

Lesson 17 Transcript: Troubleshooting

Slide 1 - Cover

Welcome to Lesson 17 of the DB2 on Campus lecture series. Today we're going to talk about troubleshooting. 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 - Overview

We are starting with the overview.

Slide 4 - Problem diagnosis

So let's say you've got a problem. These are basically some of the steps that you can follow to try and solve or diagnose this problem. So one of the main things I would do is to go to DB2, issue the help command, which is a question mark, and type "db2?" and the SQL code that you received. If that doesn't give you any idea as to what the problem is, you can continue checking with the administration notification log. If that doesn't help you, you can move on and check the db2diag.log and so on.

You can also use some operating system commands and maybe check on disk usage or CPU usage then you can also search for APARs or known problems. APARs are basically IBM's recognition that there is a bug in the software. And then you can also collect some traces and all of these will be covered in the next few slides. Now at all times you should check the DB2 online manuals also known as the Information Center and if you don't know the website for the Information Center you can always go to Google and just type DB2 Information Center. You should type DB2 as part of the search argument because there are different information centers for the different products of IBM, so try and put "db2 information center" in order to get the right hit. I will also show you in this slide how to take a look or how to work with the Information Center.

Slide 5 – Agenda-The help (?) command

Ok, so let's cover each of these steps and the first one is the help command, which is a question mark/

Slide 6 - db2 help: ?

So, let's say you get an error message like SQL0104N and you can type db2 ? and the code and you should get more information, right?

Demo:

For example, let's say I (not that window but this one), let's say I do the following, I connect to the database, right? And then I'm going to, just for illustration purposes I'm going to do a **db2 select** and then I'm going to make a mistake on purpose. On **db2 select from employee**, for example, so I'm missing the columns, right? So this will give me an error and this is the error that I get. So the code that I'm getting you should see from the error message is this one, ok? I can

copy, paste there and I can type db2 ? and then I put the code. And then I press Enter and I can get some more information about this error code. I can go to the top of that error message. It tells me exactly what it means; the error message over here, right? “an unexpected token etc etc.” And then it says an explanation: a syntax error and that’s exactly what happened, I was missing the columns. So that’s the first thing I would do whenever I get a SQL code type of error message.

Now if you are working from the Command Editor, like in this example, if I were to do the same thing here, **select from employee** and I’m missing again the columns on purpose. I will get an error message and I want to clear the results here. Execute. I get, basically I get the whole information. So when you are working from the Command Editor, you don’t need to do the “?” and the SQL code because automatically you’ll get the information that was provided like here in this case. All of the information that I got using db2 ? and the code is automatically provided when you use the Command Editor.

Ok. Now of course you can, from here if you want to take a look at any code you can also do here, SQL0104N and execute, right? And you can get information about the code. All right, so that’s how you work with the **help** command. And there is different syntax that you can use for the SQL code. So these are all the syntax that you can use and are shown in this slide.

Slide 7-Finding More Information About Error Codes

These I already showed you, from the Command Editor you can also use the “?”.

Slide 8 - SQLCODE and SQLSTATE Meaning

SQLCODE and SQLSTATE are two special variables that are set every time you issue an SQL statement. In the case of SQL code, when you issue an SQL statement that is successful you will get an SQL code of 0, if you get a statement that was successful but it returned a warning you’ll get an SQL code greater than 0, and if it’s an error it would be less than 0. SQLCODE will be more specific to DB2, while SQLSTATE is more standard and therefore the same SQLSTATE may be applicable or may mean the same in other relational database management systems. For example, in Oracle, in SQL server and DB2, the same SQLSTATE may mean the same because it follows a standard and it’s a five character string, where the first two strings or the first two characters represent or give or have some meaning. If it has a value of 00 it means the command was successful, 01 implies a warning, and 02 a not found condition and anything else is an error.

Slide 9 –Agenda - The Administration notification log

Ok, so, let’s move on to the next section of this presentation, which is the administration notification log.

Slide 10 - DB2 Administration Notification Log

This is a log that will basically put information related to any problem with DB2. And on Linux and UNIX you will find that the filename or the notification log is a file that has as name the <instance name>.nfy. So for example if the name of the instance in Linux is db2inst1 then there should be a file called **db2inst1.nfy** and that would be the administration notification log. On windows it would be stored in Event Log so you would look at it from the event viewer.

Demo:

So let me show you; let me just go to the event viewer from here. So I'm going to go to my computer, right-click, choose Manage. From here I choose Event Viewer and it should be in the application section and then from there you will see anything with DB2 is basically what we call the administration notification log and you double-click on any of these and you get more information. There is an example here; there is some warning with respect to the Health Monitor.

Ok, now the administration notification log can also be seen from the Journal. So from the Journal which can be started from the Control Center, you can look at this tab called "Notification log" and you can see the same information as what you see in the Event Viewer on Windows. If it's on Linux, again, that would be stored in the <instance name>.nfy file and also you can see that from the Journal. So let's close these two windows and let's continue.

Now there are different levels for how much information you want to collect in the administration notification log. There is this file, this parameter called notifylevel, which is a dbm cfg parameter. If you set it to these values you will get more or less logs or more or less information in the logs.

Slide 11- The db2diag.log

Ok, the next section of this presentation will talk about the db2diag.log.

Slide 12 - db2diag.log

The db2diag.log is similar to the administration notification log but actually contains more internal information. So from a user's perspective you may find maybe a little bit confusing in terms of some of the information that you find in the db2diag.log. However sometimes it's useful because thanks to Google right now you can basically Google any of these things that you can find in the db2diag.log, and you may find some interesting information. So even though you may want to first look at the administration notification log, checking the db2diag.log is not a bad idea.

Slide 13- Locating the db2diag.log

Now the db2diag.log is going to be stored in Windows under `sqllib\<instance name>\db2diag.log`, and under UNIX under `/sqllib/db2dump/db2diag.log`. This is the default path but you can also change it by changing the parameter `DIAGPATH`, which is a dbm cfg parameter, so it's at the instance level. And also you can change the parameter `DIAGLEVEL`, which is basically allowing you to specify how much information you want to store in the db2diag.log. If you set it to 4, that means you are going to be logging the most amount of information; if you set it to 0 you are going to be logging the least amount of information, and the default is 3.

Demo:

Now let me show you quickly this directory: `C:\Program Files\IBM\SQLLIB\DB2`. So that's where the db2diag.log is for the instance DB2. So if I double-click, if I open the file db2diag.log you can see an entry of the db2diag.log you can see this is the time stamp. So this is important. If you know when a problem occurred, you can search for it based on a time. It's also interesting to take a look at the level of the message. So if it says "Event" normally it's not a

problem. It's more informational. If you see that it says "severe" then you should pay attention to what the problem is. If it says 'Error' as well you want to take a look at it in more detail. And the first entry normally in the db2diag.log is going to provide you information about the system itself.

So for example, it tells you, in this case I'm using Windows system and the service pack as well, in terms of, here as well, this is the similar information as what you get when using the **db2level** command, where it tells you the db2 level that you are using as well. And then over here this is another entry, and it tells you what program or process is causing this message to happen, on which instance it's affecting, the db2 function that is being used at the time. So it's this one, for example. The probe, this probably won't be helpful to you because it tells you where within the function is where this problem happened. And these are return codes. Some of these may be useful to you in the sense that you can go to Google and you can include the function name and then search for something that may be helpful to your problem. So this is the db2diag.log.

(Back to slide 13) And moving on to the next slide.

Slide 14- CLI Tracing

We are going to now talk about CLI Tracing.

Slide 15 - CLI Tracing

Now CLI stands for Call Level Interface. And it's basically the IBM's version of ODBC. It has a little more than that but just to simplify, the explanation is that. Now many, for example, many languages or extensions or drivers that are used with DB2 are built on top of this CLI layer. So for example if you are developing a Java application using JDBC type 4 or you are using PHP or Ruby on Rails, then if you are going, or if you are connecting to a DB2 database, these connections or this work that is happening through these programming languages is going through this CLI layer. So it may still be useful to get a CLI trace even though you are using Ruby on Rails, or Java or PHP.

How do you turn on a CLI trace? Well, first of all it is stored under the directory of sqllib and the file name that is required to configure this CLI tracing is called **db2cli.ini** file and you would set that at the client, so where you execute your program, your PHP script or your Java program, etc. You will put that file in there so assuming there is a db2 client as well, then you would... this basically is the contents of that file, right, so there is a stanza that starts with "common". So that means in this case it will affect any database or if you want just for one particular database, you would put in this square bracket, the database you want to affect, and then trace equals 0 means the trace right now is off. But if you set it to 1, the trace will start, but you probably have to restart your application, your Java application, your PHP application, your ODBC/CLI application before this trace takes effect. And then there are some other different parameters that you may need to set. Some of them are optional. This one is required, tracepathname, that's where you will be writing the trace. And you may have to provide a directory that exists and that's where the information will be stored.

Slide 16 - DB2 Tracing

Ok, the next section of this presentation is db2 tracing.

Slide 17 - Low level Tracing (db2trc)

This is probably something you will not use if you just have DB2 Express-C with the unwarranted license, which means that you don't have support from IBM, while, if you have the other license of DB2 Express-C, which is the one that you have to pay for support, it's a one year term license, then this may be helpful to you. Because this is low level tracing and basically it's going over here, you get all the instructions as to how to turn on the trace and how to turn it off, but basically will collect this trace that is only helpful to db2 technical support because it has the list of functions and function calls in db2 and if you don't have access to the code of DB2, then the information is not going to help you.

Slide 18 - Dealing with DB2 Technical Support

Now as said before, if you have the DB2 Express-C unwarranted license, then you don't have access to db2 technical support. But if you have the one year subscription license of DB2 Express-C, then you do have the access to DB2 technical support and in that case, if you are dealing with DB2 technical support, before you contact them ensure that you have the same level, the same Fix pak level for your clients and servers so that they are at the same maintenance level. Then ensure that you maybe can reproduce the problem so that will help as well. And collect everything needed by DB2 tech support, normally executing the command or executing the db2support utility would be basically collecting most of the information that they need.

Slide 19 – Agenda - The DB2 Information Center

So finally, the last section of this presentation is DB2 Information Center, and this is basically the online manuals.

Slide 20 - The DB2 Information Center

There is one for version 9 as is shown in this slide and one for version 9.5. Now this is very useful. Most of the answers to your questions can be found from the db2 Information Center so don't jump into contacting DB2 technical support for example, if you can maybe just go to the manuals and find the information yourself.

Demo:

So very quickly I'm going to show you how that DB2 Information Center looks like. This is the way it looks like. In this case I have it installed locally so that's why it shows this, otherwise it would show other URLs that I mentioned in the slide. And from here you can type the whatever you are trying to look for. So let's say I'm looking for the **xmlvalidate** function, so I click on Go, and then I get different results and I click on the one I'm more interested in. And then I can see in this example the syntax of xmlvalidate.

(Back to slide 20) And then you can do the same for other different search arguments.

Slide 21 - What's Next?

With this we have finished this presentation, so congratulations for completing Lesson 17 on troubleshooting tips.

Slide 22 - What's Next?

Moreover this is actually the end of this e-learning course. So Congratulations. And as to what is next: It will be maybe to review the book, the *Getting started on db2 Express-C* book, which is a free online book. Try maybe to redo the exercises and take a look at these videos again. And then once you feel that you are comfortable with your knowledge maybe you may want to write the DB2 on Campus Exam which will be ready by July of 2008 and it will give you a completion document that basically says, you know, that you completed the DB2 on Campus program. And also if you are more interested in getting to know DB2, you can keep checking the channeldb2.com web site or the db2 Express-C website which is ibm.com/db2/express, which is mentioned here at the bottom, right? And then you can get more information about the new things that are coming up with DB2 Express-C. OK, so thank you very much and hope you enjoyed this course. Goodbye! Have a good day!