Oracle 10*g* Connection Manager

Kunde:	DOAG*News*
Ort, Datum:	Artikel im Heft Q4 / 2005
Thema / Themen:	**Artikel von merlin.zwo**
Projekt:	**Oracle 10g Connection Manager**
Autor:	**Markus Schmidt**

- Oracle & Technologien
- Systementwicklung
- Individuelle Lösungen

Schon seit langem ist der Oracle Connection Manager Bestandteil der Oracle Net-Umgebung. In diesem Beitrag soll diese Komponente genauer betrachtet und Einsatzvarianten dargestellt werden.

Einführung

Seit Oracle 8 Enterprise Edition ist der Connection Manager optionaler Bestandteil des Oracle-Netzwerkstacks. Es handelt sich hierbei um einen Netzwerkdienst, der für verschiedene Zwecke eingesetzt werden kann: als Verbindungsproxy, zur Zugriffskontrolle und zum Multiplexing von Verbindungen. Der Connection Manager ist somit ein vielseitiger Software-Router für Oracle Net. Nachfolgend wird *Oracle Connection Manager* durch *CM* abgekürzt. Soweit nicht anders angegeben, sind die Aussagen auf die Version 10.1.0.2 bezogen (Oracle 10*g* Enterprise Edition).

Features

Oracle CM ist als optionale Komponente in den Enterprise Editionen seit Version 8 verfügbar. Folgende Einsatzmöglichkeiten werden damit abgedeckt:

- Proxy: eingehende Verbindungsanfragen werden durch den CM angenommen und an die entsprechende Datenbank weitergeleitet. Dies ist vor allem in Netzwerkumgebungen interessant, bei der die Datenbank durch eine Firewall geschützt ist und direkte Verbindungen vom Client zum Datenbankserver geblockt werden.
- Session Multiplexing / Connection Concentration: mehrere Verbindungen von Clients zu einer Datenbank werden auf eine gemultiplexte Verbindung abgebildet, d.h. es werden auf der Seite des Datenbankservers weniger Verbindungen angefragt bzw. aufgebaut und somit Ressourcen eingespart.
- Regelbasierte Zugriffskontrolle: anhand von Regeln kann bestimmt werden, welche Clients Verbindungen aufbauen dürfen (Positivliste) oder welchen Clients der Zugriff verwehrt wird (Negativliste).

- Multi-Protokoll-Unterstützung: in früheren Versionen war es möglich, durch Oracle Net unterstützte Protokolle durch den CM zusammenzuführen (d.h. Anfrage über Protokoll1 zum CM, von dort über Protokoll2 zur Datenbank). Dieses Feature, bei dem CM als Protokollübersetzer fungiert, wird jedoch nicht mehr unterstützt, da der CM nur per TCP/IP oder IPC angesprochen werden kann.

Architektur

Für einen ordnungsgemäßen Betrieb sind verschiedene Bestandteile des CM definiert:

- Der CM-Listener für die Annahme von Verbindungsanfragen von Clients und CMADMIN
- CMGW (CM Gateway)
- CMADMIN (CM Administration)

Wird von einem Client eine Verbindung zu einer Datenbank angefordert, so wird diese Anfrage durch den CM Listener entgegengenommen. Dieser prüft die Verbindungsdaten gegen die definierten Regeln aus der CM-Konfiguration. Wenn keine Regel dem Verbindungsaufbau widerspricht, leitet der CM-Listener die Verbindungsanfrage zu einem CMGW-Prozess weiter. Hierbei wird derjenige GW-Prozess verwendet, der die wenigsten Anfragen verwaltet. Der GW-Prozess wiederum leitet die Anfrage direkt zur angefragten Datenbank (bzw. zum Listener der angefragten Datenbank) oder aber an einen weiteren GW-Prozess weiter. Bis die Datenbankverbindung geschlossen wird, fungiert der GW-Prozess als Proxy zwischen Client und Server. Wird bei der Konfiguration mit Multiplexing festgestellt, dass bereits eine Verbindung existiert, so wird die neue eingehende Verbindung auf die bereits bestehende aufgesetzt, eine neue Verbindung zwischen GW und Datenbank wird somit vermieden. Durch den ADMIN-Prozess wird permanent der Zustand der GW-Prozesse und der CM-Listener überwacht (aktuelle Last, Anzahl der Verbindungen). Weiterhin ist der ADMIN-Prozess für das Starten und Stoppen dieser Prozesse zuständig. Durch den Einsatz des CM und einer Shared Server-Datenbank kann die Netzwerkstruktur skalierbar gestaltet werden.

Konfiguration und Administration

Zur Konfiguration des CM sind nur wenige Schritte notwendig. Zum einen muss der CM konfiguriert werden. Die Einstellungen werden in der Datei cman.ora vorgenommen, die im üblichen Admin-Verzeichnis der Oracle Netzwerkumgebung abgelegt wird. Die Einträge umfassen die Bereiche CM-Listener (Protokoll, Adresse und Port), die Zugangsregeln sowie weitere Parameter für Performance und Logging. Leider gibt es zu diesem Zweck kein passendes Oracle-Werkzeug. Da die Datei wie andere Net-Konfigurationsdateien in Textform abgelegt ist, können die entsprechenden Eintragungen mit einem normalen Texteditor durchgeführt werden. Im zweiten Schritt wird die Konfiguration für den Client fertig gestellt: Einträge in der tnsnames.ora müssen für die Verwendung des CM leicht abgewandelt werden. Soll durch CM Sessionmultiplexing durchgeführt werden, so ist dies in den Datenbankeinstellungen ebenfalls zu berücksichtigen.
Nach Abschluss dieser Schritte kann der CM in Betrieb genommen werden.

CM-Konfiguration

Der Aufbau der cman.ora ist simpel: im ersten Teil wird der *Listening endpoint* für den CM-Listener definiert, hierbei werden nur TCP/IP (nicht TCPS) oder IPC unterstützt. Bei TCP/IP besteht der Eintrag aus der Angabe des zu verwendenden Protokolls, der Adresse des CM-Servers sowie des Ports, über den die Verbindungen des Clients aufgebaut werden. Als Port wird 1521 empfohlen (wie auch beim Datenbank-Listener).

Beispiel: auf dem Server cmserver soll der CM über TCP/IP auf Port 1521 erreichbar sein:

```
((Beginn Programmcode))
(ADDRESS=(PROTOCOL=tcp)(HOST=cmserver)(PORT=1521))
((Ende Programmcode))
```

Die Definition der Zugriffsregeln erfolgt im zweiten Teil. Einzelne Zugriffsregeln (RULE) werden in einer RULE_LIST zusammengefasst. Eine Regel gibt vor, ob ein Verbindungsaufbau zwischen zwei Knoten stattfinden darf oder abgelehnt wird. Es müssen mindestens 2 Regeln definiert sein: eine für die Clientverbindung und eine für die Verbindungen über das cmctl-Tool. Dies war in früheren CM-Versionen nicht der Fall. Wird eine Migration der Konfiguration mittels cmmigr durchgeführt, so werden diese Standardverbindungen automatisch eingeführt.
Verbindungen werden durch die Angabe einer Quelladresse (SRC) und einer Zieladresse (DST) angegeben. Der Dienst, der auf der Zielseite angesprochen wird (SRV), ergibt sich aus dem Servicenamen der Datenbank. Wird hier cmon angegeben, ist der CM und keine Datenbank gemeint.
Wildcards sind bei der Angabe von SRC und DST nicht für Teiloktette der IP-Adresse möglich: * bezeichnet lediglich eine beliebige IP-Adresse. Alternativ kann auch die

Anzahl der relevanten führenden Bits der IP-Adresse über die Notation /nn angegeben werden. Beispielsweise werden alle Subnetze aus dem Class-C-Netz 192.168.17.0 über die Angabe 192.168.17.0/24 definiert.
Mit ACT wir die durchzuführende Aktion bei einer Anfrage definiert: akzeptieren (accept) oder zurückweisen (reject).

Beispiel anhand der minimalen Regeln:

```
((Beginn Programmcode))
(RULE_LIST=
  (RULE=
    (SRC=*)
    (DST=127.0.0.1)
    (SRV=cmon)
    (ACT=accept)
  )
  (RULE=
    (SRC=*)
    (DST=*)
    (SRV=*)
    (ACT=accept)
  )
)
((Ende Programmcode))
```

Die erste Regel gibt hierbei an, dass der lokale Zugriff auf CM über cmctl (erkennbar an SRV=cmon) erlaubt ist. Alle Anfragen für andere Dienste werden über die zweite Regel grundsätzlich angenommen. Sind keine Standardregeln definiert, so werden keine Verbindungen akzeptiert (auch keine von cmctl!)
Im dritten Teil der Konfigurationsdatei werden weitere Parameter behandelt, die die Authentifizierung, Statistiken von Verbindungen, Protokollierungs- bzw. Tracingoptionen sowie Zeitbegrenzungen und Maximalwerte für Verbindungen spezifizieren. Werden die Werte hier gesetzt, so sind diese global gültig. Hierbei ist zu beachten, dass Parameter, die im Regelteil gesetzt werden, die globalen Einstellungen übersteuern.

Beispiel:

```
((Beginn Programmcode))
(PARAMETER_LIST=
  (TRACE_DIRECTORY=/oracle/network/log/)
  (TRACE_LEVEL=off)
  (LOG_LEVEL=admin)
  (CONNECTION_STATISTICS=yes)
)
((Ende Programmcode))
```

Zum Schluss müssen die drei Teile der Konfiguration nur noch zusammengeführt und mit einem Alias versehen werden. Die endgültige Datei cman.ora sieht nun folgendermaßen aus:

```
((Beginn Programmcode))
cman_cmserver=
  (CONFIGURATION=
    (ADDRESS= … )
    (RULE_LIST= … )
    (PARAMETER_LIST= … )
  )
 ((Ende Programmcode))
```

Durch den Eintrag von mehreren Konfigurationsabschnitten können verschiedene CM-Prozesse für unterschiedliche Zwecke definiert werden. Diese können voneinander getrennt administriert werden. Als Standardwert für den Alias sollte CMAN_<hostname> benutzt werden, da dieser Alias automatisch von cmctl gesucht wird und bei Existenz auch verwendet wird.

Client-Konfiguration

In der Datei tnsnames.ora muss nur noch eine Kleinigkeit geändert werden, um Verbindungen über CM herstellen zu können. Standardmäßig steht als Adresse in einem TNS-Eintrag die Adresse des Datenbankservers. Hier muss nur noch das Protokoll und die Adresse sowie der Port des CM-Servers eingetragen werden, die Einträge in Connect_Data, insbesondere der Service_Name werden unverändert an CM übergeben. Diese Konfiguration kann mit dem *Oracle Net Manager* durchgeführt werden.

Datenbankserver-Konfiguration

Die Datenbank kann sich (über den PMON-Prozess) wie bei einem üblichen Listener auch bei einem nicht auf dem lokalen System laufenden CM registrieren. Hierzu ist lediglich der TNS-Alias-Eintrag für `remote_listener` in der `init.ora` zu setzen. Der Alias wird wie üblich durch `tnsnames.ora` aufgelöst. Hierdurch wird ein direkter Informationsaustausch zwischen CM und Datenbank ermöglicht.
Wird die Multiplexing-Möglichkeit gewünscht, so ist auf der Datenbankseite der Parameter `DISPATCHERS` auf den Wert `(PROTOCOL=tcp)(MULTIPLEX=on)` zu setzen.
Zu beachten ist, dass der Listener der Datenbank trotz CM korrekt installiert und in Betrieb sein muss, da ansonsten keine Verbindungen etabliert werden können!

Administration über cmctl

Sind die Konfigurationsschritte erledigt, so kann der CM gestartet werden. Dies erfolgt mit dem Administrationstool cmctl, bei dem entweder eine Parameterübergabe per Kommandozeile oder eine interaktive Bedienung möglich ist. Für skriptgestützte Arbeiten steht ein Batch-Modus zur Verfügung, bei dem ein Skript als Parameter übergeben wird.

Für die Administration stehen verschiedene Möglichkeiten zur Verfügung: Starten und Stoppen, Ändern von Parametern, Anzeigen von Zuständen und Gateway-Kommandos. Hierbei kann entweder ein lokaler CM oder aber auch ein Remote-CM angesprochen werden.

Eine Liste mit möglichen Kommandos kann per HELP angefordert werden.

Zum Starten und Stoppen des CM werden STARTUP und SHUTDOWN verwendet:

```
((Beginn Programmcode))
oracle@cmserver> cmctl startup
CMCTL for Linux: Version 10.1.0.2.0 - Production on 09-SEP-2005 18:21:20
Current instance CMAN_cmserver is not yet started
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=192.168.0.1)(PORT=1521))
Starting Oracle Connection Manager instance CMAN_cmserver. Please wait...
TNS-04077: WARNING: No password set for the Oracle Connection Manager instance.
CMAN for Linux: Version 10.1.0.2.0 - Production
Status of the Instance
----------------------
Instance name             CMAN_cmserver
Version                   CMAN for Linux: Version 10.1.0.2.0 - Production
Start date                09-SEP-2005 18:21:20
Uptime                    0 days 0 hr. 0 min. 9 sec
Num of gateways started   2
Average Load level        0
Log Level                 ADMIN
Trace Level               OFF
Instance Config file      cman.ora
Instance Log directory    /oracle/network/log/
Instance Trace directory  /oracle/network/log/
The command completed successfully.

oracle@cmserver> cmctl shutdown
CMCTL for Linux: Version 10.1.0.2.0 - Production on 09-SEP-2005 18:29:43
TNS-04077: WARNING: No password set for the Oracle Connection Manager instance.
Current instance CMAN_cmserver is already started
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=192.168.0.1)(PORT=1521))
The command completed successfully.
((Ende Programmcode))
```

Wenn CM über die interaktive Umgebung verwaltet werden soll, muss vor dem Absetzen diverser Befehle das Kommando ADMINISTER abgesetzt werden. Hierbei wird als Parameter der Konfigurationsname aus der cman.ora verwendet. Alle nachfolgenden Kommandos beziehen sich dann auf diese Instanz.
Um Parameter während des laufenden Betriebs zu ändern, kann SET mit verschiedenen Optionen aufgerufen werden. Diese Optionen sind identisch mit denen, die in der Konfigurationsdatei in der PARAMETER_LIST angegeben werden können. Jedoch ist hierbei zu beachten, dass diese Parameter ebenfalls manuell in die cman.ora übernommen werden müssen. Ausnahme ist das Speichern des Passwortes, das den CM gegen unbefugte Zugriffe absichert. Hierzu wird über den Befehl set password und anschließende Eingabe das Passwort vergeben. Um das Passwort persistent in der cman.ora zu speichern wird anschließend save_passwd abgesetzt. Es wird ein Eintrag mit Namen password_CMAN_<alias> erzeugt, der das Passwort verschlüsselt enthält. Zu beachten ist, dass das Passwort für alle Instanzen gilt.
Werden Parameter in der cman.ora ergänzt, so können diese Einstellungen mit RELOAD direkt in einen laufenden CM übernommen werden. Dies vermeidet einen Neustart der CM-Prozesse im laufenden Betrieb.
Über den Befehl SHOW können detaillierte Informationen über den Zustand des CM abgerufen werden (ohne weitere Parameter wird eine Liste der möglichen Bereiche angezeigt). Hierbei sind im Allgemeinen wohl vor allem die Bereiche der Verbindungen, Gateways, Services und Regeln von Interesse.
Sollen Verbindungen geschlossen werden (z.B. weil sie zu lange im Status *idle* waren), so ist dies über CLOSE CONNECTION gezielt möglich. Die Auswahl der Verbindungen kann hierbei durch verschiedene Kriterien erfolgen, die abgebrochenen Verbindungen erhalten bei Weiterarbeit den Fehler ORA-03113: end-of-file on communication channel.

Szenarien

Nachfolgend soll kurz dargestellt werden, für welche Zwecke der CM eingesetzt werden kann.

- Über Regeln können nur bestimmte Rechner im Netz oder Subnetze auf die Datenbank zugreifen (z.B. nur bestimmte Abteilungen). In Abbildung 1 ist der Zugriff für das Vertriebsubnetz gestattet, alle Clients aus diesem Netz können über den CM auf die Datenbank zugreifen. Da das Subnetz der Buchhaltung über eine Zurückweisungsregel konfiguriert wurde, lehnt der CM Datenbankverbindungen ab.

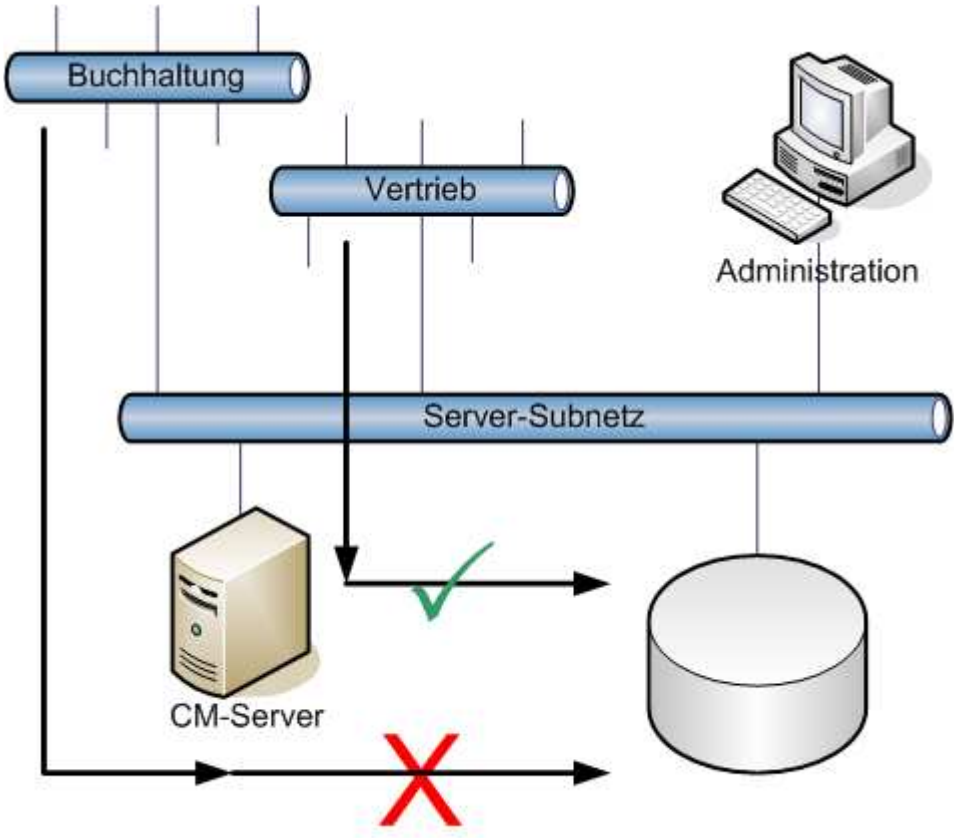

Abbildung 1: Regelbasierter Zugriff

- Bei OLTP-Datenbanken mit einer hohen Zahl von parallel arbeitenden Sessions (d.h. bei kontinuierlicher Interaktion mit der Datenbank ohne oder mit wenig Idle-Zeit) ist die Connection Concentration hilfreich. Vor allem bei Systemen, bei denen Sessions nur kurz aufgebaut, einige Statements abgesetzt und die Verbindung wieder geschlossen wird, kann CM zu besserer Bandbreitennutzung führen. Mehrere parallele Sessions werden hierbei über eine physikalische Verbindung geführt (siehe auch Abbildung 2), was den Durchsatz pro verwendetem Port erhöhen wird. Dadurch wird auch die Skalierbarkeit steigen, da eine solche gemultiplexte Verbindung mehrere Tausend Sessions bedienen kann.

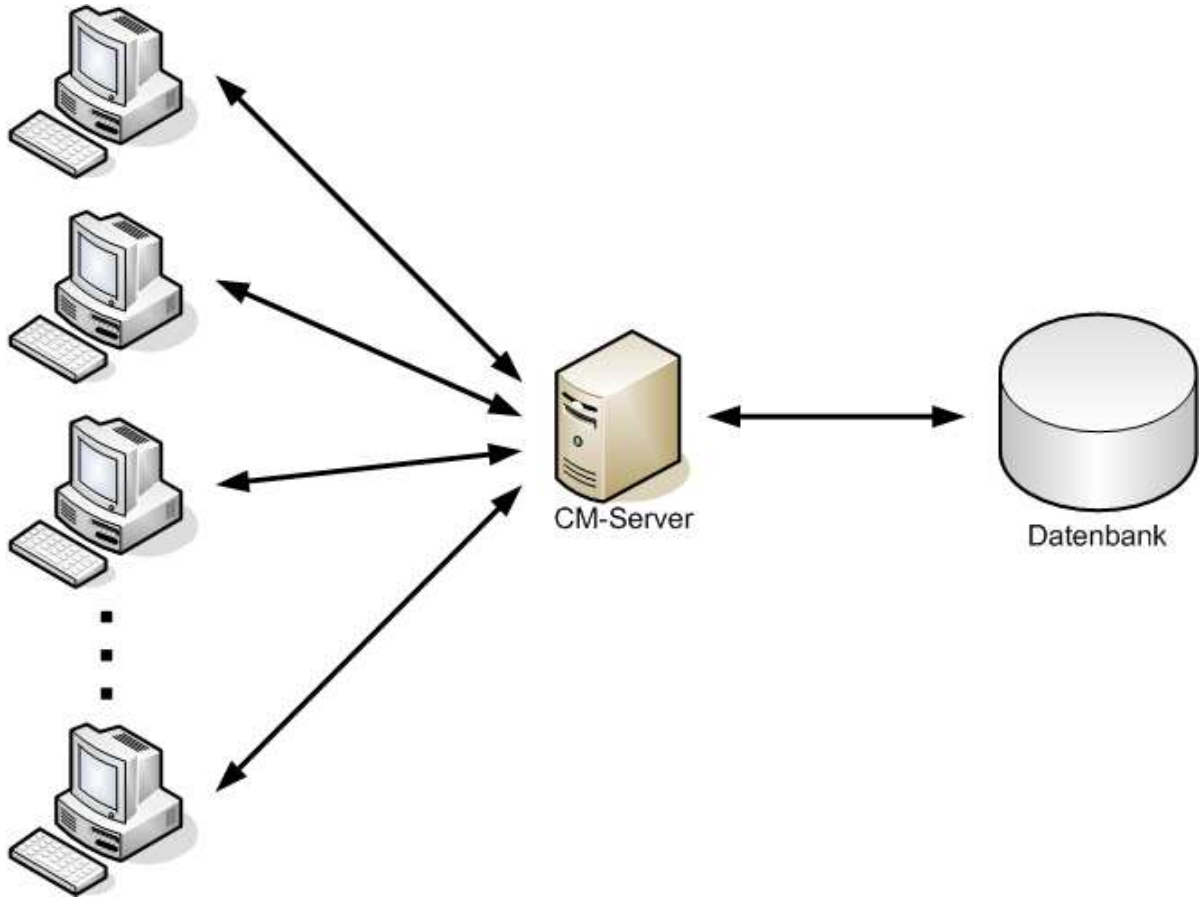

Abbildung 2: Connection Multiplexing

- Sie setzen Java Applets ein, die eine Verbindung zur Datenbank benötigen. Hierbei ist zu beachten, dass es (ohne weitere Zusatzarbeiten) nur möglich ist, auf einen Server zuzugreifen, von dem das Applet auch geladen wurde. Dieses Problem kann umgangen werden, indem auf dem Webserver ebenfalls ein CM gestartet wird. Der tatsächliche Datenbankserver kann somit auf einem eigenen Server verbleiben. Das Applet muss sich somit nur gegen den CM verbinden und nutzt dessen Proxyfunktionalität (siehe Abbildung 3).

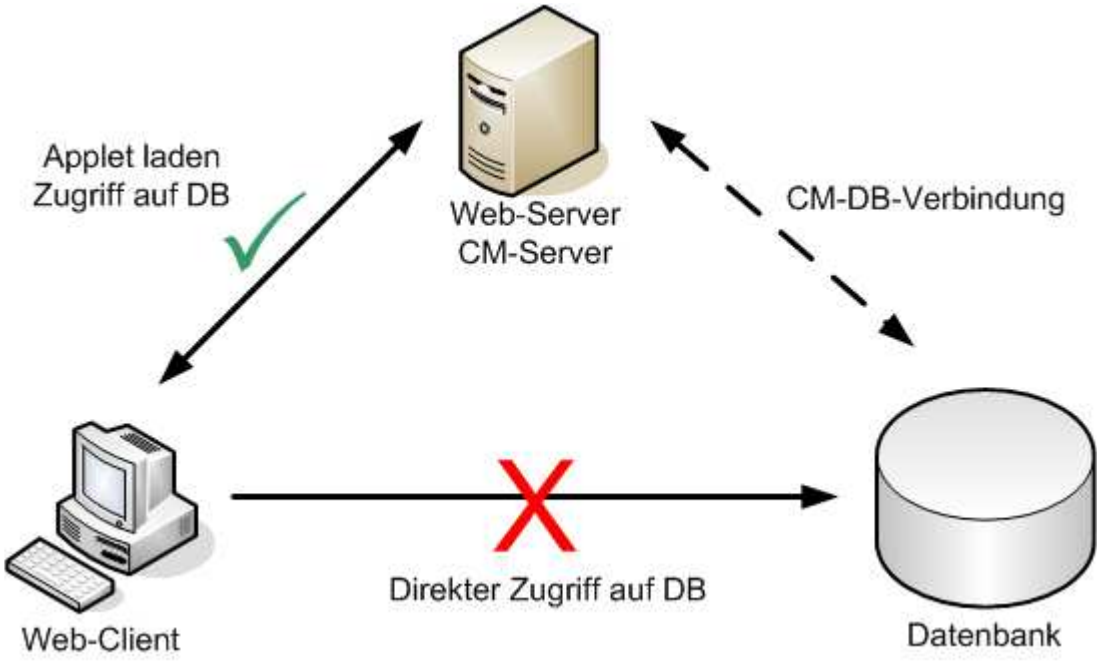

Abbildung 3: Java Applet-Zugriff

- Während einer Migration werden die Datenbanken auf neue Server mit anderen IP-Adressen umgezogen. Durch den CM sind Sie in der Lage, den Clients diesen Sachverhalt zu verbergen, d.h. es müssen keine Änderungen in tnsnames.ora vorgenommen werden. Die Clients arbeiten weiterhin mit dem CM als Verbindungsproxy.
- Die Server und Clients sind in verschiedenen Subnetzen installiert. Hierbei sind keine direkten Verbindungen zwischen Clients und Servern erlaubt, die installierte Firewall blockiert direkte Anfragen. Um jedoch die Datenbank ansprechen zu können, muss genau dies möglich sein. Die Lösung wäre hier ein Gateway-Server zwischen den Subnetzen, auf dem ein CM läuft. Die Clients müssen sich somit nicht direkt an den Datenbankserver verbinden, sondern greifen über den CM-Proxy auf den Server zu. Die Firewall muss nur (beidseitig) die Verbindungen zum Gatewayserver zulassen.

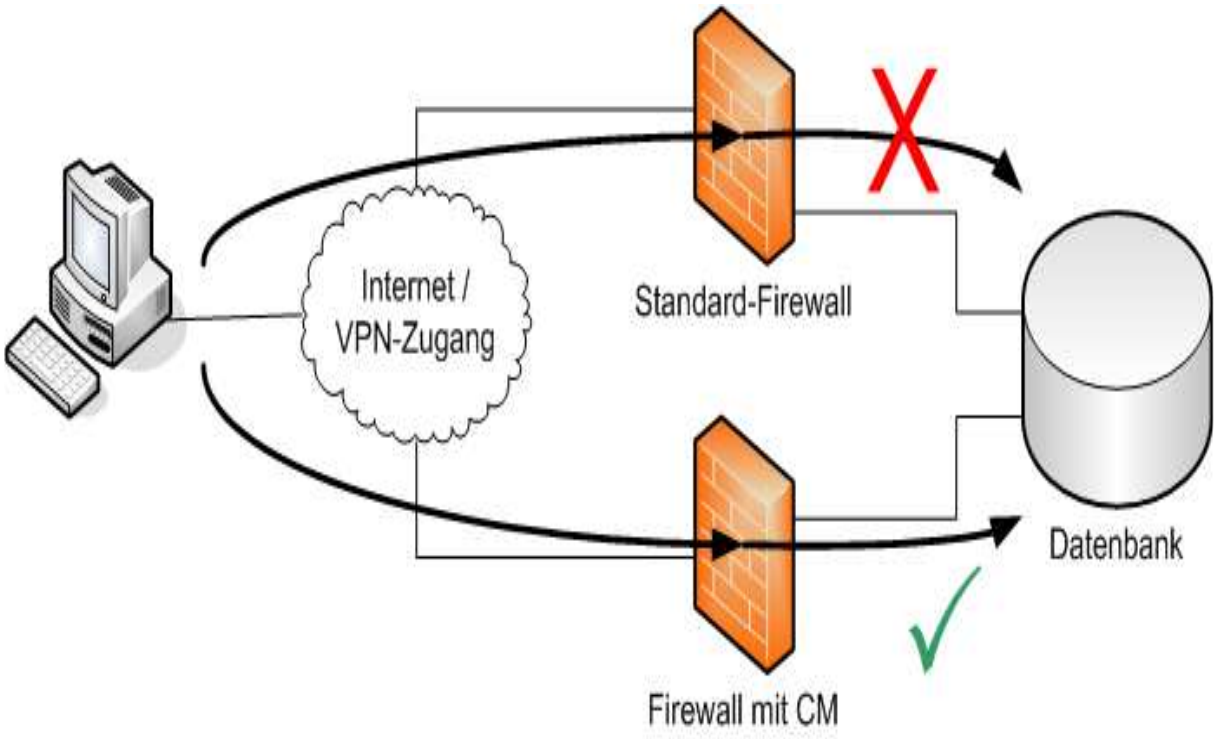

Abbildung 4: Verbindung über Firewalls

- Eine Abwandlung des vorigen Punktes ergibt sich durch den Einsatz eines VPN (Virtual Private Network). Die Mitarbeiter melden sich per Internet im Firmennetzwerk an. Durch die erhöhten Sicherheitsvorkehrungen sollen diese Mitarbeiter jedoch keinen direkten Zugriff auf andere Subnetze bekommen. Über den CM im VPN-Subnetz kann jedoch der Datenbankzugriff trotzdem gewährt werden.
- Eine Datenbankverbindung ist durch eine Firewall nicht möglich, wenn der Listener des Datenbankserver keinen *Direct Hand-off* unterstützt sondern nur über einen *Port-Redirect* Verbindungen über einen (vom Listener zur Laufzeit ausgewählten Port) aufbaut (abhängig von Betriebsystem). Es müssten somit alle Ports der Firewall geöffnet werden, da nicht vorausbestimmt werden kann, welcher Port tatsächlich verwendet werden wird. Dies wäre aber gleichbedeutend mit einer Deaktivierung der Firewall. Wird der CM hinter der Firewall betrieben, erfolgt der Port-Redirect zwischen Listener und CM, somit ist nach außen nur der definierte CM-Port zu öffnen.

Fazit

Mit dem Oracle Connection Manager steht seit langem ein Werkzeug zur Verfügung, das für die gestellten Aufgaben (Proxy, Zugriffskontrolle und Multiplexing) gut geeignet ist und sehr stabil läuft. Als Manko kann die Bündelung mit der Oracle Enterprise Edition gesehen werden. Jedoch sind bei kleinen Oracle-Umgebungen die genannten Features meist von untergeordnetem Interesse, so dass sich hier nicht unbedingt ein Nachteil ergibt. Es überwiegen die Vorteile beim Einsatz des CM, zumal sich der Administration auf den initialen Aufwand beschränkt.

Sollten Fragen zu diesem Artikel bestehen - ich stehe Ihnen gerne zur Verfügung:

Ihr Ansprechpartner:	Kontakt
Stefan Winkler	stefan.winkler@merlin-zwo.de
Geschäftsführer, **Systementwickler**	Tel.: 07052 / 933 666 Fax: 07052 / 933 670
merlin.zwo InfoDesign GmbH & Co. KG **Karmelstraße 9** **75378 Bad Liebenzell**	www.merlin-zwo.de
merlin.zwo InfoDesign GmbH & Co. KG **Taglöhnergärten 43** **76228 Karlsruhe**	