had the best reputation. Now to integrate all of this would take a lot of man power, the custom work would be out of date, or need re-programming whenever you needed to upgrade components.

Well, what do you think iAS is? It's best of breed all over again, only this time it's just with Oracle's own products. iAS is this all-encompassing beast, where you run Forms, Reports, Discoverer, J2EE Apps, OWB stuff, OEM, Portal, Wireless, OID, Single Sign-On, Web services, Collaboration Suite (which again is 4 or 5 different products) and what have you. Now instead of being separate products, they suddenly all need to fit into the whole J2EE framework and the architecture of iAS. And let's be honest here, it might all be packed into some kind of fancy J2EE wrapper, so it appears in OEM as really integrated products, but in fact most products had completely different API's and administrative interfaces, and needed a lot of manual reconfiguration if you did anything else than what was default. Just look under the hood of Forms and Reports in iAS, and tell me how much has changed since 6i or even the client/server versions?

And all of this, you could only see, if you could actually install it! I mean the install process until 10g (9.0.4) was flawed to say the least. I used to say that when you install iAS R1 or iAS R2, you will experience pain. Depending on what platform, which products you installed, it could be a small pain or a big pain, but a pain nonetheless. I had one situation at a customer site, where we ran the same installation (iAS R2) 5 times, every time on a completely clean machine, and to no surprise the install crashed 5 different places with 5 different error messages.

AS 10g (or 9.0.4) fixed a lot of the installation issues and some of the core products have been integrated much better, but there are still tons of issues as soon as you stray from default values or get into more complex setups. One thing that really confused me was that in R1 there were absolutely no way of setting up the infrastructure with any kind of HA options. This meant that the infrastructure was a single point of failure. In R2 there was one way of setting up some kind of failover option, but ONLY if you ran on Sun.

And when it comes to patching, it just shows how much all the products aren't integrated. In iAS R1 and R2, the amount of patches that you should apply is not small, and most were just separate updates to the individual products. When your patching process has to be split out for each product (and sometimes they would overwrite each other) it just shows how badly the integration was done.

To sum it all up, I think the song "The Mirror" from the Dream Theater album Awake has a part that says it all:

It's time for me to deal
Becoming all too real
living in fear-
Why did you lie and pretend?
This has come to an end
I'll never trust you again
It's time you made your amends
Look in the mirror my friend

To many people that must sound like a very harsh comment on a product from one of the World's largest software companies?

Yes, it is a harsh comment, but I think there is more than enough evidence in Oracle's bug database to merit this comment. Here is an example of how "professional" they have been with this product:

One customer site needed to install iAS R2 to run a new web based form application, and wanted to use single sign-on. As mentioned before, the HA options were basically non-existent in this version, but I did find some ways to at least duplicate the OID, or so I thought. In the docs for iAS R2, specifically Oracle Internet Directory Administrator's Guide chapter 23 (http://download-west.oracle.com/docs/cd/A97329_01/manage.902/a95192/replic.htm#-1036608) which describes just how you would install OID in a master-to-master replication environment. That manual is a direct copy of the one that comes with the RDBMS manuals. The only problem is that the OID that ships with iAS is NOT the same as the one that ships with the database! Oracle actually states that if you want to use OID for anything serious you HAVE to use the one that comes with iAS and only use the database version of OID for testing and prototyping (Metalink note#233396.1). Now as it turns out OID master-to-master replication setup is NOT supported in the iAS version, but they STILL have the docs from the RDBMS docs in there, stating it is.

And if you want all the horror stories, mail me at meg@miracleas.dk. There is simply not enough space in this interview for all the goodies.

And if you go into the bug database, you will find an endless list of just as silly documentation bugs, simple configurations issues etc.

One of my favourite ones were in one of the first versions of 10g, where part of your hostname couldn't be "web"! That specific one does seem to be fixed by

now, however.

But still my general feeling is, that the testing of the product is limited to individual products, and not how they integrate into the rest of the stack of iAS products.

What aspect of working with iAS/OracleAS do you think contributes most to you going into depressions?

In the older releases the install/patching part was the biggest contributor to my mental health. Today, it is by far, HA setup and configuration. That area is simply not well thought out, really badly documented and so different again, pending on your platform and what products you wish to install. The fact that the installation behaves differently for each customer, even though the setup is the same, is another big factor.

For instance I always feel that I leave an iAS customer only 75% satisfied no matter what. There is always some corner of a customers requirements that won't work, because it's not Oracle's default. It feels like you have to beat the horse, even though it's been dead for a long time, and when you're finished beating it, you have to figure out, how you get this dead horse back on its feet.

As someone once said: There are no present or future, only past repeating itself. And that's kind of how I feel about the iAS/AS.

Can you name specific things for consultants to watch out for before going into iAS/OracleAS projects?

There are many things that you should consider. First off is the HA issue. Many of the client sites I've been at do not need all the fancy clustering/load-balancing features. Most actually only need Forms and Reports, which cannot even take full advantage of all the clustering options because it is not a pure J2EE application.

Start out by eliminating as many products as possible. Even though it is easy to install the whole lot (I see a lot of Forms/Reports only installations with a fully configured Portal installation for instance) restrict the amount products that gets configured to the smallest number possible. The fewer integration points, the fewer issues you'll have with the product.

When it comes to upgrading from the older versions of iAS to the new 10g, DON'T. Make a fresh install of 10g, and start from scratch. It's easier, faster and will give you far fewer problems. Patching is a completely different issue, way too big to cover here.

Be prepared to learn the guts of OID and SSO. These two components are very important in most setups, and also very non-intuitive when it comes to very specialized customer setups.

Please describe the most complex setup/project you have done so far involving iAS/OracleAS, and also - if you can without starting to openly cry - describe some of the problems you faced there?

One involving four or five mid-tiers (each duplicated on two machines), all with their own infrastructure database, but everyone using the same OID. This setup is the one I talked about earlier, with the replication problem. The setup today is that all middle-tiers are clusters with cold failover, and the repository and identity management on a CFC setup as well. This project started out on R2, and this is where I've encountered most of the really bad documentation errors

that I've seen. Today it is 10g (9.0.4) and one of the only client sites where I think most things are working ~90% now.

Any other signs around you that you're working with technology that wears you down?

Well when you asked for this interview I thought it would be fun seeing what music I had been listening to during my past iAS/AS projects, and I think the band and album names says it all:

1. Faith No More – The Real Thing
2. Life of Agony – Soul Searching Sun
3. Dream Theater - Awake

And especially the Faith No More album "Real Thing" has songs listed in a way that pretty much describes how an iAS/AS consultant feels:

1. Edge of The World
2. Epic
3. Falling to pieces
4. From out of Nowhere
5. Surprise! You're dead!
6. The Morning After

Also, the book on my nightstand is about the war in Iraq.

How do the people around you react to all this?

Well it depends. Most customers are of course disappointed when Oracle doesn't deliver what it promises, and I'm an easy target to blame. I don't know how my friends have reacted to this, since I haven't talked to them for a while!

Do you have some good advice for consultants moving into advanced iAS/OracleAS projects? Before, during, and after?

Understand the differences between the various cluster options that exist, both for the middle-tier and for the infrastructure. This is one area where at least for the middle-tier the doc is pretty good. For infrastructure read the different white papers instead of the docs.

Get a detailed level of knowledge on OID and SSO. This cannot be said too often. Every Oracle product is beginning to use it more and more.

Know your DCM and OPMN tracing options!

Don't be scared of XML.

Check for patches and updates!

Don't be afraid to try out things, just don't expect success the first, second, third ... n'th time.

Why has iAS/OracleAS ended up like this? Is this something you have observed with other product collections?

The main reason for iAS/AS being this "loosely coupled" I think, is that when Oracle decided that the iAS/AS was the thing, they had to integrate so many different products, from different development groups, from different parts of the world, and in any software development group that size, you will have competition on who owns what, and how things should be done. Another reason is just the way the software business works today. If you don't release a new product version every three months, with whatever new buzz technology, you lose market share. That, unfortunately, will contribute to companies releasing software without proper testing done, leaving that to the custo-



NU

Bør du tilmelde dig Den sidste DatabaseForum.

- Det foregår på Lalandia, Rødby
- Det bedste program nogensinde i 4 parallelle spor
- Mød alle OakTablemedlemmerne og kollegerne Jonathan Lewis, Tim Gorman, Connor McDonald, James Morle, Pete Sharman, Jarel Still, Miraclerne og mange mange flere flere
- Specielt spor for SQL Server – Kimberly Tripp, Gert Drapers, Mario Broodbakker og Anjo Kolk

Tilmeld dig på www.miracleas.dk

NU

Starter Miracle SQL Server Research Team.

- Performance og internals er fokus
- Instrumentering af SQL server i samarbejde med development i Redmond
- Er du interesseret i at være med i denne gruppe, skal du kontakte Mogens Nørgaard, MNO@Miracleas.dk eller tlf: 2527 7100

NU

Bør du forhåndsregistrere dig til Miracle Master Classes 2005

- MMC Oracle med Tim Gorman, 17-19 januar
- MMC Databaser med Chris Date, dato i marts/april fastsættes snart
- MMC SQL Server med Kimberly Tripp, 2-4 maj

ODTUG-KONFERENCEN 2004



Marc de Oliveira har arbejdet som Oracle systemudvikler og system-analytiker i 14 år med primær interesse for Designer. Han sidder i ODTUGs bestyrelse, koordinerer OUGDKs Designer SIG, er skandinavisk forhandler af DesignAssist, og OracleEksperts ansvarshavende redaktør. Email: Marc@PythiaInformation.com.

Indledning

Fra den 20.-24. juni afholdt Oracle Development Tools User Group sin årlige konference. Denne gang i Scottsdale, Arizona.

For mit vedkommende begyndte konferencen allerede to dage tidligere, idet jeg som bestyrelsesmedlem også deltog i planlægningen af brugergruppens fremtid ved bestyrelsesmødet den 18. og 19. juni. Hvad der kom ud af det, kan du læse sidst i artiklen.

Business Rules

Det er efterhånden blevet en fast tradition at konferencen indledes med en hel dag om forretningsregler, kaldet The Business Rules Symposium, under kyndig styring af Dr. Paul Dorsey fra Dulcian.

Dulcians eget Business Rules produkt hedder Business Rules Information Manager (BRIM), og det lader til at være blevet ret modent. Ideen med BRIM er at alle forretningsregler registreres i et simpelt Repository, som forespørges hver gang der sker en begivenhed i et skærm billede.

Udvikler man feks sine skærm billeder i Forms, skal man sørge for at bruge en bestemt BRIM-template, men man må endelig ikke tilføje noget logik i skærm billederne. Man skal blot oprette alle nødvendige blokke og felter, og så vil BRIM-systemet sørge for at de rigtige folk med de rigtige rettigheder vil kunne se, fremsøge, opdatere mv det, som de skal kunne.

Paul Dorsey er meget inspireret af Designers strukturerede tilgang til systemudvikling, hvilket BRIM også bærer præg af. BRIM bruger dog UML til datamodellering (klassemodellering) i stedet for Designers ER-værktøj. Og til processer anvendes UseCase.

På den måde havner BRIM et sted midt mellem Designer og JDeveloper. BRIM er et objektorienteret udviklingsværktøj (som JDeveloper), men med en meget stram struktur (som Designer).

Et andet principielt element ved BRIM er at klassedia-

grammer ikke oversættes til tabel-DDL (som JDeveloper gør det), men til view-definitioner (med tilhørende tabel-DDL, selvfølgelig). Dette er en vigtig forskel, og den kære Paul Dorsey kan meget vel have fat i den lange ende. Hans holdning er, at det er umuligt at implementere et klassediagram som relationelle tabeller. Dem der forsøger falder altid i een af følgende tre faldgruber:

1. Brug af objekter. Denne løsning afvises totalt af Paul Dorsey, som hævder at ingen bruger indlejrede tabeller og lignende i virkeligheden. Oracle har implementeret det for at kunne kalde deres database Objekt-Relationel, men Oracle bruger ikke disse elementer selv. De er langsomme og fuldstændig umulige at administrere.
2. Tilpasning af designet. Det er en fundamental fejl at tvinge sig selv til at designe klassediagrammet så det kan implementeres i en relationel database. Dette vil med sikkerhed ødelægge det objektorienterede fokus.
3. Rå implementering. Modsæt punkt 2, kan man vælge bare at implementere hvert objekt som en tabel med kolonner svarende til objekternes attributter. Dette vil bevirke at man med sikkerhed vil ende med en inkonsistent datamodel, som er umulig at vedligeholde, tilpasse og udvide.

BRIMs løsning er simpel: Ud fra klassediagrammet genereres et sæt views, der afspejler klasserne (svarende til punkt 3 ovenfor). Under disse views bygger man så den korrekte normaliserede relationelle tabelstruktur.

BRIM er ikke længere et gratisprodukt, men for USD 5000 kan man købe en licens - tjek evt www.dulcian.com.

Key Note talen

Den egentlige konference begynder med Key Note talen, som denne gang blev holdt af Christophe Job, da den ellers altid underholdende Bill Dwight jo fratrådte som Vice President for Application Development Tools for et halvt års tid siden.

Christophe Job er den nye Vice President for Application Development Tools. Var udvikler hos Borland fra 1989, hvor han både var involveret i udviklingen af Turbo Pascal og Borland C++. Han begyndte hos Oracle i 1995, og har siden 2002 været leder i Application Development Tools, dvs Forms, Designer og JDeveloper.

Vel vidende, at han aldrig ville kunne underholde på samme niveau, som Bill Dwight havde gjort det før, havde han klogeligt valgt at spille "det ærlige kort". Hans tale var nedtonet og uden de sædvanlige slogans og komiske pointer. I stedet forklarede han nøgternt hvorfor han mente, at Microsofts udviklingsværktøjer havde fået meget mere



Figur 1. Christian og Birgit på terrassen foran golfbanen

succes end Oracles, og hvordan Oracle nu forsøgt at blive en seriøs spiller på markedet for udviklingsværktøj. Han fortalte dog ikke noget om, hvad der ville ske med værktøjerne i fremtiden. Han var meget bange for at komme til at love noget, som han ikke ville kunne levere.

Det var tydeligt, at han vidste mest om JDeveloper (koden til JDeveloper kom jo oprindeligt fra Borland), og sandsynligvis aldrig havde brugt Designer. Da der under Key Note talen blev spurgt ud i salen, hvem der brugte Designer rakte over halvdelen af de 500 deltagere hånden op. Ved det tilsvarende spørgsmål om JDeveloper var der ca 10 hænder at se. Et par stykker fra ODTUGs bestyrelse fik dog fat i Christophe samme aften (eller var det dagen efter), og han lod til at lade sig belære om Designers fortræffeligheder. Også Svend-Erik Jensen, som jo arbejder i Oracle Supports ledelse i Orlando, fik en personlig snak med ham, hvor dette emne bla også kom op. Han viste forståelse for behovet for, at der stadig burde foregå en vis videreudvikling af Designer, men han lovede intet konkret, hverken under talen eller under de uofficielle møder.

JDeveloper 10g

Der var en del indlæg om JDeveloper og særlig interessante var de, der handlede om JDeveloper 10g, hvor de første spor af det JRad-produkt, som har spøgt i kulissen i de sidste tre år, viste sig. Nu hedder det i stedet ADF (Application Development Framework).

Vha ADF kan man nu trække et ViewObject ud på siden med en option om at danne en input form, et søgeskærm billede, en graf mv. Herefter ser man en visuel model af det skærm billede, man vil få. Her kan man flytte rundt på felterne, rette prompter, tilføje tekster mv, hvorefter man kan køre applikationen.

ADF kan bruges til at lave JSP eller UIX. Der er endnu ikke mulighed for at generere JavaFaces via ADF.

Til sidekontrol genererer ADF Struts, som efter sigende er besværligt at gøre manuelt. Igennem ADF sker udviklingen af Struts via et PageFlow diagram, hvor man trække processerne som pile mellem de sider, man har udviklet.

JDeveloper er tydeligvis på vej til at blive et mere og mere deklarativt værktøj, men der mangler stadig helt centrale elementer for at kunne kalde det et værktøj for struktureret og modelbaseret udvikling. På dette punkt er Designer stadig det eneste anvendelige værktøj (læs mere om dette i lederen på side 2).

Designer

Så er der det kære smertens barn. Hvad skal der mon blive af Designer? Det var der en hel del interesse om.

Oracle gjorde en del ud af at understrege at Designer ikke er ved at blive desupporteret, så længe man sørger for at følge med versionerne. Der er dog ingen planer om at udvide Designers funktionalitet.

Der er ikke planer om at lave mere end een opgradering mere af Designer 6i. Det bliver patch 4.11, som er planlagt til at komme til september. Error Correction support slutter 1. januar 2005.

Der er planlagt endnu fire patches til Designer 9i og en afslutning på Error Correction support den 1. juli 2005. Der er planlagt de samme fire patches til Designer 10g, og her er der ingen planlagt afslutning på supporten.

Alle (dette inkluderer Oracle og avancerede JDeveloper-brugere) er enige om at Designer stadig er det eneste rigtige CASE-værktøj og det eneste værktøj, der understøtter struktureret og modelbaseret udvikling. Abstraktionsniveauet i JDeveloper kan med 10g sammenlignes med et RAD-værktøj som Forms.

Det eneste værktøj, der kan håndtere metadata om data og applikationer på en struktureret måde vil fortsat være Designer så længe JDeveloper ikke får et egentligt Repository.

Skulle man have behov for at udvikle Java-applikationer i JSP eller UIX, er det muligt at lave analyse og design i Designer og derefter generere skærm billederne med JHeadstart.



Figur 2. Lene og Marc ved Western-festen

JHeadstart

Selv om navnet JHeadstart lyder som om det er JDevelopers udgave af Headstart, så er det slet ikke tilfældet. Headstart er en samling templates og biblioteker mv som tilføjer avanceret funktionalitet til Forms genereret ud af Designer.

JHeadstart er en egentlig generator, der genererer JSP og UIX billeder i JDeveloper ud fra moduler i Designer.

I visse sammenhænge sælges JHeadstart som et værktøj, som kan hjælpe Designer-udviklere til at migrere deres Designer-applikationer til JDeveloper, men for mig at se, er en væsentlig bedre løsning, at beholde sine metadata i Designer og bruge JHeadstart som en almindelig generator på linie med Forms-generatoren.

Da JHeadstart generatoren ikke er lige så avanceret som Forms-generatoren, er man nødt til at tilføje visse elementer manuelt i JDeveloper efter genereringen. Dvs man skal ikke forvente at kunne lave 100% generering. Generatoren har til gengæld en forholdsvis rig funktionalitet til at kunne styre regenerering, så dan at

man kan generere ændringer i Designer modellen over i JDeveloper uden at miste de tilføjelser, man har lavet direkte i JDeveloper.

Hvis nogen er interesseret i at forsøge sig med at bruge JHeadstart som generator, så er I velkomne til at trække på min assistance.

Netværk

ODTUG er udover at være en konference med over 150 tekniske sessioner, også et forum for at udveksle erfaringer og danne netværk.

Sunset Sessions

Hvad der tidligere hed Lunch Bunch, og foregik i frokostpauserne, er nu omdøbt til Sunset Sessions og ligger i stedet efter de normale sessioner. Ideen med Sunset Sessions er at folk, der var interesseret i specifikke teknologier kunne mødes og diskutere aktuelle emner om de pågældende teknologier, efter eget valg. Hver Sunset Session handler om en bestemt teknologi, som f.eks. Data Warehousing, og der er på forhånd udpeget en ordstyrer. Ved sessionens begyndelse spørger ordstyreren om hvilke specifikke emner, der er interesse om at få diskuteret, og så er det op til alle deltagerne at få noget konstruktivt ud af de 75 minutter, som sessionerne varer.

Jeg deltog i Sunset Sessionen om Designer, SCM og Datamodellering, hvor et af de varme emner var om man skulle blive ved med at bruge Designer, eller om det var nødvendigt at skifte til JDeveloper. Udgangen på diskussionen blev at der ikke var nogen grund til at skifte fra Designer til JDeveloper så længe Designer stadig var det mest egnede værktøj til struktureret og modelbaseret udvikling. Det var dog en god ide at holde sig orienteret om udviklingen i JDeveloper, som med sikkerhed vil få flere og flere muligheder for at understøtte struktureret udvikling. Hvis det er nødvendigt for een at udvikle Java-applikationer, er det muligt at lave al modelleringen i Designer og derefter bruge JHeadstart til at generere modulet som JSP eller



Figur 3. Afslapning i poolen efter en hård dag

UIX i JDeveloper.

Danskerne

I pauserne og om aften var de danske deltagere også flittige til at netværke enten i pool-området, i baren eller på nogen af de mange restauranter, der lå i gåafstand fra hotellet. Udover netværk, så er dette også en god mulighed for at få skilt skidt fra kanel i al det man får hørt om i løbet af konferencen. Ved at vende tingene med hinanden bliver det klarere, hvilke pointer, der er vigtige, hvilke indlægsholdere man er enige med og hvem man skal undgå i fremtiden.

SCM og OracleEkspert-Prisen

Jeg var ambassadør for Lauri Boyds indlæg om "Extreme SCM - Lessons Learned", som var opfølgeren til den artikel, der sidste år vandt både ODTUGs Editors Choice Award og OracleEkspert-prisen.

Som en del af min præsentation af hende overrakte jeg hende OracleEkspert-prisen med tre af Jesper Dyjaks originale Oraclerne-strips, som hun blev meget glad for. Jeg forsøgte at overbevise hende om at strips'ene ikke kunne oversættes til engelsk, men hun sendte mig efterfølgende en mail, hvor hun insisterede på at få dem oversat, idet alle mennesker, som så prisen spurgte, hvad der blev sagt.

Desværre havde jeg ikke en fotograf med, der kunne tage et billede af overrækkelsen... :(

ODTUG Bestyrelsesmødet

Afslutningsvis kan jeg lige løfte sløret for nogle af de ting, som blev besluttet på bestyrelsesmødet i Scottsdale.

I løbet af to dage nåede bestyrelsen at diskutere og stemme om ikke mindre end 24 forslag, som var blevet fremlagt. Blandt de væsentligste kan nævnes følgende fem beslutninger (hvoraf de sidste to blev stillet af undertegnede):

1. Virtuel konference. Det blev besluttet at ODTUG skal planlægge en virtuel konference i samarbejde med Business Intelligence/Data Warehouse SIG'en. Konferencen er planlagt til at skulle afholdes i februar 2005.

2. ODTUG 2005. Trods rygter om at ODTUG ville holde en kombineret konference sammen med IOUG og AOUG i DisneyWorld i maj 2005, blev det besluttet at ODTUG afholder sin



Figur 4. Mari Cobb (fra ekspertkomitestyregruppen), Rune og Svend Erik



Figur 5. Marc, Rune og Thomas

egen konference til næste år med samme struktur som hidtil.

3. ODTUG 2006. Det blev besluttet at gå ind i en diskussion med IOUG og AOUG om vilkårene for en eventuel kombineret konference i 2006. Det er ikke besluttet, om vi ønsker at afholde en sådan fælles konference.
4. Ekspert-komité. Det blev besluttet at oprette en ny komité med følgende to formål: 1) at skabe initiativer, der kan tiltrække de dygtigste folk indenfor Oracle-udvikling, og 2) at sørge for at den ekspertviden som ODTUGs medlemmer besidder i størst muligt omfang kommer ODTUGs medlemmer til gode. Vi nåede allerede på konferencen at få skabt en styregruppe med Jeff Berkun som formand. Vi holdt hele tre møder (kl 7 om morgen - inden indlæggene begyndte), og har en stribe ideer til nye initiativer klar. Hvis du har forslag eller ønsker at blive involveret i dette udvalg, så send mig en mail.
5. Personlig komité. Det blev også besluttet at oprette en komité med det formål at få værnnet om og forstærket det personlige forhold mellem ODTUG som organisation og medlemmerne. Der er har ikke været så meget fremdrift i denne komité endnu, men det skal nok komme ind til videre er jeg selv formand og eneste medlem i udvalget, så jeg er meget interesseret i at høre fra folk, der kunne tænke sig at deltage.

Konklusion

På trods af at Oracle ikke har haft mange revolutionerende nye produkter på banen, som kunne præsenteres ved konferencen, har der alligevel været mange vigtige aktiviteter, med de tekniske indlæg som det mest centrale. Men også diskussionsmøderne, netværksmulighederne og den intime stemning, der gør det muligt at tage fat i forfattere, eksperter eller Oracle-ledere og stille dem specifikke spørgsmål.

ODTUG-konferencen er den absolut vigtigste konference for såvel nye som erfarne Oracle-udviklere.

Fortsat fra side 14

mers instead.

As I understand it, iAS/OracleAS comes in several flavours, including a Forms/Reports edition, a Java edition, etc. Are these separate products or just subsets of the Enterprise Edition?

These are separate products. The java only version, is a good lightweight apache with just the J2EE framework. The Forms/Reports only version was created because most old forms customers didn't want the whole infrastructure part, if they only wanted to run forms. Enterprise edition is the big beast with everything.

Are the customers happy about iAS/OracleAS implementations? Are you?

NO!

What are your personal future plans with Oracle's middle-tier technology?

I think my new motto in life is going to be: Back to Client/Server, Internet out of IT.

I might revisit the product at a later stage but for now, I think it would be against the Geneva Convention to send me on more iAS/AS jobs.

What's your favourite Oracle product?

For a long time my favourite product was Enterprise Manager, but right now I think my favourite product is the database again. It works, it runs and I usually don't get disappointed when I try something new.

Any final comments about Life, the Universe and Everything?

Only that I would think Websphere must have the same issues as iAS, and that it's going to take years before we get anything as stable as the database on the application server market.

Thanks for the chance to open up, and thanks for all the fish!



Kære læser!

Velkommen til den 14. udgave af Groans, som indeholder lidt sniksnak om Skelbæk-/Perregaard makkerparret, nummerering af Groans, standardkurser med eksperter, bøger, det endelige ord om iAS/OracleAS, endnu en sur bemærkning om 9.2.0.5, en riigtig bitter bemærkning om RAID-5, og så en god nyhed fra Sun til sidst.

Derudover indgår der et interview med Morten Egan, som nok skal give lidt bølgeskvulp.

Skelbæk og Perregaard

Fandens også. Det blev Ementor, der fik ansat Jan Skelbæk. Han bliver salgschef for en samlet salgsafdeling, og det skal nok gå godt. Gid man dog havde haft et firma, der var stort nok til at have ham gående.

Ementor er Oracle-partner, så på den måde er det jo godt nok for Oracle. Men Ementor oplever også stigende efterspørgsel på SAP og Microsoft, og det kommer Oracle til at mærke endnu mere på det danske marked nu.

Peter Perregaard hygger sig som freelance-konsulent, og har det vist temmelig godt.

Derudover har de to meget fornuftigt valgt at gå til kamp mod dette fjollereri fra Oracle's side. Jeg håber vitterligt, at jeg ikke er den eneste, der siger dette her til Oracle – ham fra Oracle Danmark, der oprindeligt sendte bud efter de interne revisorer fra Oracle, burde jo have en eller anden form for belønning fra Microsoft, IBM, SAP, PeopleSoft, BEA og andre konkurrenter. Eller måske en anden slags belønning fra de øvrige ansatte i Oracle Danmark. Nå, men det er vel det man har julefrokoster til. I de gode, gamle dage udnævnte vi til hver Julefrokost Jerk of the Year. Måske var det en ide at genoptage traditionen.

Læs mere på www.computerworld.dk med søgeordet 'Perregaard' eller 'Skelbæk'.

Nummerering af Groans

John Jul Jensen, tidligere Oracle Support, nu DBA Solutions, påpegede i en mail, at der er noget galt med nummereringen af mine Groans i Oracle Ekspert. Det må jeg give JohnJohnJohn (det kaldte vi ham internt i Support) ret i.

Forklaringen er naturligvis helt enkel og simpel:

På et tidspunkt kom jeg til at nummerere to Groans ens i træk, og så var redaktør Marc nødt til at skride ind, så vi sprang et nummer over.

Nummerfølgen bliver derfor sådan her:

1,2,3,4,5,6,7,8,9,10,10,11,13 – tror jeg.

Tak til JohnJohnJohn for tip'et.

Standardkurser med eksperter

Vi prøvede at lade Tuomas Pystynen (som var med til at udvikle RMAN i Oracle Development) holde et standard Oracle RMAN kursus hos Oracle, og det gik fint. Han skal gentage det i november, tror jeg, i Ballerup.

Nu prøver vi så at lade Jonathan Lewis holde et standard RAC kursus 18-20. oktober hos Oracle i Ballerup. Og Lex de Haan vil gerne have lov at holde grundlæggende SQL kurser engang i mellem. Jeg har også overtalte Anjo til at holde standard DBA-kurser nu og da.

Så bliver det da pludseligt sjovt at deltage i standardkurserne igen. Og man kan være sikker på at træffe en instruktør, der har noget real-life erfaring og kan svare på de fleste spørgsmål ☺.

Hvorfor gider de? Fordi de ikke kender Oracle's standardkursusmateriale, og synes det er sjovt at se det. Og fordi det ind imellem er sjovt og udfordrende at undervise i noget grundlæggende.

Bøger

TOTOT (Tales Of The OakTable) var en idé jeg fik sidste efterår, og nu bliver den sgu' til noget. Den fulde titel er "Oracle Insights: Tales of the Oak Table" og ISBN 1-59059-387-1. Den indeholder et kapitel af hver af disse herrer: Dave Enor, Tim Gorman, Kyld Hailey, Anjo Kolk, Jonathan Lewis, Connor McDonald, Cary Millsap, James Morle, Mogens Nørgaard, David Ruthven og Gaja Vaidyanatha. James og jeg har et appendix om BAARF, og jeg har også en intro til hele OakTable-halløjet.

Det er så den anden bog fra OakTable Press – den første var Connor's bog om PL/SQL. I kan finde mere information på www.Apress.com, som OakTable Press labelen hører under.

Jeg har også lige skrevet forord til en tyk bog om mit elskede Wait-Interface (eller OWI, som det hedder blandt de seje nu): "Oracle Wait Interface: A Practical Guide to Performance Diagnostics & Tuning" by Richmond Shee, Kirti Deshpande & K Gopalakrishnan – udkommer på McGraw-Hill.

Der er forskellige bøger på vej fra OakTable Press, deriblandt en SQL-bog af Lex de Haan (en opdateret og engelsk version af en lærebog til de mellemtnge uddannelser i Holland for nogle år siden), en bog om Oracle og Peoplesoft af David Kurtz og vistnok en performance-bog af Joakim Treugut.

Hvis I ikke har læst Cary Millsap's ynglingsbog The Goal, så gør det en dag, hvor I har lyst til at få bundet jeres viden om tidsmålinger i Oracle sammen med en helt anden verden. Det er ovenikøbet en slags roman. Anjo mener, at vi skal skrive en sådan roman om Oracle-verdenen – måske med titlen The Grey Zone ☺. Men det tager godt nok lang tid at skrive bare et enkelt kapitel til en bog. Jeg tør næsten ikke tænke på arbejdet forbundet med en hel bog.

IAS/OracleAS – The Final Countdown

Da jeg foreslog 10gAS communitiet var det – for at være sådan næsten cirka helt 100% ærlig – fordi jeg syntes at produktet havde behov for noget støtte rundt omkring, og måske en samling folk, der tilsammen kunne hjælpe og assistere hinanden.

Modsat med 10gAC, som handlede om 10g databasen – den klarer sig jo fint selv, fungerer generelt fint, også i de nye udgaver, og er et produkt, der med Morten Egan's ord kan give 100% tilfredse kunder.

I visse kredse har vi i et stykke tid joket med, at folk, der beskæftigede sig intensivt med iAS/OracleAS som troubleshooters eller blot dem, der skulle få komplekse setups til at virke, før eller siden gik ned med

flaget. Meget morsomt. Men nu står vi altså bare med et par stykker her i Miracle A/S, som mere eller mindre er gået i mild til mellemsvær depressions-mode – og måske har Morten Egan i virkeligheden identificeret den egentlige årsag: Man har som troubleshooter brug for at se en gedigen succes – at kunden bliver rigtigt godt tilfreds.

Men når forventningerne til iAS/OracleAS er de samme som til databasen, så bliver kunden aldrig rigtig glad. Morten mener selv, at han har en enkelt kunde der er oppe i ca. 90% tilfredshed. Resten ligger omkring de 75% eller lavere.

Morten er ikke hvemsomhelst i den her sammenhæng. Jeg tror han er den skrappeste på det her område vi pt har i Danmark, men hører gerne om andre. Han er bare blevet SÅ træt, at jeg tror det er en begyndende depression. Det gjorde mig alligevel lidt nysgerrig, så jeg har lavet et interview med ham, som jeg synes både er tankevækkende og morsomt. De af jer, der kender Mortens far Mogens Egan vil vide, at han ikke har sin fantastiske sans for humor fra fremmede.

Interviewet bringes separat her i bladet – til gengæld er mine Groans kortere end normalt. Interviewet er på engelsk, fordi jeg ikke orkede at lave et kæmpeinterview i to udgaver, og jeg agter altså også at udgive det her i udlandet – det skal nok give nogle (mere eller mindre forudsigelige) reaktioner fra Oracle, men så må det være sådan.

9.2.0.5

Der er altså en del underlige ting med denne release. Jeg ville holde mig fra den indtil videre, hvis det er muligt. Jeg ved godt, at man altid kan finde bugs i al software, osv. Men det ser ud som om en del kunder bliver mere ustabile af at opgradere fra 9.2.0.4 til 9.2.0.5.

Oracle's førende mand på optimizere hedder Håkon Jacobsen (jeg er ikke sikker på hans efternavns stavemåde). Han er svensk, er Princeton graduate, og regnes for meget dygtig i de cirkler. Han har endda givet navn til Håkon-faktoren i source-koden. Desværre er de hele tiden tvunget til at putte nye ting ind i optimeren, som bare medfører nye og spændende eksekveringsplaner. Det er lykkedes til overflod i 9.2.0.5. Jeg tror ikke på den endnu.

BAARF! RAID-50, 510, 551...

Det bliver til stadighed værre med det der RAID-5 pjat. Det er næsten dagligt jeg får mails fra folk rundt omkring i Verden, der fortæller mig at en kombination af sælgere (det kan man da forstå), chefer fra deres eget firma (det kan undre, men ikke forbavse), og andre teknikere uden indsigt i disk-teknologi har fundet ud af, at RAID-5 er løsningen og SAN er svaret. Og derfor er et RAID-5 SAN lykken.

Det er det sjældent, med mindre man overinvesterer i voldsom grad – men det er jo også hvad sælgere ønsker man skal gøre som kunde. Ikke noget ondt i det, hvis de dog bare ikke LØJ så meget i processen.

Jeg har i et stykke tid hørt om firmaer, der køber RAID-5 (for at spare diske), hvorefter de SPEJLER det. Nu har de så ikke alene betalt for meget, men tillige fået dårlig performance og dårlig availability i forhold til alternativerne. Sælgerne sælger flere diske og flere systemer, og kompleksiteten gavner leverandøren.

Samtidig er systemerne blevet så dyre, at der er dømt succes. Uanset hvad så er det her spejlede RAID-5

system IKKE en fiasko, men en fantastisk succes. Det var jo dyrt, så det må være godt. Loven om større tal.

Men Gud Hjælpe Mig, om ikke sælgerne er begyndt på noget nyt og rigtigt smart nu: Når DBA'en har brokket sig tilstrækkeligt meget får han (og hans chef, som typisk ikke gider høre mere vrøvl i sin rolle som Admiralen på Pinafore) at vide, at alle kan blive glade! Yessir – man kan jo lave virtuel RAID-0 ovenpå sine RAID-5 stripes!

Ja, så er alle jo glade, og performance er nu blevet ihvertfald dobbelt så dårlig. Til gengæld er availability gået yderligere ned ad bakke. Men chefen slipper for at høre mere bøvl (han orker nemlig ikke at tage kampen op mod sælgerne af RAID-5 SAN'et), og endnu en dårlig og dyr løsning er blevet installeret.

Jeg ved, at det sikkert ikke nytter noget alligevel, men det er bare svært at affinde sig med idioti. Det er lidt ligesom at acceptere det danske retursystem på udenlandske øl, som kun er til gavn for Carlsberg. Det er bare svært, ikke?

Hvorfor skal vi i de næste 10-15 år se en bunke smarte sælgere trække over i andre brancher efterhånden som folk kommer efter dem, og de ikke kan sælge oversmarte og –dyre SAN-løsninger? Hvorfor skal vi se disse sælgere skyldeskylden på teknikere i deres firmaer, som sagde god for løsningerne? Hvorfor skal vi se alt for mange IO's (og dermed penge) blive brugt?

Det er sjovt at se, hvordan de unge, ambitiøse mennesker strømmer ind i ejendomsmæglerbranchen lige nu – fordi der er hurtige og store peng på tjene. I IT-verdenen samles de omkring SAN'er. Det burde få enhver til at tænke sig ekstra godt om.

Fremfor alt: Hvorfor skal vi i de næste 10-15 år høre chefer (som er blevet narret til at tage en dyr og dårlig beslutning) sige ligesom en tidligere, dansk politiker Jens Busk: 'Ja, hvis det er fakta, så benægter A fakta' og bede os om at finde en anden årsag til performance-problemerne i systemet? Dødssygt.

Så Jesper Haure har foreslået at tage det her pjat til sin logiske konklusion: Lad os få RAID-10 (den rigtige løsning) virtuelt ovenpå RAID-5 (RAID-510). Eller RAID-5 ovenpå RAID-5 stripsene og derpå spejling (RAID-551)? Loven om større tal viser jo, at det må være bedre.

Eller brug RAID-6 (som måske hedder det, fordi en enkelt skrivning medfører hele seks IO'er) – så kan vi endelig få RAID-666.

Der kommer et par nye medlemmer til BAARF-partiet om ugen i øjeblikket (fra hele Verden). Andre må opleve det samme pjat. Der sælges ud af SAN'er lige nu, og der gives meget store rabatter. Kunne det være fordi der er en ny generation af billige, rigtige SAN'er på vej? Ja! ISCSI og SATA teknologierne vil gøre det muligt at levere det rigtige til en ordentlig pris.

God fornøjelse.

Dtrace fra Sun

Sun har annonceret Dtrace, som muliggør at man linker sig op på programmer og får at vide, hvor tiden går hen. De har været et par år om at lave det, og det er lige præcist den slags instrumentering af kode på Unix vi har brug for.

Det er desværre nok for lidt og for sent, for Windows og Linux ruller deruda' – men ros til Sun for at gøre det rigtige.

Sidste retsmøde i PeopleSoft-sagen

2004-07-20

Så er det sidste retsmøde blevet afholdt i PeopleSoft-sagen. Man regner dog ikke med at få dommens afgørelse før slutningen af året.

En del analytikere tror, at dommen kan falde ud til Oracles fordel.

Oracle og Macromedia partnerskab om E-Learning

2004-07-15

Oracle og Macromedia er gået sammen om at forbedre integrationen af macromedias produkter i undervisningsmateriale.

Samarbejdet vil ifølge Oracle gøre det væsentlig nemmere at udvikle funktionsrig undervisningsmateriale.

Oracle får ny finansdirektør

2004-07-13

Harry L You fra Accenture er blevet Executive Vice President og Chief Financial Officer hos Oracle med

direkte reference til Larry Ellison.

Harry L You afløser Jeff Henley, der fratrådte ifm sin udnævnelse som formand for Oracles bestyrelse (hvor han afløste Larry Ellison).

Perregaard og Skelbæk stævner Oracle

2004-07-12

Peter Perregaard og Jan Skelbæk kræver erstatning for uberettiget bortvisning, samt den løn, som de var sikrede ifølge deres ansættelseskontrakter.

Oracle bekræfter ønske om at opkøbe PeopleSoft

2004-07-12

Oracle har forlænget fristen for opkøb af PeopleSoft-aktier frem til den 27. august.

10g kommer til Apple-platformen

2004-06-30

Oracle har annonceret, at de vil frigive både Oracle 10g databasen og JDeveloper 10g på Mac OS X platformen.

Vinnere af ThinkQuest er udpeget

2004-06-16

Oracle har udpeget 12 ud af 400 uddannelseshjemmesider udviklet af studerende fra 29 lande.

De vindende hold er beskrevet i nedenstående link.

<http://www.thinkquest.org/winners>

4. kvartals regnskab

2004-06-15

I 4. kvartal øgede Oracle nettoindtjeningen med 15% til USD 990 mill, svarende til USD 0.19 pr aktie. Sidste år var indtjeningen i 4. kvartal USD 858 mill (USD 0.16 pr aktie).

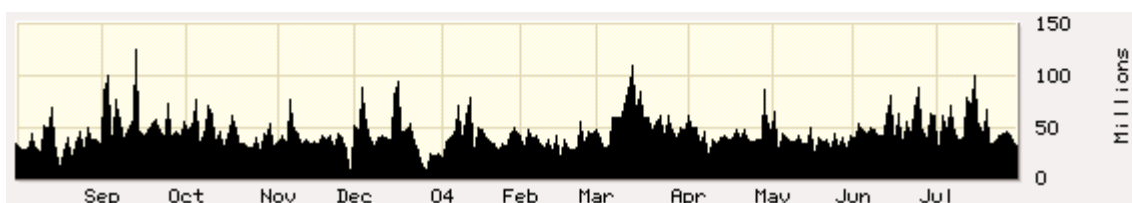
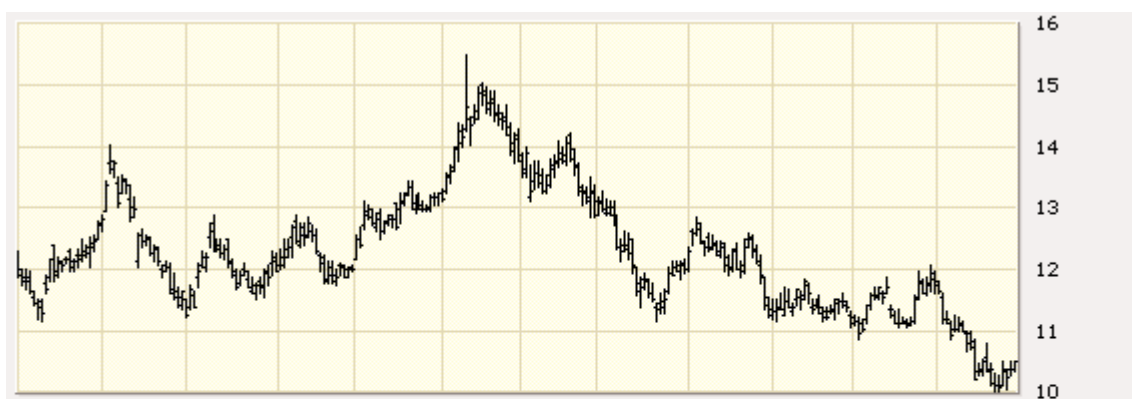
For hele året var nettoindtjeningen steget med 16% til USD 2.7 mia. Dvs at indtjeningen pr aktie er steget fra USD 0.43 til USD 0.50.

Onsætningen er steget 7% til USD 10.2 mia for hele året. Softwareindtægterne steg med 12%, mens service-intægterne faldt med 8%.



Spred det glade budskab!

Referer abonnenter og få
signerede og numererede
tryk af **Oraclerne**





Deadline for Artikler til
OracleEkspert nr 26
er den 15. september 2002

Vi betaler 700 kr pr side

se mere på:

www.OracleEkspert.dk

Eventkalenderen

Her kan alle abonnenter annoncere deres Database-relaterede arrangementer.

Følg med i de nyeste databaserelaterede arrangementer og opret helt nye arrangementer direkte på vores hjemmeside www.OracleEkspert.dk under punktet Eventkalender.

Det er gratis for vores abonnenter at oprette arrangementer i Eventkalenderen.

DesWeb SIG

Dato: 2004-08-18

Kl 13:00 - 16:00

Emnerne er

- 1) Objekt Orienteret udvikling
- 2) The traveling salesman problem

Pris: Gratis

Sted: Oracle Danmark, Ballerup

Kontakt: Tilmelding sker hos jesper.n.larsen@oracle.com

OracleEkspert-konferencen 2004

Dato: 2004-08-25

Kl 9:00 - 16:00

Årets OracleEkspert Databasekonference med stribevis af indlæg om databaseteknologi for både udviklere og databaseadministratorer.

Pris: dkk 2550. OracleEkspert-abonnenter får yderligere

dkk 500 i rabat. Ved tilmelding af fem fra samme firma, deltager den ene gratis.

Sted: Trekroner ved Langelinie

Kontakt: info@OracleEkspert.dk / 2683 9991

Database Forum 2004

Dato: 2004-10-28

Den 28. - 30. oktober 2004

Tre dage med sessioner om databaseteknologi

Pris: se www.MiracleAS.dk

Sted: Lalandia, Rødby

Kontakt: mno@miracleas.dk

Oracle Open World 2004

Dato: 2004-12-05

Den 5. - 10. december 2004

Sted: San Francisco

Kontakt: Se mere på <http://www.oracle.com/openworld>

Miracle Master Class med Tim Gorman

Dato: 2005-01-17

3 dage

Kontakt: www.miracleas.dk

Live



When is good enough
just **NOT GOOD ENOUGH?**

For database management, nothing's better than SmartDBA.

Native database tools. They're cheap. They're easy to buy. And if you listen to some people, they're "good enough." But is that the level of quality you want when your mission-critical data is on the line? If good enough just isn't good enough for your business, you want SmartDBA™ from BMC Software. It ensures your data stays manageable, protected and constantly available. That's why over 80% of the Fortune 500™ rely on SmartDBA. It's the most

comprehensive, cost-effective data management solution in the industry—and the only one that works across all major database platforms. What's more, it's from the company ranked #1 worldwide in Data Management Facilities by IDC for the seventh year in a row. So if the phrase "good enough" doesn't fit into your business strategy, see what SmartDBA can do for your organization. Visit www.bmc.com/smartdba today.