

Ontwikkelingshulp voor PL/SQL (4)

De stille kracht van Java

In een serie van vier artikelen zal Lucas Jellema, technisch consultant bij AMIS, een overzicht geven van ideeën, patronen, richtlijnen en tools die PL/SQL ontwikkelaars kunnen overnemen uit de wereld van Java/J2EE. Hij gaat onder meer in op concepten uit de Java programmeertaal die een toepassing hebben in PL/SQL, maar ook op open source-tools voor onder meer logging, unit testen, automatiseren van batch-operaties en het genereren van documentatie. Let wel: dit artikel is bestemd voor PL/SQL ontwikkelaars; kennis van Java is niet vereist.

We zullen in dit artikel kennismaken met het door Oracle als 'vlaggenschip voor database design en development' naar voren geschoven tool JDeveloper. We kijken naar de betekenis van JDeveloper als ontwikkelgereedschap voor SQL en PL/SQL development. Ook kijken we naar enkele open source alternatieven voor ondermeer TOAD en PL/SQL Developer.

Database browsers

Database browsers zijn tools waarmee een ontwikkelaar of DBA in een GUI (Grafische User Interface) kunnen bladeren door de objecten in een database. Er zijn veel van dit soort database browsers beschikbaar - meestal met links een tree-navigator met per object-type een node en rechts een property pallet met de eigenschappen van het momenteel geselecteerde object, al dan niet editable. Bekende commerciële aanbiedingen zijn TOAD (Tool for Oracle Application Developers) en PL/SQL Developer. Oracle zelf biedt dit soort browsers binnen ondermeer Oracle Warehouse Builder en Oracle Enterprise Manager. In het open source domein is ook een flink aantal Oracle database browsers beschikbaar. Zie hiervoor bijvoorbeeld OracleEditor.php (<http://tim.digicol.de/oraedit/>), SchemaMap (<http://sourceforge.net/projects/schemamap/>), Oracle Explorer (<http://sourceforge.net/projects/oraexp/>), JDBExplorer (<http://www.davemanton.com/>) en Oracle PL/SQL Code Editor (<http://sourceforge.net/projects/oracle/>). Twee tools die er enigszins bovenuit lijken te steken zijn Tora en SquirrelL.

Tora Toolkit for Oracle

Tora is ontwikkeld voor Linux als tegenhanger voor de vele tools die bestonden voor Windows. Inmiddels is Tora geadopteerd door Quest Software, de producent van TOAD, die het onder de GPL licentie als open source beschikbaar stelt het op Linux en onder een commerciële licentie op Windows. Tora biedt een database explorer (of browser) inclusief data browser en editor, een SQL Worksheet en een PL/SQL Editor en Debugger. Daarnaast bevat Tora een vrij omvangrijke verzameling DBA hulpmiddelen. Tora lijkt veel op TOAD - maar is op een Linux systeem gratis.

Squirrel SQL

Een andere Oracle database browser uit het open source domein is SquirrelL. SquirrelL is een stand-alone Java tool dat op diverse platforms -Windows, Macintosh and X Window System (X11) - beschikbaar is. SquirrelL biedt een database browser voor iedere willekeurige database die JDBC ondersteunt, waaronder Oracle, MySQL, DB2, SQL Server etc. Naast de browser bevat SquirrelL ook een SQL Worksheet. Ook is er ondersteuning voor het exporteren van data.



Figuur 1. Tora biedt een explorer inclusief data browser en editor, een SQL Worksheet en een PL/SQL Editor en Debugger

Adv. Quest

Tora karakteristieken

Homepage	http://www.globecom.net/tora/ en
Bestaat sinds	December 2000
Meeste recente release	1.2.4 (Juni 2002), 1.3.14 (Juli 2004 - Development Release)
Status	Productie
Onderliggende technologie	C++, PL/SQL
Ontwikkelteam	3 + Quest Software
Zip-grootte	8,2 MB
Licentie	GPL (General Public License) (zie ook : http://www.opensource.org/licenses/gpl-license.php) behalve op Windows: daar moet een commerciële licentie voor verkregen worden.

Voor een aantal databases zoals Oracle en MySQL zijn speciale plug-ins beschikbaar die specifieke functionaliteit voor een bepaalde database toevoegen aan de basis-set in SquirrelL. Daarnaast zijn er plugins die extra functionaliteit bieden als een geavanceerde text-editor, een code bibliotheek, een DDL (Create Table en Insert) generator en een SQL Validator. Zeker voor organisaties die verschillende databases door elkaar gebruiken is SquirrelL een aardig tool, aangezien hetzelfde tool voor die databases gebruikt kan worden.

Oracle JDeveloper

Een tool waar je wellicht niet direct aan zou denken bij data-

SquirrelL SQL karakteristieken

Homepage	http://squirrel-sql.sourceforge.net/
Bestaat sinds	Mei 2001
Meeste recente release	0.32 (Augustus 2004), 1.2 Beta (Juni 2004)
Status	Productie
Onderliggende technologie	Java (Swing), JDBC
Ontwikkelteam	25
Zip-grootte	1,9 MB
Licentie	GPL of LGPL(GNU (Lesser) General Public License) (zie ook: http://www.opensource.org/licenses/gpl-license.php)
Gerelateerde technologie en concepten	JFaceDbc (Commerciële licentie) Eclipse plugin op basis van SquirrelL (http://www.pratocity.com/index.jsp?mod=/jface/jfacedbc.jsp)



Figuur 2. SquirrelL biedt een database browser voor iedere willekeurige database die JDBC ondersteunt

base browsing en PL/SQL Development is Oracle JDeveloper. Toch is JDeveloper ook voor Database Designers en PL/SQL Developers zeer de moeite waard. Niet in de laatste plaats omdat Oracle heeft aangekondigd JDeveloper verder te gaan optuigen als volwaardig PL/SQL ontwikkelplatform tot het als concurrent van tools als TOAD en PL/SQL Developer gepositioneerd kan worden. Daarbij zal JDeveloper ook Database Design en Generatie functionaliteit gaan bieden die te vergelijken is met Oracle Designer.

JDeveloper is geen open source en is ook niet gratis. Het maakt onderdeel uit van de 9i of 10g Developer Suite, waar ook de andere Oracle Development Tools zoals Oracle Designer en Oracle Forms in zitten. Ook wordt JDeveloper geleverd als onderdeel van de 9iAS of 10g Application Server. Er is een goede kans dat een organisatie via een van beide licenties ook al beschikt over JDeveloper. In elk geval is JDeveloper voor evaluatie doeleinden te downloaden van OTN. Installatie van JDeveloper omvat niet meer dan downloaden en unzippen van een - betrekkelijk grote - zip-file. (NB: voor dit artikel is gebruik gemaakt van de 10.1.3 Preview release van december 2004).

Schema Browser

Als JDeveloper is geïnstalleerd kan je naar de Connections Navigator navigeren. Daar kan je allereerst Connecties definiëren naar user accounts in iedere database waarvoor een JDBC driver beschikbaar is. Een connectie kan vervolgens gebrowsewd worden, zie bijvoorbeeld de LocalORCL connectie in onderstaande figuur. JDeveloper toont nodes voor de schema's waarin zich objecten bevinden waarop het user account rechten heeft, daaronder nodes voor alle object types en onder die nodes de daadwerkelijke database objecten. Het rechterdeel van het scherm toont de eigenschappen van het geselecteerde object. Voor tabellen kunnen ook de data in de tabel worden getoond; deze kunnen niet worden gewijzigd.

Oracle JDeveloper karakteristieken

Homepage	http://www.oracle.com/technology/products/jdev/index.html
Bestaat sinds	1998
Meeste recente release	10.1.2 (januari 2005) en 10.1.3 Beta (december 2004)
Status	Productie
Onderliggende technologie	Java (Swing), JDBC, SVG, XML, XSLT
Ontwikkeltteam	Tientallen
Zip-grootte	180 MB
Licentie	Via Oracle Developer Suite of Application Server

Voor het geselecteerde object kan een Wizard worden gestart - via de Edit optie in het rechtermuisknop menu. In deze wizard kunnen de relevante eigenschappen en kinderen van het object worden bekeken en gewijzigd. Voor een tabel gaat het dan om ondermeer de kolommen, indexen en de constraints.

De browser bevat ook een SQL Worksheet. Hierin kunnen SQL statements worden uitgevoerd. Dit worksheet houdt een historie bij van alle eerder uitgevoerde SQL statements. Ook biedt het de mogelijkheid om van een SQL statement een Explain Plan uit te voeren. Op dit moment is dit worksheet tamelijk rudimentair en beter te vergelijken met SQL*Plus dan met TOAD. Er zijn geen mogelijkheden om de resultaten van een statement te sorteren, te filteren of te exporteren.

PL/SQL Development

Als het geselecteerde object een PL/SQL Function, Procedure of Package is, toont JDeveloper een PL/SQL Editor waar direct de PL/SQL code kan worden geëdit, gecompileerd, gerund en gedebugged.



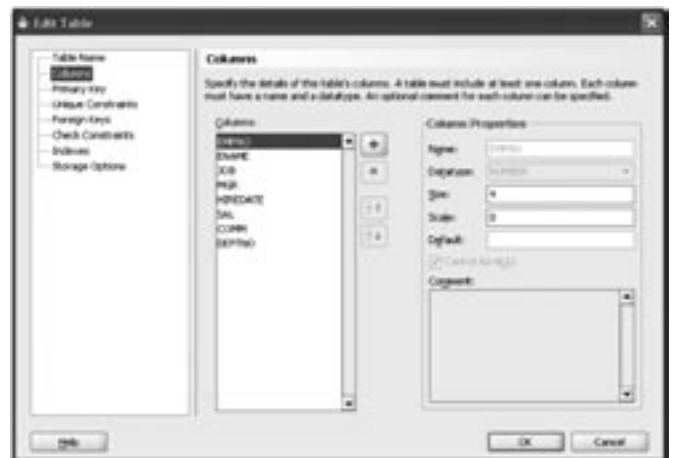
Figuur 3. In JDeveloper kunnen connecties gedefiniëerd worden naar user accounts in iedere database waarvoor een JDBC driver beschikbaar is.

De PL/SQL Code editor bevat een aantal features, zoals Code Insight: type het begin van een database object zoals een tabel naam of een package en gebruik CTRL+Space; JDeveloper toont nu een poplist van alle beschikbare objecten die beginnen met de reeds ingetypte letters. Hiermee kunnen snel en zonder fouten de namen van gerefereerde database objecten in code worden opgenomen. Jammer genoeg biedt de JDeveloper PL/SQL Editor geen navigatie naar de gerefereerde objecten. Ook is er geen ondersteuning bij het opvoeren van de parameters in een aanroep van een PL/SQL routine.

De compiler geeft een duidelijk overzicht van de gevonden fouten. Dubbelklikken op een fout brengt je direct naar de betreffende regel in de code. Uiteraard wordt door compilatie ook direct de gewijzigde PL/SQL source code geladen in de database. Wat mij betreft is PL/SQL editing met JDeveloper aanzienlijk beter dan met SQL*Plus maar bepaald nog niet van het niveau van PL/SQL Developer en TOAD.

PL/SQL Debugging

Debugging PL/SQL code wordt door JDeveloper bijzonder goed ondersteund. In de PL/SQL code kunnen in JDeveloper breakpoints worden gedefiniëerd. Dit zijn locaties in de code waarop de uitvoering van de code wordt gestopt. Als de code wordt gerund in debug-mode zal zodra een breakpoint bereikt wordt de executie worden onderbroken. JDeveloper toont dan het betreffende breakpoint en de bijbehorende code alsook de callstack. Aan de callstack kan je zien welke PL/SQL programma eenheden zijn aangeroepen alvorens de code die het breakpoint bevat werd aangeroepen. De ontwikkelaar kan er dan voor kiezen om: stap voor stap door de verdere code heen te lopen of de normale uitvoering (tot aan het volgende breakpoint) vervolgen. De ontwikkelaar kan ook de waarden van alle variabelen en parameters inspecteren en zelfs wijzigen. Dit betekent wat mij betreft het eind van de dbms_output.put_line



Figuur 4. Voor een geselecteerd object kunnen de relevante eigenschappen en kinderen van het object worden bekeken en gewijzigd

aanroepen in packages om een soort tracing en logging te implementeren!

Breakpoints kunnen op slimme manieren worden gedefinieerd. Zo kan je ze conditioneel maken. Dat wil zeggen dat de code executie alleen onderbroken wordt als aan een bepaalde voorwaarde wordt voldaan. Bijvoorbeeld als een FOR-LOOP door 10.000 iteraties heengaat zou je een breakpoint kunnen definiëren dat alleen na 5.000 en 10.000 iteraties actief is als echt breakpoint. Ook kun je een breakpoint koppelen aan een bepaald type Exception; iedere keer als een PL/SQL Exception van dat type - bijvoorbeeld NO_DATA_FOUND of NUMERIC OR VALUE ERROR - optreedt gaat het breakpoint af en krijgt JDeveloper controle over de PL/SQL executie.

Het meest interessante deel van de debug ondersteuning in JDeveloper betreft Remote Debugging. Bij Remote Debugging wordt PL/SQL code uitgevoerd, ergens in een database. In die database is aangegeven, ergens gedurende de sessie die de PL/SQL code uitvoert, dat er door de database contact moet worden gezocht met een debugger proces. Dit doe je door in de PL/SQL code - bijvoorbeeld de ON-LOGON trigger voor de betreffende sessie - de volgende aanroep uit te voeren: `DBMS_DEBUG_JDWP.CONNECT_TCP('<IP ADRES van de machine waarop de debugger (JDeveloper) draait>', '4000')` Hierbij is 4000 het poortnummer waarop de debugger luistert naar de remote database. De database server legt nu contact met de JDeveloper PL/SQL Debugger en krijgt te horen op welke punten in de PL/SQL code breakpoints zijn geplaatst. Als de database server tijdens de uitvoering van PL/SQL code op

een breakpoint stuit, wordt tijdelijk gestopt met verdere uitvoering en krijgt het debug-proces in JDeveloper de controle. De database wacht met verdere executie tot JDeveloper daar een indicatie voor geeft.

Met Remote Debugging kan je een willekeurige PL/SQL applicatie die waar ook wordt uitgevoerd gaan debuggen. De voorwaarden waar je aan moet voldoen: voor volledige functionaliteit tenminste een 9iR2 database waarin alle PL/SQL code is gecompileerd met debug informatie (alter package <package naam> compile debug body). Het database account van waaruit de debug sessie wordt gestart heeft de privileges EXECUTE ON DBMS_DEBUG_JDWP en DEBUG ANY PROCEDURE en DEBUG CONNECT SESSION nodig.

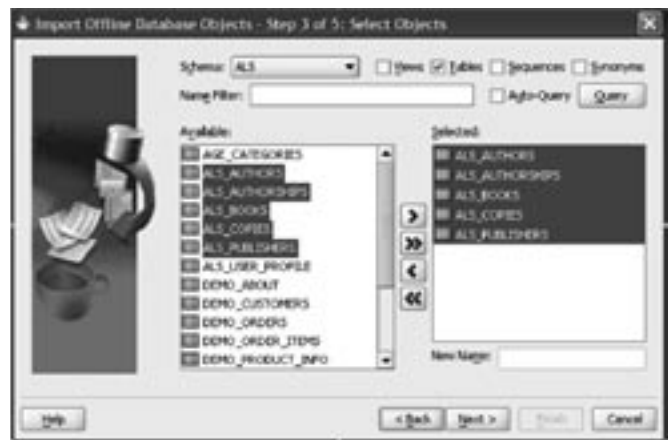
Database modellering

JDeveloper biedt de mogelijkheid om ook off-line met database objecten te werken. Dat wil zeggen dat de objecten waaraan je aan het sleutelen bent, niet rechtstreeks in de database gewijzigd worden. Je zou dit ook een meta-niveau of een ontwerp-fase kunnen noemen. In deze offline toestand kan je diagrammen tekenen die sterk lijken op de Server Diagrammen van Oracle Designer. Je kunt gebruik maken van een soort Design Capture tool - de Import Offline Database Objects wizard - om de definities van bestaande database objecten in het JDeveloper project te importeren. Vervolgens kan je die objecten gaan bewerken, bijvoorbeeld in een diagram.

Offline Objecten kunnen naar een database diagram worden geslept. Nieuwe objecten kunnen rechtstreeks in het diagram worden gecreëerd door de betreffende icoontjes uit het Components Palette te selecteren. Er zijn allerlei lay-out opties beschikbaar om het diagram visueel te verbeteren: kleuren, lijnstijl, Straighten Lines, Font instellingen, Layout Shapes (om de objecten op het diagram volgens vijf verschillende algoritmes te



Figuur 5. Het meest interessante deel van de debug ondersteuning in JDeveloper betreft Remote Debugging



Figuur 6. Met een wizard kunnen de definities van bestaande database objecten in het JDeveloper project geïmporteerd worden



Figuur 7. Een diagram kan gepubliceerd worden als PNG- of JPEG-plaatje of als SVG-document

positioneren) en View as Compact (met een simpel icoon per object).

Een diagram kan gepubliceerd worden als PNG of JPEG plaatje of als SVG (Scalable Vector Graphics) document. Deze formaten kunnen allemaal eenvoudig in websites worden opgenomen. Daarmee zijn deze diagrammen een stuk toegankelijker dan de diagrammen in Oracle Designer.

Offline database objecten kunnen worden ge-edit in Wizards, dezelfde als die voor online objecten worden gebruikt. Vervolgens kunnen de objecten worden gegenereerd, dat wil zeggen: kunnen er DDL scripts voor geselecteerde objecten worden gegenereerd. Objecten kunnen ook worden 'gereconciled' met een database schema; in dat geval vergelijkt JDeveloper de object definitie met het vergelijkbare object in het opgegeven database schema en genereert de Alter statements die nodig zijn om het database object gelijk te maken aan de JDeveloper definitie. Met een simpele Refresh from Database kan de off-line definitie gelijk worden gemaakt aan de database definitie. Overigens kun je er ook voor kiezen om niet DDL scripts te genereren maar de DDL direct in een aangegeven database schema uit te voeren.

Vergelijking met Oracle Designer

Een belangrijk verschil met Oracle Designer is het gebruik van een Repository in een Database. Oracle Designer legt alle diagrammen en object definities vast in een database. Daarmee zijn alle meta-data direct na het vastleggen van wijzigingen voor alle ontwikkelaars toegankelijk. JDeveloper daarentegen slaat alle metadata op in (XML-)files, die over het algemeen op het lokale filesystem staan. Gebruik van een Source Code Beheer systeem - zoals CVS, ClearCase of eventueel Oracle SCM - is zeer sterk aanbevolen, wanneer meerdere ontwikkelaars gezamenlijk in een project werken. Het SCM systeem vormt dan de repository waarin gewijzigde of nieuwe files moeten worden

ingecheckt en waar vandaan lokale filesystemen regelmatig moeten worden ververst.

Al met al is de Database Modelling in JDeveloper 10g al vrijwel op het niveau van de Design Editor en Server Generator van Oracle Designer, tenminste voor gewone tabellen en views. Er is geen ondersteuning voor zaken als TAPI packages en minder frequent gebruikt object types als Materialized Views en Queues.

Overige faciliteiten

- Publiceren PL/SQL Webservices - JDeveloper ondersteunt de publicatie van PL/SQL Packages als webservices. De ontwikkelaar selecteert binnen de PL/SQL webservice wizard - te starten vanuit het rechtermuisknop menu op de packages in de database browser - de procedures en functies die binnen de webservice ondersteunt moeten zijn. JDeveloper genereert vervolgens - met behulp van JPublisher - de noodzakelijke Java Classes en Webservice Definition File (wsdl). De wizard genereert ook een Deployment Profile waarmee de webservice direct gedeployed kan worden op OC4J.
- Laden van Java Classes, verpakt in PL/SQL: binnen JDeveloper kunnen classes worden geselecteerd binnen een Load Java en Java Stored Procedures deployment profile. Deze classes kunnen eenvoudig geladen worden in een database schema. Ook kan JDeveloper een PL/SQL package genereren met Functions en Procedures gebaseerd op de (static) methodes van de class. Daarmee komt de functionaliteit van de betreffende classes beschikbaar binnen PL/SQL. Deze functionaliteit kan bijvoorbeeld worden ingezet bij de consumptie van WebServices vanuit PL/SQL code of de benadering van het file systeem vanuit de database.
- Oracle BPEL (Business Process Execution Language) Plugin voor JDeveloper - deze komt in de loop van 2005 beschikbaar en zal naar verwachting 'short-cuts' voor PL/SQL gaan ondersteunen waarbij PL/SQL procedures kunnen worden aangeroepen vanuit BPEL processen zonder de overhead van complete WebServices.

Resources

- "Designing, Modelling And Generating Database Tables": http://otn.oracle.com/obe/obe9051jdev/dataModeling/lesson_dm.htm
- "Developing SQL and PL/SQL with JDeveloper": <http://otn.oracle.com/obe/obe9051jdev/PISqlOBE/OBEplsql.htm>
- Publish PL/SQL as Webservice <http://www.oracle.com/technology/tech/webservices/htdocs/series/plsql/index.html>
- Calling a Web Service From The Database http://www.oracle.com/technology/products/jdev/howtos/10g/WSDBCallout/DBCAlloutWS_HowTo.htm
- JPublisher User's Guide: http://download-west.oracle.com/docs/cd/B14117_01/java.101/b10983/toc.htm

Conclusies

Uit bovenstaande blijkt dat Oracle langzamerhand van JDeveloper een serieus stuk gereedschap maakt voor PL/SQL ontwikkelaars. Weliswaar blijft JDeveloper nu nog op diverse punten achter bij gevestigde tools als TOAD en PL/SQL Developer, toch zijn er al gebieden waar JDeveloper die twee voorbij steekt - met name bij de integratie van Java- en database-technologie en bij het ontwerp en de generatie van databases. Gezien Oracle's ambitie met JDeveloper lijkt de verwachting gerechtvaardigd dat we de komende tijd nog meer gaan zien aan PL/SQL en SQL functionaliteit in JDeveloper. Naast JDeveloper blijkt het open source domein ook een flinke serie database browsers te bieden. Hieronder bevinden zich

diverse tools die zeker de moeite van het overwegen en evalueren waard zijn; bijvoorbeeld SquirrelL en TOra (met name op Linux).

Lucas Jellema (jellema@amis.nl) is sinds 2002 werkzaam bij AMIS Service in Nieuwegein, als Expertise Manager Technologie en Technisch Consultant. Daarvoor werkte hij ruim acht jaar bij Oracle, ondermeer binnen het iDevelopment Center of Excellence. Hij houdt zich onder meer bezig met Java, XML/XSLT en andere webtechnologie als ook de Oracle database en tools voor applicatie ontwikkeling. Lucas verzorgt presentaties, workshops en schrijft regelmatig artikelen in ondermeer Optimize en het AMIS Technology Weblog (<http://technology.amis.nl/blog>).

N I E U W S

SPSS workbench ondersteunt predictive modeling met Oracle Data Mining

De op open standaarden gebaseerde architectuur van Clementine 9.0, de nieuwste versie van de data mining workbench van SPSS, vormt vanaf nu de basis voor de integratie met Oracle Database 10g, de eerste relationele database ontworpen voor grid computing. Oracle Database 10g ondersteunt predictive modeling met Oracle Data Mining, zodat modellen direct binnen de database kunnen worden gebouwd en toegepast. Met Clementine 9.0 kunnen analisten de Clementine interface gebruiken om modellen te bouwen, te doorzoeken en te scoren met Oracle Database 10g, met behulp van Oracle Data Mining technieken. Oracle Data Mining algoritmen, zoals Naïve Bayes, Adaptive Bayes Network en Support Vector Machines, verschijnen als nodes in de Clementine interface. Deze technieken kunnen nu net als iedere andere techniek die ingebouwd is in Clementine gebruikt worden.

“Oracle Database 10g scoort goed op beheersbaarheid, hoge efficiency en productiviteit,” aldus Peter Caron, SPSS senior product manager. “Oracle Database 10g en Oracle Data Mining hebben die-

zelfde drie kwaliteiten met Clementine gemeen. Analisten kunnen daardoor efficiënter en doeltreffender werken met modellen die ze via Oracle 10g verkrijgen en kunnen ze eenvoudiger bouwen, gebruiken en beheren.”

Verdere integratie SAP NetWeaver en Microsoft .NET

SAP en Microsoft verdiepen hun samenwerking met als doel om SAP-gebruikers meer rendement te laten halen uit SAP-oplossingen binnen een Windows-omgeving. Hiervoor kunnen zij gebruikmaken van de Portal Development Kit for Microsoft .NET (PDK for .NET), die nu binnen SAP NetWeaver beschikbaar is. Met de realisatie van PDK for .NET ontstaat verdere integratie tussen SAP NetWeaver en de ontwikkelomgeving Microsoft Visual Studio .NET 2003. Deze ontwikkelomgeving wordt ingezet voor het bouwen en integreren van op XML gebaseerde webdiensten en -applicaties. De totstandkoming van de Portal Development Kit for Microsoft .NET is het directe resultaat van deelname van SAP aan het Microsoft Visual Studio Industrie Partner (VSIP) programma. Microsoft richt zich daarmee nadrukkelijk

tot de 40.000 gebruikers van SAP-oplossingen die met het Windows-platform werken.

Met behulp van PDK for .NET kunnen SAP-gebruikers hun SAP Enterprise Portal - de portal-component van SAP NetWeaver - nog beter inzetten voor de integratie van hun bedrijfsprocessen. Hierbij kunnen zij verschillende ontwikkeltalen gebruiken, waaronder Microsoft Visual C# ('C-sharp') en Microsoft Visual Basic .NET.

De oplossing richt zich vooral op interactieve processen, waarbij PDK .NET de mogelijkheden van het Microsoft .NET Framework optimaal benut. Hiermee kunnen veelgebruikte portal-functionaliteiten worden verbeterd. Hierbij valt te denken aan gebruikersbeheer, SAP Single Sign On (SAP SSO) en toegang tot gespecialiseerde content. Voor de bouw van de gebruikersinterface kunnen ontwikkelaars de standaard Microsoft .NET besturing en de interfacebesturing van SAP NetWeaver door elkaar heen gebruiken. Content op basis van .NET kan eenvoudig worden gebruikt met behulp van SAP NetWeaver en Portal Runtime for .NET, waarmee de .NET-code wordt aangeemaakt en de resultaten worden geïntegreerd in SAP Enterprise Portal.