

# Lesson 5 Transcript: Client Connectivity

## Slide 1: Cover

Welcome to lesson 5 of the DB2 on Campus Series. Today we are going to talk about client connectivity. My name is Raul Chong, and I'm the DB2 on Campus Program Manager.

## Slide 2: Agenda

This is the agenda for today.

## Slide 3 DB2 Directories

And we will start with DB2 Directories.

## Slide 4: DB2 Directories

What are DB2 directories? These are binary files where we store information about databases that we can connect to. There are four directories. The first one is called the *system database directory*, and you can think of it as the table of contents of a book. So it will list all the databases that you can connect to, whether they are local or remote. Then we have a *local database directory*. This will list the databases that you can connect to locally. That means, the database is in the same server where you are issuing the commands. Then there's the *node directory*. It's a directory that is used to store information about, let's say, if you are using TCP/IP, you'll have the IP addresses and port numbers. And then we have, finally, the *DCS directory*, which will not be covered in this course, but it is used when you have DB2 Connect, which is another software product that is required to connect to the mainframe. That means DB2 for Z/OS, or to connect to what used to be called AS400, for DB2 for i5/OS.

## Demo

So let me show you how we can take a look at these directories. So, if I start a DB2 command window, or if you are on Linux it could be your Linux shell, you could issue the command **db2 list db directory**. And this will show me, from this computer, which databases I can connect to. Now, these are all the databases I can connect to. And we have one database; the name of the database is TOOLSDB. This is the database alias, which is exactly the same. And when we see here it says "Indirect", it means that it's a local database. So that means the database resides on this computer, and it resides starting from, or is in drive C.

If we look at another entry here we see this other entry for database SALESDB, and the alias for this one is also SALESDB. And now this one says "Remote", so this means this database is not in this server where I'm issuing the command, but it's in some other server. And it also gives me a node, a node name, here, with the name of MYNODE. So this is basically a hyperlink to the node directory, and when we looked at the node directory, you will see that we have to look at this entry and we will see the information on connectivity for that particular server. Then we have other entries, another one for the RESEARCH database, which is also a local database because it says "Indirect" there. The MYDB database is also local, also because it says "Indirect". And SAMPLE is also local. So, now, that's the system

database directory, which, like I said before, is like a table of contents. Now, if you want to look at the local database directory, I just have to put **db2 list db directory on** <the drive where it is>, and that information I got from the entry before, right? If it's Linux you'll probably have to put a path instead of a drive. So now, if I issue this command, I will get similar information, but, as you can see, the SALESDB directory doesn't appear. So there are four entries instead of five, because it's showing me only the four local databases, the "R" part means they are all residing on the server.

Ok, so now so far I've shown you the system database directory and the local database directory, and now I'm going to show you the node directory. If I do **db2 list node directory**, then this will show me the entry that I mentioned before for the SALESDB, so we are looking at SALESDB. And it's showing me that I'm using TCP/IP. And this is the hostname or IP address. And this is the port that I need to reach that server and then reach that port for that instance where my database is. Ok, so this was a very quick overview of the different directories in DB2.

### **Slide 5: Setup required at the server**

Now let's start working on connectivity, and we will talk about how to setup, and what is required in terms of setup at the server. Later on I will talk about what's required in terms of setup at the client. So what I'll do is I'll show you a demo first on how to do these two steps, and then I will go back to the presentation, or the PowerPoint material. And just go through them very quickly.

### **Demo**

So, going back here, you could start a tool, called the Configuration Assistant using this command **db2ca**. If I press Enter, I will start the Configuration Assistant. You can also start it by going to Start > Programs > IBM DB2 > DB2COPY1 (or whichever copy it is) > Set-up Tools > Configuration Assistant. So this is the long way. The short way will be just to type **db2ca** as I did before. Ok, now that I've typed this, I executed it, I get here the Configuration Assistant. You can use the Configuration Assistant from either the client or the server.

So right now, I'm going to show you the setup at the server. We were talking about setup at the server. Alright, so let us assume right now you are looking at the server computer. So what I'll need to do is I need setup two things. The first one is called DB2COMM, and for that I will go to DB2 Registry. And then I will look for DB2COMM. DB2COMM is a DB2 profile registry, which is what you have to specify which protocol you want to use. Most people nowadays use TCP/IP, but from here, if I click on this button I could change the protocol. Now, in this particular case DB2COMM is already set up, but if it wasn't set up, you'll have to type the value, and you may have to restart the instance, that means you issue **db2stop** and **db2start**. And by setting this value to DB2COMM, you're basically going to have DB2 turn on the TCP/IP listeners, so they are aware of incoming requests.

Ok, so that is the setup, I was working on this server and DB2COMM is set to "TCP/IP", so that's fine. So I click OK. The second configuration on the server is to set up the service name,

or the port numbers. So if I click Configure > DBM Configuration... By the way, these... actually, let me show you that you could also get DBM Configuration from the Control Center. So if I type **db2cc** to go to the Control Center... I will start the Control Center, and from here you could also set... from my computer you could set, at the instance level, the parameters that you will need. In this case you are looking for the parameter SVCENAME... which is... so let me shrink these two categories. And then you can see here, that is SVCENAME. So if you click this value you can change it, which is a string.

Now, you may be wondering what is this string about, right? Well, you could have either a string, like in this case, or a number, which would be the port number. Now SVCENAME stands for service name, and this applies to TCP/IP. So, let's say if you go to... I have shortcut here, so let me go quickly to the... to look for the "services" file. So in Windows, if you go to this directory, C:\Windows\System32\drivers\etc, you can find the "services" file. If you were on Linux you would have to go to "/etc/services". And if we take a quick look at, I'm just going to change quickly the extension so that I go and take a look at this file, here you will see this entry, "db2c\_DB2", "5000". So basically this is the service name that maps to this port number. So given that I have this service name mapping to the port number, I can use the service name. If I didn't have this file with this entry, then I would not be able to use this string, and then I would have to type directly the number, port 50000. Ok, so let me shut down, shut this file, and make sure I put back the original extension, which was none.

And now, we go back here to the Control Center and this one happens to be set up correctly to "db2c\_DB2", which means it's using port 50000. If I had to make a change as I have shown before, I would type the port number here, or I could have put the name in the "services" file and put the string for that mapping entry. OK. So I'm just going to cancel here because everything was set up. And everything was set up because I'm looking at the db2 instance, which is the instance that is created by default. So normally communication is set up at installation time, and when this instance was created. But when you start working with other instances that you create yourself, you may have to manually input this information for the service name, the port number, and DB2COMM to be set up to "TCP/IP". And you could also make this like... we set this from... using right-click and Configure Parameters, but you can also use it from the command line, using the **db2 update dbm cfg**. Or, you can also use it from the Configuration Assistant by going to Configure > DBM Configuration and it would open the same panel that I showed you before for configuring parameters. Ok, so the set-up has been done for the server, so the server is ready. It's accepting, or the listeners are waiting for requests, incoming requests. And the port is set up. So everything is fine from the server. So let me go back to the presentation here.

### **Slide 6: I) Setup required at the server**

So the set up is required from the server. We did it from the Configuration Assistant, for the set up of DB2COMM, and for the setup also of SVCENAME, the service name. Or you could also do from the Control Center. Actually we did the SVCENAME part from the Control Center.

### **Slide 7: Setup required at the server**

And, you can do it manually as I mentioned before, using the **update dbm cfg** command.

### **Slide 8-Slide9: Launching Configuration Assistant- Configuration Assistant**

Ok, now this is information on how to launch the Configurations Assistant that I already showed you.

### **Slide 10-Slide11: Verifying the DB2COMM registry variable- DB2COMM Registry Variable**

And we have already set up DB2COMM.

### **Slide 12: Agenda**

So now let's continue with the setup required at the client.

### **Demo**

Now, for this we also use the Configuration Assistant. So now let's imagine that we are looking at the client machine. And what we see here is the same output as if issuing a **list db directory** command. So basically you are looking at the system database directory here and these are all the databases that you can connect to. Now, if you want to connect to another, a new database, you would right-click on any white space and choose "Add Database Using Wizard". Then you will get this wizard where you have three options. We will choose, first, "Manually configure a connection to a database" option, and later on I'll explain the other two options. Now we click Next. We have to choose the protocol.

Normally "TCP/IP" is the most common one, so that's the one we will use. This section here requires that you have DB2 Connect, and this is for connecting to a host, meaning DB2 for Z/OS, or to the AS/400, used to be called OS/400, which is now System i. Now we click on Next, here we have to specify a host name. Now, given that I'm not right now on a network, I'm just going to type "localhost", which means that I'm going to connect to my own computer using TCP/IP. I really don't need to do this, but this is just for demonstration purposes so that you can see that TCP/IP connection to myself can work. Now here I can put the port, which is 50000 in my own computer for that instance of DB2, and the service name, you can put either the service name, or the port number. In this case I just decide to put the port number. Then I can click Next. Here you put the database name, in this case I'm going to connect to the SAMPLE database, but I'm going to use a different alias. So let's say I'm going to use the "sampleYY" alias. Then I click Next. Here this is optional but this is required if you want to register your database as... for CLI or ODBC applications as a data source. So I'm just going to leave the default and click Next.

Then here you choose the operating system of the server you are trying to reach. If you don't know the operating system of the server you are trying to reach, you can choose the last option, which is "Unknown DRDA". But it is better if you choose, or if you know the operating system, because that way the connection may be a little bit faster. So, I'm going to choose Windows, I'm going to click Next, This is just like a summary that you just need to

leave the default there. Click Next. Here you get your authentication. The default for now is ok, we'll talk more about this later in the Security lesson. So I will click Finish. Now, you will get this dialog box where it asks you to... you can see there's a "Test Connection" so I just test the connection. I'm going to choose all the options here. Well, my user ID is this one. In your case it may be **db2admin**, or any other user that you specified at the beginning or may be Windows Logon administrator. And then from here I'm going to put the password. And I click "Test Connection". And then we wait for a few seconds for the connections to occur.

So, as we can see from here, all the connections were successful. You see CLI or ODBC, or OLE DB, JDBC and so on. Right. So we just close this now, and then from this chart we can see we have here SAMPLEYY, which we connect to the database SAMPLE. And this is my previous entry. This one is using a local connection to the SAMPLE database. Now, the alias has to be unique. So that's why we had to use another name here for connecting to the same SAMPLE database, because the alias part has to be unique within the same system. Alright, so as you can see it's a fairly easy way to configure client connectivity between a client and a server. And you will practice in the lab, as long as you have or as long as you work on a network, you can practice connecting to another machine that has DB2, so that you can connect from a client... from a DB2 client to a DB2 server machine as long as you know the IP address and the port number.

### **Slide 13: II) Setup required at the client**

Moving on, well, so basically... I covered this information using that Configuration Assistant. But if you want to do this using commands, you could use these commands here, the **catalog** command, the **catalog tcpip node** command, and the **catalog database** command. And this is the way they relate to the directories that I mentioned before. So when you use a **catalog database mydb1 at node mynode** command, it will be entering information into the system database directory. When you use a **catalog tcpip node** command, you will be entering information into the node directory. You normally have to issue the first one, which is **catalog tcpip node**, and then **catalog database**, because, as you can see in red, these two values are what links each other. Ok, very well, and then when you want to connect, you have to use a "user", normally you have to use a "user" and a "password" of the server that you are trying to connect to.

### **Slide 14: Putting both setups together: At the client and server**

And then, this is putting together information. So there is a setup client, this is a setup server, where you had to set up DB2COMM, and the port, which is a service name. And this is the user and the password on the server, which is the same as the one that is put here. That's based on the authentication parameter, which, by default, is sent to server. But we will talk more about this on lesson about security.

### **Slide 15: Setup required at the client**

And you could do the setup of the client through the Configuration Assistant. But you could also do it through the Control Center or do it using commands as shown here. These are the commands.

## **Slide 16- Slide 17: Launching the Add Database Wizard from the CA- Add Database Wizard**

We already discussed about this.

### **Slide 18: Caution**

And, if you have noticed, the Configuration Assistant doesn't really check if you are putting valid information or not. So it will not check if I put any IP address in the host name or any host name, or any port number. The Configuration Assistant will just take those values and put those values into the correct database directories, But it will not check if they are valid or not. The check will happen when you try to do a test to connect to the databases. So you test to connect and you get an error, then probably something is wrong with the configuration. And if your configuration is perfect, then it may be some other issues, maybe a network issue. So maybe you have to do a "ping", to make sure you can reach the other host. Or, maybe there is a firewall problem.

### **Slide 19: Agenda**

Ok, moving on to the last section of this presentation, "Other methods to setup connectivity."

### **Slide 20: Search and Discover Modes**

There are two other methods. One is called "Search" and the other one is called "Discover".

#### **Demo**

So that can also be seen from the Configuration Assistant, when you right click and choose "Add Database Using Wizard". You can choose here, "Search the network", the second option, then if you click Next, you have "Known systems", and you have "Other systems". If you choose "Known systems", that means that you actually know what's the name of system and you know the IP address, for example, so you could double click here and it will show you the list of systems that right now recognizes. But you can click on "Add System", and then you have to specify this information. And from there, I'm not going to do it, as I said before I am not in a network. But once you put this information, these two will show you from the system, which instances there are, and which databases there are, and then you have to basically just scroll down to the database you want to choose to connect to it.

The other method, which is the "Search" method, would be to choose this other option. Now, if I choose this other option and double click on it, what's going to happen is DB2 will start searching across the network. And this would not be recommended if it is a large network, because, you know, it may take a long time. And it may not be too reliable as well because it may have to go through several hoops. So this option is not recommended when you have a large network. But it would be the same process. It would search on each computer of the network. If they have DB2 instances, within or... many instances, and within each instance it will check which databases it can take a look at, and you have the choice to select which database from the chart that appears, or from the tree that appears. Then I want to Cancel from here. So those are the "Search" and "Discover" modes.

### **Back to Slide 20**

So as you can see, you have the "Search" method where you can go around the network, and you have the "Known" method where you actually have to know at specific computer or system in the network.

### **Slide 21: Discovery by DB2 Clients**

Now, let me move on to this other slide, and in this slide, what it is showing is that, there are some parameters that you can specify at different levels, for example, at the DB2 administration server level, at the instance level, or even at the database level. And based on these parameters you can restrict the access to a given database that could be discovered. So, for example, let's say I want to... let's say I'm a regular user. If I want to use these "Search" or "Known" method, then potentially I could be able to access private databases, like a database containing salary information, right? So, let's say, my DBA would not want any user to have access to this. So what he can do is use these parameters at different levels and by configuring them in a different way he can restrict the access so that any private or confidential database would not be seen by this "Discover" procedure.

### **Slide 22: Client and Server Profiles**

OK. So in this next other slide, we are talking about client/server profiles. I'm not going to talk about this in detail, but basically what you do here is you can save into a file your client profile. So let's say, if I work on a machine, I can get all the information that is stored in my database directories, I can save that into... or I can import that into a... sorry, export that into a client profile, which is basically a file. Then I can take that file to other computers and then I can import that into those other clients. And that way all of those clients are setup to connect to the same databases that my client can connect to. Similarly for a server profile.

### **Slide 23: Quicklab # 6 - Using the Configuration Assistant**

So, before we move on to that slide, those profiles could be setup by, again, from the Configuration Assistant.

### **Demo**

You can click "Add Database Using Wizard", and then you can use this "Use a profile", for example, to work with client and server profiles.

### **Back to slide 23**

And now, you can pause this presentation and take a look at Quicklab #6, where you can practice using the Configuration Assistant to connect to other database servers.

### **Slide 24: Good references**

This is good reference. There are several... these are a series of articles called "DB2 Connectivity Cheat Sheet", and these are available in developerWorks. And, they provide many many cases for connectivity and also troubleshooting.

**Slide 25: What's Next**

So, we have reached the end of this lesson number 5. Congratulations because you've completed the lesson. And as to what is next, it would be Lesson 6, Working with database objects. Thank you.