

Was man zu Oracle 11g R2 und .NET wissen sollte

Claus Jandausch, ORACLE Deutschland B.V. & Co. KG

Dieser Artikel zeigt, was neben dem reinen Datenbankzugriff mit Oracle 11g R2 und .NET noch alles möglich ist.

In den Anfangszeiten von Microsoft .NET war die im Oracle-Umfeld vorherrschende Frage meist ein ungläubiges „Geht das überhaupt?“. Heute weiß praktisch jeder, dass es geht – nur nicht, wie weit die Kombination aus Oracle-Technologie und .NET reicht. Und immer noch schwelt dabei im Hintergrund der auch unausgesprochene Vergleich zum Microsoft SQL Server. Kann es denn wirklich sein, dass man sich als .NET-Entwickler mit der Oracle-Datenbank keine gravierenden Nachteile gegenüber einem Microsoft SQL Server einhandelt? Hat Microsoft nicht erst im vergangenen Jahr das „Aus“ für die Weiterentwicklung des hauseigenen Microsoft „.NET Data Provider für Oracle“ erklärt? Soll man es wirklich glauben, wenn in einem Vergleich zwischen der Oracle-Datenbank und dem SQL Server sogar von zahlreichen Vorteilen für .NET-Anwendungen gesprochen wird? Die Antwort auf all diese Fragen lautet schlicht „Ja“. Da aber der Stachel einer früheren Verunsicherung meist tief sitzt, genügen schon kleine Hürden, um alte Zweifel wieder erneut auf die Bühne zu rufen.

Um vorab gleich eine der eingangsgestellten Fragen zu beantworten: „Ja, Microsoft hat tatsächlich vor einem Jahr erklärt, in Zukunft auf einen eigenen Oracle Data Provider für das .NET Framework zu verzichten.“ Doch was im ersten Moment nach einem Bruch in der .NET-Strategie beider Hersteller klingt und negative Folgen für .NET-Anwendungen auf Basis von Oracle befürchten ließ, ist jetzt als eine gute Nachricht zu sehen. „Endlich!“, war demnach auch die Reaktion auf diese Ankündigung im Oracle-Produkt-Management und -Development in den USA. Endlich muss keine Überzeugungsarbeit mehr geleistet werden, um Kunden zum Einsatz des „Oracle Data

Providers for .NET“ (ODP.NET) zu bewegen. Endlich hat sich die Frage von allein geklärt, welcher Data-Provider zu bevorzugen ist – und die bessere Technologie darstellt.

Fragen hierzu gab es in der Vergangenheit nämlich zuhauf. In einer gemeinsam mit Microsoft durchgeführten Veranstaltungsreihe (Oracle Developer Days) und einem Vortrag auf der DOAG 2009 Konferenz war zu beobachten, dass genau dieses scheinbar simple Detail im Brennpunkt der Kunden und Partner stand. „Welchen Provider soll man denn jetzt einsetzen?“ lautete die immer wiederkehrende Frage. Als selbige dann von einem Microsoft-Spezialisten beantwortet wurde, war die Verunsicherung am Ende schnell durch Vertrauen ersetzt. „Niemand kennt die eigene Datenbank-Technologie besser als Oracle selbst“, lautete eine der Kernaussagen.

Oracle nutzt offene und standardisierte Schnittstellen, um den nativen Zugriff von .NET-Anwendungen auf Oracle-Datenbanken zu realisieren. Es sind übrigens dieselben, die auch der Microsoft SQL Server für den Datenzugriff verwendet. Die vom Oracle-eigenen Data-Provider (ODP.NET) bereitgestellten Möglichkeiten übertreffen jedoch den Funktions- und Leistungsumfang des von Microsoft bisher gelieferten Data-Providers um Längen. Es ist Microsoft ja verständlicherweise auch gar nicht möglich, die aus ihrer Sicht teils proprietären Features und Functions der Oracle-Datenbank adäquat abzubilden. Nachdem man sich im Microsoft-Development den „Oracle Data Provider for .NET“ genauer angesehen und mit der eigenen Entwicklung verglichen hatte, war das Aus für die Weiterentwicklung deshalb eine logische Konsequenz und im Sinne der Kun-

den. „Setzen Sie den Oracle Data Provider for .NET ein“, war deshalb auch die eindeutige Antwort auf die Frage nach der Wahl des Data-Providers. Auch unabhängige Spezialisten und MVPs haben diese Empfehlung gegeben. Ein Artikel in den DOAG News Q1/2010 beschreibt das genaue Vorgehen beim Umstellen der Data-Providers („Oracle und .NET – Die neue ODAC-Version“).

Verwirrungen um ODAC und andere Begrifflichkeiten

Nicht selten ist im direkten Gespräch mit Anwendern eine regelrechte Verwirrung festzustellen, wenn es im Zusammenhang mit Oracle und .NET um Begriffe wie „ODAC“ oder „Instant Client“ geht. Der Autor musste vor Kurzem sogar eine richtige Verunsicherung in diesem Zusammenhang aufklären.

„Was bedeutet eigentlich ODAC genau?“, „Was ist der Unterschied zu dem hier enthaltenen Instant Client und dem Full Client?“, „Was hat es mit der XCopy-Version auf sich?“, „Wie müssen wir uns die Versionsplanung vorstellen?“, „Gibt es zu allen Varianten parallele Patches und gegebenenfalls Bug-Fixes?“ und „Wir sind verunsichert – welche Client-Version soll nun im Rollout verwendet werden, nachdem wir Diskrepanzen festgestellt haben?“ lauteten die Fragen eines Kunden, der seit Jahren anspruchsvolle .NET-Entwicklungen auf Basis von Oracle betreibt. Es handelte sich also keineswegs um einen Neustarter, der erstmalig mit Oracle-Produkten konfrontiert wurde. An der Fülle der Fragen lässt sich der Grad der Verunsicherung leicht ablesen. Die Ursachen hierfür sind einerseits verständlich, andererseits aber auch schnell aus dem Weg geräumt.

ODAC – Oracle Data Access Components

Oracle Data Access Components (ODAC) orientieren sich in ihrer Begrifflichkeit an den im Microsoft-Umfeld bekannten MDAC – den sogenannten „Microsoft Data Access Components“. Beides stellt nicht mehr und nicht weniger als eine Sammlung unterschiedlicher Software-Komponenten für den Datenzugriff dar. Die MDAC enthalten beispielsweise ODBC, OLE DB, ADO und RDS. Entgegen der landläufigen Meinung ist nämlich ODBC – um es am bekanntesten Vertreter deutlich zu machen – kein inhärenter Bestandteil des Windows-Betriebssystems, sondern ein Teil dieser Komponentensammlung. Spitzfindig – aber wichtig für das Verständnis. In gleicher Form ist auch ODAC zu verstehen und es muss deshalb ganz folgerichtig als Paket installiert werden, um die hierin enthaltenen Komponenten für den Datenzugriff unter .NET nutzen zu können. Es sind insgesamt vier:

- Oracle Developer Tools for Visual Studio
- Oracle Data Provider for .NET
- Oracle Providers for ASP.NET
- .NET Stored Procedures

Das Einsatzgebiet dieser Komponenten ergibt sich bereits aus dem Namen und bedarf eigentlich keiner weiteren Erklärung. Sowohl die Implementierung als auch die besonderen Vorteile wurden ebenfalls im zuvor schon erwähnten Artikel in den DOAG News Q1/2010 auf technischer Ebene ausführlich dargestellt. Der hier dargestellte Mehrwert gilt uneingeschränkt auch für die Komponenten der aktuellen ODAC Version 11.2.0.1.2.

Obwohl an dieser Stelle nur von insgesamt vier Komponenten die Rede ist, wird man eine ganze Reihe weiterer Bestandteile finden, wenn es konkret an den Download geht. Diese haben jedoch keine unmittelbare Relevanz für reine .NET-Entwicklung und sind auch nicht erklärungsbedürftig, wie beispielsweise der Oracle ODBC Driver oder Oracle SQL*Plus.

Ganz anders sieht es dagegen mit dem ebenfalls enthaltenen „Oracle

Instant Client“ und seiner Abgrenzung gegenüber dem an anderer Stelle ebenso zum Download angebotenen „Oracle Client“ aus. Hinzu kommen noch zwei in Umfang und Größe unterschiedliche ODAC-Versionen, von denen sich eine als „XCOPY-Version“ bezeichnet. Welcher Download ist nun der Richtige? Welcher ist besser geeignet für die .NET-Entwicklung? Welche ODAC-Version soll man wählen?

All dies führte in Summe zu der eingangs bereits erwähnten Verunsicherung. Aber sind das überhaupt die richtigen Fragen? Aus diesem Grund hier noch einmal die Unterscheidung der auf dem Oracle Technology Network (OTN) bereitgestellten Downloads.

Oracle „Full“ Client

Der klassische Oracle Client – in seiner aktuellen Version offiziell als „Oracle Database 11g Release 2 Client“ bezeichnet – wird manchmal auch „Full Client“ genannt. Im erwähnten Kunden-Szenario war dies beispielsweise der Fall. Mag sein, dass sich diese Bezeichnung von der Größe des Downloads ableitet, der immerhin weit über 680 MB umfasst, oder den Umfang der angebotenen Tools zum Ausdruck bringen soll. Dieser geht weit über das hinaus, was für einen .NET-Entwickler von Interesse ist. Paradoxaer Weise wird er dagegen sogar für ihn wichtige Komponenten vermissen – wie die Oracle Developer Tools for Visual Studio (ODT.NET).

Ganz anders sieht die Welt dagegen aus Sicht eines Oracle DBAs aus, der den hier enthaltenen „Net Configuration Assistant“ ebenso benötigt wie den „Net Manager“ oder der sich gerne auch des „Oracle Snap-In für die Microsoft Management Console“ bedient, um Oracle-Anwender unter Windows damit schnell und bequem für den Zugriff auf Oracle-Datenbanken per Betriebssystem-Authentifizierung zu berechtigen. Dieser Full Client umfasst außerdem die „Oracle Performance Counter“ für den Windows Performance Monitor und einen Wallet Manager. Mit diesem Client lässt sich auch die Integration mit dem Active Directory zur Auflösung von „Oracle-Net-Service-Namen“ konfigurieren.

Kurzum – das Angebot des Oracle „Full“ Clients übertrifft bei Weitem das, was für einen reinen Zugriff von Oracle auf .NET benötigt wird.

Oracle Instant Client

Der im ODAC-Paket enthaltene „Instant Client“ kommt ohne Registry-Einträge oder ORACLE_HOME-Einstellungen aus und kann auf einfachste Art und Weise aktualisiert werden. Wichtig zu wissen ist, dass er eine freie Komponente darstellt und sowohl Entwicklung als auch Verteilung einer ODP.NET-Applikation deshalb kostenlos sind.

ODAC-Nutzer sind im Allgemeinen Windows-Entwickler (.NET inklusive). Demzufolge beinhaltet ODAC auch nur jenes Subset an Funktionalität, das für diese Zielgruppe von Bedeutung ist. Entsprechend wird man hier vergeblich nach den zuvor erwähnten, zahlreichen Oracle-Assistenten suchen, findet dagegen jedoch jetzt die zuvor im „Full Client“ vermissten Oracle Developer Tools for Visual Studio (ODT.NET).

Ein wichtiger Unterschied besteht damit auch in der Oracle-Release-Planung. Diese kann zur Verwirrung führen – bis man die Hintergründe besser versteht und damit eine plausible Erklärung an die Hand bekommt. Ein neues ODAC-Release orientiert sich primär an den Interessen der Windows-Entwickler, wie beispielsweise im aktuellen Fall an Visual Studio 2010 und Microsoft .NET 4.0, beziehungsweise wird dann veröffentlicht, wenn das ODAC-Team zur Auslieferung neuer Features bereit ist. Das Microsoft Entity Framework oder LINQ sind gute Beispiele dafür. Der Oracle (Full) Client hingegen wird als neues Release zur Verfügung gestellt, wann immer ein neues Release der Datenbank oder ein Patchset bereitstehen.

Oracle XCOPY Client

Eigentlich gibt es gar keinen „XCOPY Client“, sondern nur eine Xcopy-Version von ODAC. Doch so wie schon zuvor beim „Full Client“ wird in der Praxis aus einem in der ODAC-XCOPY-Version enthaltenen Instant Client schnell ein sogenannter „XCOPY Cli-

ent“. Man braucht nicht lange, um den Hintergrund dieser Wortschöpfung zu verstehen. Oracle bietet zwei ODAC-Versionen zum Download an. Beide enthalten den zuvor beschriebenen Instant Client. Die eine ist jedoch ungefähr 270 MB groß, die andere lediglich 50 MB. Erstere enthält die Oracle Developer Tools for Visual Studio, die zweite nicht. Ebenso fehlen hier die Database Extensions for .NET, ODBC und SQL*Plus. Der Unterschied liegt im gewünschten Software-Verteilmechanismus in den Unternehmen. Während die erste Version mit dem Oracle Universal Installer ausgeliefert wird und deshalb eigentlich „Oracle ODAC Universal Installer Version“ heißen muss, erfolgt die Installation der ODAC-XCopy-Version im Batchverfahren, um System-Administratoren zu unterstützen, die den Rollout von Client-Software auf eine große Anzahl von Rechnern im Unternehmen planen und durchführen müssen. Aus diesem Grund ist die XCopy-Version deutlich kleiner und bietet Administratoren eine fein granulierbare Kontrolle über Installation und Konfiguration. Was aber ist nun neu in der aktuellen Version ODAC 11.2.0.1.2?

Die Oracle Developer Tools und Visual Studio 2010

Die Oracle Developer Tools for Visual Studio (ODT.NET) sind ein nahtlos in Microsoft Visual Studio integriertes Add-in. Die in ODAC 11.2.0.1.2 enthaltene aktuelle Version unterstützt nun auch Visual Studio 2010 – neben Visual Studio 2008 und Visual Studio 2005. ODT.NET unterstützt die Entwickler im gesamten Development-Lifecycle.

Die Features der Vorgängerversion 11.1.0.7.20 sind ebenfalls bereits in der DOAG News Q1/2010 beschrieben. Besonders interessant sind hier unter anderem die Einführung eines neuen „SQL Tuning Advisor“ und die „Advanced Queuing Administration“-Tools.

Wer an einer genauen Beschreibung aller in ODT.NET enthaltenen Features interessiert ist, findet weiterführende Informationen, Video-Demonstrationen und schrittweise Anleitungen (Oracle By Example) auf der ODT.NET-Homepage (<http://www.oracle.com/technetwork/developer-tools/visual-studio/overview/index-097110.html>).

Der Oracle Data Provider for .NET und das .NET Framework 4

Mit dem Oracle Data Provider for .NET (ODP.NET) profitieren .NET-Anwendungen vom optimierten Datenzugriff auf die Oracle-Datenbank im Sinne von Skalierbarkeit, Hochverfügbarkeit und Performance. Insbesondere die umfassenden Performance-Features stellen für .NET-Entwickler einen hohen Mehrwert dar.

Hier zeigen sich die besonderen Vorteile gegenüber dem früher parallel zur Verfügung gestellten und im .NET-Framework ausgelieferten Microsoft .NET Oracle Data Provider. Auch dieser sehr interessante und für .NET-Entwickler wichtige Bereich wurde in dem bereits erwähnten Artikel der DOAG News Q1/2010 technisch ausführlich beschrieben (Oracle und .NET – Die neue ODAC-Version). Mit der aktuellen ODAC-Version 11.2.0.1.2 unterstützt der ODP.NET nun sowohl das .NET Framework Client Profile als auch die Vollversion des Microsoft .NET Frameworks 4.

Ein Upgrade von ODP.NET 11.1.0.7.0 auf die neuste Version ist auf jeden Fall sinnvoll und erfordert keine Code-Anpassungen. Out-of-the-Box und ohne Tuning ergaben sich in einem konkreten Kundenbeispiel bei identischer Last im Vergleich zu ODP.NET 11.1.0.7.0 folgende Performance-Steigerungen:

- Eine schnellere Ausführung (24 Prozent)
- Mehr Durchsatz (60 Prozent)
- Bessere Antwortzeiten (38 Prozent)
- Weniger Netzwerk-Traffic (21 Prozent)
- Weniger CPU, sowohl auf dem Client (19 Prozent) als auch auf dem Server (17 Prozent)
- Mehr Transaktionen pro Sekunde auf dem Client (99 Prozent) und auf dem Server (92 Prozent)

Wer an den technischen Details von ODP.NET interessiert ist, findet weiterführende Informationen und aktuelle technische Beschreibungen der einzelnen Mehrwerte auf der ODP.NET-Homepage (siehe oben).

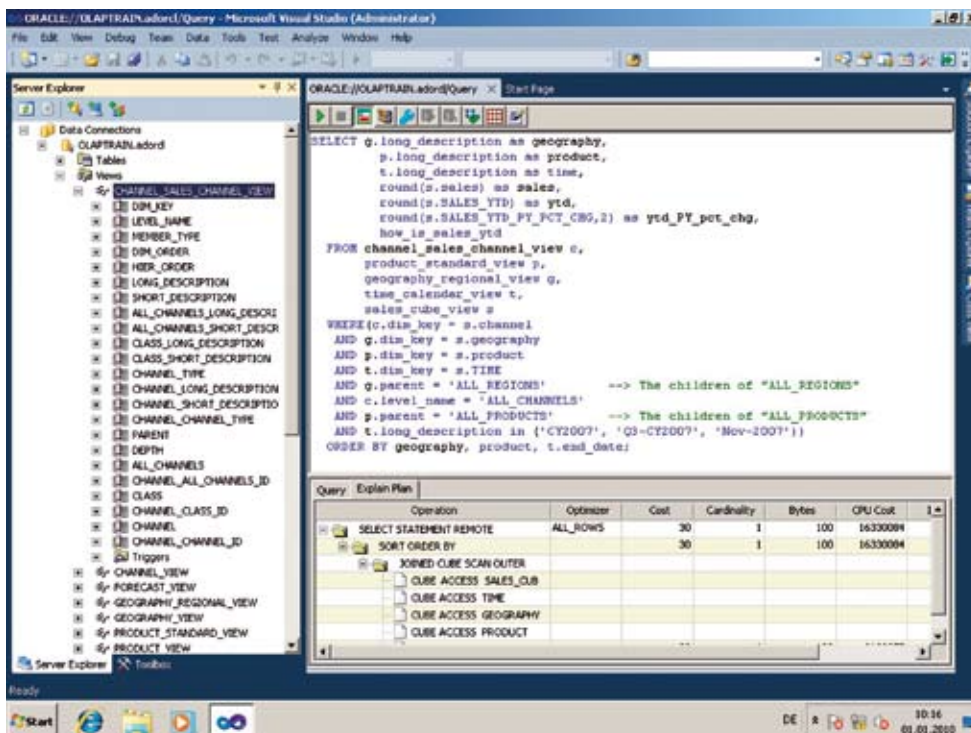


Abbildung 1: Die ODT.NET in Visual Studio 2010 und eine OLAP Cube Query

ODT.NET und Oracle OLAP für den .NET-Entwickler

Die besten Tools und Data Provider besitzen keinen Selbstzweck, am Ende müssen immer Anwendungen geschrieben werden. Da die Integration zwischen Oracle OLAP und Microsoft .NET weniger bekannt ist und in den zahlreichen klassischen Beispielen selten Berücksichtigung findet, sind die Möglichkeiten nachfolgend beschrieben.

OLAP-Applikationen können mit den Oracle Developer Tools for .NET (ODT.NET) designed und OLAP-Queries im Oracle Query Window oder Microsoft Query Window ausgeführt werden. Microsoft Query Windows erstellt OLAP-Queries grafisch. Der Oracle Data Provider for .NET (ODP.NET) wird genutzt, um OLAP-Applikationen zu erstellen und OLAP-Queries via SQL oder PL/SQL auszuführen. Ebenso kann Microsoft Office zum Erstellen von OLAP-Applikationen eingesetzt werden. Zum grafischen Design von OLAP-Cubes steht der Oracle Analytical Workspace Manager (AWM) zur Verfügung.

Abbildung 1 zeigt die ODT.NET in Visual Studio 2010 mit Cube Views und einer im Query Window geöffneten OLAP Cube Query. Darunter ist der Ausführungsplan (Explain Plan) erkennbar.

Für die .NET-Entwicklung ergeben sich mit der von Oracle angebotenen Integration zahlreiche Vorteile. MDX wird nicht benötigt, stattdessen kommt SQL mit Oracle-OLAP zum Einsatz. Die Wartungskosten sind reduziert, da Oracle-OLAP keinen separaten Datastore erfordert. Wenn die OLAP-Cubes immer weiter wachsen, kann die Oracle 11g Release 2 Grid Infrastructure genutzt werden, um Skalierbarkeit, Performance und Hochverfügbarkeit fortlaufend und unterbrechungsfrei zu gewährleisten. Mit Active Data Guard lässt sich die Auslastung gering genutzter Server erhöhen, Materialized Views verbessern die Performance von Summen-Abfragen gegen Detail-Tabellen und OLAP-Daten lassen sich durch zahlreiche Security-Features der 11g R2-Datenbank absichern – wie beispielsweise Virtual

Private Database und fein granulierte Zugriffskontrolle.

Die Oracle-Provider for ASP.NET

Beginnend mit dem .NET-Framework 2.0 beinhaltet ASP.NET dedizierte Service-Provider, um den Status einer Web-Applikation in der Datenbank zu speichern und somit die Verfügbarkeit der Web-Daten zu gewährleisten und für alle Web-Server vorzuhalten. Insbesondere die Oracle 11g Release 2 Grid Infrastructure erlangt hier eine besondere Bedeutung, um beispielsweise einen Warenkorb hochverfügbar zu machen, was im Endeffekt nichts anderes bedeutet, als den zu erwartenden Umsatz zu schützen.

Die in der aktuellen ODAC-Version 11.2.0.1.2 enthaltenen Oracle Provider for ASP.NET unterstützen diese Service-Provider nun bei der Nutzung der Oracle-Datenbank bis hin zu ASP.NET 4. Folgende ASP.NET-Provider sind unterstützt:

- Membership-Provider
- Role-Provider
- Site-Map-Provider
- Session-State-Provider
- Profile-Provider
- Web-Events-Provider
- Web-Parts-Personalization-Provider
- Cache-Dependency-Provider

Weitere Informationen dazu stehen auf der „Oracle Providers for ASP.NET“-Homepage (<http://www.oracle.com/technetwork/topics/dotnet/index-087367.html>).

Das Ende der Fahnenstange ist lange nicht erreicht

In vielen Gesprächen scheint an dieser Stelle die Integration zwischen Oracle und .NET meist ihr Ende gefunden zu haben. Eventuell unterhält man sich noch über die Möglichkeit zur Nutzung der .NET Stored Procedures, aber mehr Diskussionspunkte ergeben sich meist nicht. Doch dieser Schein trügt und hinterlässt vorschnell den Eindruck, als könne die hauseigene Microsoft-Datenbank dann doch noch an der einen oder anderen Stelle punkten.

Probleme mit Ihren Geschäftsprozessen?



Sprechen Sie uns an - wir optimieren Ihre IT!

Strategieberatung

Mit optimierten Geschäftsprozessen und cleveren IT-Strategien die Zukunft aktiv gestalten. Wir zeigen Ihnen wie!

Projektmanagement

Von der Vision bis zur Einführung: GPM zertifizierte Projektleiter garantieren Projektmanagement auf höchstem Niveau.

Technologie

Das Beste aus beiden Welten mit Oracle und Open Source. Richtungsweisende Internet Applications mit Adobe Flex, ADF, APEX und Grails.

Erleben Sie uns live auf der DOAG Konferenz + Ausstellung
16. - 19. November - Nürnberg

Profitieren Sie von unserem Oracle Know-how in den neuen esentri Workshops:

APEX Einsteiger	2 Tage
APEX Allrounder	1 Tag
APEX Champion	2 Tage

Nutzen Sie auch den individuellen IT-Fitness Workshop und erfahren Sie, welche Potenziale in Ihrer IT stecken.

esentri

Stephanienstr. 36
76133 Karlsruhe
Tel. 0721 / 7540 7530
Fax 0721 / 7540 7539
www.esentri.com
info@esentri.com

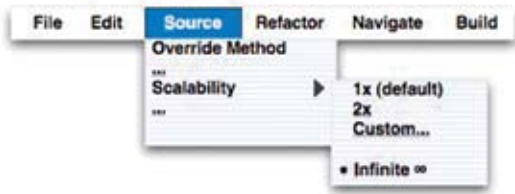


Abbildung 2: Ein Scale-Out der Application-Tier auf Knopfdruck ist nicht möglich

Insbesondere das Beispiel des „ASP.NET Session State“ führt zu einer interessanten Alternative im Zusammenhang mit .NET und aktuellster Oracle-Technologie – die im Zuge von 11g R2 mit der neuen Grid-Infrastruktur für Standalone Server und Cluster nicht unerwähnt bleiben sollte. Die Rede ist von „Oracle Coherence for .NET“.

Oracle Coherence 3.6 für das Microsoft .NET-Framework

Oracle Coherence ist das führende „In-Memory Data Grid“-Produkt – oder einfacher ausgedrückt, der leistungsfähigste verteilte Datencache am Markt. Coherence wird im sogenannten „Extreme Transaction Processing“ (XTP) eingesetzt und bietet die Möglichkeit zur Skalierung kritischer Applikationen, indem regelmäßig benötigte Daten näher an die Applikation gebracht und schnell zugreifbar gemacht werden.

Alle diese Möglichkeiten von Coherence stehen mit „Oracle Coherence

for .NET“ auch der .NET-Entwicklung zur Verfügung und können im zuvor verwendeten Beispiel des Session State als alternativer Speicherort verwendet werden (ASP.NET Session State Provider), um ihn nicht nur über mehrere Knoten hinweg skalierbar und ausfallsicher zu machen, sondern in dieser Rolle die Performance der Web-Applikationen insgesamt deutlich zu verbessern.

Wie und wo soll man diesen verteilten Cache nun einordnen? Als Erstes muss man sich von der Vorstellung trennen, es handele sich hier um eine ähnliche Technologie wie die im Einsatz mit den Oracle Real Application Clusters bekannte Implementierung des Cache Fusion. Aus mehreren Gründen ist dies nicht der Fall: Coherence adressiert eine ganz andere Ebene als die der Datenbank und fungiert dabei als Daten-Service für die Application-Tier. In dieser Aufgabe bringt eine „Coherence-Wolke“ idealerweise solche Daten nahe an die Applikationen, die vor allem dem häufigen, lesenden Zu-

griff ausgesetzt sind. Stellt man sich ein ERP-System wie beispielsweise SAP vor, lässt sich auf diese Art eine elegante Möglichkeit realisieren, um für häufig lesende Zugriffe die relevanten Daten – oder ein für die Anwender bereits fertig aufbereitetes Subset an Informationen – in eine Coherence-Wolke zu verlagern und somit das ERP-System zu entlasten. Der ansonsten notwendige direkte Zugriff wird dadurch sogar obsolet. Diese Lastreduzierung gemeinsam genutzter Datenquellen im Back-end stellt eine hochinteressante Möglichkeit dar, die Skalierbarkeit einer gesamten Infrastruktur unter Kostengesichtspunkten möglichst effektiv zu gestalten anstatt sich auf abgetrennte Segmente zu fokussieren. Denn eines ist unangefochten: Eine durchaus wünschenswerte Möglichkeit, wie in Abbildung 2 dargestellt, um ein Scale-Out der Application-Tier auf Knopfdruck zu realisieren, gibt es nicht, sondern muss in der Gesamtarchitektur geplant und umgesetzt werden.

Daneben stellen Real Time Clients eine interessante Möglichkeit dar, um es Java-, JavaEE- und .NET-Applikationen zu ermöglichen, sich mit einem „Coherence Data Grid“ zu verbinden und somit als Brücke für Plattform- und Sprachen-Interoperabilität zu fungieren. Dies ist für heterogene Umgebungen höchst interessant. An dieser Stelle muss allerdings auf die Verwechslungsgefahr aufgrund parallel verwendeter Begriffe aufmerksam gemacht werden. „Grid“ besitzt hier eine andere Bedeutung als die „Grid-Infrastruktur“ im Zusammenhang mit den Oracle Real Application Clusters. Denn Coherence bildet eine Art „Peer-to-Peer“-Netzwerk, dessen Member ad-hoc auf beliebigen Rechnern gestartet werden können und sich ohne weitere Konfiguration in dieses Netzwerk einklinken. Jedes Mitglied in diesem Netzwerk hält mindestens eine Kopie von Daten eines anderen Mitglieds. Alle überwachen sich gegenseitig und garantieren die verlustfreie Verfügbarkeit der Daten (siehe Abbildung 3).

Weitere Informationen zur Nutzung und Architektur von Coherence mit .NET stehen auf der Oracle Coherence Homepage (<http://www.oracle.com/>

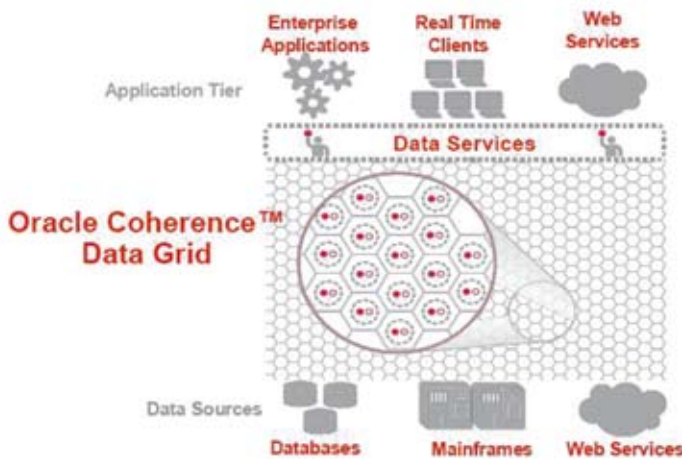


Abbildung 3: Oracle Coherence als Skalierungs-Infrastruktur für die Application-Tier

technetwork/middleware/coherence/overview/index-087514.html).

Manch einer mag sich jetzt fragen, ob hier nicht eine Überschneidung mit Oracle Times Ten besteht. Doch aus unterschiedlichen Gründen ist auch dies nicht der Fall. Zugegeben – man muss sich tiefer mit der Technologie beschäftigen, um die dedizierten Mehrwerte richtig unterscheiden zu lernen, die dann aber umso deutlicher zu Tage treten. Ein unmittelbar erkennbarer Unterschied zeigt sich dagegen, wenn man den Datenzugriff betrachtet, denn Coherence arbeitet ausschließlich mit Objekten und muss deshalb auch vollkommen anders angesprochen werden als Oracle Times Ten in seiner Rolle als „In-Memory-optimierte“, relationale Datenbank. Die nachfolgenden Code-Beispiele einer Abfrage gelten für eine Implementierung unter .NET und verdeutlichen die Unterschiede bei der Arbeit mit Coherence.

Als Erstes wird ein Index angelegt, um die Abfrage effizienter zu machen:

```
cache.AddIndex(new ChainedExtractor("getHomeAddress.getState"), /*fOrdered*/ false, /*comparator*/ null);
```

Anschließend sollen alle Kontakte in Massachusetts gefunden werden:

```
ICacheEntry[] aCacheEntry = cache.GetEntries(new EqualsFilter("getHomeAddress.getState", "MA"));
```

Für das gesamte Cache-Dataset werden Kontakte gezählt, die älter sind als „nAge“:

```
Console.WriteLine("count > " + nAge + ": " + cache.Aggregate(new GreaterFilter("getAge", nAge), new Count()));
```

Im konkret verwendeten „Coherence for .NET“-Projekt werden auf Basis der vorhandenen Daten mehr als 10.000 Kontakte geliefert. Abbildung 4 zeigt ein Beispiel der Ausgabe.

Dieses und weitere Beispiele stehen auf der Coherence Website (<http://coherence.oracle.com/display/EXAMPLES/Coherence+Features+Examples#CoherenceFeaturesExamples-basic>).

Coherence+Features+Examples#CoherenceFeaturesExamples-basic).

Oracle Times Ten Data Provider for .NET

Times Ten ist eine „In-Memory-optimierte“, relationale Datenbank und bietet als solche volle SQL-Unterstützung und eine tiefe Integration mit der Oracle-Datenbank. Der Oracle Times Ten Data Provider for .NET bietet .NET-Anwendungen den schnellen Datenzugriff auf Times Ten und unterstützt dabei ODP.NET-Klassen, Enumerationen, Interfaces, Delegates und Strukturen. Weitere Informationen und konkrete Beispiele zu Oracle Times Ten sind auf der Oracle Times Ten Homepage (<http://www.oracle.com/technetwork/database/timesten/overview/index.html>). Dort stehen auch die 32-Bit- und 64-Bit-Version des Oracle Times Ten Data Providers for .NET als XCopy-Version zum Download bereit.

Oracle-Unterstützung für LINQ und das Microsoft Entity Framework

Im Zusammenhang mit der aktuellen Datenbank 11g R2 dürfen auch Microsoft LINQ und das Microsoft Entity Framework nicht unerwähnt bleiben. Das Entity Framework bietet ein objekt-relationales Mapping (ORM oder O/RM) für .NET und abstrahiert das logische (relationale) Datenbankschema, um es der Applikation als spezifisches, konzeptuelles Schema zu präsentieren. Dabei eliminiert es die Probleme des sogenannten „Impedance Mismatch“, die sich in der Software-Entwicklung aus den unterschiedlichen Strukturen relationaler Datenbanken und objekt-orientierter Klassen-Hierarchien ergeben.

Die kommende Oracle-Unterstützung wird das .NET Framework 4 und Visual Studio 2010 erfordern. Der Zeitrahmen sieht derzeit eine Beta-version gegen Ende 2010 vor, eine produktive Version folgt in 2011. Diese Angaben sind jedoch ohne Gewähr und können sich jederzeit ändern. Ein offizielles Statement of Direction zu diesem Thema steht unter <http://www.oracle.com/technetwork/topics/dotnet/whatsnew/oracle-entity-framework-sod-130214.pdf>. Neuigkeiten zu dieser Unterstützung und verwandten Themen wer-

Houston, wir haben ein Problem!

Besuchen Sie uns auf der DOAG 2010 Konferenz + Ausstellung

...und wir die Lösung!

IT im Alarmzustand! Fachbereiche rufen nach neuen und besseren IT-Services. Governance, Risikomanagement & Compliance sind auf der Tagesordnung.

Zeit, mit einem verlässlichen Partner zu sprechen, der mit Ihnen gemeinsam wirtschaftliche Lösungen entwickelt. Höchste Zeit, mit PROMATIS zu sprechen!

PROMATIS verbindet intelligente Geschäftsprozesse mit modernen Oracle Technologien und Applikationen:

- Oracle E-Business Suite und CRM On Demand
- Oracle Accelerate Mittelstandslösungen
- Out of the box Appliances: BPM, ECM und WebCenter
- Business Intelligence und Data Warehouse
- BPM, Oracle SOA und Application Integration (AIA)
- Bewährte Projekt-Vorgehensmodelle und Best Practice-Prozesse

Profitieren Sie von der PROMATIS Lösungskompetenz und unserer internationalen Erfahrung. Sprechen wir darüber!

PROMATIS

Knowledge Powered Business Processes

PROMATIS software GmbH

Tel.: +49 7243 2179-0 · Fax: +49 7243 2179-99

www.promatis.de · hq@promatis.de

Ettlingen/Baden · Hamburg · Berlin

den auch zeitnah über die Deutsche Microsoft Community Website der Oracle Business Unit ST PCM (<http://apex.oracle.com/url/communities>) veröffentlicht.

Objektrelationales Mapping mit Oracle und NHibernate

Im Zusammenhang mit dem Entity Framework und der ausstehenden Unterstützung durch Oracle muss man sich auch ins Gedächtnis rufen, dass ORM beileibe keine neuartige Erfindung von Microsoft darstellt. Das Gegenteil ist der Fall und wie so oft befindet sich Microsoft in diesem Terrain wieder einmal auf Aufholjagd. Denn es existieren bereits seit langer Zeit erprobte Lösungen wie beispielsweise Hibernate. Dabei zählt nicht nur die Tatsache, dass mit NHibernate ein ORM für .NET angeboten wird, es besitzt darüber hinaus mehrere gute Merkmale. NHibernate ist eine Open-Source-Lösung. Des Weiteren stellt es – wieder im Gegensatz zum Microsoft Entity

Framework – schon heute eine ausgereifte ORM-Implementierung dar.

Wer das gesamte ORM-Thema aus der Vogelperspektive betrachtet, wird schnell feststellen, dass am Rande der Diskussion im Entwicklerumfeld nicht nur Art und Anzahl der möglichen Abstraktionslayer zwischen Codierung und Daten als wichtig gelten, sondern am Ende immer einzig und allein Benutzerfreundlichkeit und Performance einer fertigen Applikation im Praxiseinsatz aus Anwendersicht maßgeblich sind. Es zählt nur, was am Ende dabei herauskommt. In unserer heutigen Welt, in der Prozesse immer schneller werden und das Datenvolumen explosionsartig wächst, darf eine Abstraktion keine nennenswerten Performance-Einbußen nach sich ziehen. Wenn sie den Zugriff auf unterschiedliche Datenbanken nach dem Prinzip des kleinsten gemeinsamen Nenners bewirkt, werden proprietäre Features zur Performance-Steigerung und Sicherung von Skalierbarkeit und Hochverfügbarkeit wie das sprichwörtliche Kind mit dem Bade ausgekippt.

Im Vergleich zu kommerziellen Lösungen ist dieser Aspekt mit einer Open-Source-Implementierung wie NHibernate als potenziell geringer einzuschätzen. Hier wird kaum zu erwarten sein, dass die eine oder andere Datenquelle bevorzugt behandelt oder im Gegenzug ausgegrenzt wird. Idealerweise sorgen die jeweiligen Hersteller selbst für die optimierte Nutzung.

Bei diesem Stichwort stellt sich die Frage nach der praktischen Anwendung des Oracle Data Providers for .NET (ODP.NET) mit NHibernate, um sich optimierter Datenzugriffe sicher zu wissen und den Oracle-Mehrwert für .NET-Anwendungen nutzen zu

können, wie beispielsweise Connection Pooling oder Skalierbarkeit auf Basis einer Oracle 11g Release 2 Grid Infrastructure. Der maßgebliche Schritt ist also folgerichtig, NHibernate dazu zu bewegen, ODP.NET zu nutzen. Dies geschieht durch einen Eintrag in der NHibernate-Konfigurationsdatei:

```
<property name="connection.driver_class">
NHibernate.Driver.OracleDataClientDriver
</property>
```

Gute Aussichten: 100 Prozent „Managed ODP.NET“

In Zukunft wird es von Oracle einen sogenannten „Managed Provider“ geben, der auch den Oracle Client beinhaltet und vollständig ADO.NET-konform sein wird. Managed ODP.NET wird 2011 ein Thema sein – hier ist dann auch eine Betaversion zu erwarten.

Weitere Informationen

Die Oracle Business Unit ST PCM der ORACLE Deutschland B.V. & Co. KG veröffentlicht auf der „Deutschen Microsoft Community Website“ (<http://apex.oracle.com/url/communities>) fortlaufend Tipps und Tricks zum Einsatz von Oracle-Technologie im Microsoft-Umfeld. Daneben gibt es weitere deutschsprachige Communities zu den Themen „Modernisierung“, „Streams“, „SAP“, „Apex“ sowie „DBA- und BI-Publisher“.

Kontakt:

Claus Jandausch
claus.jandausch@oracle.com

```
MA Residents
ConverterEntry{Key=John Scqngqda“,
Value=“John Scqngqda
Addresses
Home: 265 Beaconm St.
Oaskxm, MA 88259
US
Work: Yoyodyne Propulsion Systems
330 Lectroid Rd.
Grover’s Mill, OK 95744
US
Phone Numbers
work: +11 88 903 8991283
home: +11 98 553 5878221
Birth Date: 1960-01-03”}
```

Abbildung 4: Ein Ergebnis der Abfrage

Oracle-Newsticker

Oracle mit sattem Gewinn

Im ersten Finanzquartal 2011, das zum 31. August 2010 endete, erzielte die Oracle Corporation einen GAAP-Umsatz von 7,5 Milliarden US-Dollar. Das entspricht einem Wachstum um 48 Prozent im Vergleich zum Vorjahreszeitraum. Der Non-GAAP-Gewinn stieg im ersten Finanzquartal 2011 um 50 Prozent auf 7,6 Milliarden US-Dollar.

Die Software-Erlöse aus Neulizenzen nach GAAP stiegen um 25 Prozent auf 1,3 Milliarden US-Dollar. Zugleich stiegen die Umsätze aus Software-Lizenz-Updates und aus dem Produkt-Support nach GAAP und nach Non-GAAP um 11 Prozent auf 3,5 Milliarden US-Dollar. Die betrieblichen GAAP-Einnahmen stiegen um 10 Prozent auf 1,9 Milliarden US-Dollar, die betriebliche GAAP-Gewinnspanne betrug 26 Prozent. Die betrieblichen Non-GAAP-Einnahmen stiegen um 27 Prozent auf 2,9 Milliarden US-Dollar, die Non-GAAP-Marge aus dem operativen Geschäft lag bei 39 Prozent. Der Reingewinn nach GAAP stieg um 20 Prozent auf 1,4 Milliarden US-Dollar, während sich der Reingewinn nach Non-GAAP um 38 Prozent auf 2,1 Milliarden US-Dollar erhöhte. Der GAAP-Gewinn pro Aktie stieg im Vergleich zum Vorjahreszeitraum um 20 Prozent auf 0,27 US-Dollar, der Non-GAAP-Gewinn pro Aktie um 38 Prozent auf 0,42 US-Dollar. Der betriebliche Cashflow nach GAAP lag über die vergangenen zwölf Monate gerechnet bei 8,8 Milliarden US-Dollar.