

Knowledge Base Article: Article 218 – Revision 2
How to connect BAI to a Remote SQL Server Database?

Date: January 11th, 2011

Last Update: January 21st, 2013 (see Section 2, C, 4)

Problem:

You want to create a Secured Data Source that will connect to a remote SQL SERVER using Windows Authentication.

Pre-requisites:

SQL Server Management Studio has to be installed on both machines corresponding to the same SQL Server version.

If you are using SQL Server Express, you can download and install Microsoft SQL Server Management StudioExpress (SSMSE) from Microsoft web site:

- SQL Server 2005: <http://www.microsoft.com/downloads/details.aspx?FamilyID=5D76230D-580D-4874-8C7D-93491A29DB15&displaylang=en>
- SQL Server 2008: <http://www.microsoft.com/downloads/details.aspx?FamilyID=08e52ac2-1d62-45f6-9a4a-4b76a8564a2b&displaylang=en>

Solution:

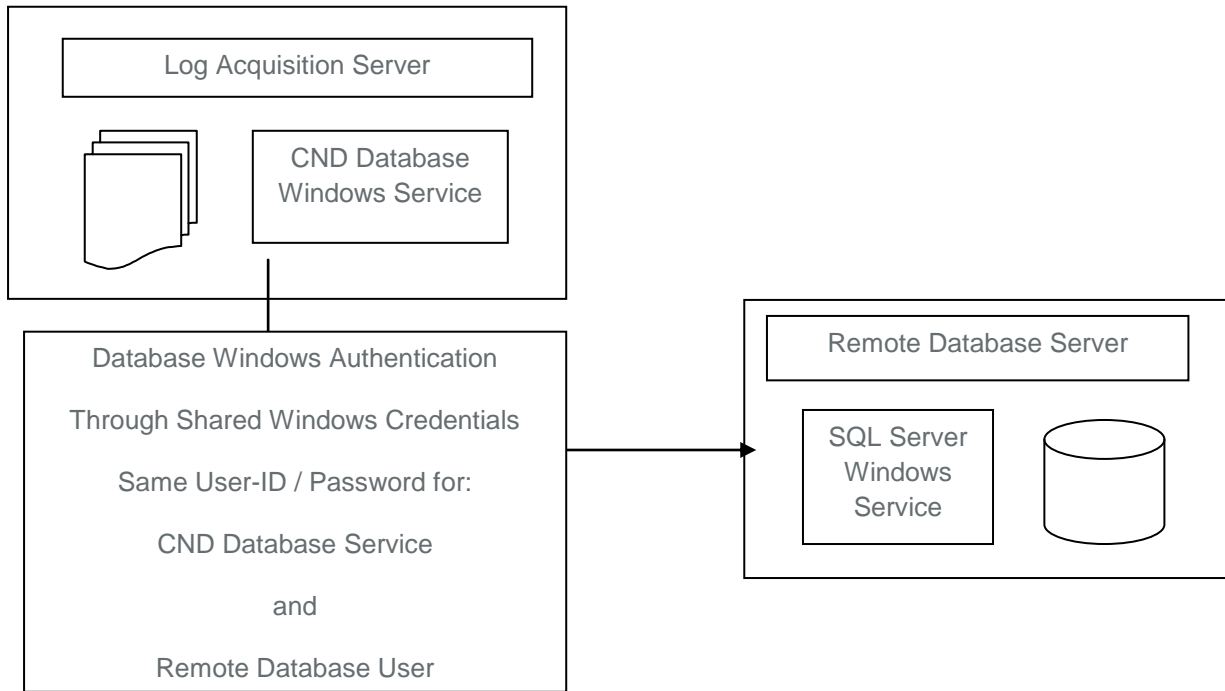
Follow the steps below

Table of Contents

Architecture.....	2
Section 1: Configure the Remote SQL Server Machine.....	3
A. Create a Custom User	3
B. Map this user to the database	3
C. Test the Database Configuration	4
D. Prepare the SQL Server for Remote Connections	6
Section 2: Prepare the Click &DECiDE-BAI Machine	11
A. Create Custom User	11
B. Test the Remote Database Connection	11
C. Configure BAI Services.....	12

Architecture

Net Report uses Windows Authentication to connect to the database. As a consequence you need to provide a Windows User credential that is able to log into your SQL Server Database.



Section 1: Configure the Remote SQL Server Machine

A. Create a Custom User

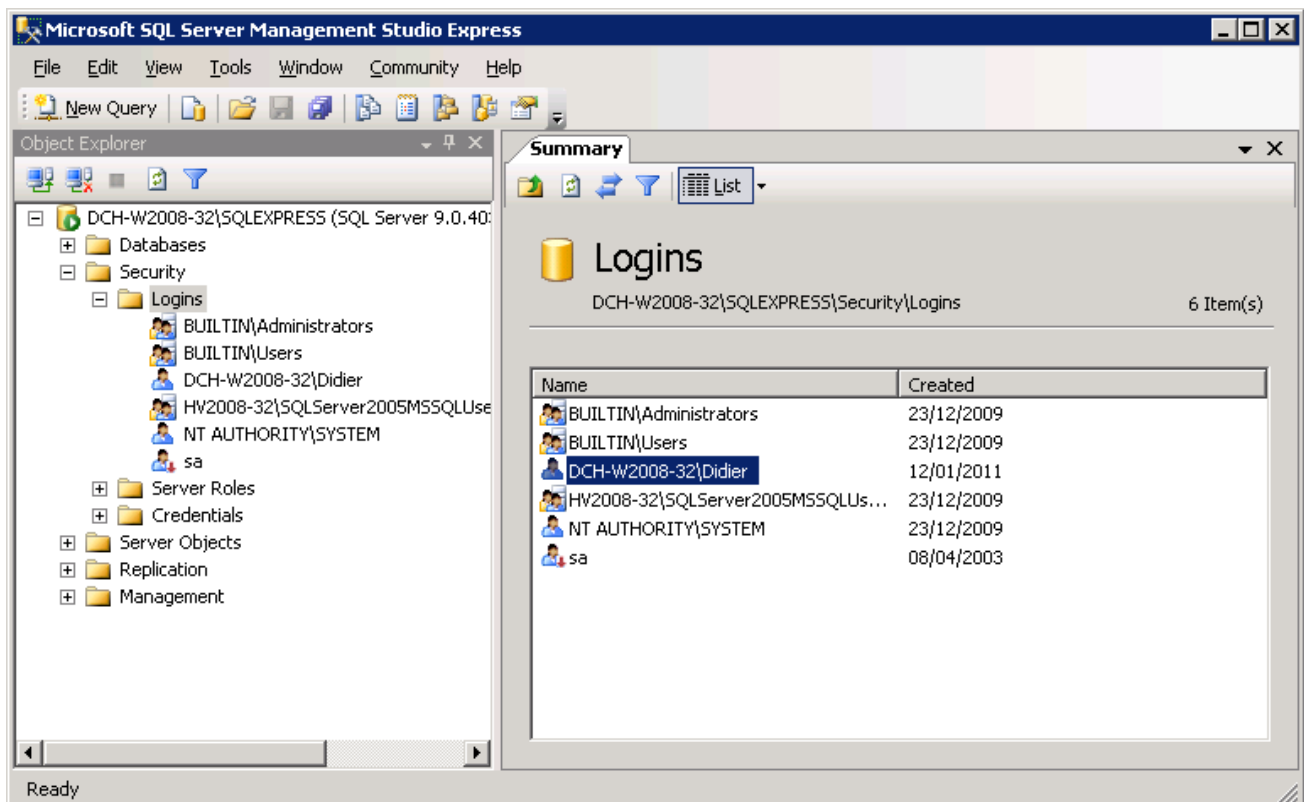
1. Connect to the machine where the remote SQL Server is running
2. Create a User that will be used by Click & DECiDE-BAI to connect to your database.

WARNING: You must choose a password that complies with the password policy of both the remote SQL Server machine and the machine that runs Click & DECiDE-BAI.

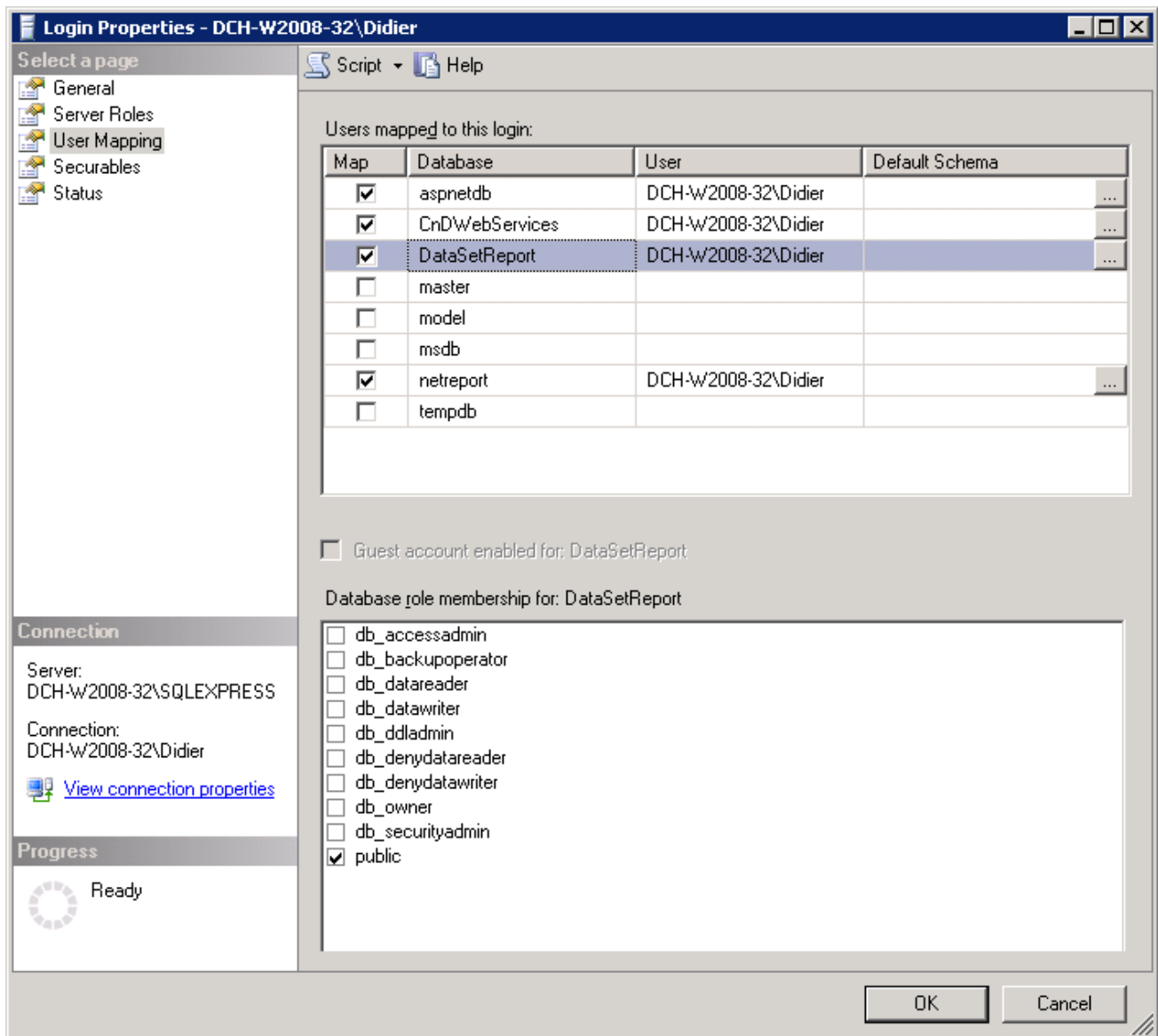
B. Map this user to the database

1. Connect to the machine where the SQL Server is running and launch the SQL Server Management Studio. (Run As Administrator)

From the Security node, add the user created below as a login



2. Then, right click on the just created login, select Properties and User Mapping to set rights to the login on the database(s) you want to access.



3. Click **OK**.

C. Test the Database Configuration

1. Open a Command Prompt: **Start>All Programs>Accessories>Command Prompt**.
2. Go to the following directory (with the CD command):

For SQL Server 2005

```
"C:\Program Files\Microsoft SQL Server\90\Tools\Binn\VSShell\Common7\IDE"
```

For SQL Server 2008 32 bits

```
"C:\Program Files\Microsoft SQL Server\100\Tools\Binn\VSShell\Common7\IDE"
```

For SQL Server 2008 64 bits

"C:\Program Files (x86)\Microsoft SQLServer\100\Tools\Binn\VSShell\Common7\IDE"

And enter the command:

For SQL Server 2005:

```
>runas /user:<domain>\<user>ssmsee.exe
```

Or, if no domain is defined:

```
>runas /user:<computername>\<user>ssmsee.exe
```

For SQL Server 2008:

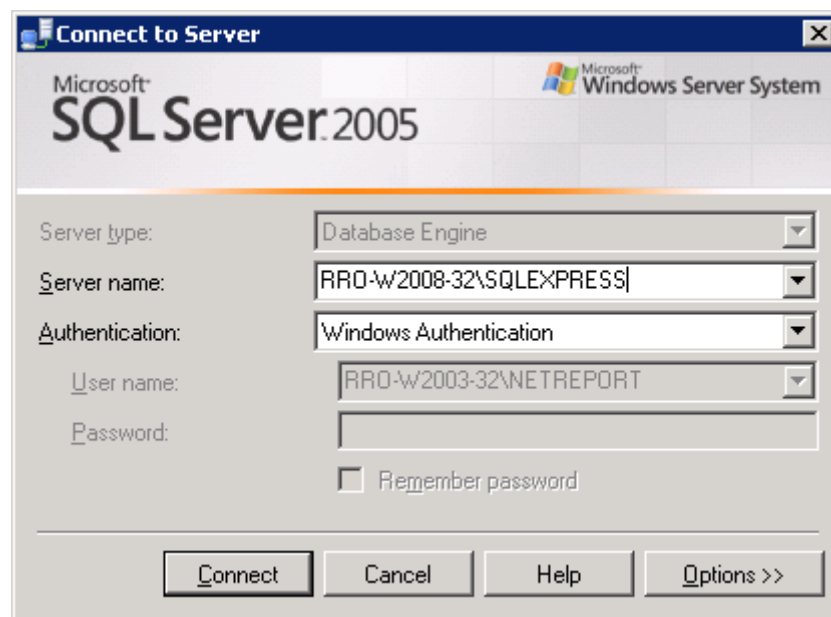
```
>runas /user:<domain>\<user>ssms.exe
```

Or, if no domain is defined:

```
>runas /user:<computername>\<user>ssms.exe
```

Replace **<domain>** by your domain name, **<user>** by the user name created in step 2. (if no domain is defined, just enter the **<computername>\<user>**.)

3. Type **<ENTER>**.
4. Enter the password of the user created in step 2.
5. Type **<ENTER>**.
6. Enter the **Server Name (<machine>\<instance>)** of the SQL Server you want to connect to.



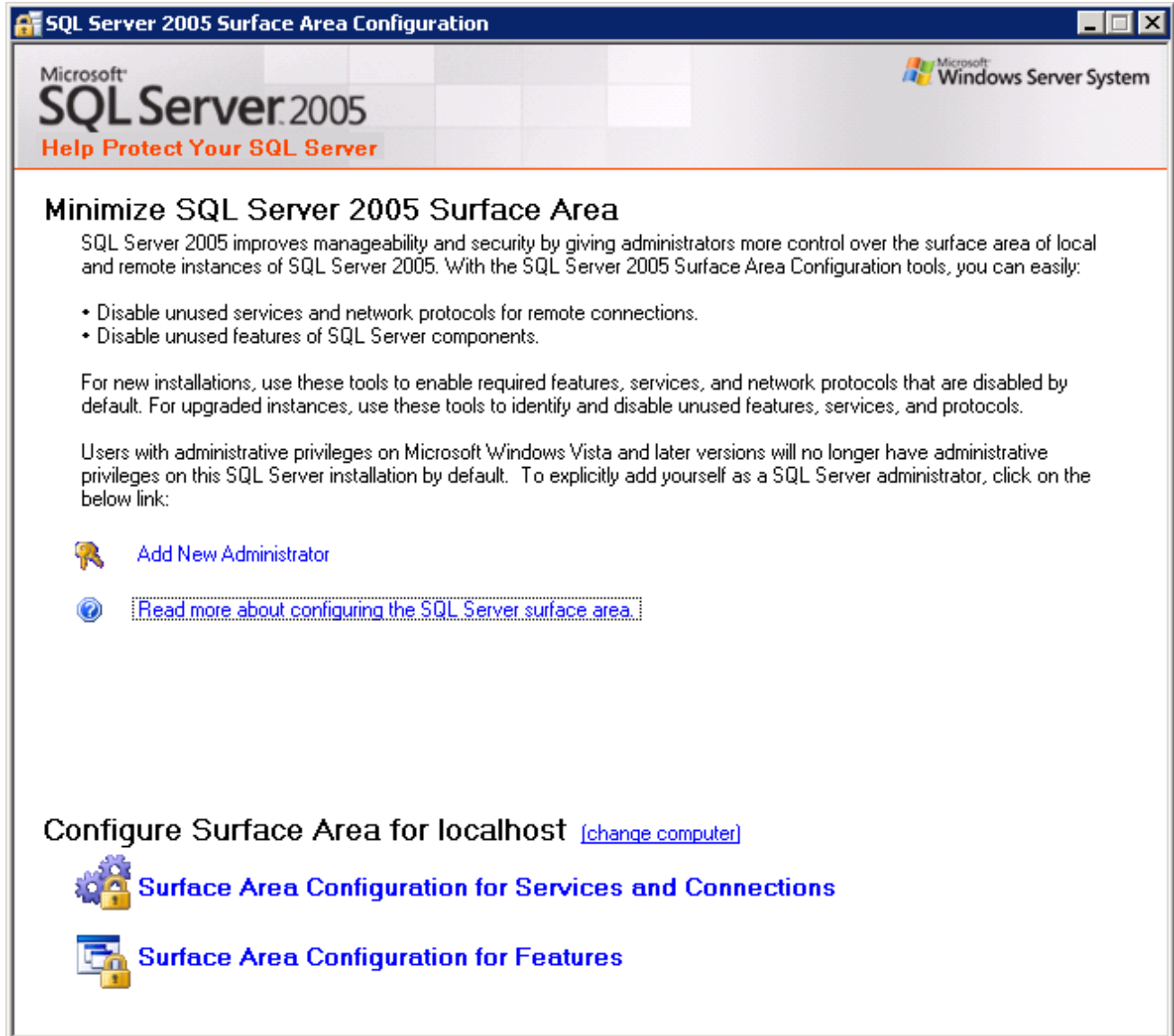
7. Click **Connect**.
8. You should be connected to the remote SQL Server and access to the database.
9. Note that a Firewall on the remote Server can forbid the connection. Change the Firewall status if needed.



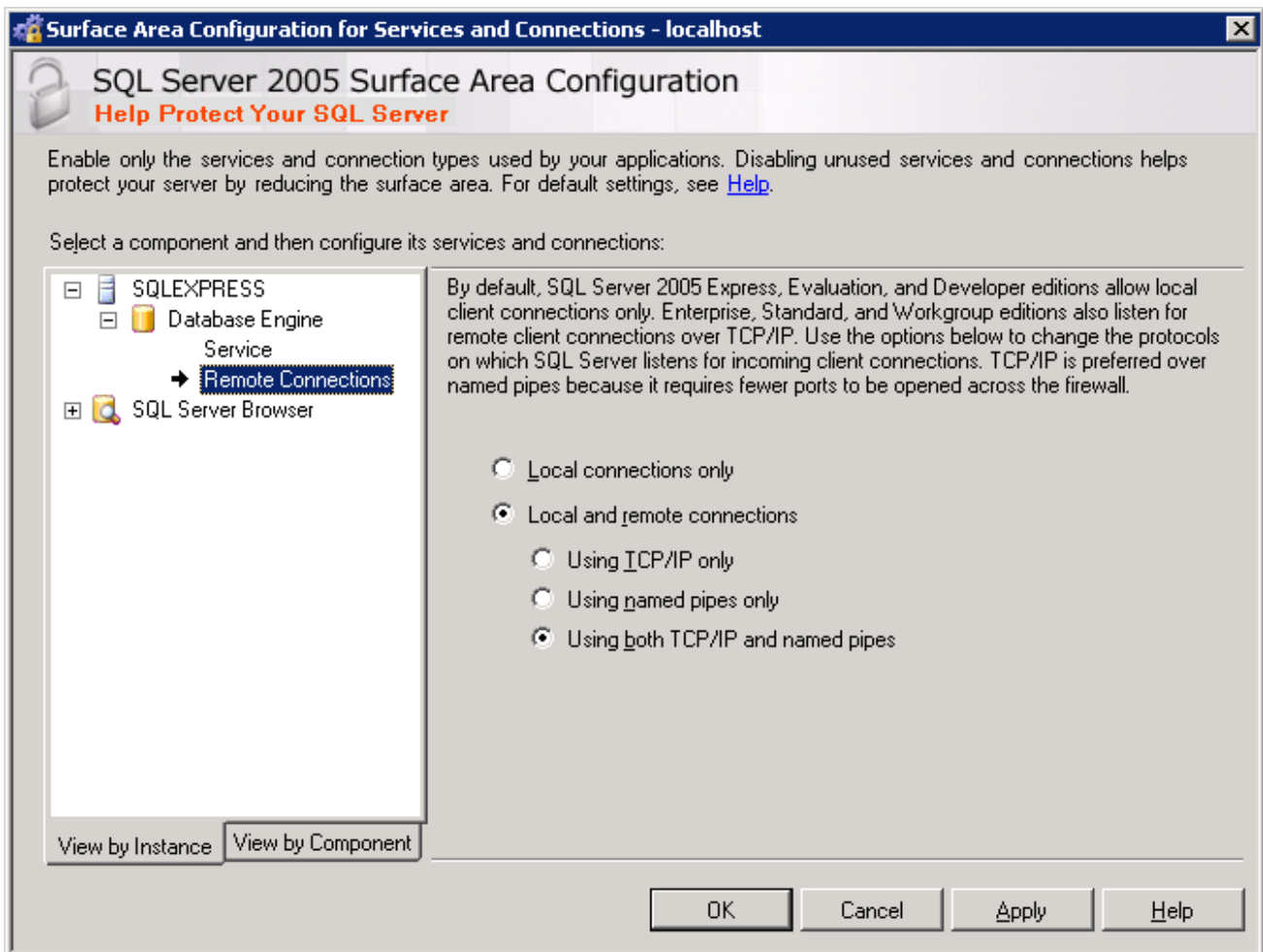
D. Prepare the SQL Server for Remote Connections

1. Verify that the SQL Server is accepting remote connection.

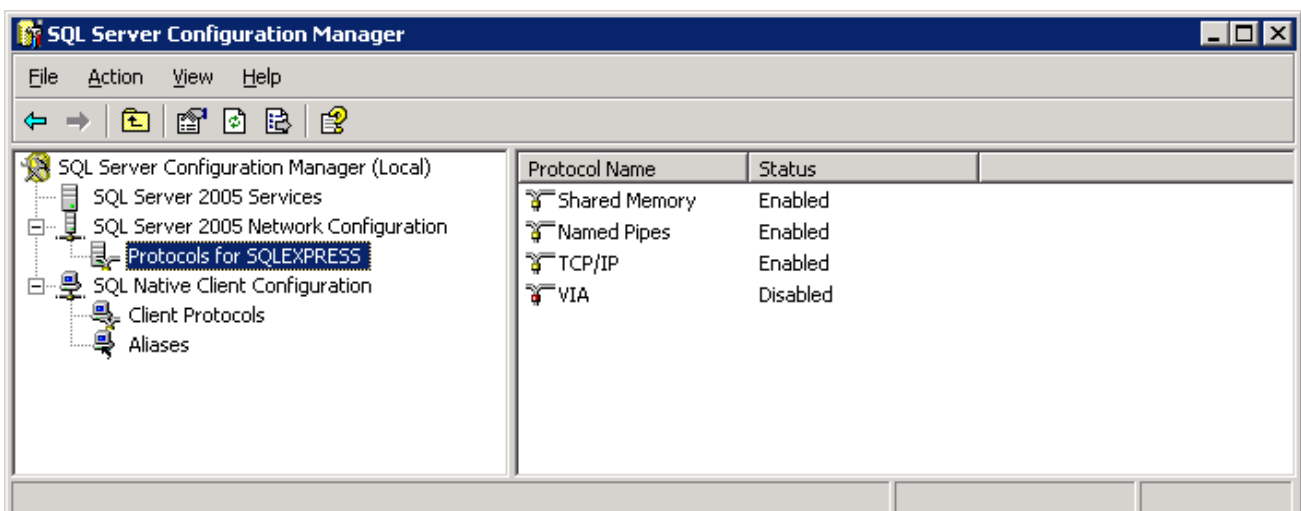
For SQL Server 2005, start the SQL Server Surface Area Configuration from All Programs / Microsoft SQL Server 2005 / Configuration Tools.



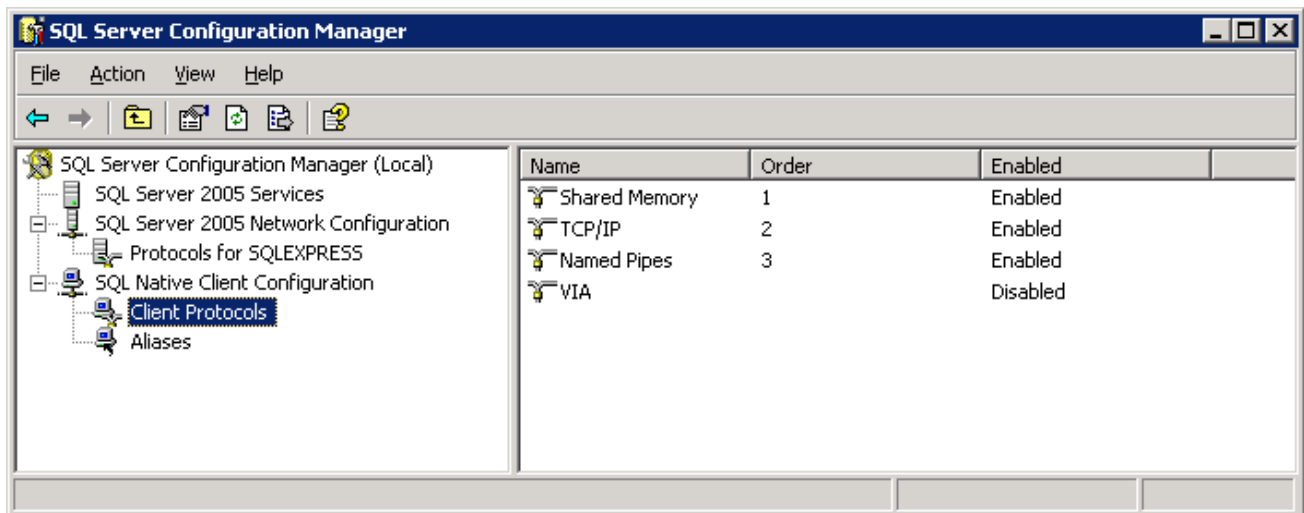
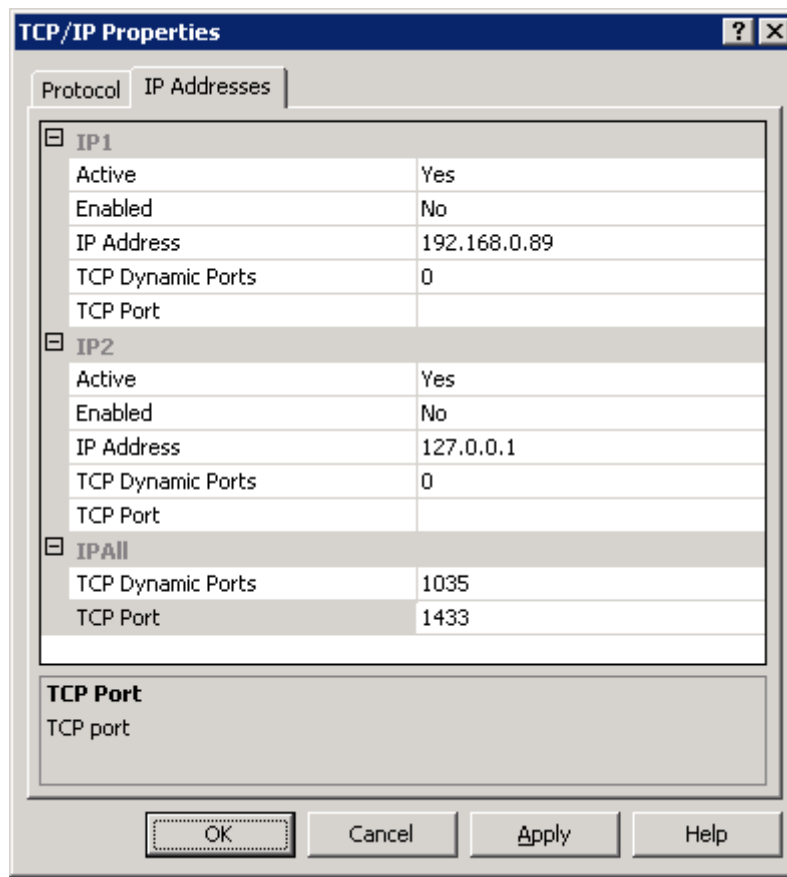
Then click on Surface Area Configuration for Services and Connections and check that remote connections are allowed as below:



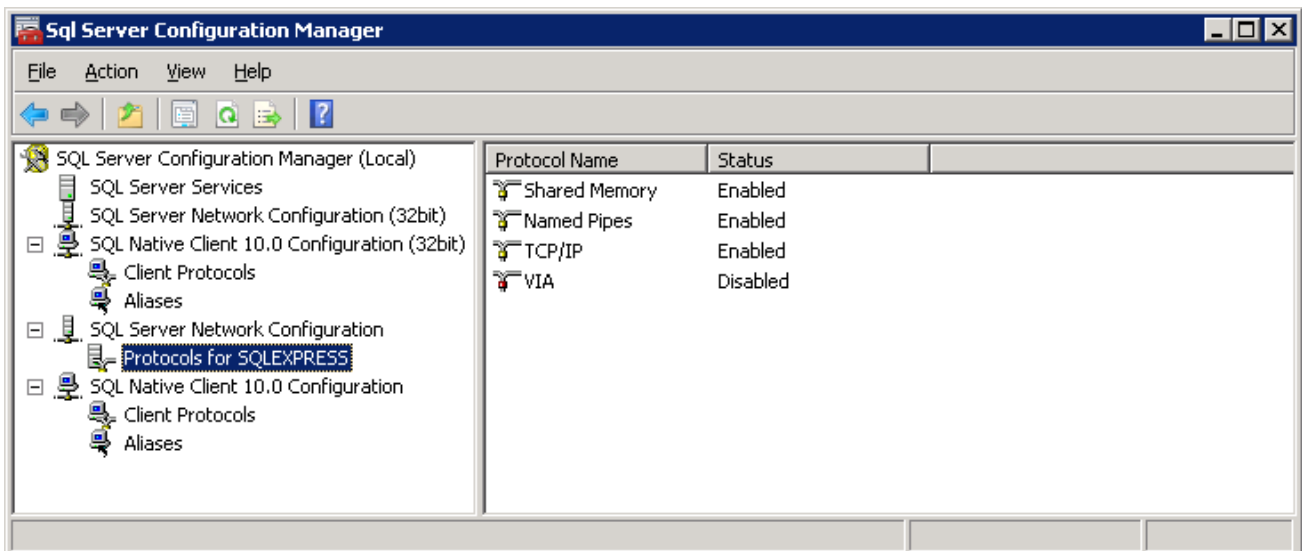
Then start the SQL Server Configuration Manager from All Programs / Microsoft SQL Server 2005 / Configuration Tools and check protocols are enabled as below:



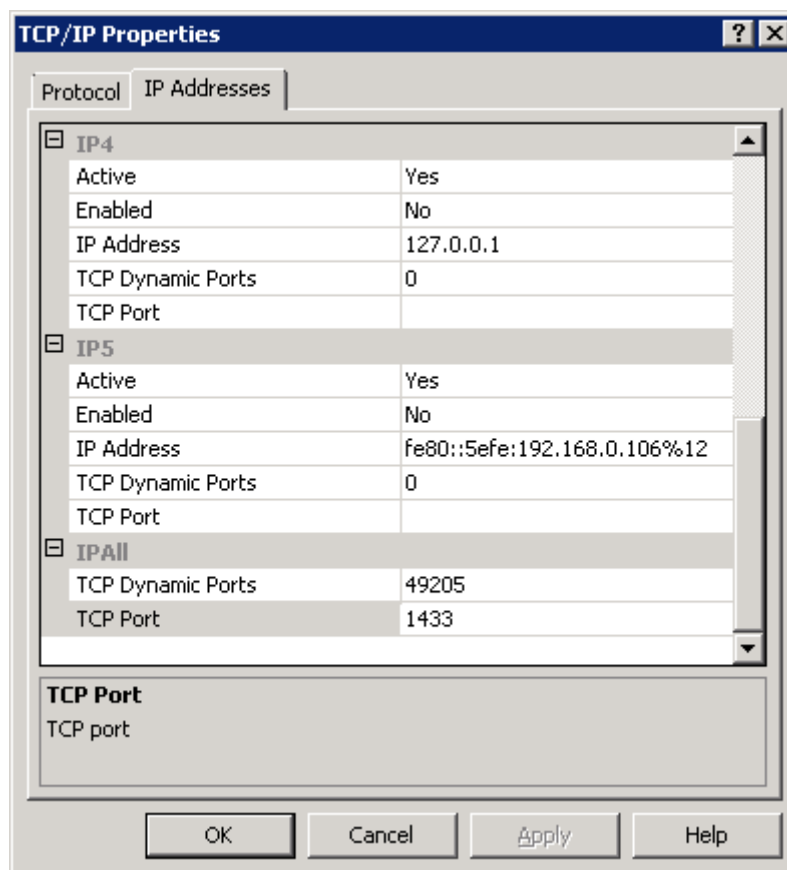
Then right click on TCP/IP properties and set a fixed TCP port for IPAll at 1433 as below:



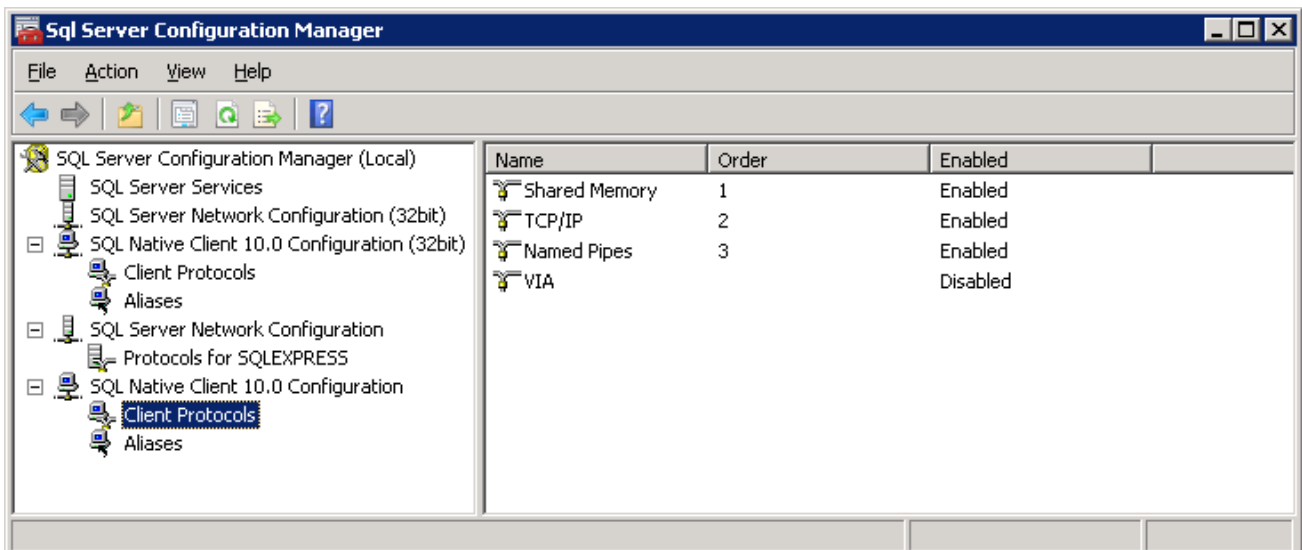
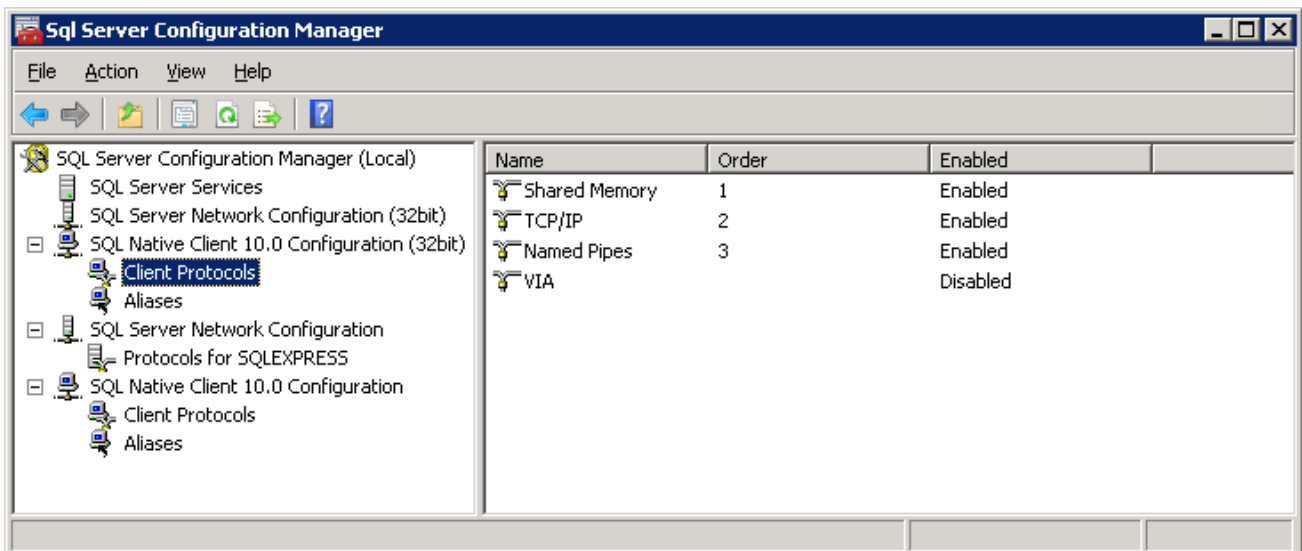
For SQL Server 2008, start the SQL Server Configuration Manager from All Programs / Microsoft SQL Server 2008 / Configuration Tools and check protocols are enabled as below:



Then right click on TCP/IP properties and set a fixed TCP port for IPAll at 1433 as below:



The following window will appear only in 64 bits version.



2. Verify that no FIREWALL setting is blocking the communication between the two machines.

- SQL Server: TCP 1433
- SQL Browser: UDP 1434

Section 2: Prepare the Click &DECiDE-BAI Machine

A. Create Custom User

1. Connect to the machine where Click &DECiDE-BAI is installed.
2. Create a User that will be used by Click &DECiDE-BAI to connect to your database.
 - a. This user must have the same login and password as the one created in section 1.A.
 - b. If the machine is located in a domain, this user must belong to the domain.
3. Grant the **log on as a batch job** and **log on as a service** rights to this user.

From **Administrative Tools**, start the **Local Security Policy**.

Then, in **Local Policies / User Rights Assignment**, add this user to entries below:

- Log on as a batch job
 - Log on as a service
4. Give full access to this user to the NSI directory located by default at:
C:\Program Files\Click and DECiDE\NSI.
 5. Open a Command Prompt and enter the following command:

```
>C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe -ga<domain>\<user>
```

Or, if no domain is defined:

```
>C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe -ga<computer name>\<user>
```

Replace **<domain>** by your domain name, **<user>** by the user name created in step 2(if no domain is defined, just enter the <computer name>\<user>.)

6. Type<ENTER>.

B. Test the Remote Database Connection

Do the same test as you did on the remote machine from this machine. (Section 1, Step C).

C. Configure BAI Services

If you only have the Click&DECiDE BAI software and not the Click&DECiDE NSI software, you need to download a file at: (**applicable only for version 10.02.01**)

<http://support.clickndecide.com/downloads/patches/faq218.zip>

The unzip the content into a c:\temp directory, then you can run the DVSqlServerConfig.exe program according to the next step:

1. Open a Command Prompt and enter the following command:

```
DVSqlServerConfig.exe -u <domain>\<user> -p <password> -s "DataSelect Connect Service;d7VEMService;DataSet Web Objects Manager Service;NRFilterEngine" -d "d7LEOService;D7TaskManager;DataSet Server Helpers Service;DataSet Web Objects Manager Service;d7vnrt;NRExecutor"
```

Or, if no domain is defined:

```
DVSqlServerConfig.exe -u <computer name>\<user> -p <password> -s "DataSelect Connect Service;d7VEMService;DataSet Web Objects Manager Service;NRFilterEngine" -d "d7LEOService;D7TaskManager;DataSet Server Helpers Service;DataSet Web Objects Manager Service;d7vnrt;NRExecutor"
```

Replace **<domain>** by your domain name, **<user>** by the user name created in section 2 (if no domain is defined, just enter the **<computer name>\<user>**).

2. Type**<ENTER>**

3. Stop and Restart the NSI and BAI Services using the **stopsvc.bat** and **startsvc.bat** located in:

```
C:\Program Files\Click and DECiDE\BAI
```

And

```
C:\Program Files\Click and DECiDE\NSI
```

4. When using the Builder project files (*.wfv) through the Web Portal, make sure that the directory containing these project files has the read rights for the specified user created in Section 2. If it is not the case you could get an error message such as "Error when opening the projects - Access denied".