

DOAG 2010

```
select * from dual = 42
```

Dipl.-Inform. Frank Beutelschiess

Übersicht

- Kapitel 1 – Vorstellung
- Kapitel 2 – Sinn
- Kapitel 3 – Pakete
- Kapitel 4 – Verbindung
- Kapitel 5 – Proxy

Kapitel 1

- Kapitel 1 – Vorstellung
- Kapitel 2 – Sinn
- Kapitel 3 – Pakete
- Kapitel 4 – Verbindung
- Kapitel 5 – Proxy

Wer bin ich?

- Dipl.-Informatiker aus Leidenschaft
- 18 Jahre freiberuflich tätig
- 11 Jahre im Bereich Siebel und OracleDB
- Seit 3 Jahren BzYxS.com

Was macht BzYxS.com?

- Schulungen & Consulting
 - u.a. für die Oracle University
- Siebelumfeld
 - Konfigurationsreviews
 - Komponententuning
- Oracle Datenbankumfeld
 - Tuning, bevorzugt unter Verwendung der Tracefileanalyse mit QueryAdvisor



Kapitel 2

- Kapitel 1 – Vorstellung
- **Kapitel 2 – Sinn**
- Kapitel 3 – Pakete
- Kapitel 4 – Verbindung
- Kapitel 5 – Proxy

Worum geht es?

```
SQL> select * from dual;
```

```
D
```

```
-
```

```
X
```

```
SQL> select * from DUAL;
```

```
42
```

```
-----
```

```
1
```

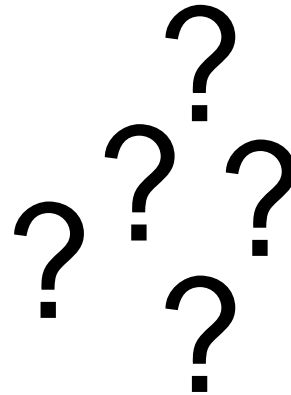
```
SQL> |
```

Worum geht es?

```
SQL> select count(*) as „42“ from dual;
```

```
         42  
-----  
         1
```

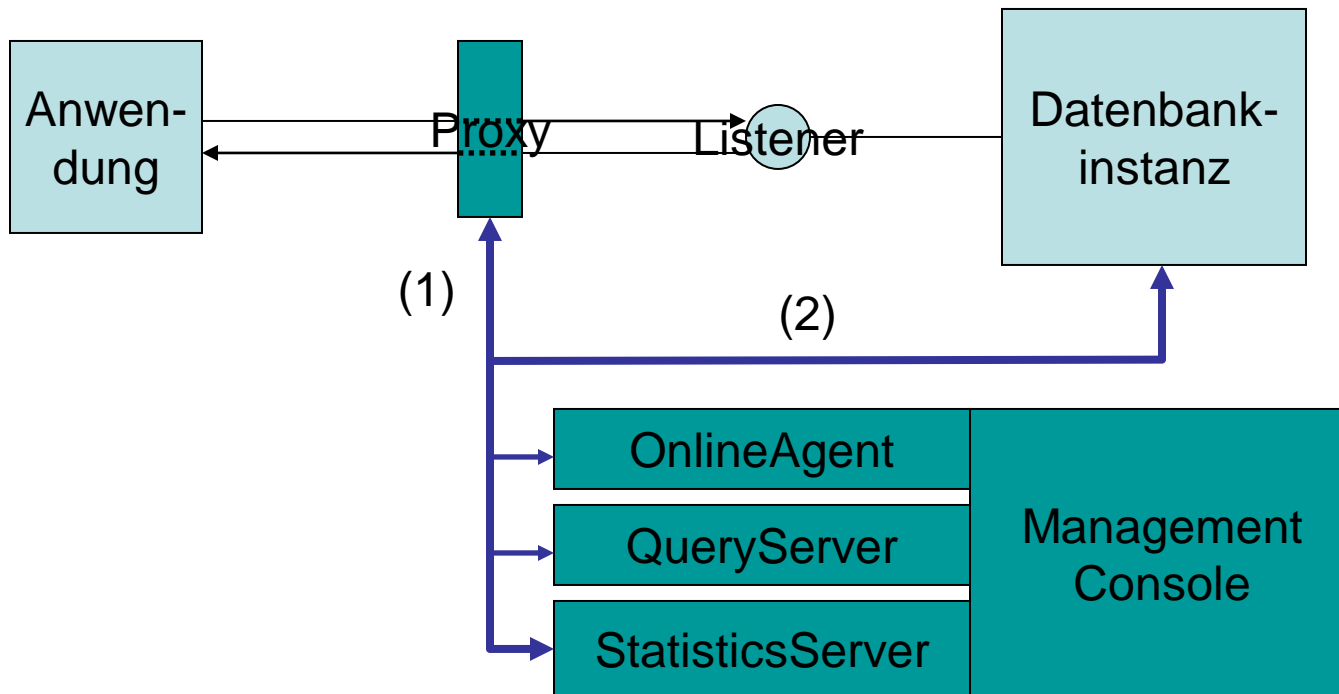
```
SQL>|
```



Was passiert?

- Manipulation des Datenaustauschs zwischen Client und Server
- Das SQL Kommando **select * from DUAL;** wurde auf dem Weg zum Server dynamisch durch das SQL Kommando **select count(*) as "42" from dual;** ersetzt.
- Die Ergebnisse wurden an den Client zurückgegeben.
- Der Anwender wundert sich.
- Die Datenbankinstanz ist ahnungslos.

Wie sieht dies praktisch aus?



Wie passiert es?

Synchrone Paketfolge



Der Client sendet eine Anfrage und wartet, bis die vollständige Antwort von dem Server eingetroffen ist.

In diesem Fall verursacht der Proxy sowohl auf dem Hin- als auch auf dem Rückweg eine geringe Verzögerung, welche sich nach meinen Laufzeitmessungen aber pro Zyklus deutlich unter <1ms bewegt.

Wie passiert es?

Asynchrone Paketfolge



Der Client sendet eine Anfrage, wartet aber nicht bis die vollständige Antwort vom Server eingetroffen ist.

Die Anzahl der maximal als Vorlauf gesendeten Blöcke wird im Rahmen des Verbindungsaufbaus festgelegt.

Solange die Auswertung $>$ Proxydelays ist gibt es nur einen konstanten Offset für die erste Anfrage ($\ll 1\text{ms}$).

Sinn/Unsinn des Ganzen

- Extraktion der SQL Kommandos zum Aufbau eines Nutzungsprofils (im Gegensatz zu Event 10046 keinerlei Belastung der Instanz und keine Problematik der Sessionzuordnung im Shared Server Umfeld, 100% InTime).
- Berechnung der effektiven Auswertungszeit innerhalb der Datenbank (Proxy muß auf der selben Hardware wie die Instanz liegen, 100% InTime).
- Austausch des gesendeten SQL Kommandos, d.h.
 - Automatisches Hinzufügen von Hints (Grenzen von Outlines/Baseplans bei zugekauften Anwendungen wie Siebel CRM).
 - Einführung von Bindvariablen bei einzelnen SQL Kommandos (Grenzen von Forced, Manipulation von Statistiken/Clusterfaktoren, ein typischer Fall wären einzelne SQL Queries bei Oracle BI).

Sinn/Unsinn des Ganzen

- Neben der Sammlung von Daten und der Optimierung von SQL Kommandos

Gibt es noch eine weitere Verwendungsmöglichkeit:

- Zugriffsschutz
durch das automatische Hinzufügen von zusätzlichen Bedingungen in dem Where-Teil des SQL Kommandos.

All dies geschieht 100% in Echtzeit (für alle die Neartime bevorzugen – gerne auch dies) und ermöglicht die schnelle Erkennung von Langläufern und Hinzunahme von Korrekturmustern.

Kapitel 3

- Kapitel 1 – Vorstellung
- Kapitel 2 – Sinn
- **Kapitel 3 – Pakete**
- Kapitel 4 – Verbindung
- Kapitel 5 – Proxy

Paketaufbau

Ein Paket besteht aus einem Header und einem Body. Die 8 identischen Bytes des Headers setzen sich zusammen aus:

- Paketlänge Header & Body 2 Byte
- Prüfsumme Paket 2 Byte
- Pakettyp 1 Byte
- <Reserviert> 1 Byte
- Prüfsumme Header 2 Byte

Die Prüfsumme berechnet sich in beiden Fällen aus dem 16 Bit Einerkomplement des 16 Bit Einerkomplements der 16 Bit Summe.

Pakettypen

Die 11 verschiedenen Pakettypen in 3 Gruppen eingeteilt werden:

- **Verbindung**

- Connect Type: 0x01
- Accept Type: 0x02
- Refuse Type: 0x04
- Redirect Type: 0x05

Pakettypen

- Austausch der von dem Sender benötigten minimalen Versionsnummer.
- Aktuelle Versionsnummer auf welcher der Verbindung zugrundeliegt.
- Transport Data Unit & Session Data Unit Größe.
- Anzahl der gesendeten Pakete bis eine Antwort des Servers erfolgen muss.
- Testbyte um Byte Order festzustellen (0x0001).
- Eigentliche Verbindungsdaten.
- Informationen über den neuen Port auf welchen die Verbindung umzieht.
- Grund warum eine Verbindung abgelehnt wurde.

Pakettypen

Connect Pakete enthalten u.a.:

```
(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=127.0.0.1)
(PORT=1521)) (CONNECT_DATA=
(SERVICE_NAME=test.bzyxs.com) (... <siehe TnsNames.ora>
```

Redirect Pakete enthalten u.a.:

```
ADDRESS=(PROTOCOL=tcp)(HOST=127.0.0.1)(PORT=4886)
```

Pakettypen

- **Datentransfer**

- Data Type: 0x06
- Null – Empty Data Type: 0x07

- Eigentlich zu übertragende Daten mit einem Hinweis bei Aufteilung.
- Sicherheitsrelevante Informationen.
- Testdaten zur serverseitigen Erkennung toter Verbindungen.

Pakettypen

- **Kontrollfluß**

- Abort Type: 0x09
 - Resend Packet Type: 0x0B
 - Marker Packet Type: 0x0C
 - Attention Type: 0x0D
 - Control Info Type: 0x0E
-
- Übermittlung protokollinterner Informationen.
 - Übertragung von CTRL-C synchron / asynchron.
 - Nochmalige Anforderung eines Paketes.

Theoretisch ist als Wertebereich des Pakettyps der Bereich von 0x01 ... 0x12 (einschließlich) zulässig.

Pakettypen

Eine detaillierte Beschreibung der einzelnen Pakettypen findet sich z.B. unter

MetaLink Doc ID: 1007807.6

SQL*Net Package Structure: NS Packet Header

Kapitel 4

- Kapitel 1 – Vorstellung
- Kapitel 2 – Sinn
- Kapitel 3 – Pakete
- **Kapitel 4 – Verbindung**
- Kapitel 5 – Proxy

Verbindungsaufbau

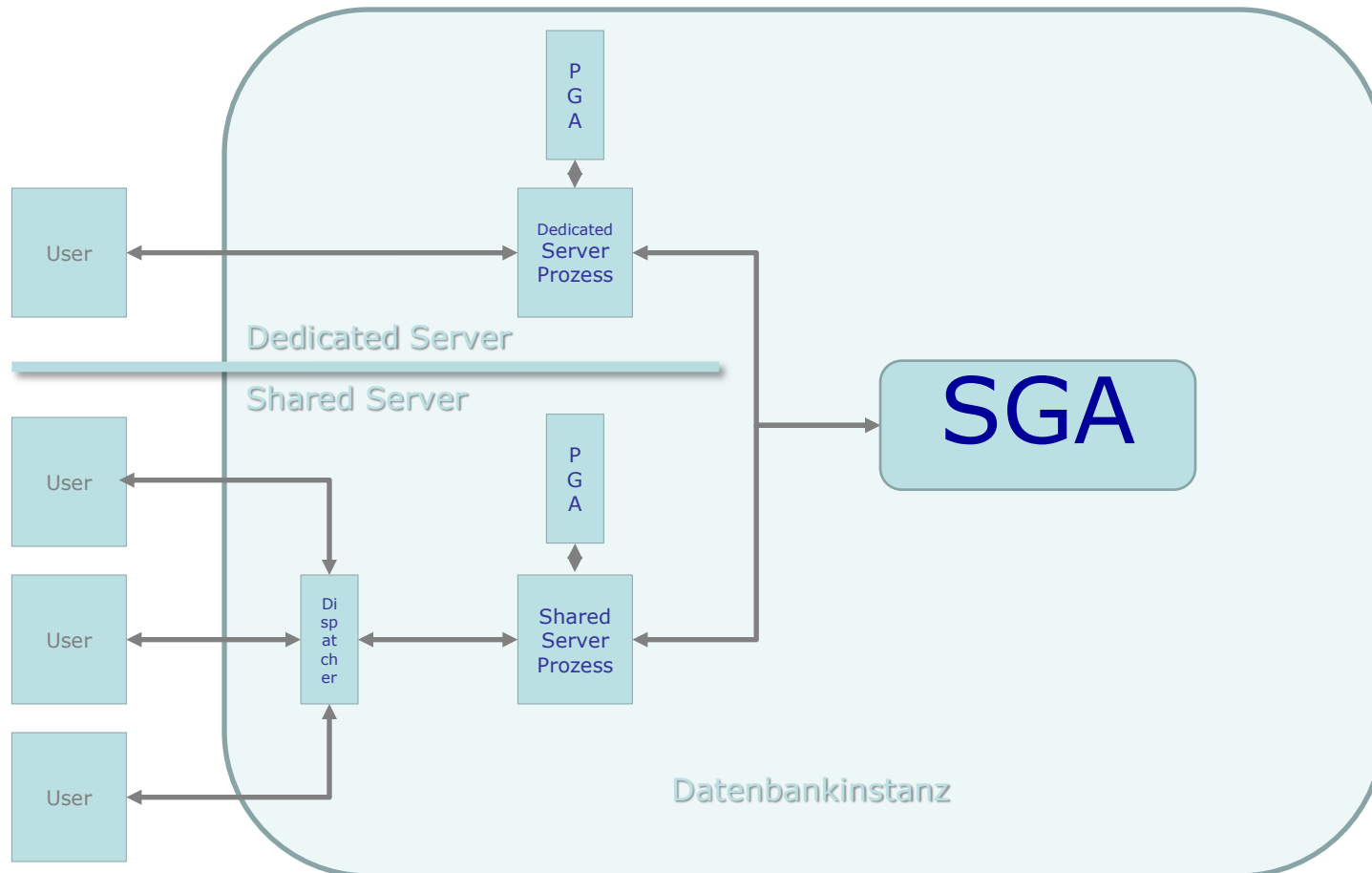
Dedicated & Shared Server

- Der Client sendet an den genannten Port an der spezifizierten Adresse ein Connect Paket (siehe TnsNames.ora).
- Der Listener muss nun entscheiden, ob für die gestellte Verbindungsanfrage ein eigener Hintergrundprozess erzeugt werden muss (Fall des Dedicated Servers) oder der bereits vorhandene Dispatcherprozess (Fall des Shared Servers) verwendet werden kann.
- Im Fall eines Shared Servers kann somit direkt ein Accept Paket zurückgeschickt werden.

Verbindungsaufbau

- Im Fall des Dedicated Servers muss nun zuerst ein Redirect Paket zurückgeschickt werden, um den Anfragenden den Weg zu seinem eigenen Hintergrundprozess zu weisen.
- Die weitere Kommunikation wie der Austausch von Datenpaketen erfolgt nach dem Redirect im Rahmen des Verbindungsaufbaus an identisch.
- Nachdem die neue Session initiiert wurde, kann die eigentliche Benutzeridentifikation, d.h. der Login nun in Form des Austauschs von Datenpaketen erfolgen.
- An dieser Stelle soll der Hinweis, dass im Falle eines unmöglichen oder fehlgeschlagenen Verbindungsaufbaus ein Refused Paket von dem Server an den Client gesendet wird nicht fehlen.

Verbindungsaufbau



Verbindungsaufbau

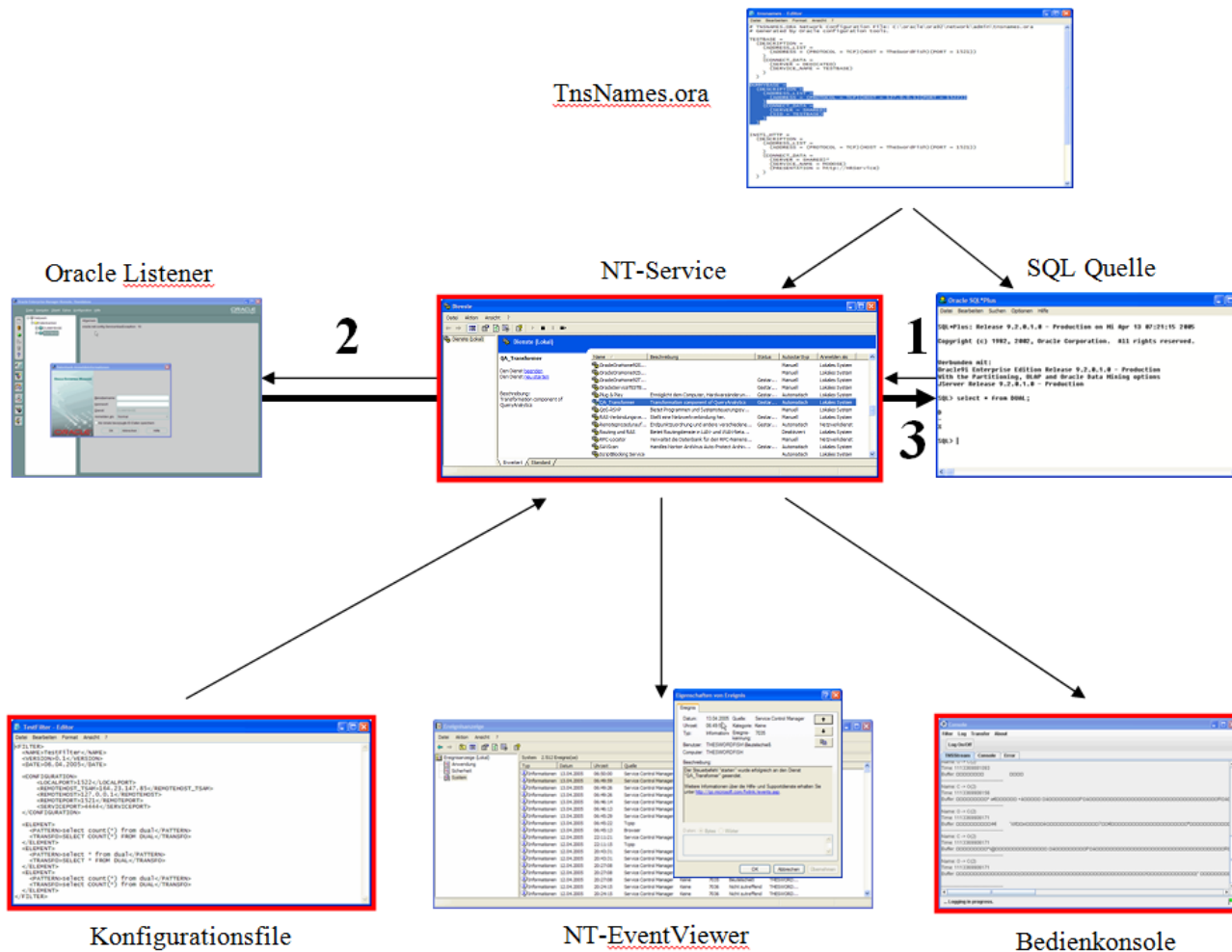
Sonderfall Bequeath

- Client und Server befinden sich auf der selben Hardware.
- Sehr kurze Zykluszeit durch Verwendung der von dem jeweils OS unterstützen Interprozeßkommunikation.
- Da kein Servicename angegeben wird, stehen weder eine Netzwerkadresse noch Port zur Verfügung. Eine Zuordnung zu einem Dispatcher oder Hintergrundprozess ist somit nicht möglich.
- Ein neuer Hintergrundprozess wird für den anfragenden Client gestartet und mit der SGA der in der globalen Systemvariablen ORACLE_SID referenzierten Datenbankinstanz verbunden.

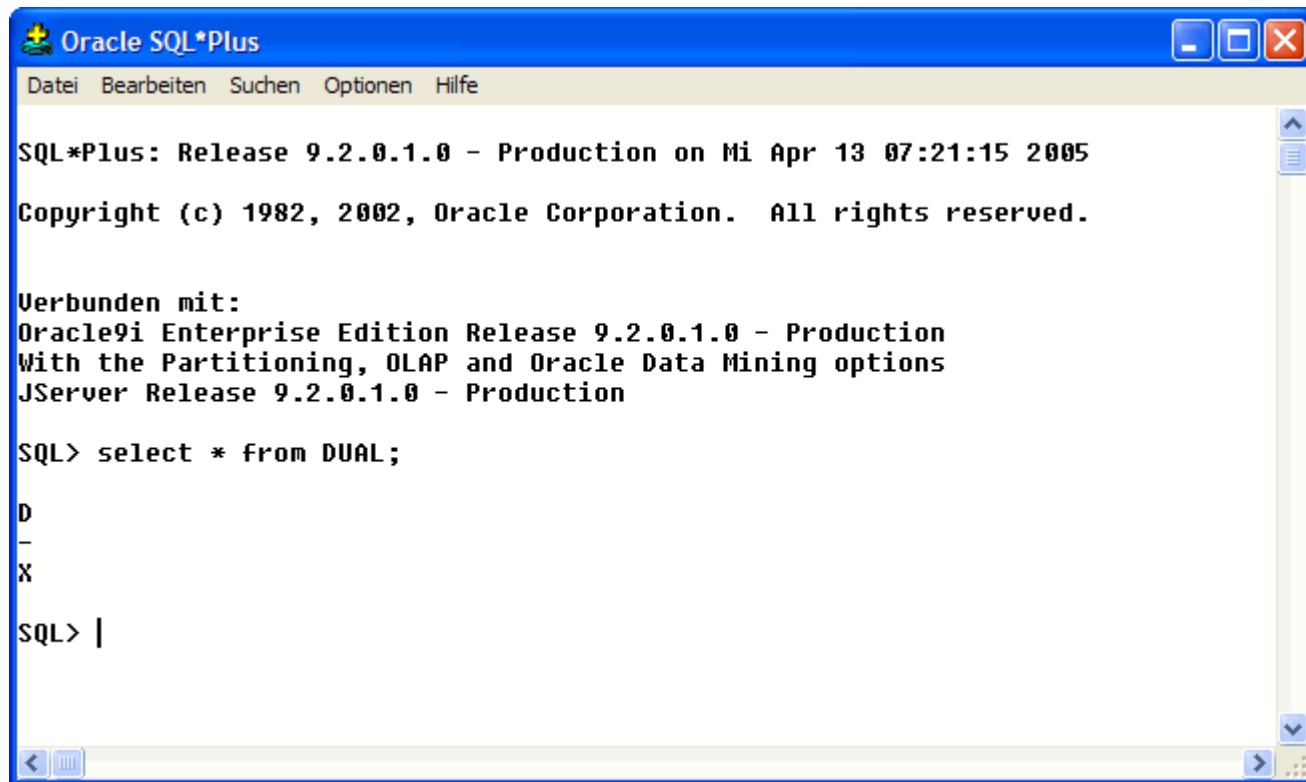
Kapitel 5

- Kapitel 1 – Vorstellung
- Kapitel 2 – Sinn
- Kapitel 3 – Pakete
- Kapitel 4 – Verbindung
- **Kapitel 5 – Proxy**

Proxy - Übersicht



Proxy – SQL PLus



```
Oracle SQL*Plus
Datei Bearbeiten Suchen Optionen Hilfe

SQL*Plus: Release 9.2.0.1.0 - Production on Mi Apr 13 07:21:15 2005

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

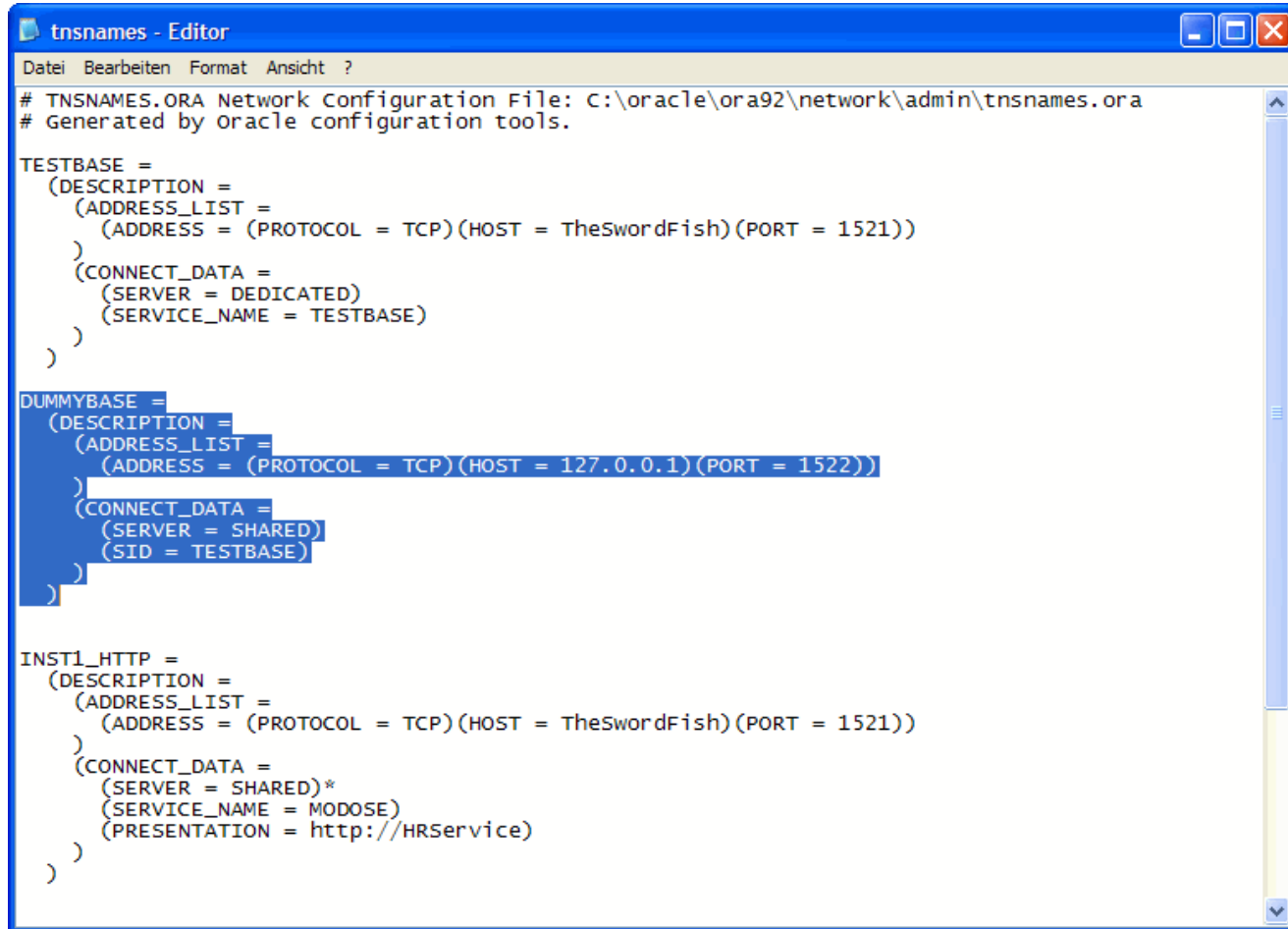
Verbunden mit:
Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production

SQL> select * from DUAL;

D
-
X

SQL> |
```

Proxy – TnsNames.ora



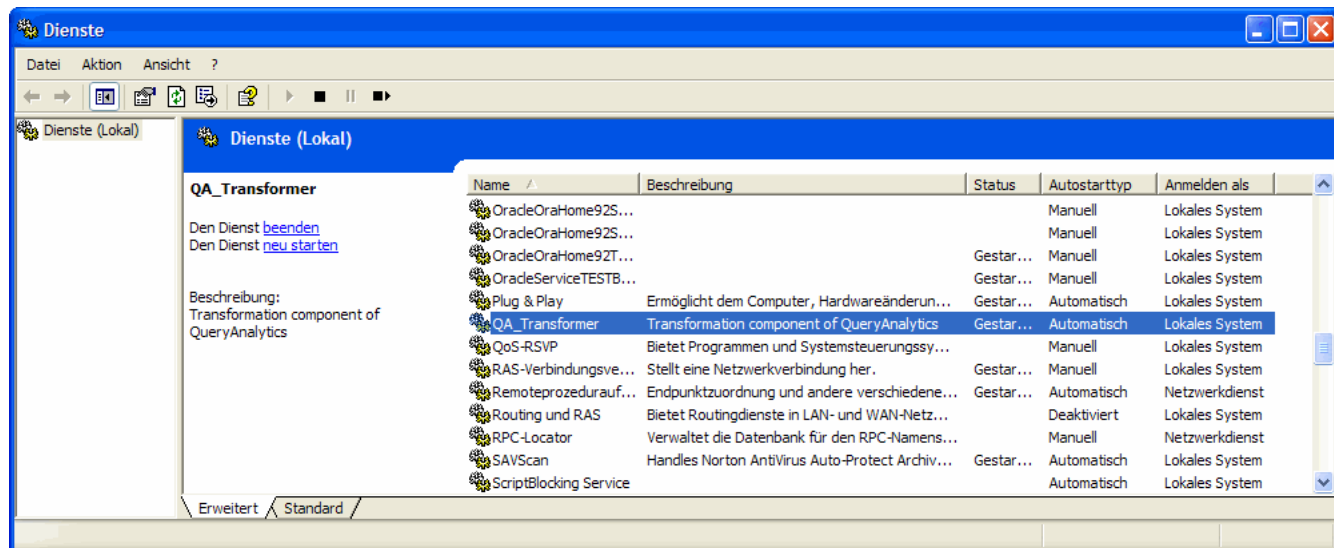
```
tnsnames - Editor
Datei Bearbeiten Format Ansicht ?
# TNSNAMES.ORA Network Configuration File: C:\oracle\ora92\network\admin\tnsnames.ora
# Generated by oracle configuration tools.

TESTBASE =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = TheswordFish)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = TESTBASE)
    )
  )

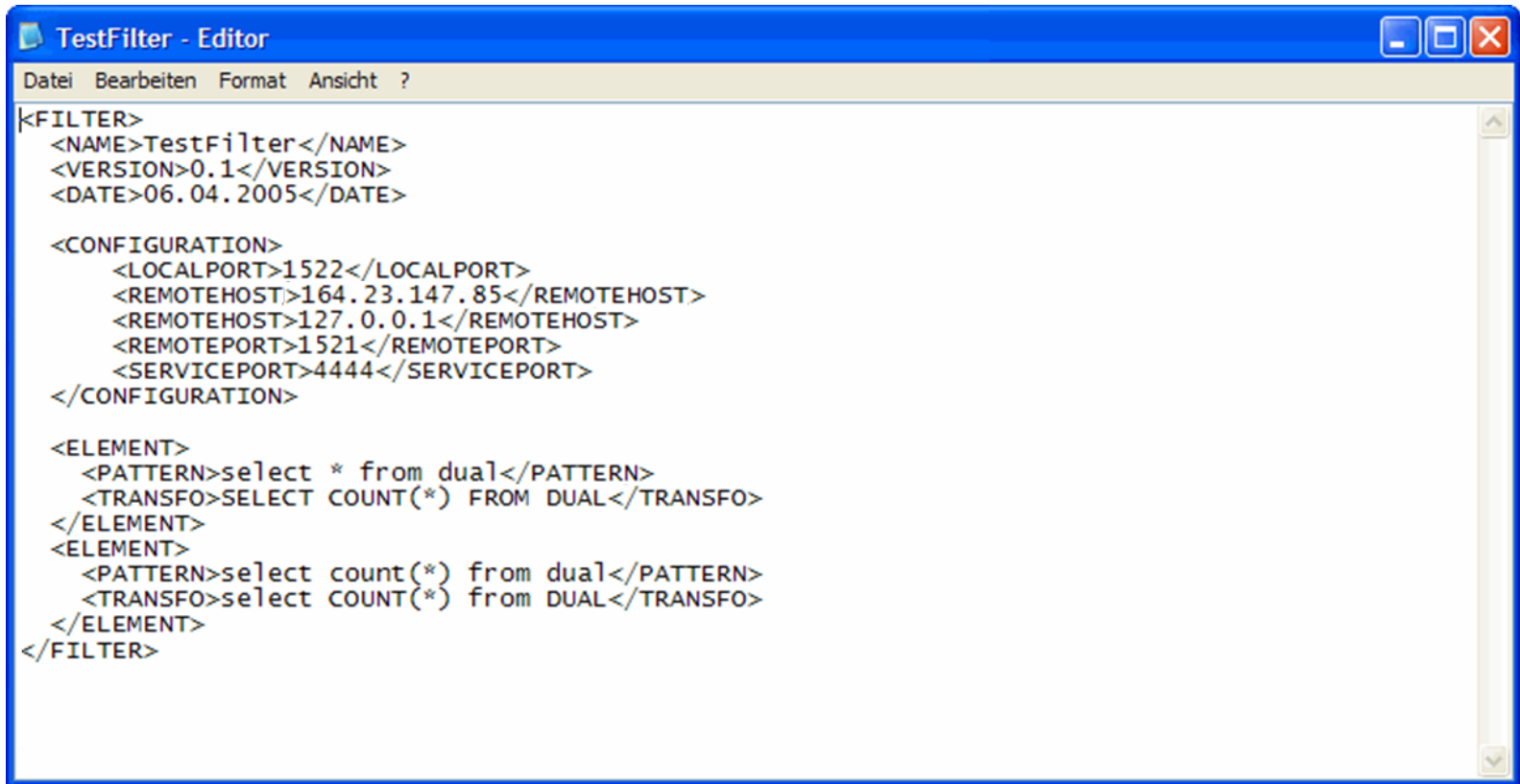
DUMMYBASE =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = 127.0.0.1)(PORT = 1522))
    )
    (CONNECT_DATA =
      (SERVER = SHARED)
      (SID = TESTBASE)
    )
  )

INST1_HTTP =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = TheswordFish)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = SHARED)*
      (SERVICE_NAME = MODOSE)
      (PRESENTATION = http://HRService)
    )
  )
```

Proxy – NT-Service



Proxy – Transformationsregeln



```
TestFilter - Editor
Datei Bearbeiten Format Ansicht ?
<FILTER>
  <NAME>TestFilter</NAME>
  <VERSION>0.1</VERSION>
  <DATE>06.04.2005</DATE>

  <CONFIGURATION>
    <LOCALPORT>1522</LOCALPORT>
    <REMOTEHOST>164.23.147.85</REMOTEHOST>
    <REMOTEHOST>127.0.0.1</REMOTEHOST>
    <REMOTEPORT>1521</REMOTEPORT>
    <SERVICEPORT>4444</SERVICEPORT>
  </CONFIGURATION>

  <ELEMENT>
    <PATTERN>select * from dual</PATTERN>
    <TRANSFO>SELECT COUNT(*) FROM DUAL</TRANSFO>
  </ELEMENT>
  <ELEMENT>
    <PATTERN>select count(*) from dual</PATTERN>
    <TRANSFO>select COUNT(*) from DUAL</TRANSFO>
  </ELEMENT>
</FILTER>
```

Analyse – Statische Teile

SQL Kommandos:

```
select * from dual
```

```
select count(*) from DUAL
```

```
select * from test where start > to_date('20040615','YYYYMMDD')
```

Generierter Baum:

```
„select “      +      „* from “      +      „dual“  
                +      +      „test...”  
                +      „count(*) drom DUAL“
```

Analyse – Variable Teile

```
select * from test where start > to_date('20040618','YYYYMMDD')  
select * from test where start > to_date('20040615','YYYYMMDD')
```

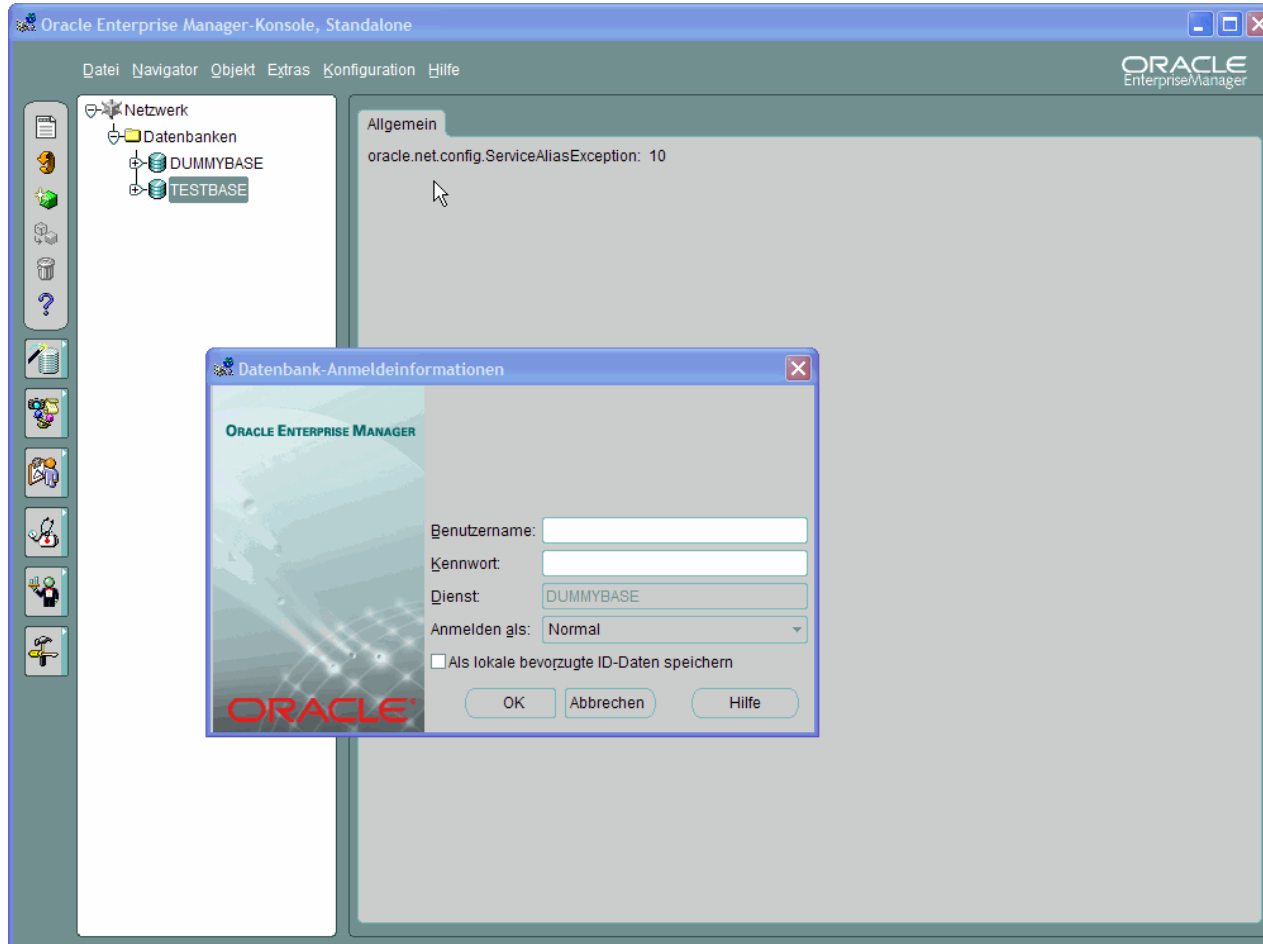
Isolation des Präfixes und Einführung von variablen Elementen um fehlende Bindvariablen bei der Analyse zu kompensieren.

```
„select * from test where start > to_date(„ +  
<String> + „','YYYYMMDD')“
```

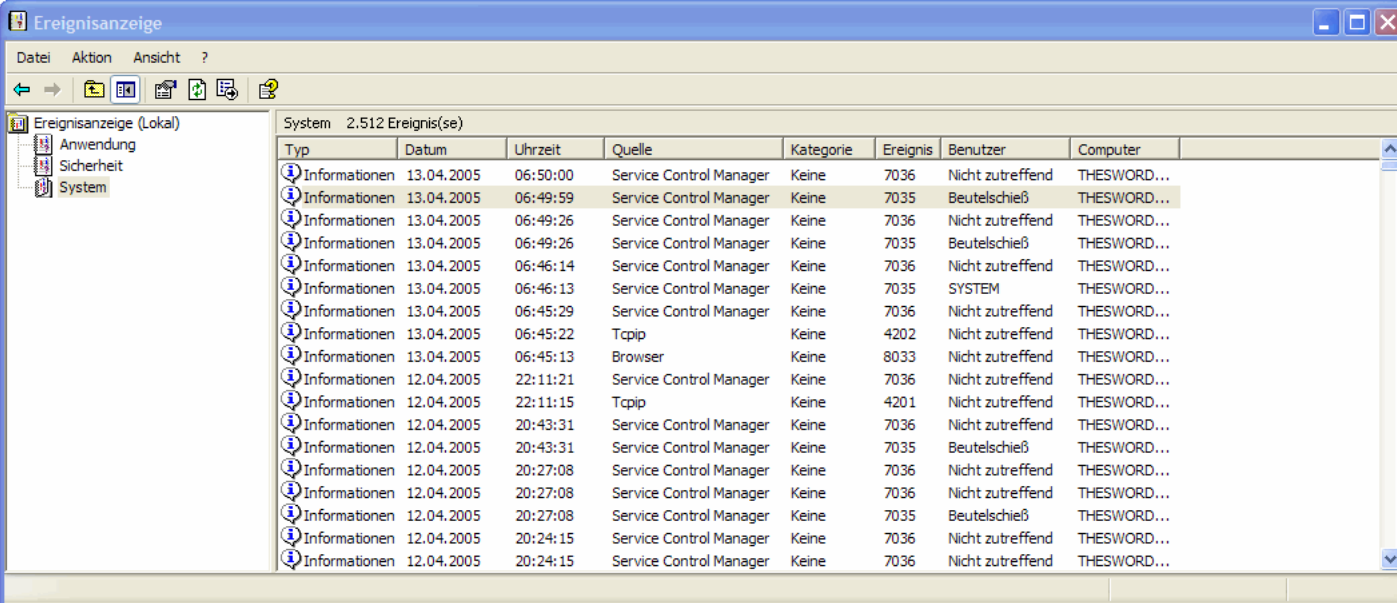
Es kann nun der folgende String eingesetzt werden:

```
„select * from test where start > to_date(:1,'YYYYMMDD')“  
<String> entspricht dabei dem Bindung für :1
```

Proxy – Instanz



Proxy – Eventlogger



The screenshot shows the Windows Event Viewer window titled 'Ereignisanzeige'. The left pane shows the tree view with 'System' selected under 'Sicherheit'. The right pane displays a table of 15 system events. The table has columns for Typ, Datum, Uhrzeit, Quelle, Kategorie, Ereignis, Benutzer, and Computer. The events are sorted by date and time, showing a sequence of events from April 12, 2005, to April 13, 2005.

Typ	Datum	Uhrzeit	Quelle	Kategorie	Ereignis	Benutzer	Computer
Informationen	13.04.2005	06:50:00	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	13.04.2005	06:49:59	Service Control Manager	Keine	7035	Beutelschieß	THESWORD...
Informationen	13.04.2005	06:49:26	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	13.04.2005	06:49:26	Service Control Manager	Keine	7035	Beutelschieß	THESWORD...
Informationen	13.04.2005	06:46:14	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	13.04.2005	06:46:13	Service Control Manager	Keine	7035	SYSTEM	THESWORD...
Informationen	13.04.2005	06:45:29	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	13.04.2005	06:45:22	Tcpip	Keine	4202	Nicht zutreffend	THESWORD...
Informationen	13.04.2005	06:45:13	Browser	Keine	8033	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	22:11:21	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	22:11:15	Tcpip	Keine	4201	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	20:43:31	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	20:43:31	Service Control Manager	Keine	7035	Beutelschieß	THESWORD...
Informationen	12.04.2005	20:27:08	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	20:27:08	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	20:27:08	Service Control Manager	Keine	7035	Beutelschieß	THESWORD...
Informationen	12.04.2005	20:24:15	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...
Informationen	12.04.2005	20:24:15	Service Control Manager	Keine	7036	Nicht zutreffend	THESWORD...

Literatur

1) SQL*Net Packet Structure: NS Packet Header

Doc ID 1007807.6

2) Examining Oracle Net Trace Files

Kevin Reardon, Oracle Corporation

3) SQL*Net Performance Tuning Utilizing Underlying Network Protocol

Gamini Bulumulle, Oracle Corporation

4) Display Filter Reference: Transparent Network Substrate Protocol

<http://www.wireshrk.org/docs/dfref/t/tns.html>