

Oracle Designer 2.1

Grote verbeteringen, gemiste kansen

Een cursuslokaal. Acht ervaren Designer/2000 ontwikkelaars volgen de workshop 'Oracle Designer Upgrade'. Voordat de deelnemers thuis hebben kunnen raken in het nieuwe produkt, wordt de eerste opdracht verstrekt: "Leg een sequence-definitie vast met startwaarde 1 en eindwaarde 1000. Er wordt geklokt hoe lang jullie er over doen." Meteen klinkt er zelfverzekerd gegrinnik. "Mmm, tien seconden misschien?", mompelt een deelnemer. Maar al gauw verstomt het gemompel. Na tien minuten is het nog steeds muisstil en wordt de irritatie voelbaar. Pas na twintig minuten spreekt een deelnemer de verlossende woorden: "Nou, het is me gelukt hoor. Maar als iemand me kan vertellen wat ik heb gedaan, dan hoor ik dat graag!". Oracle Designer 2.1. Om een reclameslogan te gebruiken: "het is even wennen, maar daarna wil je niets anders meer".

De laatste jaren zijn we er aan gewend geraakt dat de marketingafdeling van Oracle haar produkten regelmatig een nieuwe naam geeft. Zo luidt de nieuwe naam van Designer/2000 met ingang van release 2.1 : Oracle Designer 2.1. Maar zoals door de geschetste praktijksituatie wordt geïllustreerd, is er deze keer geen sprake van 'oude wijn in nieuwe zakken'. Het mag dan wel zo lijken dat alleen de gebruikersinterface van Designer is aangepast en de bugs zijn verholpen, het tegendeel is echter waar. Wie de moeite neemt zich te verdiepen in de nieuwe functionaliteit treft een veel volwassener produkt aan, met enorme mogelijkheden, maar een produkt dat wel een andere manier van werken vereist dan voorheen.

In dit artikel zullen verschillende facetten van Designer 2.1 aan de orde komen. Allereerst wordt uitgelegd waarom je überhaupt zou willen overschakelen van release 1.3.x op release 2.1. Vervolgens worden de consequenties opgesomd van het kiezen van de nieuwe release. Hierbij wordt o.a. gekeken naar onderhoud van reeds ontwikkelde informatiesystemen en gebruik van de Headstart Template Package.

In het resterende deel van het artikel wordt een beschrijving gegeven van de nieuwe features van alle onderdelen van Designer (en waar nodig van Developer) en de juiste manier van werken om hier efficiënt gebruik van te maken.

Om de beschrijving compleet te maken worden de onmisbare features die in release 2.1 toch nog ontbreken opgesomd.

Waarom migreren?

De vraag waarom je zou willen migreren van Designer/2000 1.3.x naar Oracle Designer 2.1 is eigenlijk verkeerd. Het kiezen voor Oracle Designer 2.1 is namelijk onlosmakelijk verbonden met het kiezen voor Oracle Developer 2.1. Als je na gaat denken over de migratieproblematiek kunnen er dus beter twee vragen worden geformuleerd:

waarom zou je Developer/2000 1.x willen migreren naar Oracle Developer 2.1, en waarom zou je Designer/2000 1.3.x willen migreren naar Oracle Designer 2.1

Dit betekent dat als één van beide nieuwe produkten aanzienlijke voordelen oplevert voor de onderneming waarin ze worden toegepast, dat migreren dan al de moeite waard is.

Verbeteringen in Developer

We zullen beginnen met een belangrijke verbetering in Oracle Developer 2.1, die met name bij grote bedrijven (en daar wordt Oracle Designer het meest gebruikt) voordelen biedt. De verbetering is terug te vinden in het Runtime Reports tool, dat in release 2.1 de mogelijkheid biedt rapporten op een aparte Reports Server te draaien. Deze Reports Server draait fysiek op een andere machine in het netwerk. Op deze server kunnen rapporten bijvoorbeeld gescheduled worden en kan aan rapporten een soort houdbaarheidstermijn meegegeven worden. Indien binnen deze termijn eenzelfde rapport gestart wordt met dezelfde parameterwaarden, dan zal direct de (bewaarde) output worden gedistribueerd. Een en ander betekent een aanzienlijk lagere belasting van de clients en kortere responstijden op gebruikersacties.

Naast deze verbetering is de nieuwe release van Developer aanzienlijk gebruikersvriendelijker geworden voor ontwikkelaars door de toepassing van vele wizards voor specifieke taken. Voor de Reports component van Developer betekent dit dat je met minder kennis van het tool toch fraaie rapporten kunt bouwen. Uiteraard blijft het ultieme doel het genereren van rapporten vanuit Designer. De verbeteringen in de Forms component van Developer liggen met name op het vlak van de verder doorgevoerde mogelijkheden van object oriëntatie. Kwam release 4.5 van Forms niet verder dan property-classes, met ingang van de huidige release is het concept van sub-classing geïntroduceerd. Hierbij zijn alle Forms-objecten te bewaren in een nieuw soort library: de objectlibrary. Bij het starten van Oracle Forms worden alle objectlibraries automatisch geopend en de objecten daarin zijn direct bruikbaar voor alle forms. Door drag-and-drop vanuit de libraries naar de forms kunnen deze objecten gekopieerd danwel gesubclassed (gerefereerd en eventueel aangepast) worden. Deze verbeteringen kunnen allemaal geschaard worden onder de noemer kostenbesparing door kortere ontwikkeltijden en grotere onderhoudbaarheid.

Verbeteringen in Designer

Een belangrijke verbetering in Designer is de ondersteuning voor de subclassing mogelijkheden van Developer. Moest vroeger bijvoorbeeld voor het genereren van een modal window een specifiek template worden gebruikt, met de huidige release kan worden volstaan met het definiëren van een modal window in een objectlibrary, en het in de repository opgeven dat het te genereren window op dat specifieke window uit de objectlibrary gebaseerd moet zijn. Wat algemener gesteld: in de vroegere releases werkte je met een template form, tegenwoordig kun je op object niveau als het ware een template object opgeven, wat vooraf in een objectlibrary opgenomen dient te zijn. Als een dergelijk gegenereerd form wordt geopend met Developer, dan is te zien dat het betreffende object is gesubclassed vanuit de objectlibrary.

Een andere belangrijke verbetering ligt op het vlak van het versiemangement op niveau van applicatiesystemen. Zo dient eindelijk niet langer de naam van het applicatiesysteem (in feite een bug), maar de combinatie van naam en versienummer uniek te zijn binnen de repository. Daarnaast behoren de problemen met betrekking tot dummy applicatiesystemen die soms optraden bij archive en restore tot het verleden.

Wellicht de belangrijkste verbetering die is doorgevoerd in Designer is de mogelijkheid tot daadwerkelijk 100% generatie van diverse soorten modules, waaronder forms. Was er in de vorige release nog een uitgebreide structuur nodig van template forms, referenced forms en libraries (zoals bijvoorbeeld de structuur van de Headstart Template Package) om tot 100% generatie te komen, met de huidige release kan

programmalogica -bijvoorbeeld de code van een when-validate-item trigger van Oracle Forms- direct in de repository worden vastgelegd. Ook voor Designer kunnen de verbeteringen allemaal geschaard worden onder de noemer kostenbesparing door kortere ontwikkeltijden en grotere onderhoudbaarheid.

Samenvattend kun je stellen dat zowel voor de eindgebruikersorganisatie als voor de ontwikkelafdelingen van ondernemingen tal van argumenten zijn op te sommen om tot migratie over te gaan.

Consequenties migratie?

Als het overstappen op release 2.1 louter voordelen zou bieden, zou de overweging om al dan niet te migreren slechts een 'hamerstuk' zijn voor de verantwoordelijke managers in een onderneming. De realiteit ligt uiteraard iets minder simpel.

Wat zijn globaal de consequenties van de migratie? Deze zijn onder te verdelen in een aantal categorieën: infrastructuur, opleiding en consequenties voor onderhoud van reeds ontwikkelde informatiesystemen, die we kort de revue zullen laten passeren.

Infrastructuur

Zoals eerder opgemerkt betekent kiezen voor Designer 2.1 tevens kiezen voor Developer 2.1. Deze verandering is echter niet alleen merkbaar op de werkvloer van de ontwikkelafdelingen. De runtime Oracle software voor Forms en Reports dient op de eindgebruikerswerkplekken eveneens een upgrade te ondergaan van resp. Forms 4.5 en Reports 2.5 naar Forms 5.0 en Reports 3.0. Indien deze software op een fileserver te vinden is, heeft deze upgrade niet veel te betekenen. Bij een groot aantal lokale installaties kan dit een ander verhaal worden.

Bij het ontwerp van Designer 2.1 is beoogd zoveel mogelijk validaties van de vastgelegde modellen vanuit de client te kunnen doen om zo het netwerkverkeer te verminderen. Met betrekking tot de ontwikkelwerkplekken kan er daarom het best worden gekozen voor zwaardere clients, met name met meer memory. Een intern geheugen van 64Mb mag best, en een van de jongere generaties Pentium processoren is ook zeker niet verkeerd. Alleen dan is de performance van Designer 2.1 vergelijkbaar met de vorige release.

Opleiding

Met betrekking tot opleiding blijft de eindgebruiker ditmaal buiten schot. Het enige dat merkbaar is voor de eindgebruiker is dat de ontwikkelde schermen nog meer de echte Windows look-and-feel krijgen (met name het gebruik van tabbladen in de gegenereerde schermen).

De ontwikkelaars dienen in ieder geval een upgrade cursus te volgen voor de nieuwe Developer en voor

de nieuwe Designer. Indien het Headstart Template Package wordt gebruikt, is ook een cursus voor de nieuwe versie daarvan op z'n plaats. Zonder cursus aan de slag gaan met release 2.1 kan uiteraard wel maar leidt al snel tot 'ontwikkelen met 2.1 op de 1.3.2 manier'. Zonde.

Consequenties voor onderhoud van reeds ontwikkelde informatiesystemen

Strikt genomen hoeft de migratie van reeds ontwikkelde informatiesystemen voor verder onderhoud met de huidige release niet veel consequenties te hebben. Dat wil zeggen: indien men uitsluitend van de 1.3.2 features gebruik blijft maken, ook al wordt er verder ontwikkeld met de 2.1 release. Hetzelfde geldt voor het gebruik van het Headstart Template Package.

De vorige versie Headstart (3.4.x) kan met een paar kleine aanpassingen worden 'doorgebruikt' qua libraries, templates etc. Dit is de zogenaamde Simple Upgrade. Het onderdeel Utilities is dan niet meer geheel bruikbaar.

De Full Upgrade behelst het aanschaffen van Headstart 5.x (eind oktober 1998 limited production, begin 1999 productie) waardoor de nieuwe features van Designer en Developer 2.1 volledig worden benut. Wel betekent dit het aanzienlijk aanpassen van moduledefinities van reeds ontwikkelde informatiesystemen om hier efficiënt gebruik van te maken.

Het verstandigst is dan ook de Simple Upgrade te kiezen voor onderhoud van reeds ontwikkelde informatiesystemen, en het gebruik van de nieuwe Headstart voor nieuwe ontwikkeling.



Afbeelding 1 Het oude Designer/2000 Launchpad (l) en het nieuwe Oracle Designer Frontpanel (r)

Nieuwe features in Oracle Designer 2.1

De belangrijkste wijzigingen in Designer en Developer zullen kort en bondig worden behandeld. Opgemerkt dient te worden dat, gezien de hoeveelheid nieuwe functionaliteit van Oracle Designer, de opsomming slechts een summier weergave is van de nieuwe eigenschappen.

Oracle Designer 2.1 Front Panel

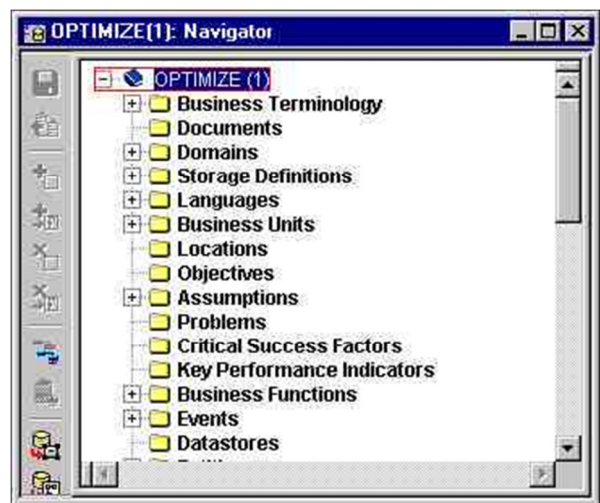
Het Oracle Designer 2.1 Front Panel bevat veel minder iconen dan het Designer/2000 1.3.x launchpad (zie afbeelding 1). Een vijftal

diagrammers en een zestal generators is opgegaan in de Design Editor. Dit is de ontwikkelomgeving voor de gehele 'System Design and Generation' fase van Oracle's CDM. CDM staat hier voor Custom Development Method, de gestructureerde projectaanpak met richtlijnen en standaarden voor het ontwikkelen van maatwerk informatiesystemen.

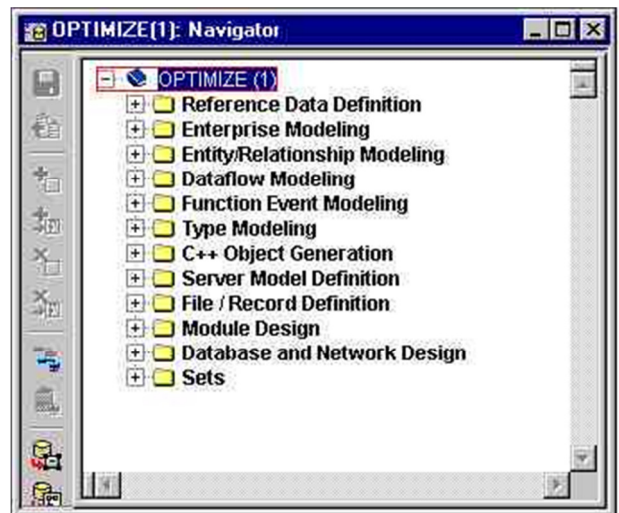


Repository Object Navigator

De navigatortree van de Repository Object Navigator (RON) kan op verschillende manieren worden weergegeven (zie afbeelding 2a t/m 4b):

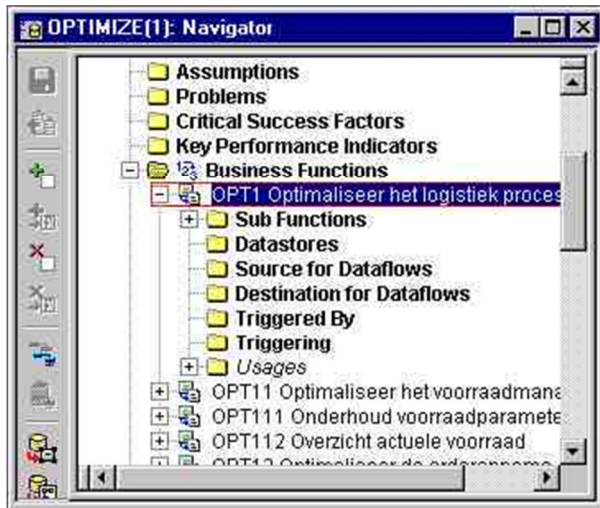


Afbeelding 2a De RON: ungroup view



Afbeelding 2b De RON: group view

· Ungroup view (per objecttype zoals in versie 1.3.2) versus Group view (taak georiënteerd);



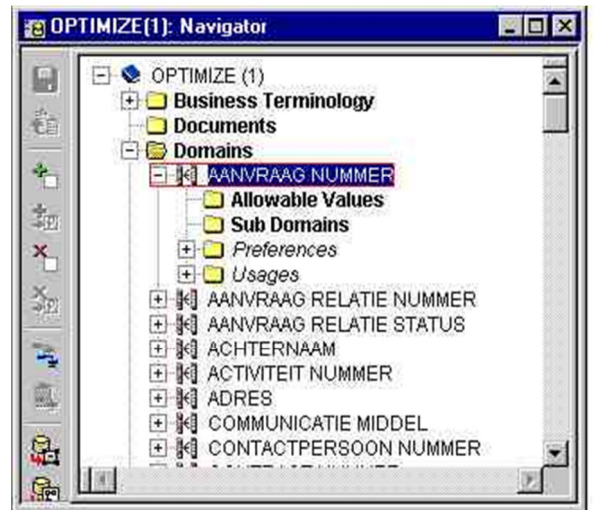
Afbeelding 3a De RON: association view



Afbeelding 4a De RON: split view



Afbeelding 3b De RON: hierarchy view



Afbeelding 4b De RON: remove split view

- Association view (associaties zoals in versie 1.3.2) versus Hierarchy view (associaties hiërarchisch weergegeven). Zo kan bijvoorbeeld de structuur van modules worden weergegeven waardoor de Module Structure Diagrammer overbodig is geworden);

- Split view (elk voorkomen van een geselecteerd objecttype wordt met een eigen tab weergegeven. Zo kan er bijvoorbeeld een tabblad worden weergegeven met voor elke module een tab) versus Remove Split View.

Naast de verschillende weergavemogelijkheden voor de navigatortree is deze zelf ook te customizen. Er kunnen eigen groups (taakgroep met verschillende objecttypen) en views (filters op een specifiek objecttype, zoals modules van het type 'Oracle Forms' met status 'Completed') worden gedefinieerd en geïnclose in de navigator tree.

Het toevoegen van groups leidt tot extra nodes in de tree bij de Group View weergavewijze en het toevoegen van views leidt tot extra nodes in de tree bij zowel de Group View als de Ungroup View weergavewijze.

Voorts is het kopiëren van een object in de RON sterk verbeterd. Het effect van de copy is te beïnvloeden door het wijzigen van de zogenaamde 'copy rules' en de 'expanded copy list'. Daarnaast is ook Force Delete uitgebreid. Dit laat nu vooraf zien wat echt weg wordt gepooid volgens delete rules en expanded delete list (hiervoor geldt hetzelfde principe als voor de copy actie).

Zoals eerder is beschreven bij 'verbeteringen in Designer' zijn de mogelijkheden bij archive en restore sterk uitgebreid met betrekking tot het omgaan met dummy applicatiesystemen. Nog altijd is het zo dat, als een applicatiesysteem zogenaamde gesharede objecten van andere applicatiesystemen bevat, een archive van dat applicatiesysteem (zonder de gesharede

applicatiesystemen expliciet ook mee te nemen) leidt tot het ontstaan van dummy applicatiesystemen. Deze dummy applicatiesystemen bevatten dan de 'losse uiteinden' van de verwijzingen. Met de huidige versie kunnen archives die dummy applicatiesystemen bevatten bij het restoren weer worden 'ingesmolten'. Hierbij worden de verwijzingen naar de dummy applicatiesystemen verlegt naar de gelijknamige objecten in reeds aanwezige applicatiesystemen.

Om foutsituaties aangaande de werking van Designer zo goed mogelijk te kunnen onderzoeken zijn de debuggingfaciliteiten sterk uitgebreid. Via een menuoptie kan een scherm worden getoond waarmee enkele tientallen debugopties in en uit te schakelen zijn. De hiermee te genereren informatie varieert van traceinformatie die inzicht verschaft in de werking van de clientsoftware tot het uitvoeren van SQL-tracing op de server.



Business Process Modeler
Entity Relationship Diagrammer
Dataflow Diagrammer
Function Hierarchy Diagrammer

De uppercase modeling tools zijn vrijwel ongewijzigd. De belangrijkste toevoeging is het Broadcast feature. Dit houdt in dat een stip wordt weergegeven indien een object in een ander tool of door een andere gebruiker is gewijzigd.



Database Transformer

Van de Database Transformer (voorheen Database Design Wizard) zijn alle settings en opties opnieuw ingedeeld en bereikbaar via tabs. Voor minder ervaren gebruikers is er een default modus waarin vrijwel niets ingesteld hoeft en kan worden. Een nieuw feature is de mogelijkheid tot 'Reordering Columns', ofwel het bepalen van de kolomvolgorde op basis van bijvoorbeeld het onderdeel zijn van een unieke sleutel of het al dan niet optioneel zijn. Deze reordering was voorheen een van de Headstart Utilities.



Application Transformer

De Application Transformer (voorheen Application Design Wizard) is vrijwel ongewijzigd. De bug met betrekking tot 'name clashes' (het genereren van modules met namen die reeds bestaan) wordt nu voorkomen met de button 'Find Highest'. Deze button zoekt naar de module met het hoogste volgnummer binnen de opgegeven prefix i.p.v. dat de gegenereerde namen standaard beginnen bij 0010.



Matrix Diagrammer

In de matrix diagrammer zijn veel nieuwe combinaties mogelijk. Veel ervan kunnen bestaande Repository Reports in versie 1.3.2 vervangen. Deze rapporten zijn dan ook verdwenen in 2.1. Een belangrijk aspect is dat nu ook het element 'Application System' als dimensie is op te geven. Op die manier kunnen nu met behulp van de matrix diagrammer shares worden vastgelegd. Tenslotte zijn voor specifieke combinaties van elementtypes nu 3-D matrices mogelijk. De derde as is dan te vinden in de statusregel van de diagrammer.



Repository Reports

Evenals in de RON kan ook hier de navigatortree op verschillende manieren worden weergegeven: Group view versus Hierarchy view versus Report Name view. Daarnaast is ook de navigatortree weer te customizen. Er kunnen eigen groups, reports en parameters worden gedefinieerd en geinclude in de navigatortree. Hierbij kan een eigen report worden gebaseerd op een bestaand Repository Report. Op de achtergrond wordt er dan fysiek een kopie gemaakt van de report definition file (.rdf) van het standaard repository report naar de opgegeven naam. Door dit nieuw ontstane report aan te passen met Developer kun je eenvoudig je eigen rapporten, compleet met de juiste parameterafhandeling vanuit Designer, in de Repository Reports navigatortree opnemen.

In versie 2.1.0 en 2.1.1 kunnen niet alleen de toegevoegde reports maar ook de standaard reports uit de tree worden verwijderd. Met behulp van de General Settings van het Options-menu is op te geven waar de fysieke rapporten moeten staan en waar de reports-output moet komen. Verder is er een aanzienlijk aantal rapporten bijgekomen en is er een aantal verdwenen (zie matrix diagrammer).



Repository Administration Utility

De belangrijkste wijziging in de RAU is de mogelijkheid statistieken te bepalen op de tabellen van de repository. Hierdoor wordt cost-based-optimizing toegepast voor de SQL statements waarmee de Designer clientsoftware de repository benadert.



Design Editor

De genereerbare objecttypen uit de RON zijn in de design editor onderverdeeld in vier aandachtsgebieden, elk terug te vinden onder een

eigen tab. Sommige objecttypen zijn opgeknipt en verspreid terug te vinden onder de verschillende tabs.

De aandachtgebieden zijn:

- Server Model
- Module Applications
- Database Administration
- Distribution en Replication

Het server model bevat de definitie van server objecten exclusief implementatie specifieke zaken zoals sizing. Module applications bevat de definitie van de client modules. Database administration bevat de definitie van implementatiespecifieke zaken zoals rechtenstructuren en sizing. Distribution en Replication spreekt voor zich.

Zoals in de beschrijving van het Front Panel al is beschreven vervangt de Design Editor een vijftal diagrammers en een zestal generators. In dit onderdeel van Designer zijn dan ook de grootste wijzigingen doorgevoerd.

In de volgende uitgave van Optimize zal de Design Editor en het genereerproces uitgebreid worden beschreven.

Gemiste kansen

De grote afwezigheid in Oracle Designer 2.1 is nog steeds de mogelijkheid tot versiebeheer op objectniveau in de repository. Een voorbeeld: als aan een tabeldefinitie in de repository een kolomdefinitie wordt toegevoegd, dan wil je dat deze wijziging leidt tot een hogere versie van de tabeldefinitie. Bij het genereren van een DDL-script (het 'CREATE TABLE'-statement) wil je vervolgens kunnen kiezen welke versie van de tabeldefinitie gegenereerd wordt.

Eerlijk is eerlijk. Er zijn maar weinig softwareproducenten die centraal in hun ontwikkelomgeving een repository hebben en bovendien versiebeheer op objectniveau hebben geïmplementeerd. Het zal technisch gezien dan ook ongetwijfeld zeer ingewikkeld zijn. Maar dat lost het probleem van de repositorygebruiker niet op.

Bij het uitvoeren van projecten van enige omvang met de Oracle Designer omgeving, moet je toch al gauw veel dingen procedureel regelen, om het 'versiebeheerprobleem' in de hand te hebben. Het incrementeel ontwikkelen, waarbij geïntegreerde deelsystemen als afzonderlijke applicatiesystemen parallel of volgordelijk worden ontwikkeld, is met de huidige faciliteiten qua versiebeheer in Designer vragen om moeilijkheden.

Omdat de consultancyafdeling van Oracle Nederland deze problematiek al geruime tijd geleden onderkende, heeft zij hier een aparte dienst voor ontwikkeld: Echo/2000. Als onderdeel van deze dienst wordt een produkt geïnstalleerd dat verschillende tools bevat om versiemangement te doen in de Repository. Zo kent het produkt bijvoorbeeld een module die definitieverschillen kan rapporteren van repositoryobjecten in verschillende versies van applicatiesystemen. Een andere module ondersteunt bij het restoren het insmelten van applicatiesystemen met verwijzingen naar dummy applicatiesystemen. Zoals bij de beschrijving van de RON te lezen was, is de 'insmelt'-functionaliteit inmiddels standaard opgenomen in Designer 2.1. De status van Echo is daarom nogal onduidelijk. Het is te hopen dat Oracle op korte termijn overgaat tot het daadwerkelijk implementeren van fijnmazig versiemangement in de repository.

De voortekenen zijn wat dat betreft positief. Enige tijd geleden heeft Oracle Corporation haar strategie aangekondigd met betrekking tot de repository. Het is de bedoeling dat er in de loop van 1999 een apart produkt op de markt komt, Oracle Repository, dat de opslag moet gaan vormen van metagegevens van zowel Oracle Designer, Oracle Applications, Decision Support Systemen als van Datawarehouse toepassingen. Daarbij zal Oracle Repository de volledige life-cycle ondersteunen van individuele objecten. We zullen zien. Maar mocht dat inderdaad allemaal worden waargemaakt, dan hebben we nog maar heel weinig te wensen over.

Kijk voor meer publicaties op

<http://www.anewlink.nl/ict/nl/publicaties/>.

(c) Copyright 1999 A New Link bv