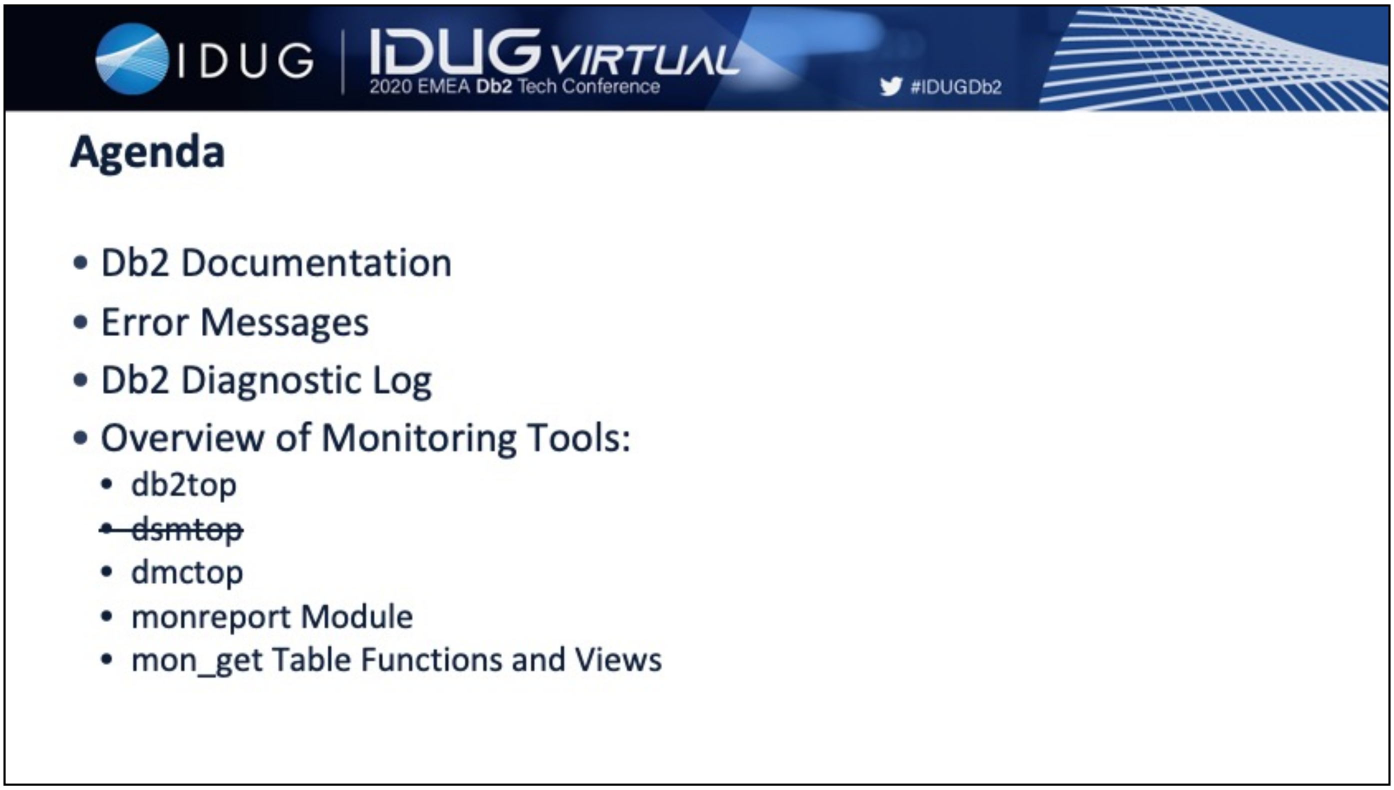
#IDUGDb2

Db2 Basics: Monitoring and Troubleshooting Db2

Ember Crooks

Session code:

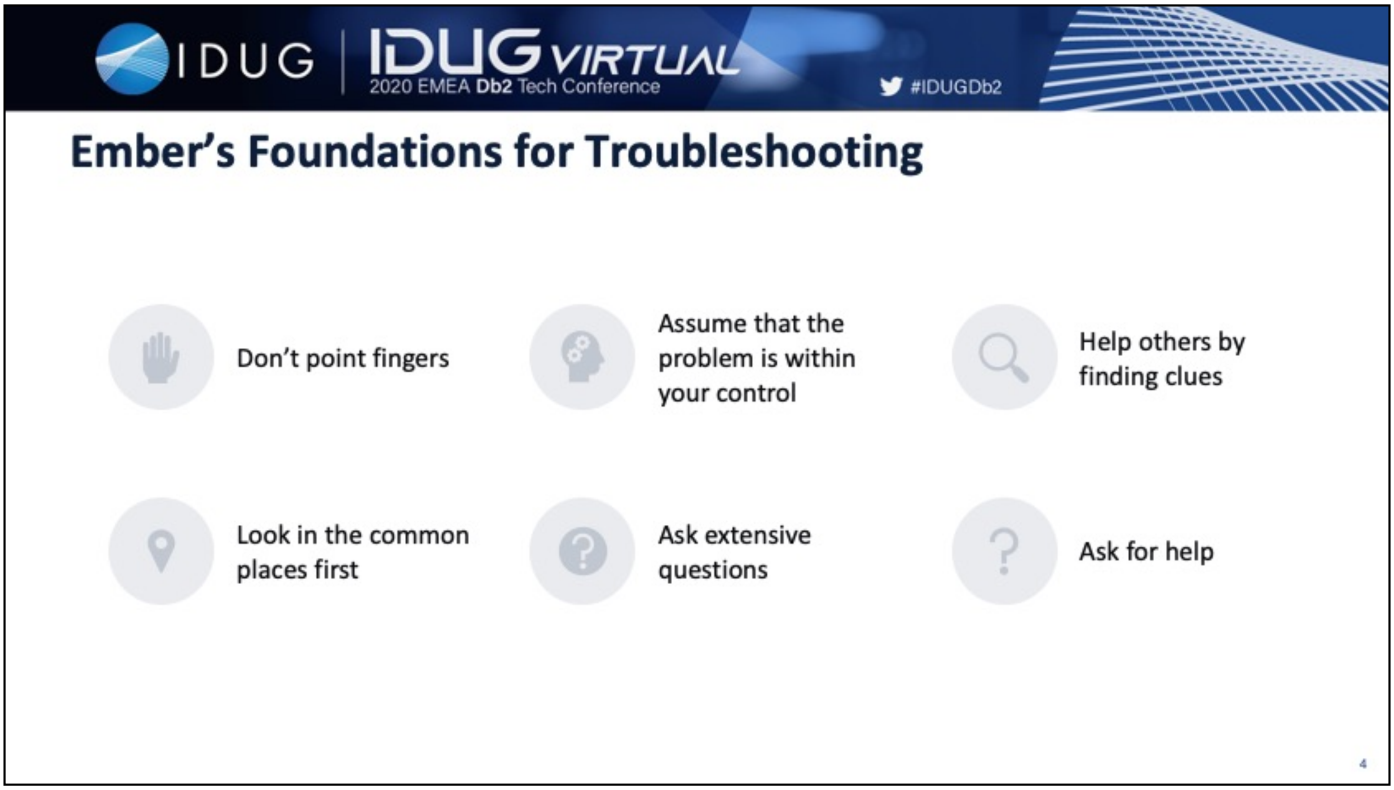


Agenda

- Db2 Documentation
- Error Messages
- Db2 Diagnostic Log
- Overview of Monitoring Tools:
  - db2top
  - ~~dsmtop~~
  - dmctop
  - monreport Module
  - mon_get Table Functions and Views

Troubleshooting Methodology

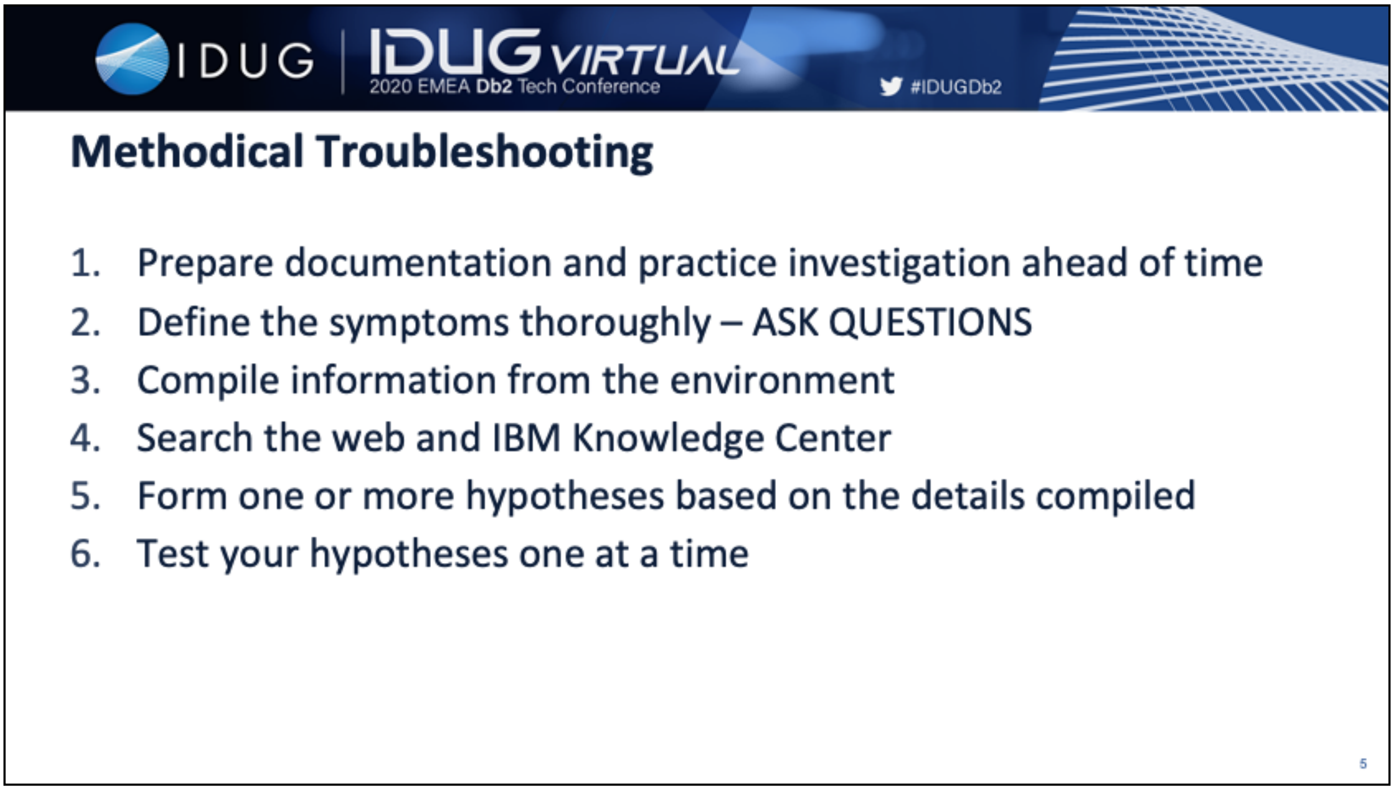
Take a Systemic Approach



Ember's Foundations for Troubleshooting

- Don't point fingers
- Assume that the problem is within your control
- Help others by finding clues
- Look in the common places first
- Ask extensive questions
- Ask for help

When troubleshooting you're rarely able to immediately say "this is a database problem", "this is an application problem", or "this is a network problem". The key is that if you're working with experts in other areas or even if you're trying to cover all the areas yourself, don't rule things out prematurely. If someone calls me in for a problem, I do my best not to just say "That's not the database.", even if it clearly isn't. Even if it isn't the database, often there are clues that you can find within the database that may help others locate the problem in their areas. My approach is to start from "This is in the database, but where?", and then once I have significant proof it isn't, provide others with the details that will help them in their own troubleshooting efforts.



IDUG | IDUG VIRTUAL 2020 EMEA **Db2** Tech Conference #IDUGDb2

Methodical Troubleshooting

1. Prepare documentation and practice investigation ahead of time
2. Define the symptoms thoroughly – ASK QUESTIONS
3. Compile information from the environment
4. Search the web and IBM Knowledge Center
5. Form one or more hypotheses based on the details compiled
6. Test your hypotheses one at a time

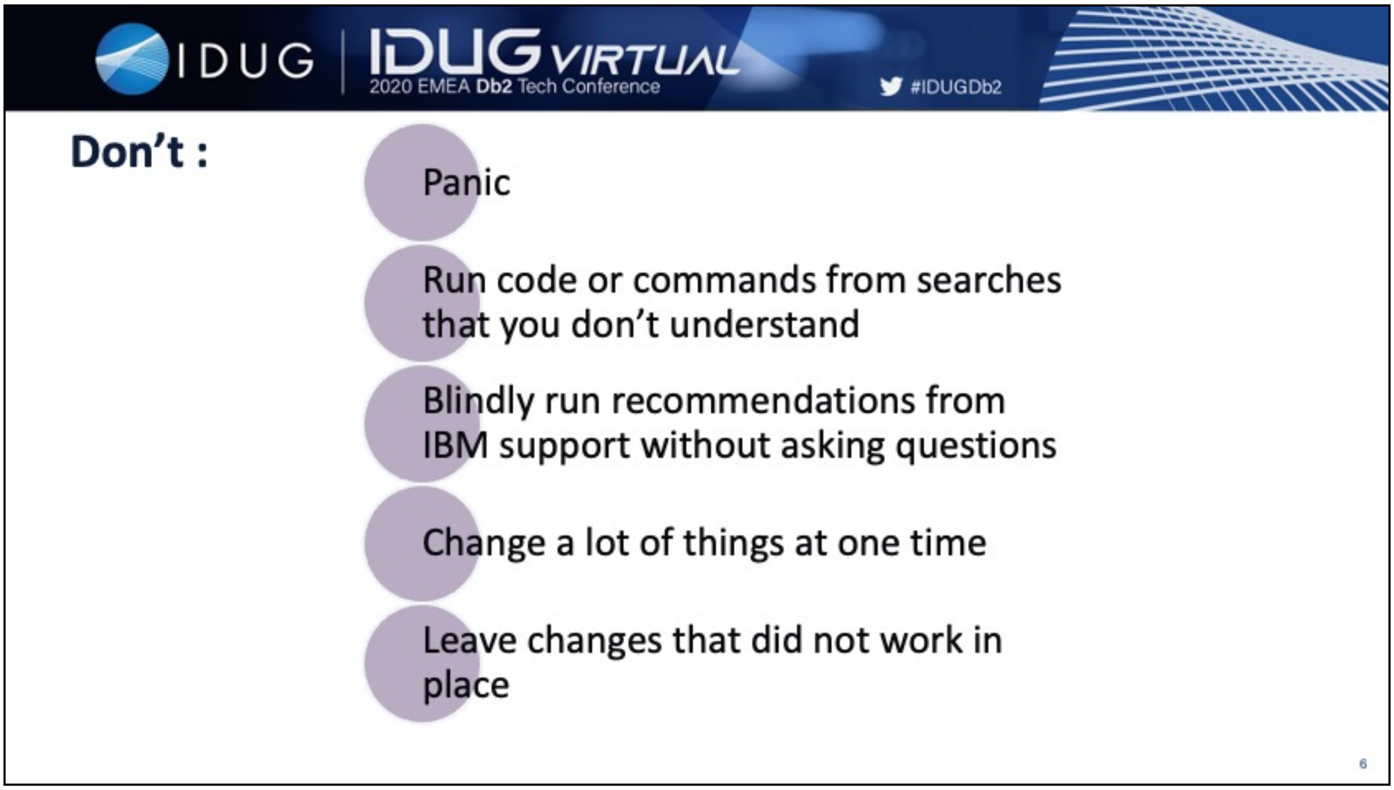
It is hard to perform proper troubleshooting if you don't have the documentation to understand what is in place, and if you haven't prepared with basic skills ahead of time. It is hard to practice troubleshooting, but you can at least practice using the tools. I worked with a junior DBA once who would pass his laptop to me and say "break my database!". He was running Db2 on a VM, and this was an excellent way for him to come across at least a few scenarios to practice on.

Asking questions is one of the most important troubleshooting skills. The first symptoms of a problem may come in through user complaints and "the database doesn't work" simply isn't enough information. Gather all the information you can, as you never know which details are going to be relevant.

It is easy in the heat of troubleshooting to start chasing what you think is the problem and overlook some basic. The two things that come to mind here are checking the diagnostic log for errors and checking to see when the last runstats was. These are both fundamental things that may be critical to the problem. Never forget to check the basics.

I well remember the days when searching a problem online didn't return many results, and we spent a lot more time with the manuals and asking our colleagues for advice and expertise. Now there is a wealth of information online, and even if someone doesn't have the exact same problem, their information may point you in the right direction to look or try things.

It is very helpful if you can formulate one or more hypotheses as to why the problem is occurring and why now. With those in hand, you can move forward to actions or information gathering that can either rule out or confirm your ideas.


IDUG
IDUG VIRTUAL
2020 EMEA Db2 Tech Conference
#IDUGDb2

Don't :

- Panic
- Run code or commands from searches that you don't understand
- Blindly run recommendations from IBM support without asking questions
- Change a lot of things at one time
- Leave changes that did not work in place

6

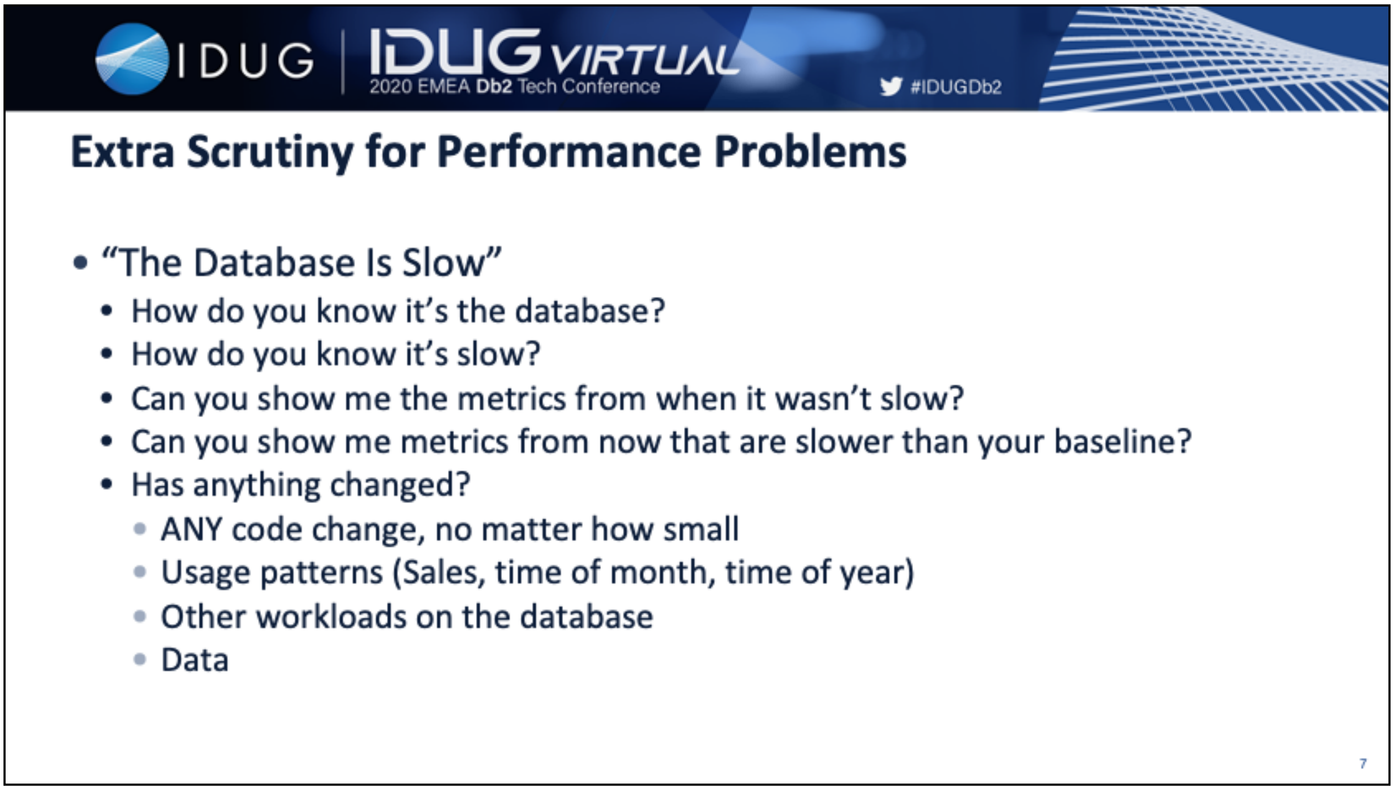
It is very important to stay calm, positive, and focused while troubleshooting. If someone called you in the middle of the night, and you had trouble logging in, and you're nervous, just taking a moment to take a deep breath can be hugely beneficial.

There's an internet legend of someone who was having query performance problems and found a post online recommending truncating the table. They truncated the table, and only then learned that truncation meant deleting all the data, and their data was now gone. Maybe that's an extreme example, but it is easy to run across commands that you don't understand while searching. Be cautious and make sure you understand the impacts before running these commands. Many commands can impact performance or availability further, and some can cause harm that is hard to undo.

As absurd as it may seem, this also applies to recommendations from IBM support. More than once, I've had support recommend something that will cause an outage without telling me it will. More than once, I've had support recommend something that was not appropriate for my environment. Ask support questions about the impact of their recommendations, and if they cannot answer those questions, either insist on answers before you run them, or research to understand the impact yourself.

It can be tempting when troubleshooting to change a number of things at once, particularly if you have more than one thing you need to try that requires an outage. Try to resist this and change only one thing at a time, so you understand what it is that really resolved the problem.

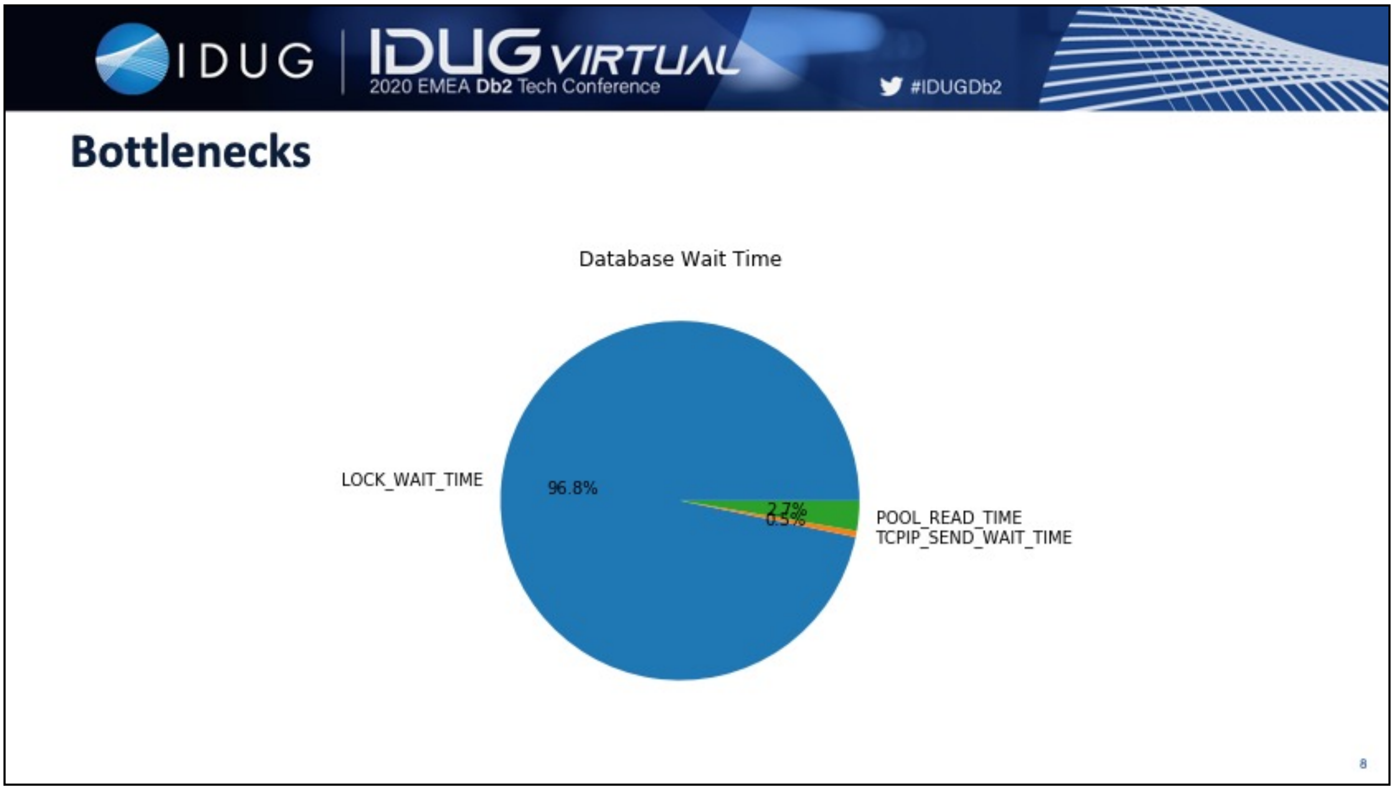
If possible, remove changes that did not fix the problem. Sometimes you try things, and it's best to remove them if they did not work.



Extra Scrutiny for Performance Problems

- “The Database Is Slow”
  - How do you know it’s the database?
  - How do you know it’s slow?
  - Can you show me the metrics from when it wasn’t slow?
  - Can you show me metrics from now that are slower than your baseline?
  - Has anything changed?
    - ANY code change, no matter how small
    - Usage patterns (Sales, time of month, time of year)
    - Other workloads on the database
    - Data

One of the keys for troubleshooting is to ask questions. This is more important for performance troubleshooting than any other area. I once had a developer come to me with a performance problem, complaining that simply returning the CURRENT TIMESTAMP from SYSIBM.SYSDUMMY1 was taking 800 milliseconds – nearly a whole second. This should return very quickly. As I started to investigate, he sent me a screen shot showing the time, and it said 800 microseconds, not milliseconds, which was much more reasonable. Always ask questions, always ask to see what someone is looking at and dig into the details



IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

Bottlenecks

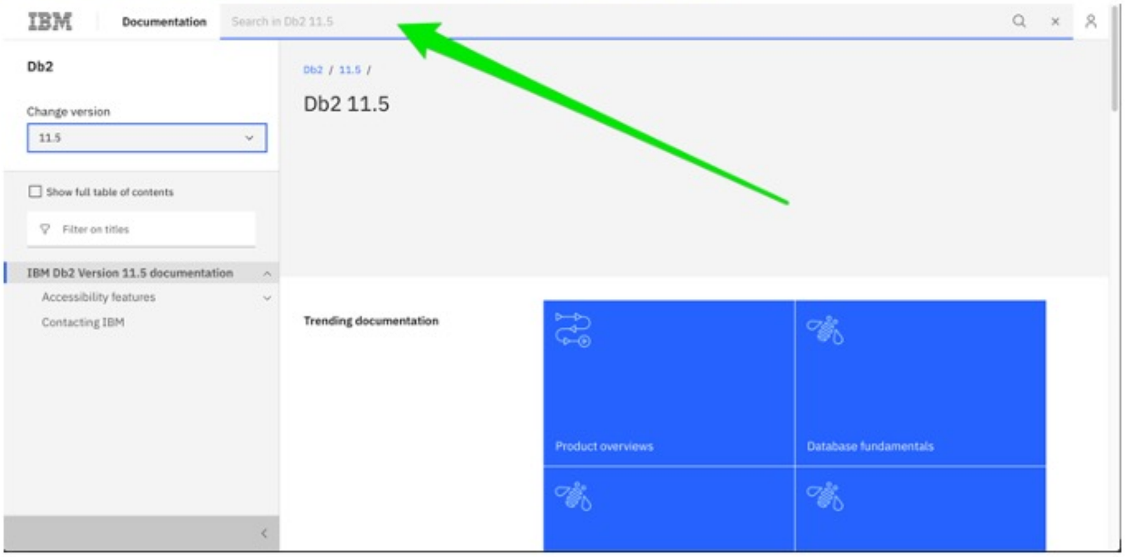
It is sometimes easy to start off investigating a problem, find a promising direction, and loose sight of the bigger picture. For example, if I have a graphic like this one representing wait time within my database, and I go off looking into a network problem, the most I can decrease database wait time by is 0.5%. In this example, my bottleneck is clearly LOCK_WAIT_TIME, and if I spend too much time in other areas, I'm probably wasting my time. Even if I completely eliminated all pool read and all network time, I'd barely touch the overall wait time. That doesn't mean that there cannot be some other area contributing to the lock wait time, just that if whatever I'm working on doesn't reduce lock wait time in some way, it is not addressing my bottleneck.

There are often bottlenecks in database systems. As soon as we eliminate one, something else shows up as the new bottleneck. I remember when a network bottleneck was eliminated once between application servers and the database server – the network went from 10M to 100M. And that made it so database performance was the new focus. Had database performance changed? No. But with the old bottleneck gone, the database was the new bottleneck.

Db2 Documentation

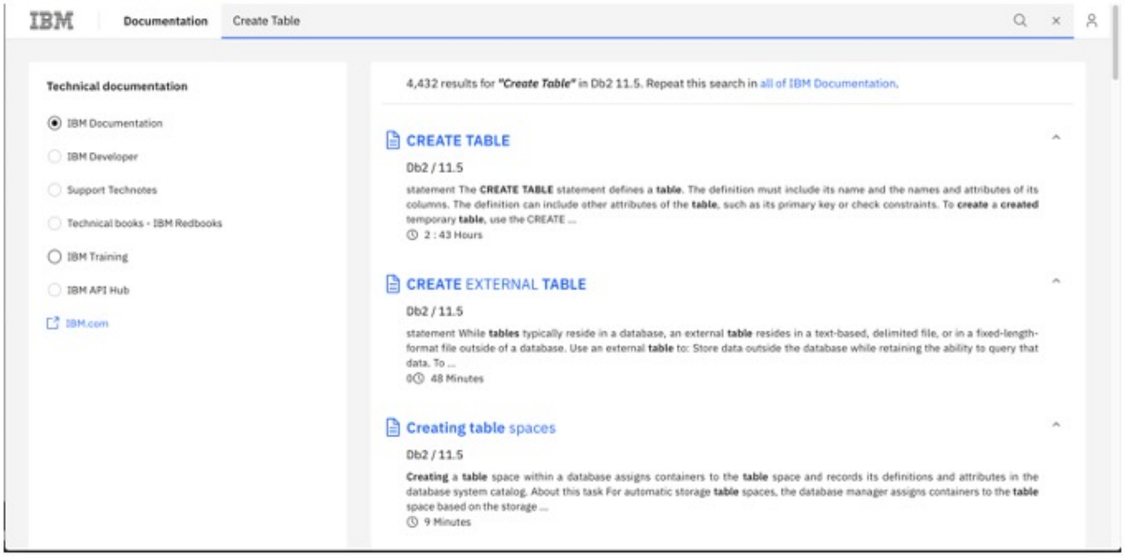
Navigating the IBM Db2 Documentation

IBM Db2 Documentation – Search Box



There is a search box at the upper right. The search in the IBM Db2 Knowledge Center is an order of magnitude better than it used to be. Google or your favorite search engine will also return pages from the Knowledge Center, but sometimes in different languages.

Search – Results



Create Table

4,432 results for ***"Create Table"*** in Db2 11.5. Repeat this search in all of IBM Documentation.

CREATE TABLE

Db2 / 11.5

statement The **CREATE TABLE** statement defines a **table**. The definition must include its name and the names and attributes of its columns. The definition can include other attributes of the **table**, such as its primary key or check constraints. To **create** a **created** temporary **table**, use the CREATE ...

2 : 43 Hours

CREATE EXTERNAL **TABLE**

Db2 / 11.5

statement While **tables** typically reside in a database, an external **table** resides in a text-based, delimited file, or in a fixed-length-format file outside of a database. Use an external **table** to: Store data outside the database while retaining the ability to query that data. To ...

0 48 Minutes

Creating table spaces

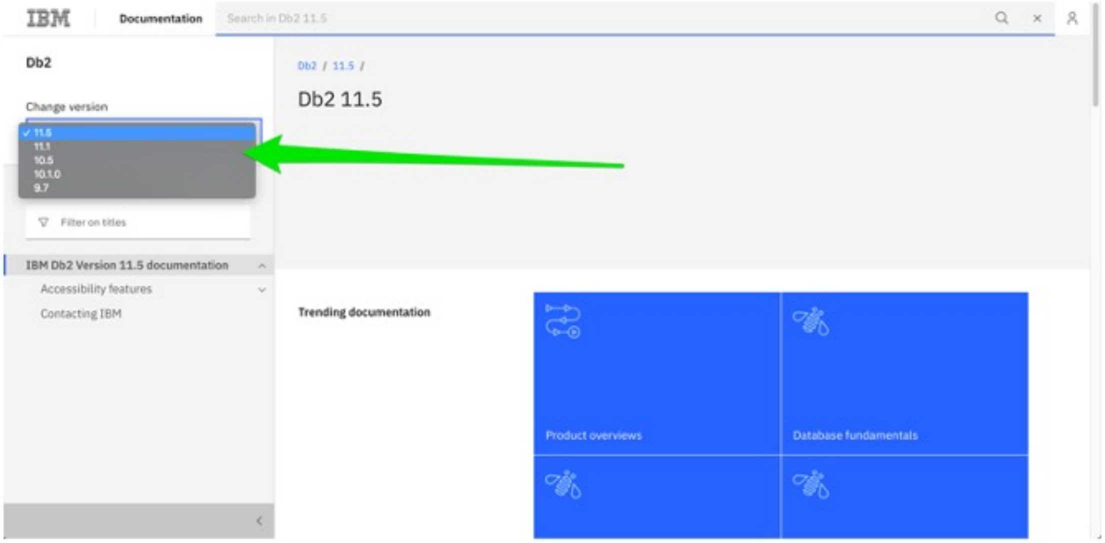
Db2 / 11.5

Creating a **table** space within a database assigns containers to the **table** space and records its definitions and attributes in the database system catalog. About this task For automatic storage **table** spaces, the database manager assigns containers to the **table** space based on the storage ...

9 Minutes

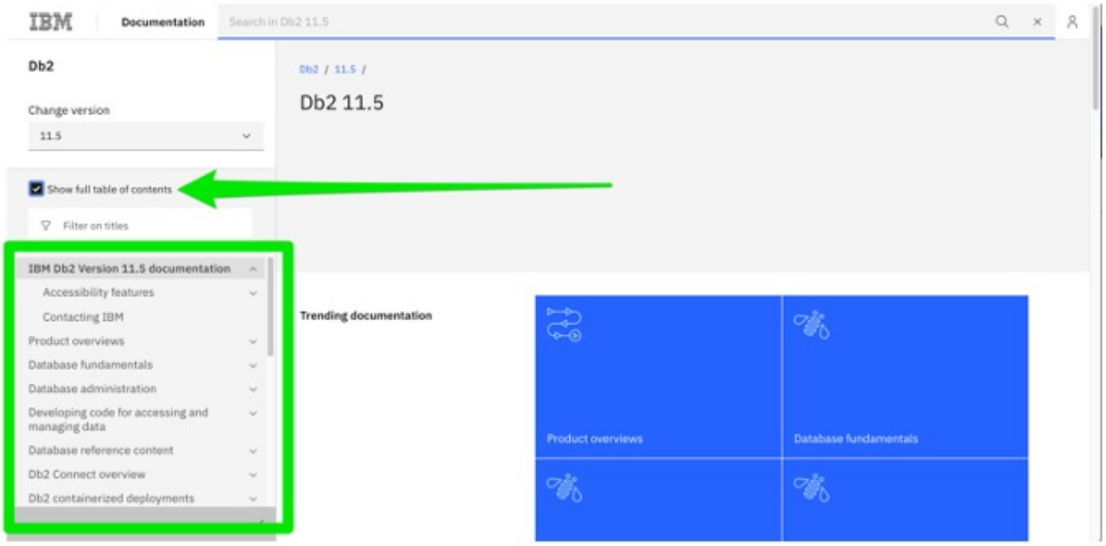
By clicking the plus-sign at the left of any search result, you can get an idea of what the result looks like to decide if it is the one you actually want to click on.

IBM Db2 Documentation – Version Dropdown



Some of the changes between versions are drastic. It is critical to make sure you're looking at the documentation for the version you're working with. On every page, there is a drop-down box that allows you to select different versions of that page. For some versions, some pages don't exist. For example, BLU functionality was added in 10.5, and so the pages related to 10.5 don't exist in 10.1 and before. Documentation is also removed for the oldest versions.

IBM Db2 Documentation – Table of Contents



Db2

Change version

11.5

☑ Show full table of contents

Filter on titles

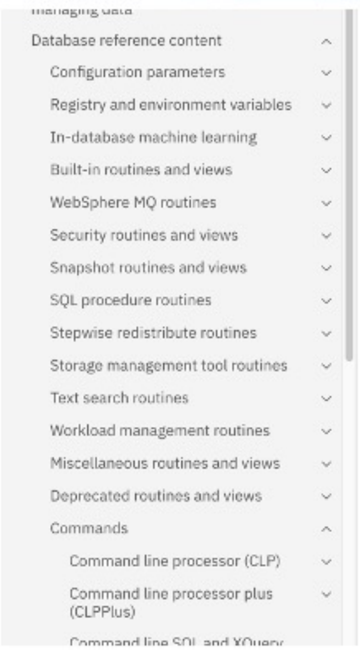
Db2 / 11.5 /

Db2 11.5

Trending documentation

For any page you're on, you can also pull up the table of contents. This will show all the different areas, as well as where the page you currently have up is within the hierarchy. For old-school dbas, this is really easy to navigate because we used to have to use an entire shelf's worth of Db2 books to find this information.

IBM Db2 Knowledge Center – Navigating from the Table of Contents



Managing data

Database reference content

- Configuration parameters
- Registry and environment variables
- In-database machine learning
- Built-in routines and views
- WebSphere MQ routines
- Security routines and views
- Snapshot routines and views
- SQL procedure routines
- Stepwise redistribute routines
- Storage management tool routines
- Text search routines
- Workload management routines
- Miscellaneous routines and views
- Deprecated routines and views
- Commands
  - Command line processor (CLP)
  - Command line processor plus (CLPPlus)
  - Command line SQL and XQuery

It is amazing how much is in the Db2 Knowledge Center. Even DBAs who have been working with Db2 for a while don’t always realize the depth of the documentation. The “Database fundamentals” and “Database administration” sections have some more conceptual Db2 topics. The “Database Reference” is where DBAs spend most of their time, but the “SQL” section there has a lot of details for developers, too. Some of the structure here may be faster to navigate than search. If you’re new to Db2, just perusing some of these topics may be interesting.

Command Page

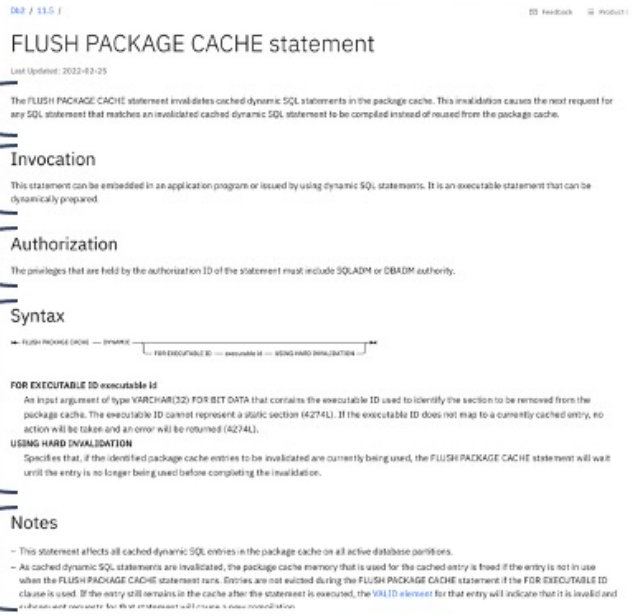
Plain text description of the command

Details about how this command can be executed

What permissions or authorities are needed to run this command

Syntax diagram

Additional tips on the effects of this command and how it might be useful



FLUSH PACKAGE CACHE statement

Last Updated: 2022-02-25

The FLUSH PACKAGE CACHE statement invalidates cached dynamic SQL statements in the package cache. This invalidation causes the next request for any SQL statement that matches an invalidated cached dynamic SQL statement to be compiled instead of reused from the package cache.

Invocation

This statement can be embedded in an application program or issued by using dynamic SQL statements. It is an executable statement that can be dynamically prepared.

Authorization

The privileges that are held by the authorization ID of the statement must include SQLADM or DBADM authority.

Syntax

```
►► FLUSH PACKAGE CACHE — DYNAMIC ─┬──────────────────────────────────────────────────────────────┬──►◄
                                  └─ FOR EXECUTABLE ID — executable id — USING HARD INVALIDATION ─┘
```

FOR EXECUTABLE ID executable id
An input argument of type VARCHAR(32) FOR BIT DATA that contains the executable ID used to identify the section to be removed from the package cache. The executable ID cannot represent a static section (4274L). If the executable ID does not map to a currently cached entry, no action will be taken and an error will be returned (4274L).

USING HARD INVALIDATION
Specifies that, if the identified package cache entries to be invalidated are currently being used, the FLUSH PACKAGE CACHE statement will wait until the entry is no longer being used before completing the invalidation.

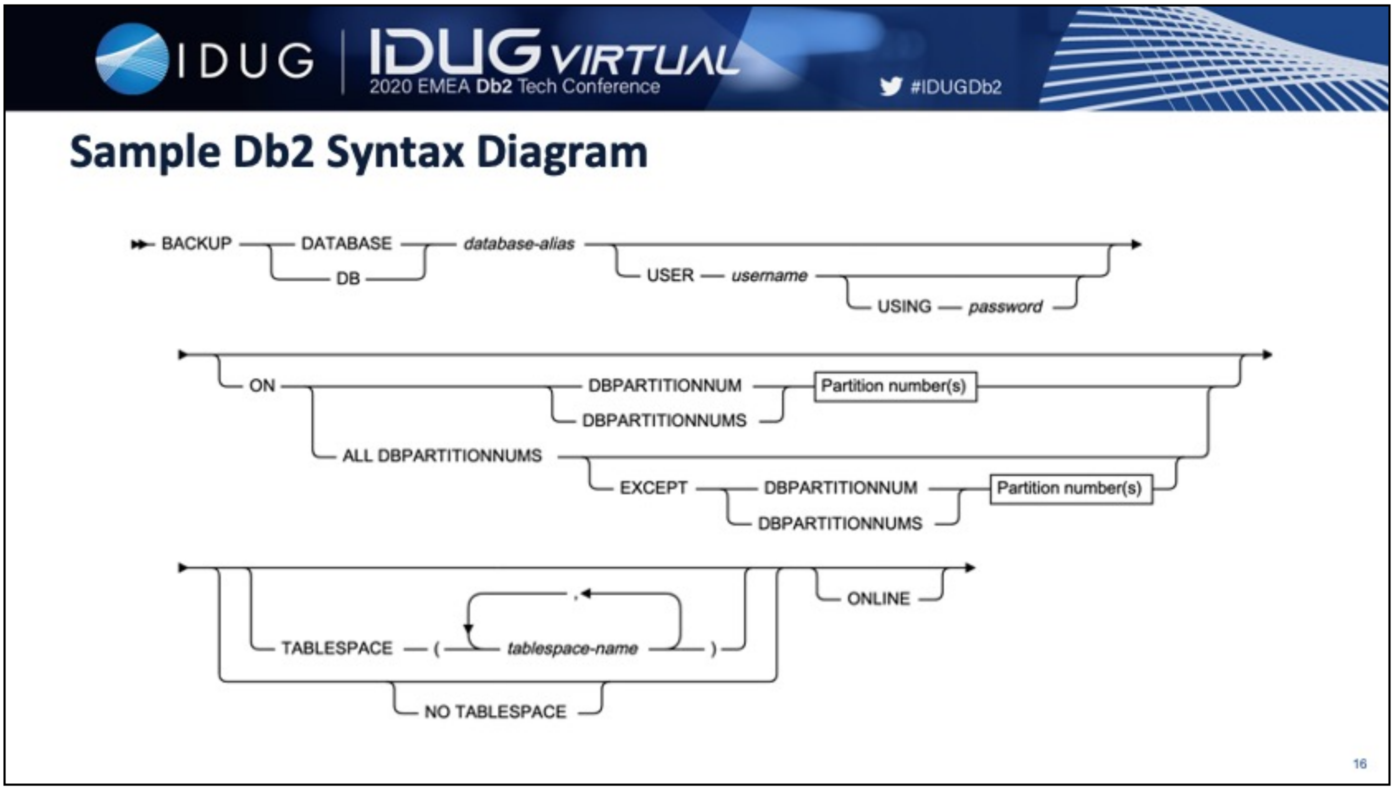
Notes

- This statement affects all cached dynamic SQL entries in the package cache on all active database partitions.
- As cached dynamic SQL statements are invalidated, the package cache memory that is used for the cached entry is freed if the entry is not in use when the FLUSH PACKAGE CACHE statement runs. Entries are not evicted during the FLUSH PACKAGE CACHE statement if the FOR EXECUTABLE ID clause is used. If the entry still remains in the cache after the statement is executed, the VALID element for that entry will indicate that it is invalid and subsequent requests for that statement will cause a new compilation.

15

https://www.ibm.com/docs/en/db2/11.5?topic=statements-flush-package-cache

Each command is laid out in detail in the IBM Db2 Knowledge Center. Note the authorization section which specifies what authorities or permissions are needed to execute a command. The syntax diagram follows a specific format covered on future slides. At the bottom of this page, there are often examples of executing this command.



Sample Db2 Syntax Diagram

BACKUP DATABASE | DB *database-alias* USER *username* USING *password*

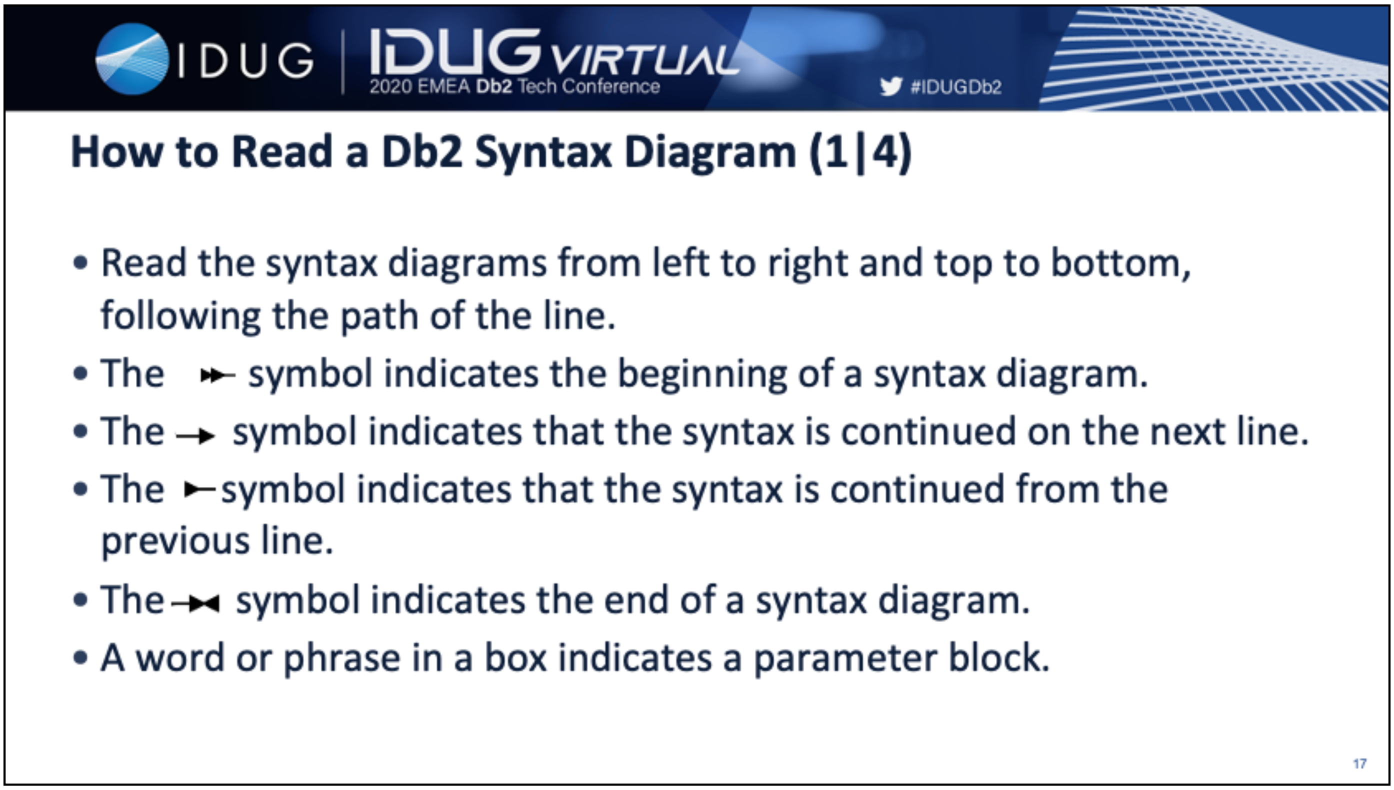
ON DBPARTITIONNUM | DBPARTITIONNUMS Partition number(s)

ALL DBPARTITIONNUMS EXCEPT DBPARTITIONNUM | DBPARTITIONNUMS Partition number(s)

TABLESPACE (*tablespace-name* ,)

NO TABLESPACE

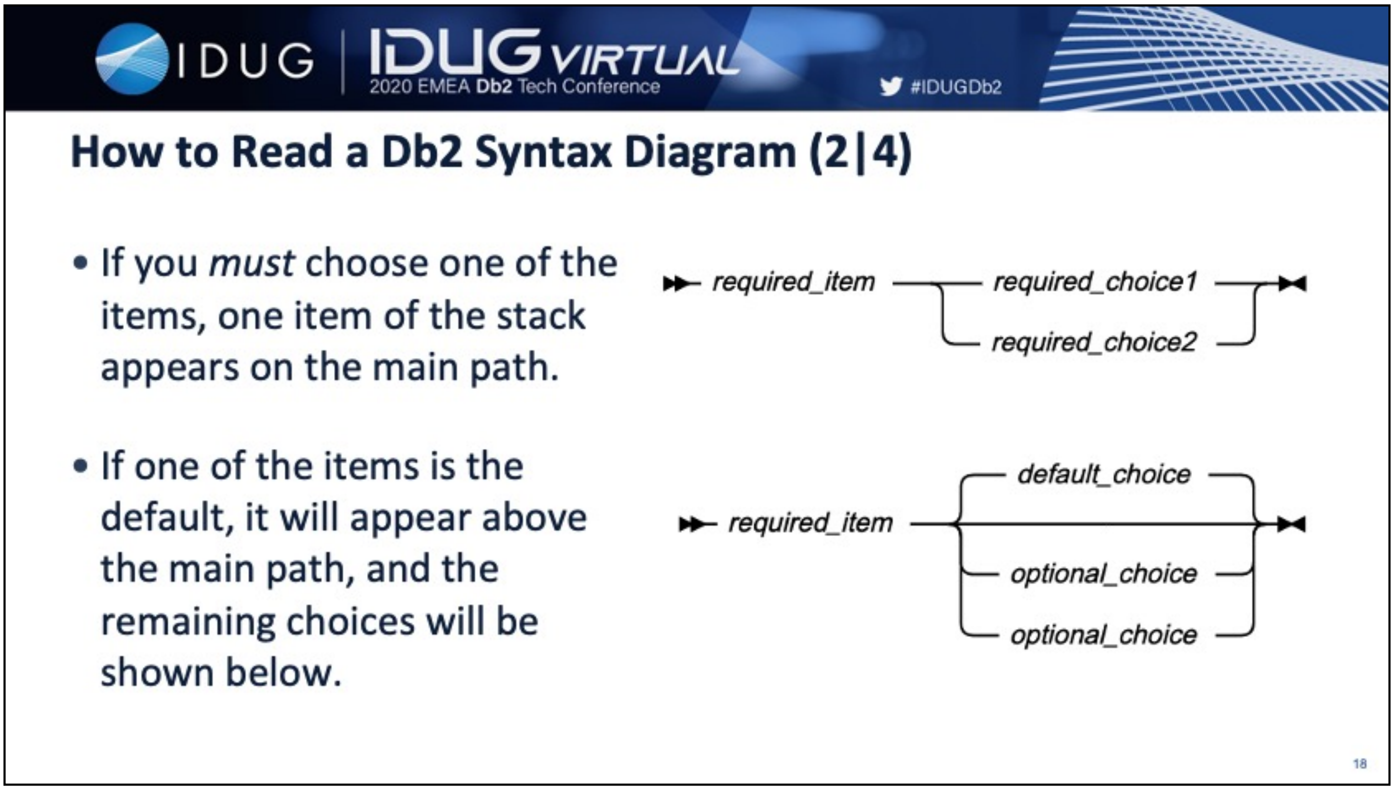
ONLINE



How to Read a Db2 Syntax Diagram (1|4)

- Read the syntax diagrams from left to right and top to bottom, following the path of the line.
- The ►►─ symbol indicates the beginning of a syntax diagram.
- The ─► symbol indicates that the syntax is continued on the next line.
- The ►─symbol indicates that the syntax is continued from the previous line.
- The─►◄ symbol indicates the end of a syntax diagram.
- A word or phrase in a box indicates a parameter block.

https://www.ibm.com/docs/en/db2/11.5?topic=sql-how-read-syntax-diagrams



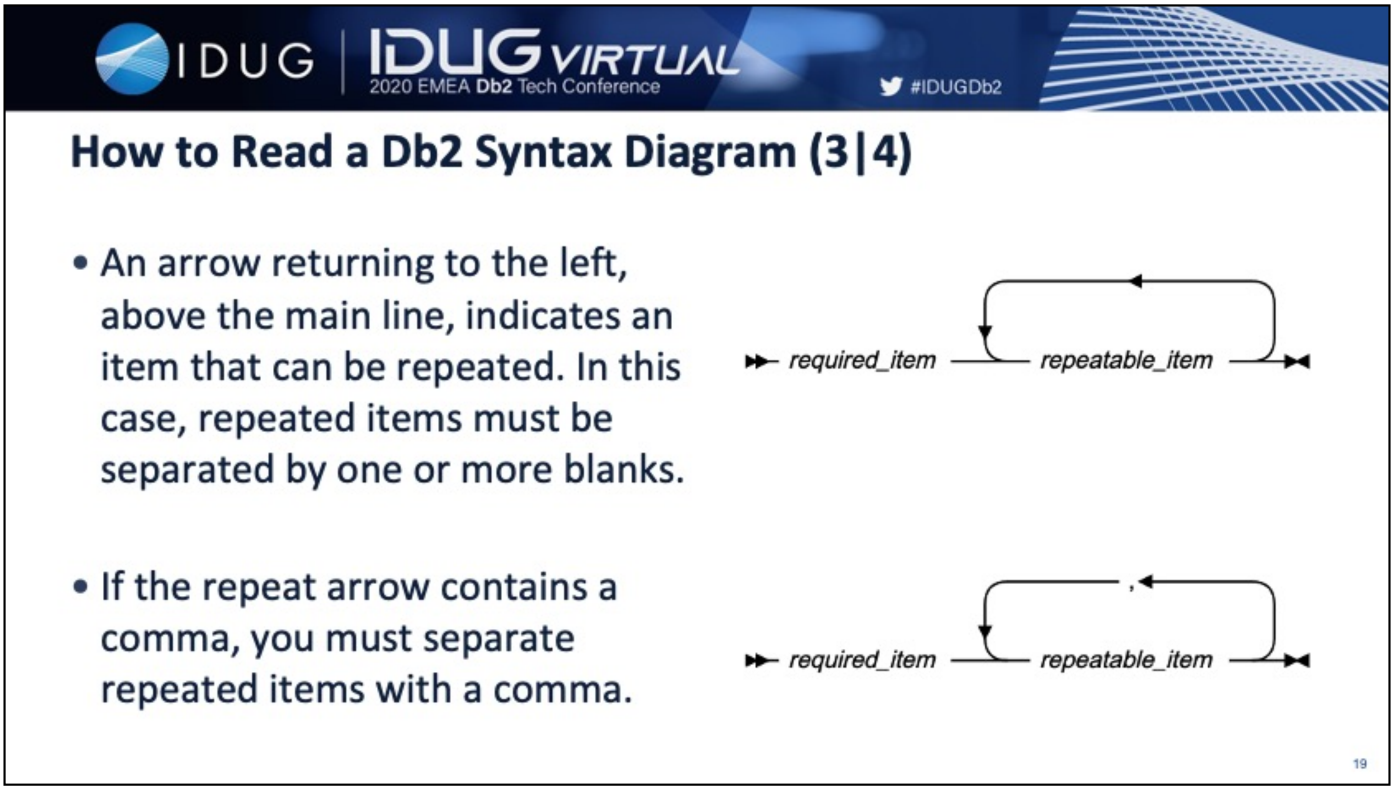
How to Read a Db2 Syntax Diagram (2|4)

- If you *must* choose one of the items, one item of the stack appears on the main path.

required_item required_choice1 required_choice2

- If one of the items is the default, it will appear above the main path, and the remaining choices will be shown below.

required_item default_choice optional_choice optional_choice



How to Read a Db2 Syntax Diagram (3|4)

- An arrow returning to the left, above the main line, indicates an item that can be repeated. In this case, repeated items must be separated by one or more blanks.

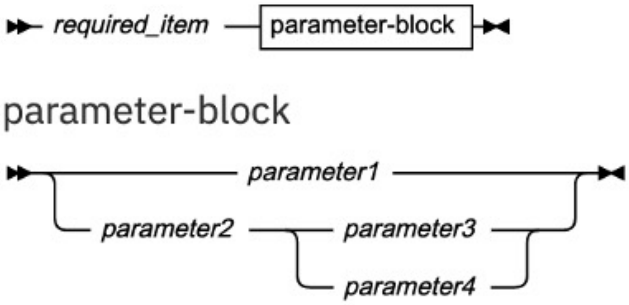
required_item
repeatable_item

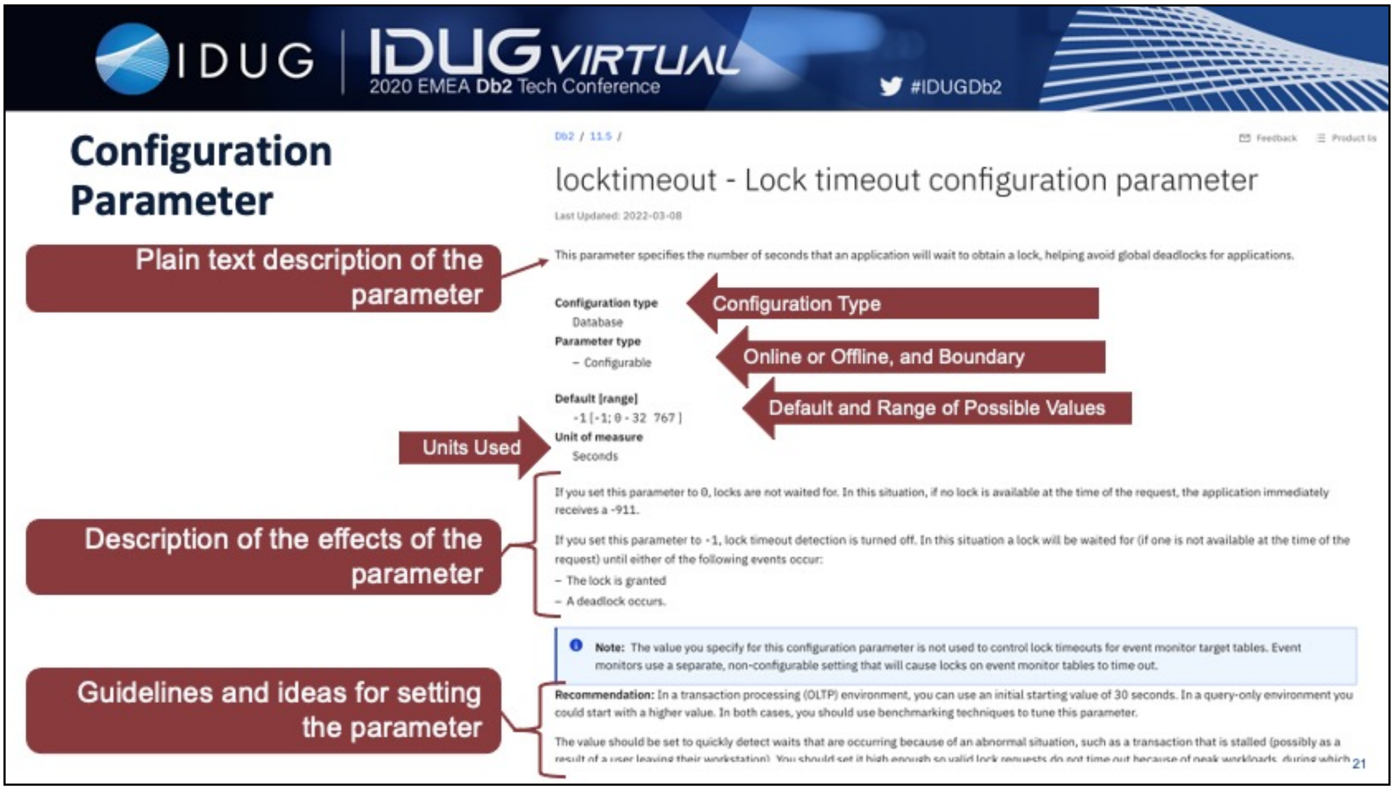
- If the repeat arrow contains a comma, you must separate repeated items with a comma.

,
required_item
repeatable_item

How to Read a Db2 Syntax Diagram (4|4)

- Sometimes a single variable represents a larger fragment of the syntax. For example, in the following diagram, the variable parameter-block represents the whole syntax fragment that is labeled **parameter-block**:




IDUG
IDUG VIRTUAL
2020 EMEA Db2 Tech Conference
#IDUGDb2

Configuration Parameter

Db2 / 11.5 /
Feedback
Product lis
Plain text description of the parameter
Configuration Type
Online or Offline, and Boundary
Default and Range of Possible Values
Units Used
Description of the effects of the parameter
Guidelines and ideas for setting the parameter

locktimeout - Lock timeout configuration parameter

Last Updated: 2022-03-08

This parameter specifies the number of seconds that an application will wait to obtain a lock, helping avoid global deadlocks for applications.

Configuration type
Database

Parameter type
- Configurable

Default [range]
-1 [-1; 0 - 32 767]

Unit of measure
Seconds

If you set this parameter to 0, locks are not waited for. In this situation, if no lock is available at the time of the request, the application immediately receives a -911.

If you set this parameter to -1, lock timeout detection is turned off. In this situation a lock will be waited for (if one is not available at the time of the request) until either of the following events occur:

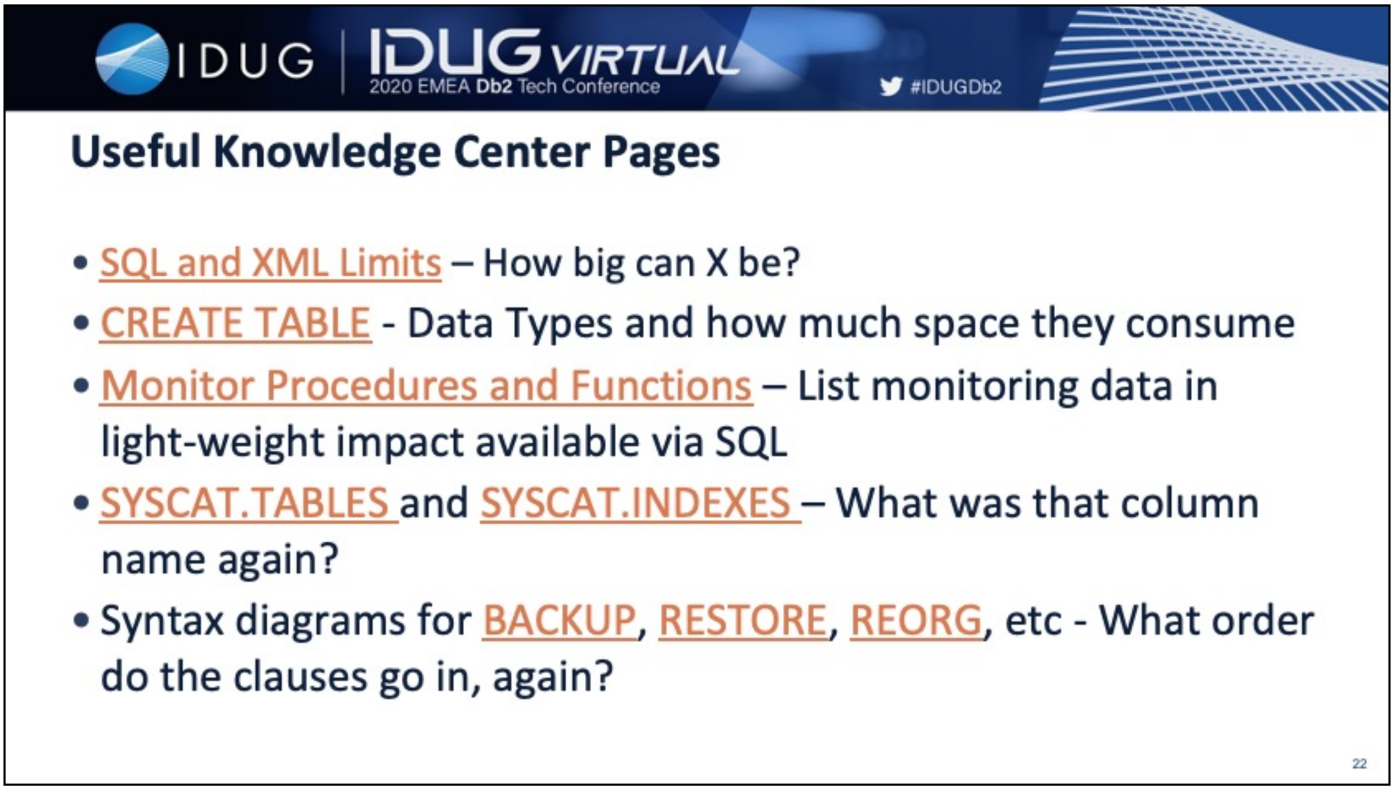
- The lock is granted
- A deadlock occurs.

Note: The value you specify for this configuration parameter is not used to control lock timeouts for event monitor target tables. Event monitors use a separate, non-configurable setting that will cause locks on event monitor tables to time out.

Recommendation: In a transaction processing (OLTP) environment, you can use an initial starting value of 30 seconds. In a query-only environment you could start with a higher value. In both cases, you should use benchmarking techniques to tune this parameter.

The value should be set to quickly detect waits that are occurring because of an abnormal situation, such as a transaction that is stalled (possibly as a result of a user leaving their workstation). You should set it high enough so valid lock requests do not time out because of peak workloads, during which

Each configuration parameter is described in great detail in the IBM Knowledge center. Note particularly the Parameter type, which tells you if a parameter can be changed online or not. The recommendation section often provides a bit of advice on things to consider when setting a parameter.

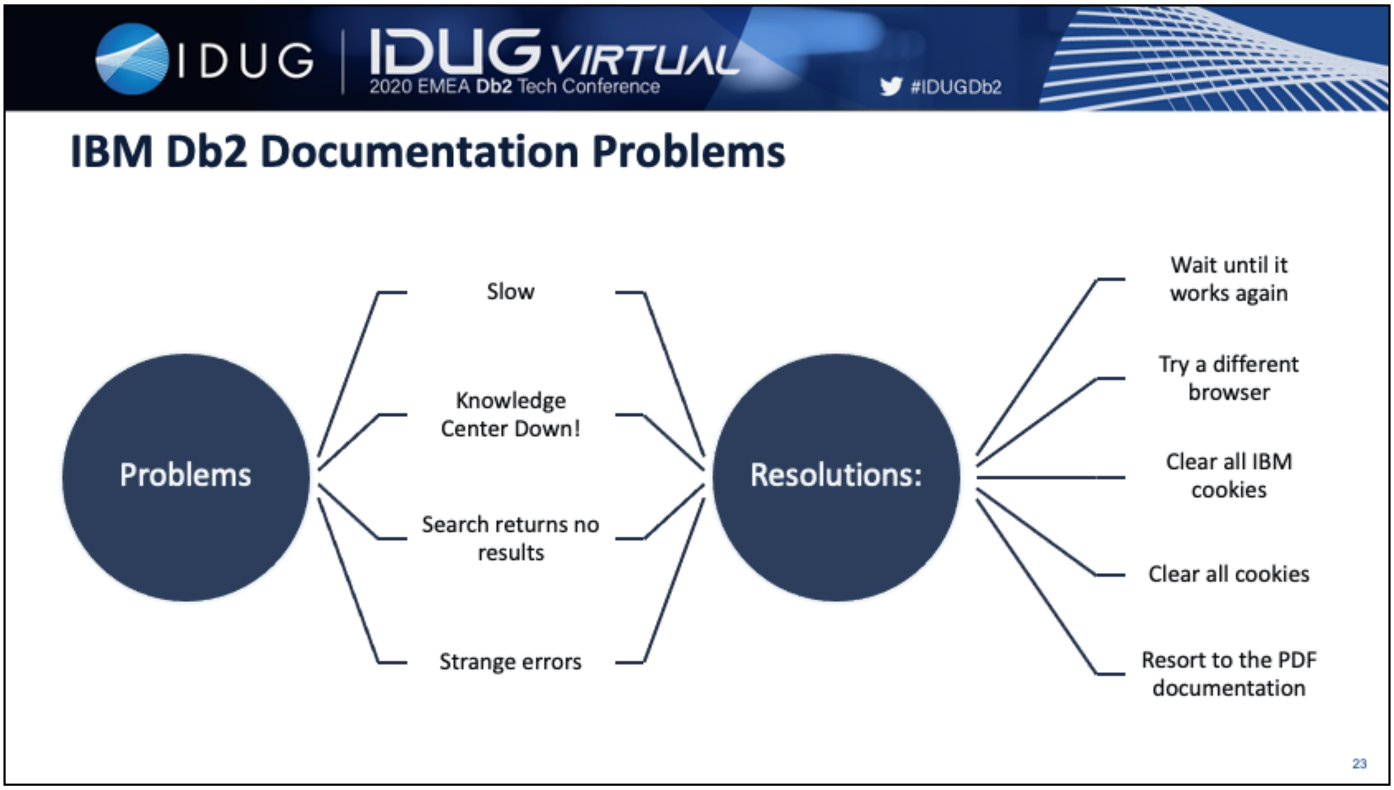

IDUG
IDUG VIRTUAL
2020 EMEA Db2 Tech Conference
#IDUGDb2

Useful Knowledge Center Pages

- SQL and XML Limits – How big can X be?
- CREATE TABLE - Data Types and how much space they consume
- Monitor Procedures and Functions – List monitoring data in light-weight impact available via SQL
- SYSCAT.TABLES and SYSCAT.INDEXES – What was that column name again?
- Syntax diagrams for BACKUP, RESTORE, REORG, etc - What order do the clauses go in, again?

22

There are some pages that you find yourself visiting again and again. These are a few of them that I find myself returning to over and over again, even after 19 years of experience.



IBM Db2 Documentation Problems

Problems

- Slow
- Knowledge Center Down!
- Search returns no results
- Strange errors

Resolutions:

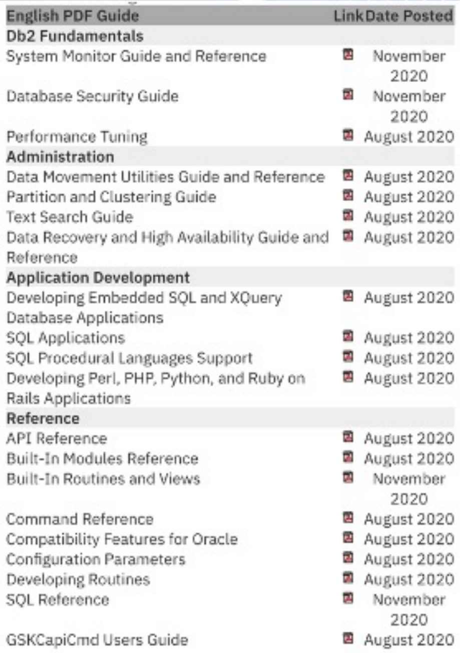
- Wait until it works again
- Try a different browser
- Clear all IBM cookies
- Clear all cookies
- Resort to the PDF documentation

Some places do not allow access to the internet while connected to their VPN. Sometimes the IBM Db2 Knowledge Center is unavailable – even if this is for short periods of time in the middle of the night, it always seems to be when you're in the middle of a major problem or change. DBAs don't always sleep.

IDUG | IDUG VIRTUAL 2020 EMEA **Db2** Tech Conference #IDUGDb2

PDF Documentation

- Download before you need it
- Great for when:
  - Internet access is not available
  - Online documentation is down
- https://www.ibm.com/support/pages/node/627743
- Different set of PDFs for each version



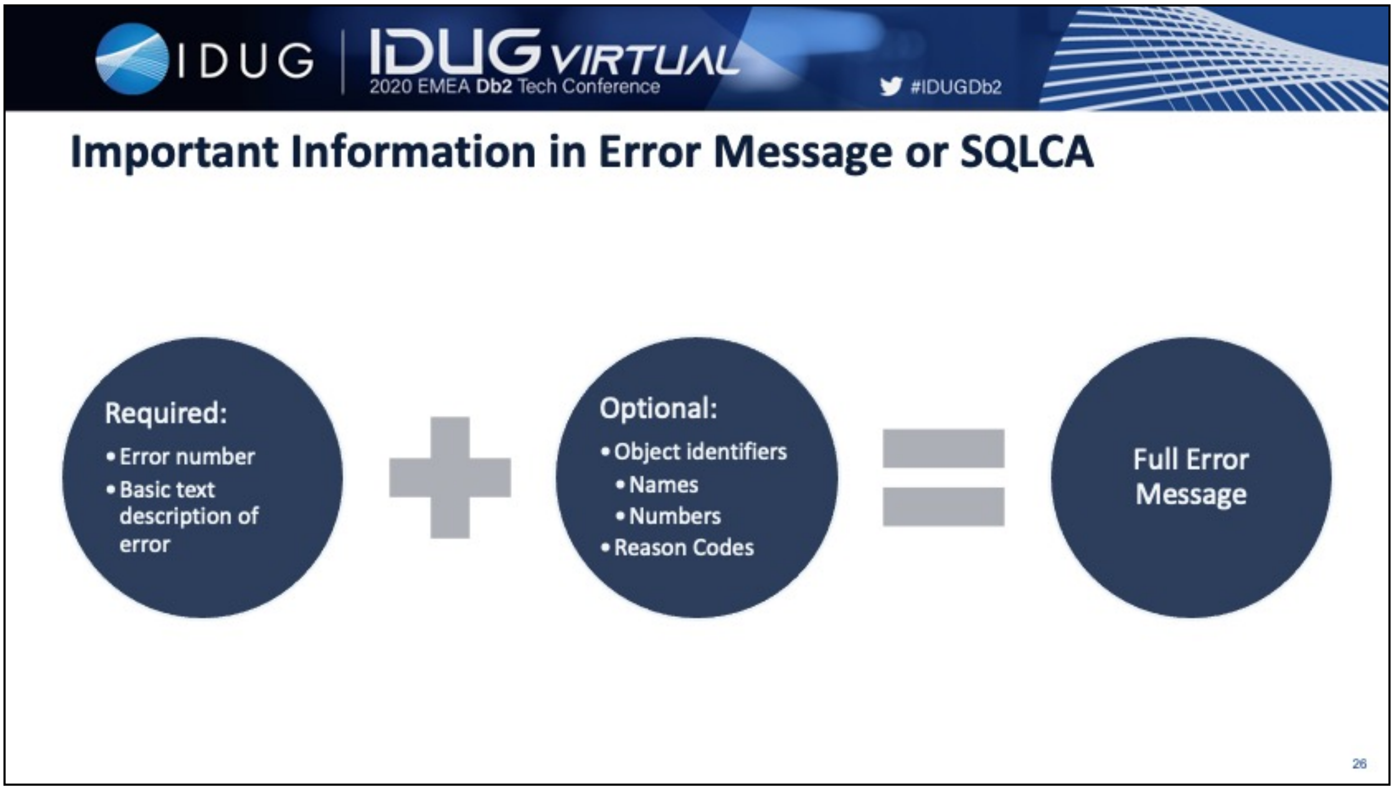
English PDF Guide	Link	Date Posted
Db2 Fundamentals		
System Monitor Guide and Reference	📄	November 2020
Database Security Guide	📄	November 2020
Performance Tuning	📄	August 2020
Administration		
Data Movement Utilities Guide and Reference	📄	August 2020
Partition and Clustering Guide	📄	August 2020
Text Search Guide	📄	August 2020
Data Recovery and High Availability Guide and Reference	📄	August 2020
Application Development		
Developing Embedded SQL and XQuery Database Applications	📄	August 2020
SQL Applications	📄	August 2020
SQL Procedural Languages Support	📄	August 2020
Developing Perl, PHP, Python, and Ruby on Rails Applications	📄	August 2020
Reference		
API Reference	📄	August 2020
Built-In Modules Reference	📄	August 2020
Built-In Routines and Views	📄	November 2020
Command Reference	📄	August 2020
Compatibility Features for Oracle	📄	August 2020
Configuration Parameters	📄	August 2020
Developing Routines	📄	August 2020
SQL Reference	📄	November 2020
GSKCapiCmd Users Guide	📄	August 2020

24

IBM was hesitant to continue offering PDF and offline forms of the Db2 documentation with 11.1. However, an RFE (Request for Enhancement) was opened, and it was the most popular one. IBM decided to start offering offline documentation again, in the form of PDFs. Some places do not allow access to the internet while connected to their VPN. Sometimes the IBM Db2 Knowledge Center is unavailable – even if this is for short periods of time in the middle of the night, it always seems to be when you're in the middle of a major problem or change. DBAs don't always sleep. It is great to have another option available. The key is to download these before you need them.

Db2 Error Messages

A Wealth of Information in a Small Package



Important Information in Error Message or SQLCA

Required:
- Error number
- Basic text description of error

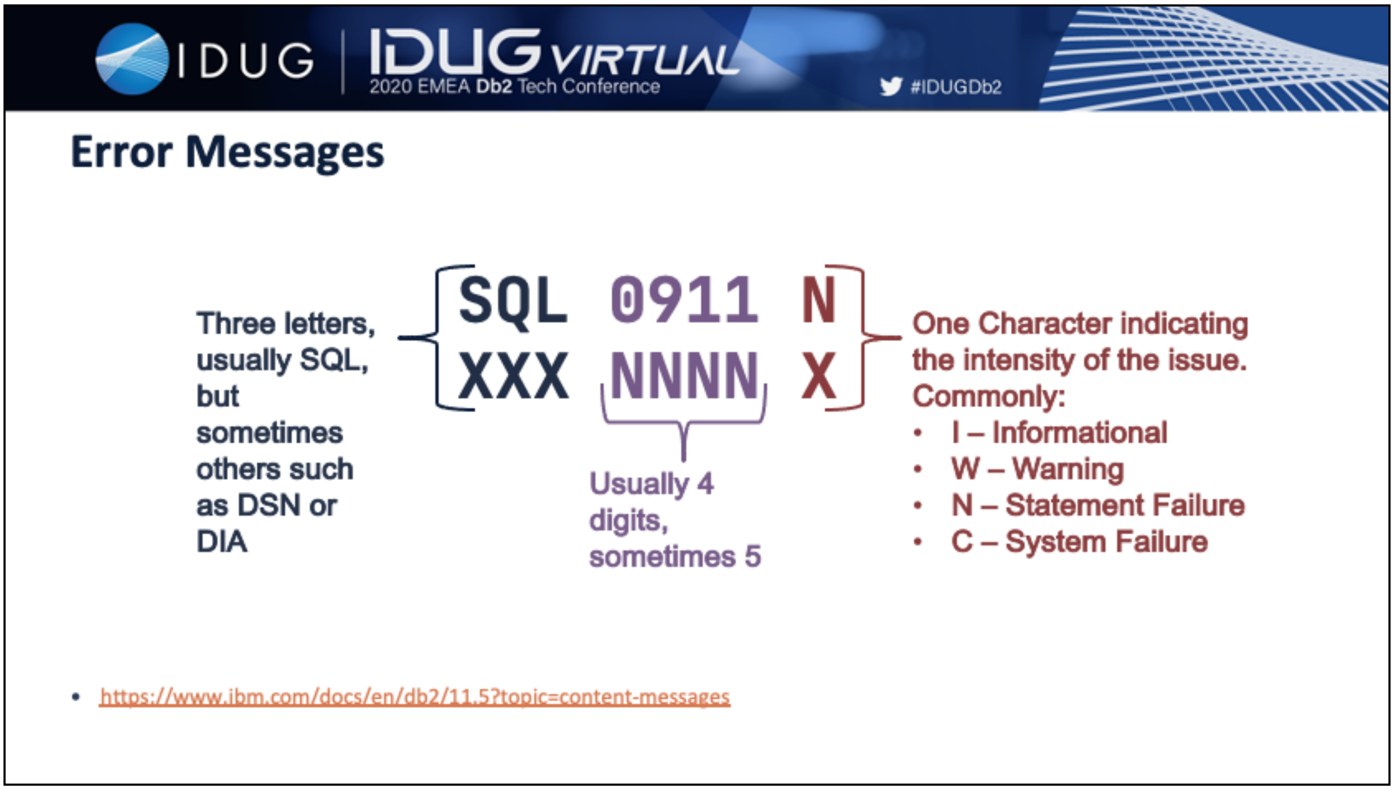
+

Optional:
- Object identifiers
  - Names
  - Numbers
- Reason Codes

=

Full Error Message

There is a wealth of information available in Db2 error messages.



Error Messages

SQL 0911 N

XXX NNNN X

Three letters, usually SQL, but sometimes others such as DSN or DIA

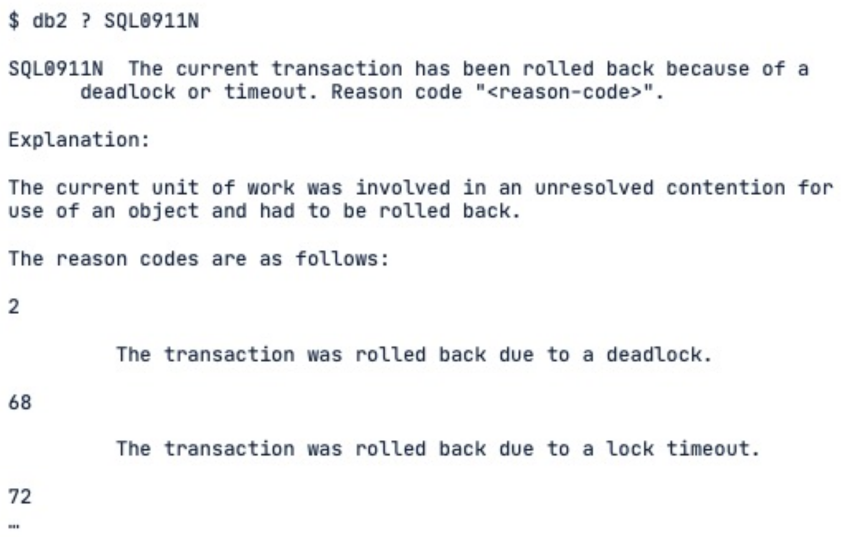
Usually 4 digits, sometimes 5

One Character indicating the intensity of the issue. Commonly:

- I – Informational
- W – Warning
- N – Statement Failure
- C – System Failure

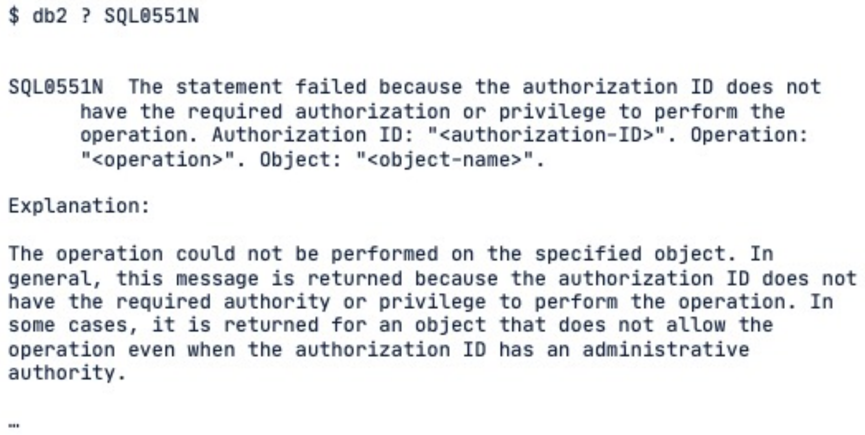
- https://www.ibm.com/docs/en/db2/11.5?topic=content-messages

Getting Information on Errors – Command Line (1|2)



```
$ db2 ? SQL0911N

SQL0911N  The current transaction has been rolled back because of a
      deadlock or timeout. Reason code "<reason-code>".

Explanation:

The current unit of work was involved in an unresolved contention for
use of an object and had to be rolled back.

The reason codes are as follows:

2

         The transaction was rolled back due to a deadlock.

68

         The transaction was rolled back due to a lock timeout.

72
…
```

Getting Information on Errors – Command Line (2|2)



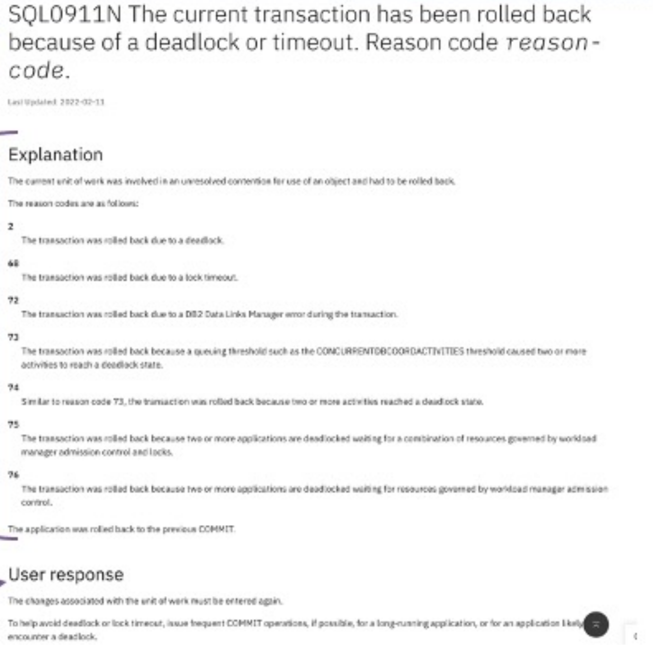
```
$ db2 ? SQL0551N


SQL0551N  The statement failed because the authorization ID does not
      have the required authorization or privilege to perform the
      operation. Authorization ID: "<authorization-ID>". Operation:
      "<operation>". Object: "<object-name>".

Explanation:

The operation could not be performed on the specified object. In
general, this message is returned because the authorization ID does not
have the required authority or privilege to perform the operation. In
some cases, it is returned for an object that does not allow the
operation even when the authorization ID has an administrative
authority.

…
```

IBM Knowledge Center: Anatomy of an Error

Error number and generic error text

Plain text description of the error and any reason codes

What to try, sometimes specific to reason codes, and other details



SQL0911N The current transaction has been rolled back because of a deadlock or timeout. Reason code *reason-code*.

Last Updated: 2022-02-11

Explanation

The current unit of work was involved in an unresolved contention for use of an object and had to be rolled back.

The reason codes are as follows:

2
The transaction was rolled back due to a deadlock.

68
The transaction was rolled back due to a lock timeout.

72
The transaction was rolled back due to a DB2 Data Links Manager error during the transaction.

73
The transaction was rolled back because a queuing threshold such as the CONCURRENTDBCOORDACTIVITIES threshold caused two or more activities to reach a deadlock state.

74
Similar to reason code 73, the transaction was rolled back because two or more activities reached a deadlock state.

75
The transaction was rolled back because two or more applications are deadlocked waiting for a combination of resources governed by workload manager admission control and locks.

76
The transaction was rolled back because two or more applications are deadlocked waiting for resources governed by workload manager admission control.

The application was rolled back to the previous COMMIT.

User response

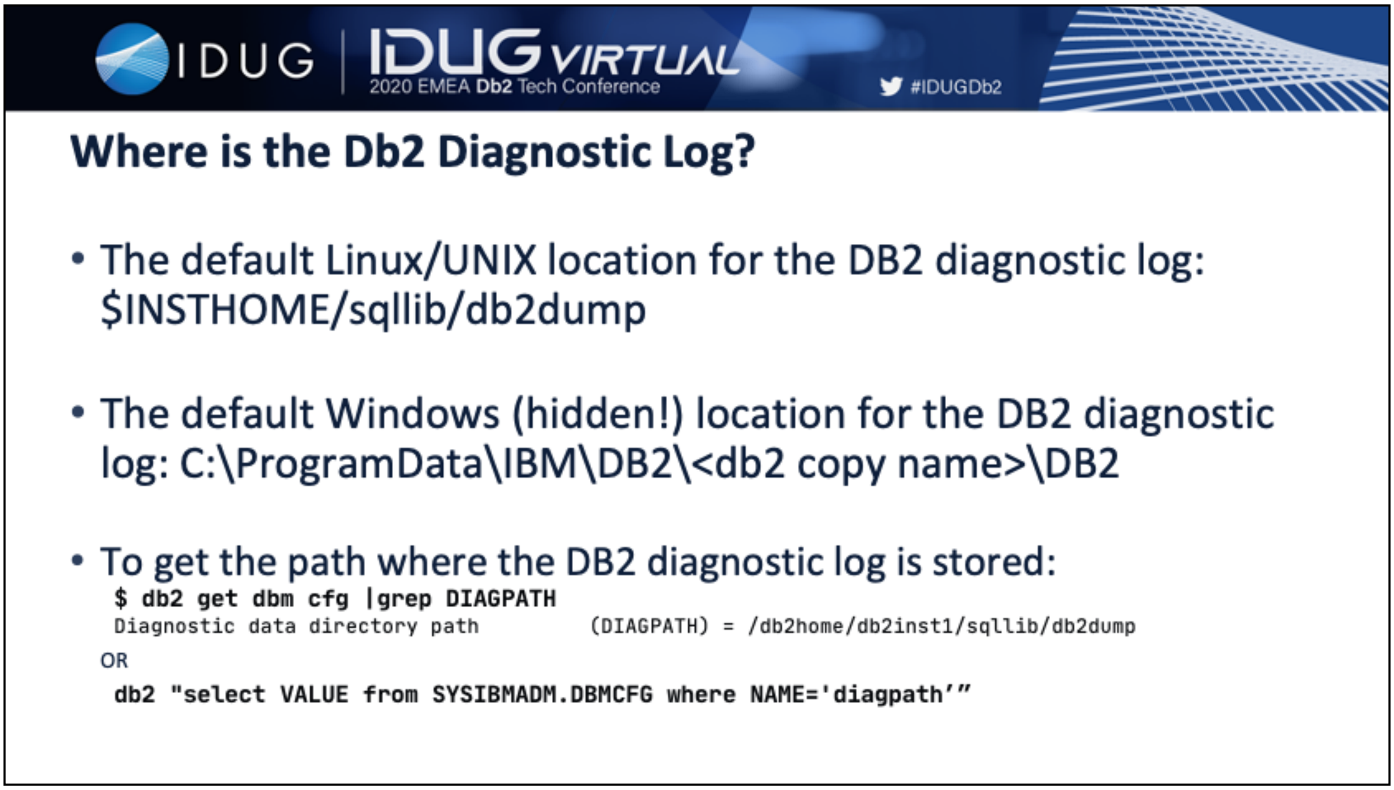
The changes associated with the unit of work must be entered again.

To help avoid deadlock or lock timeout, issue frequent COMMIT operations, if possible, for a long-running application, or for an application likely encounter a deadlock.

Each Section for an error on an error page has a well-defined structure. Knowing that structure can help you find the data you’re looking for more quickly. The first part is the actual error message in its generic format. Next, in the explanation section, there is a plain text description of the error and descriptions of each reason code, if there are any. Finally, in the user response section, there are ideas for things to try to deal with the error, including suggestions specific to different reason codes.

Db2 Diagnostic Log

The Preeminent Location for Information About System-Level Issues



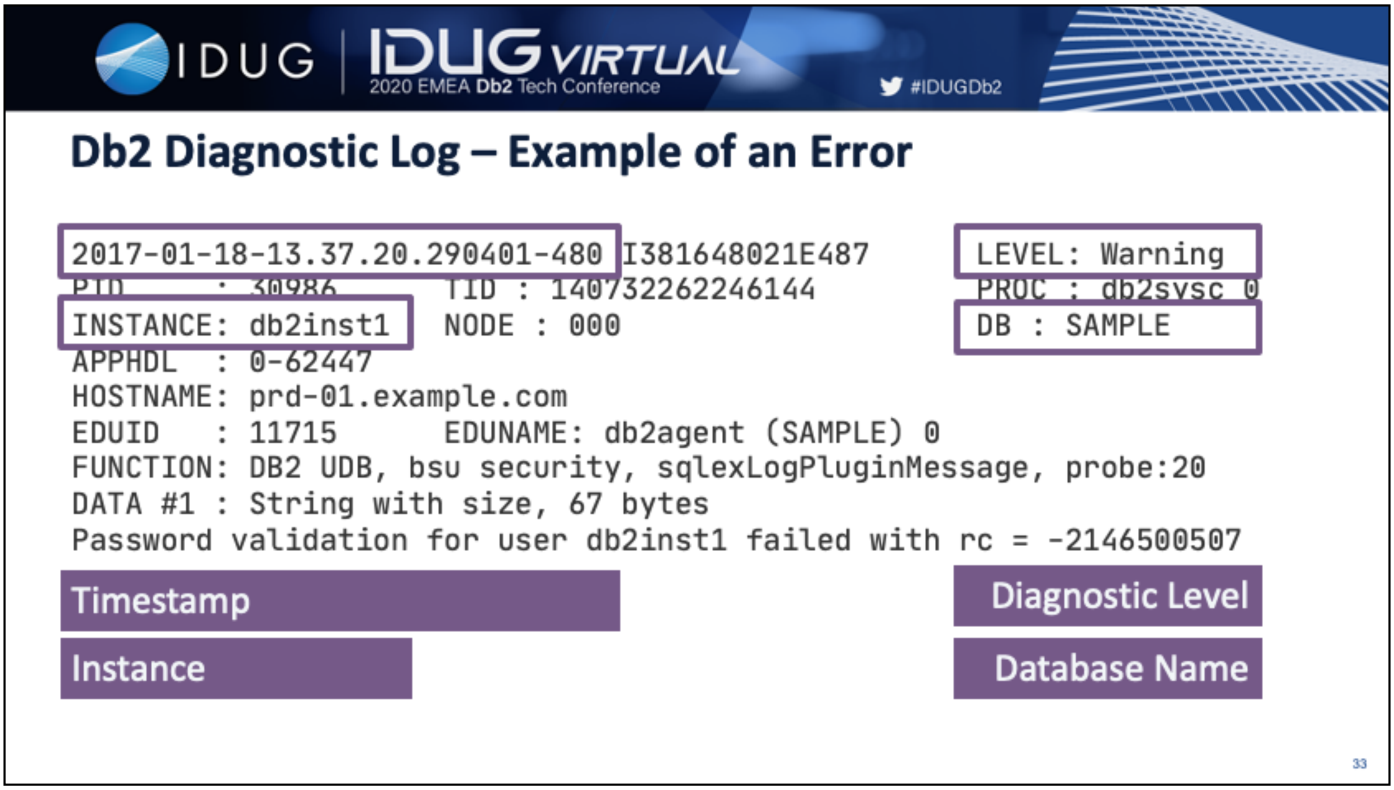
Where is the Db2 Diagnostic Log?

- The default Linux/UNIX location for the DB2 diagnostic log: $INSTHOME/sqllib/db2dump
- The default Windows (hidden!) location for the DB2 diagnostic log: C:\ProgramData\IBM\DB2\<db2 copy name>\DB2
- To get the path where the DB2 diagnostic log is stored:

```
$ db2 get dbm cfg |grep DIAGPATH
Diagnostic data directory path         (DIAGPATH) = /db2home/db2inst1/sqllib/db2dump
```

OR

```
db2 "select VALUE from SYSIBMADM.DBMCFG where NAME='diagpath’”
```



Db2 Diagnostic Log – Example of an Error

```
2017-01-18-13.37.20.290401-480 I381648021E487       LEVEL: Warning
PID     : 30986           TID : 140732262246144     PROC : db2sysc 0
INSTANCE: db2inst1        NODE : 000                DB : SAMPLE
APPHDL  : 0-62447
HOSTNAME: prd-01.example.com
EDUID   : 11715           EDUNAME: db2agent (SAMPLE) 0
FUNCTION: DB2 UDB, bsu security, sqlexLogPluginMessage, probe:20
DATA #1 : String with size, 67 bytes
Password validation for user db2inst1 failed with rc = -2146500507
```

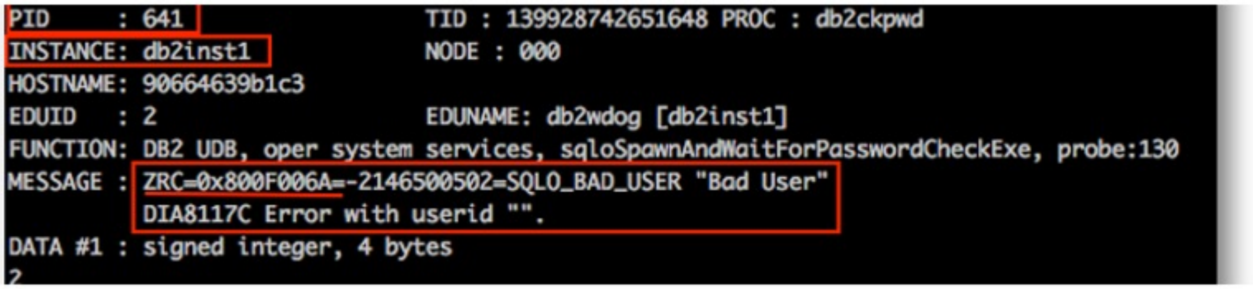
Timestamp

Diagnostic Level

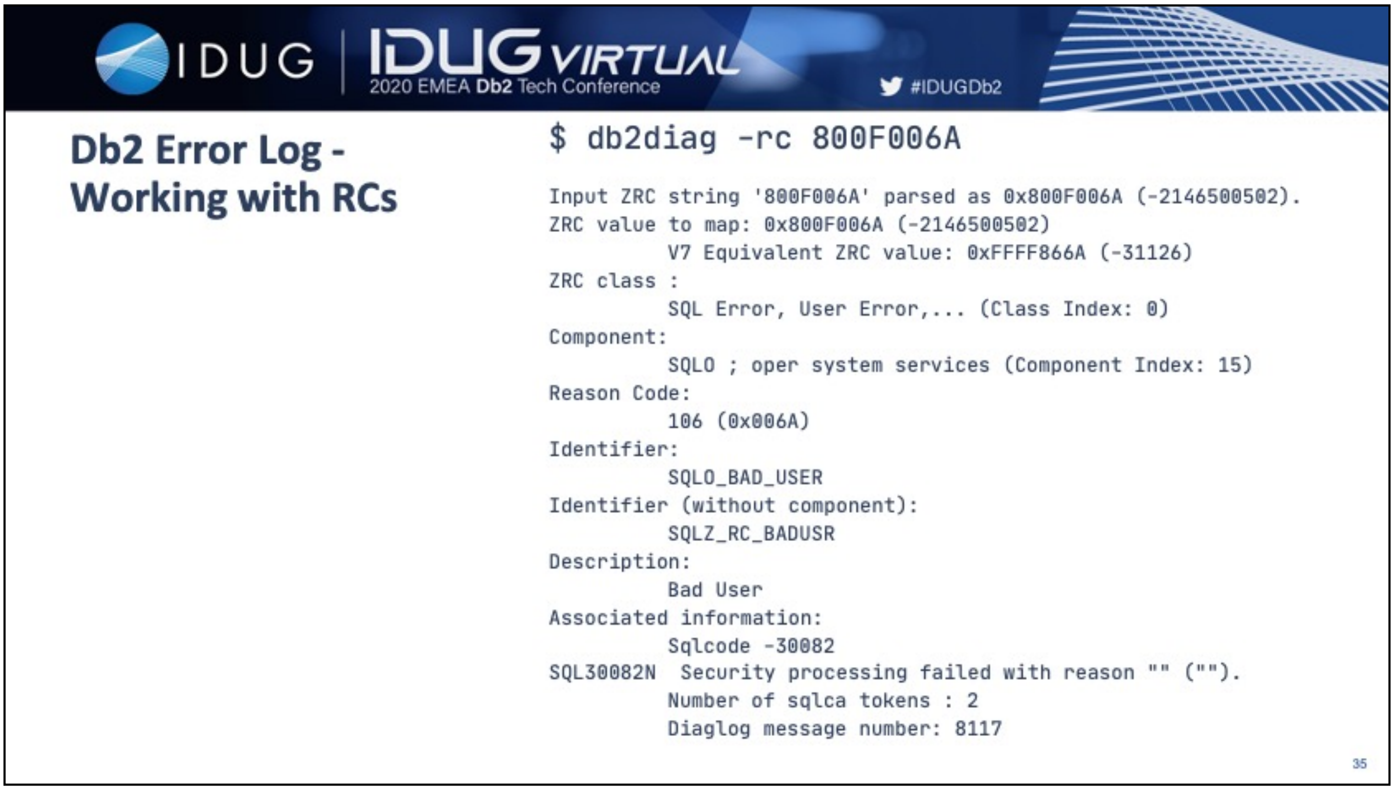
Instance

Database Name

Db2 Error Log – Working with RCs



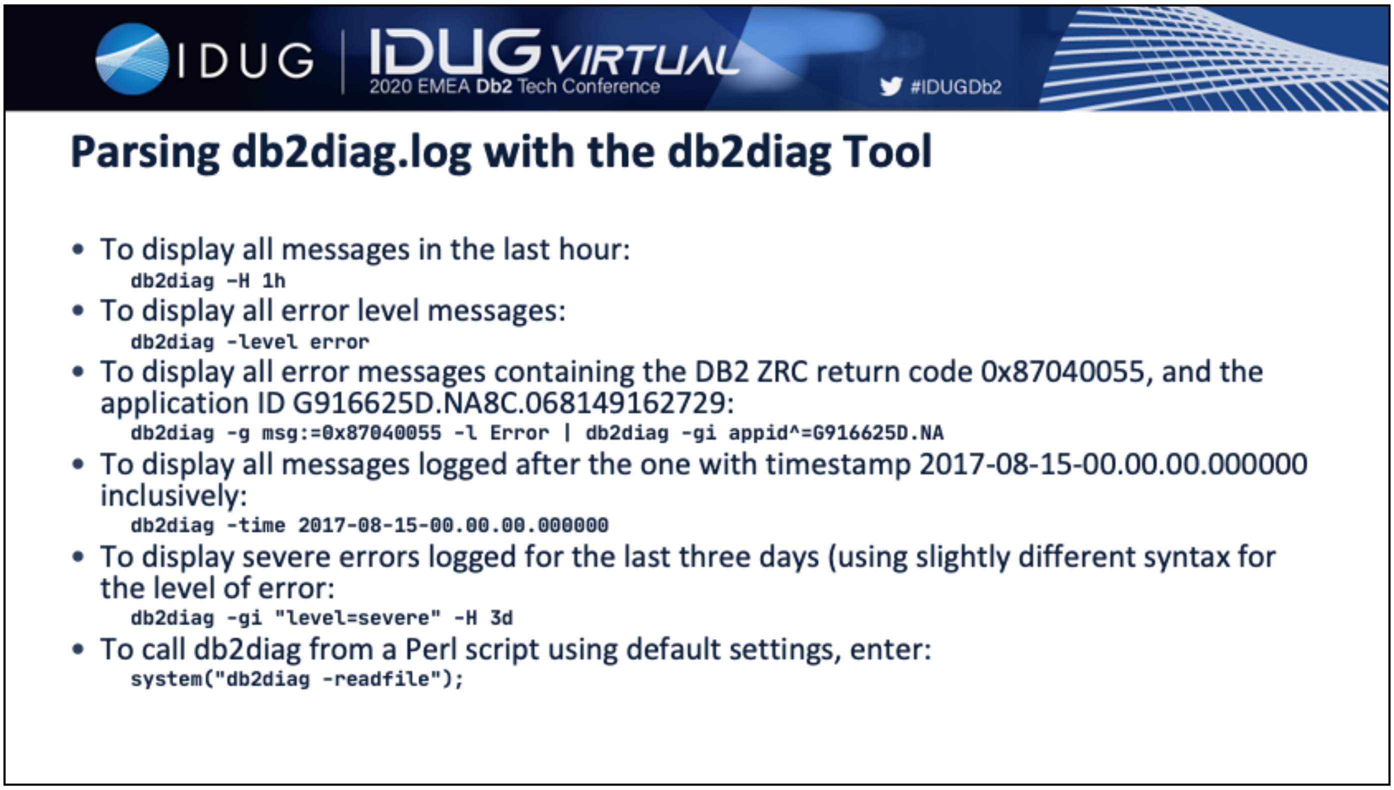
```
PID     : 641                   TID : 139928742651648 PROC : db2ckpwd
INSTANCE: db2inst1              NODE : 000
HOSTNAME: 90664639b1c3
EDUID   : 2                     EDUNAME: db2wdog [db2inst1]
FUNCTION: DB2 UDB, oper system services, sqloSpawnAndWaitForPasswordCheckExe, probe:130
MESSAGE : ZRC=0x800F006A=-2146500502=SQLO_BAD_USER "Bad User"
          DIA8117C Error with userid "".
DATA #1 : signed integer, 4 bytes
2
```



Db2 Error Log - Working with RCs

```
$ db2diag -rc 800F006A

Input ZRC string '800F006A' parsed as 0x800F006A (-2146500502).
ZRC value to map: 0x800F006A (-2146500502)
          V7 Equivalent ZRC value: 0xFFFF866A (-31126)
ZRC class :
          SQL Error, User Error,... (Class Index: 0)
Component:
          SQLO ; oper system services (Component Index: 15)
Reason Code:
          106 (0x006A)
Identifier:
          SQLO_BAD_USER
Identifier (without component):
          SQLZ_RC_BADUSR
Description:
          Bad User
Associated information:
          Sqlcode -30082
SQL30082N  Security processing failed with reason "" ("").
          Number of sqlca tokens : 2
          Diaglog message number: 8117
```

Some of the error messages in the Db2 diagnostic log can seem a bit cryptic. The db2diag system command can be used to help you translate odd error messages into something you can more easily look up. In this case, 800F006A translates to SQL30082N.


IDUG
IDUG VIRTUAL
2020 EMEA Db2 Tech Conference
#IDUGDb2

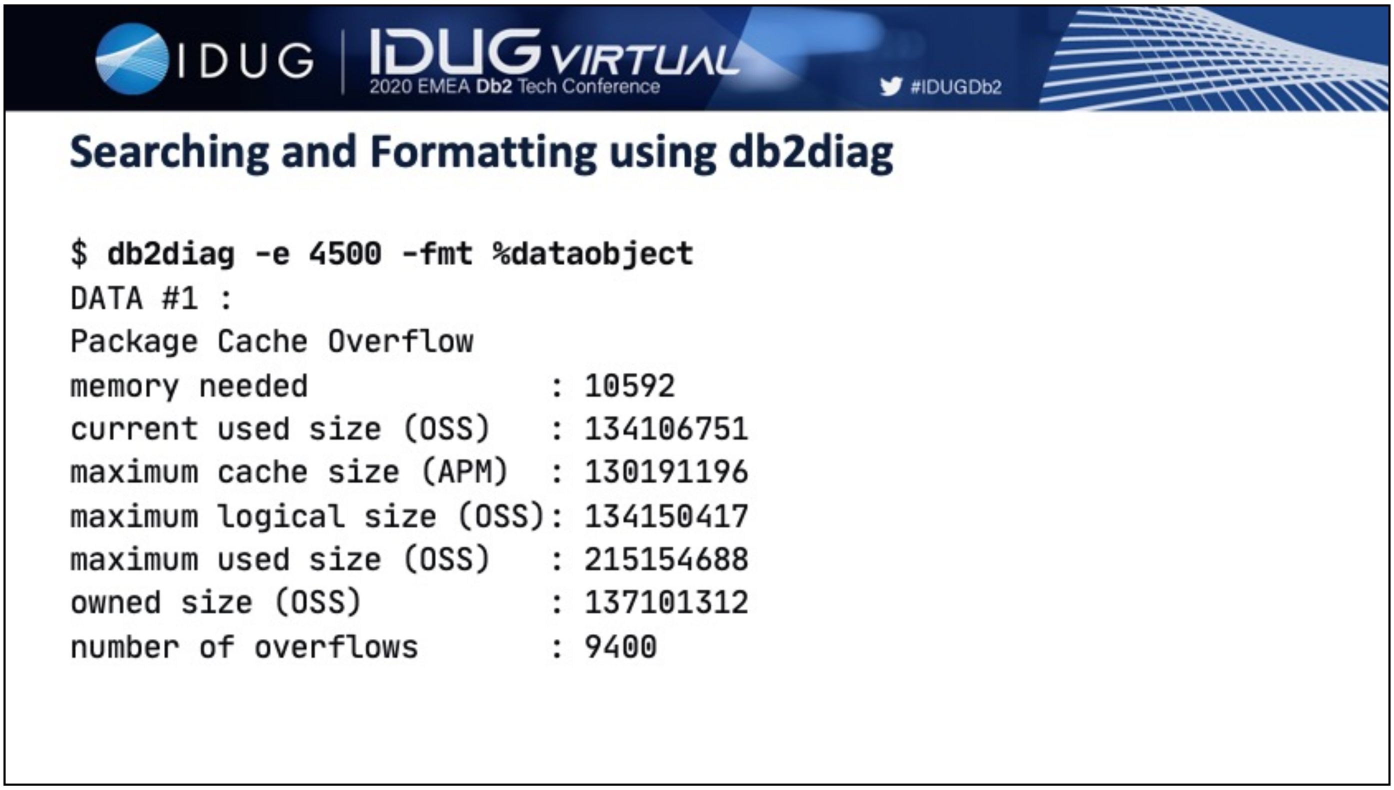
Parsing db2diag.log with the db2diag Tool

- To display all messages in the last hour:
  ```
  db2diag -H 1h
  ```
- To display all error level messages:
  ```
  db2diag -level error
  ```
- To display all error messages containing the DB2 ZRC return code 0x87040055, and the application ID G916625D.NA8C.068149162729:
  ```
  db2diag -g msg:=0x87040055 -l Error | db2diag -gi appid^=G916625D.NA
  ```
- To display all messages logged after the one with timestamp 2017-08-15-00.00.00.000000 inclusively:
  ```
  db2diag -time 2017-08-15-00.00.00.000000
  ```
- To display severe errors logged for the last three days (using slightly different syntax for the level of error:
  ```
  db2diag -gi "level=severe" -H 3d
  ```
- To call db2diag from a Perl script using default settings, enter:
  ```
  system("db2diag -readfile");
  ```

https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.admin.cmd.doc/doc/r0011728.html

https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.admin.trb.doc/doc/c0020701.html

The db2diag tool is excellent for parsing the Db2 diagnostic log and sorting through the information available there. You could also choose to query what's in the Db2 diagnostic log using the PD_GET_LOG_MSGS table function.

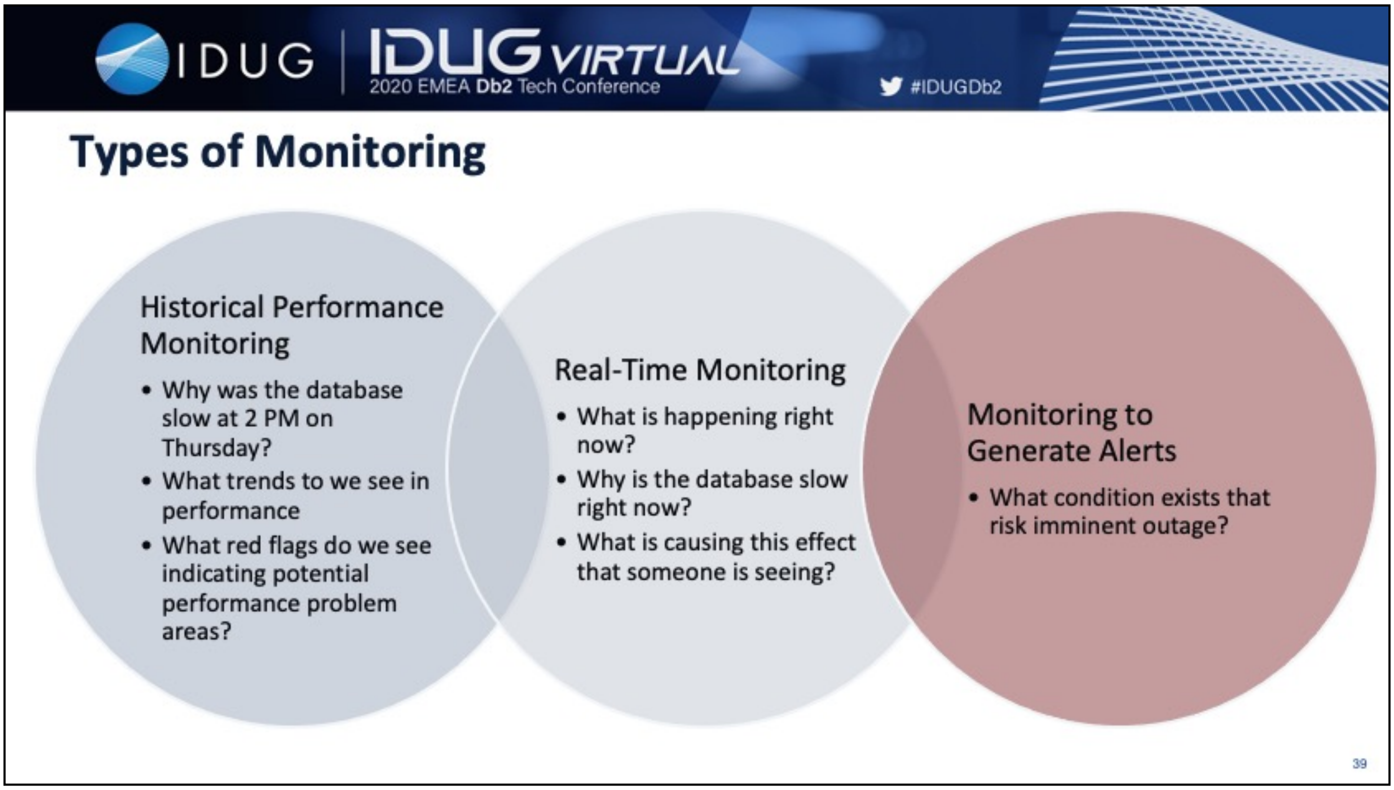


Searching and Formatting using db2diag

```
$ db2diag -e 4500 -fmt %dataobject
DATA #1 :
Package Cache Overflow
memory needed               : 10592
current used size (OSS)     : 134106751
maximum cache size (APM)    : 130191196
maximum logical size (OSS): 134150417
maximum used size (OSS)     : 215154688
owned size (OSS)            : 137101312
number of overflows         : 9400
```

Monitoring Tools

Many Methods to Understand what is Happening



IDUG | IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

Types of Monitoring

Historical Performance Monitoring

- Why was the database slow at 2 PM on Thursday?
- What trends to we see in performance
- What red flags do we see indicating potential performance problem areas?

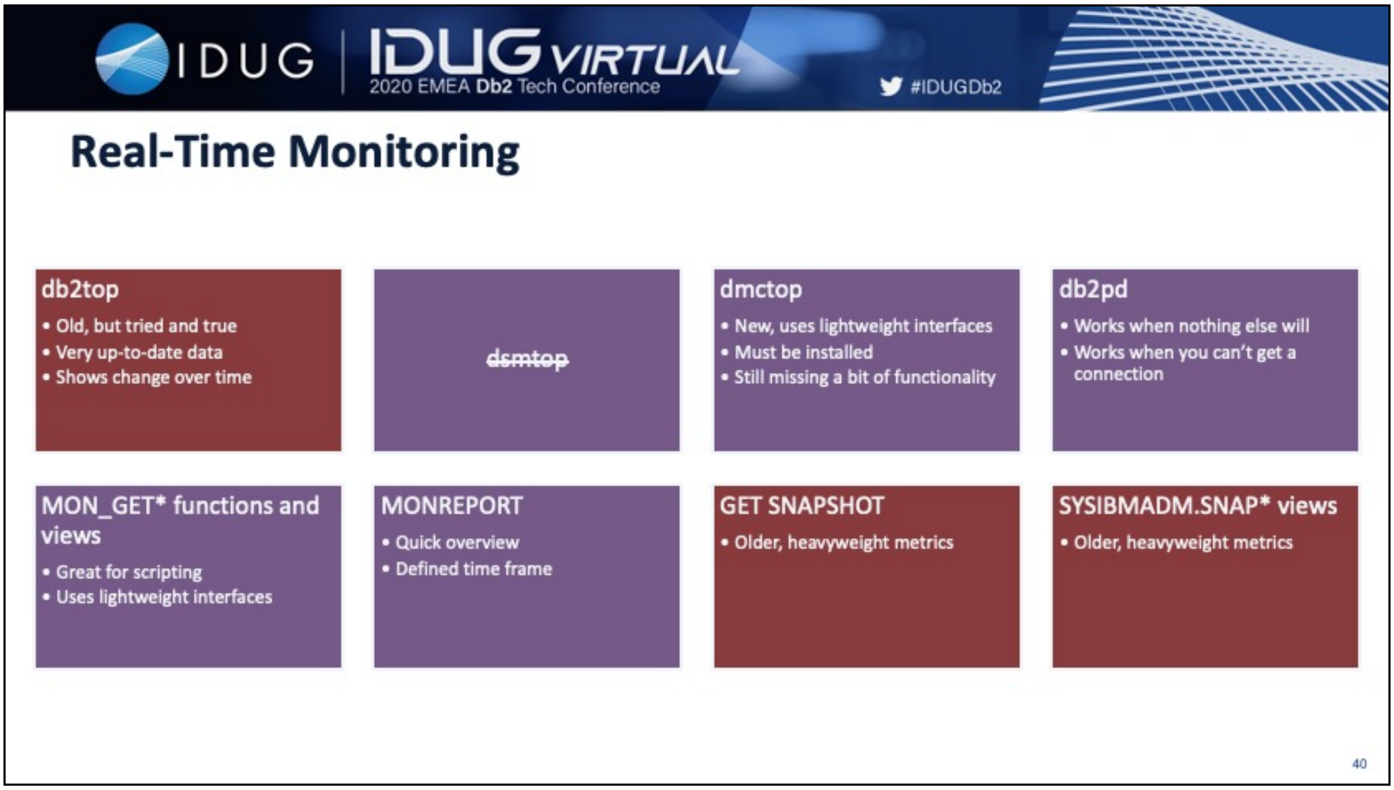
Real-Time Monitoring

- What is happening right now?
- Why is the database slow right now?
- What is causing this effect that someone is seeing?

Monitoring to Generate Alerts

- What condition exists that risk imminent outage?

Monitoring is a general term that may include a lot of different potential reasons and actions. In the case of this presentation, we’re talking about real-time monitoring to see what is going on in the database at the moment and to troubleshoot problems.



IDUG | IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

Real-Time Monitoring

db2top
- Old, but tried and true
- Very up-to-date data
- Shows change over time

~~dsmtop~~

dmctop
- New, uses lightweight interfaces
- Must be installed
- Still missing a bit of functionality

db2pd
- Works when nothing else will
- Works when you can't get a connection

MON_GET* functions and views
- Great for scripting
- Uses lightweight interfaces

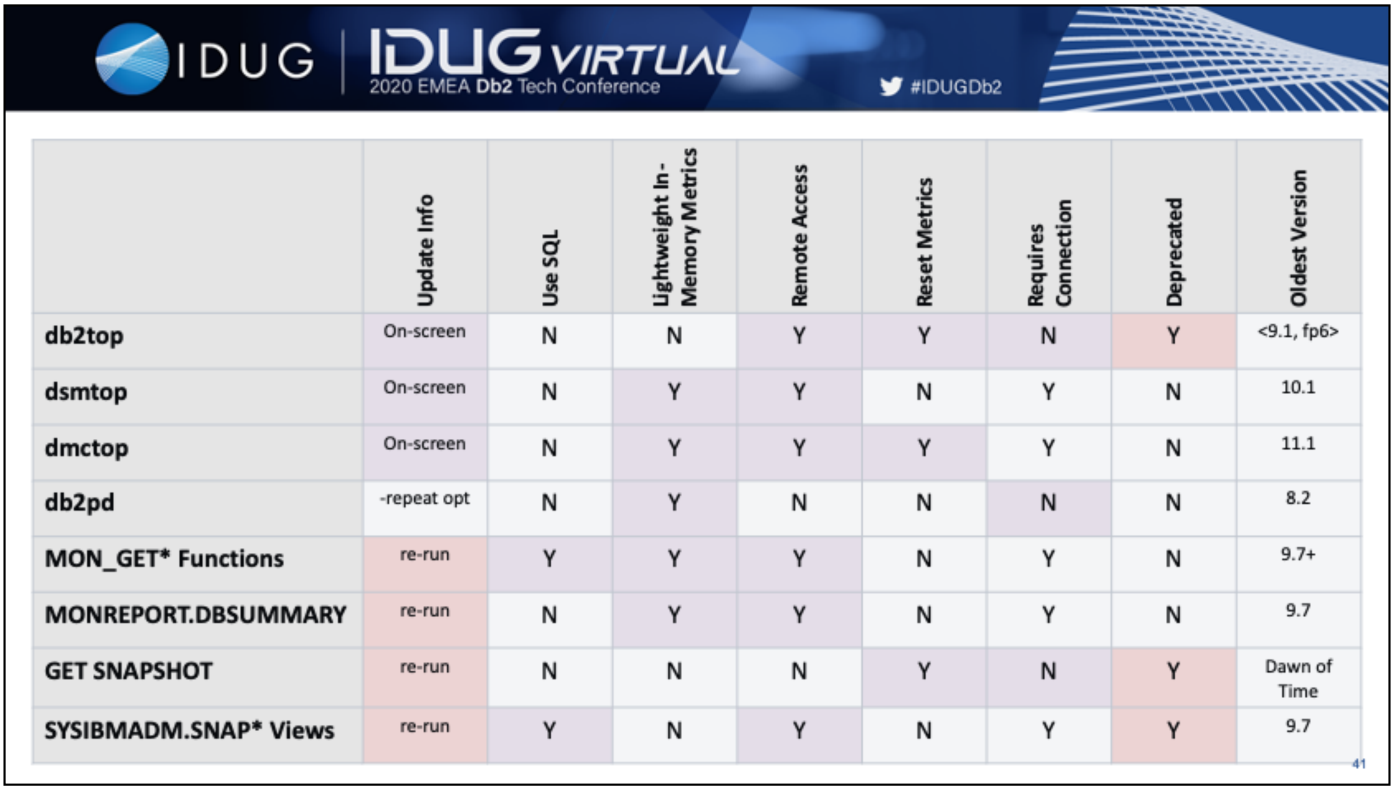
MONREPORT
- Quick overview
- Defined time frame

GET SNAPSHOT
- Older, heavyweight metrics

SYSIBMADM.SNAP* views
- Older, heavyweight metrics

40

There are a lot of choices in real-time monitoring. Which you choose depends on your purposes, constraints, and preferences.



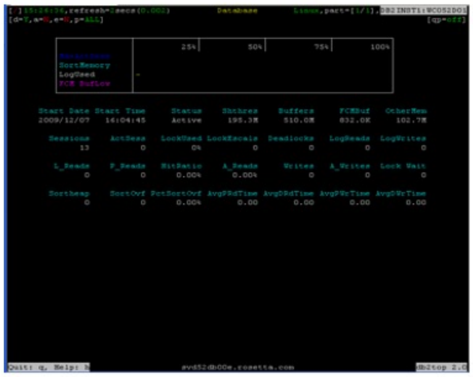
IDUG | IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

	Update Info	Use SQL	Lightweight In-Memory Metrics	Remote Access	Reset Metrics	Requires Connection	Deprecated	Oldest Version
db2top	On-screen	N	N	Y	Y	N	Y	<9.1, fp6>
dsmtop	On-screen	N	Y	Y	N	Y	N	10.1
dmctop	On-screen	N	Y	Y	Y	Y	N	11.1
db2pd	-repeat opt	N	Y	N	N	N	N	8.2
MON_GET* Functions	re-run	Y	Y	Y	N	Y	N	9.7+
MONREPORT.DBSUMMARY	re-run	N	Y	Y	N	Y	N	9.7
GET SNAPSHOT	re-run	N	N	N	Y	N	Y	Dawn of Time
SYSIBMADM.SNAP* Views	re-run	Y	N	Y	N	Y	Y	9.7

41

This is a summary of the major options for real-time monitoring and the areas in which they succeed or fail.

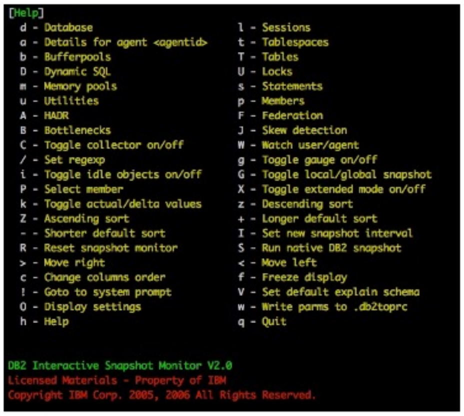
db2top



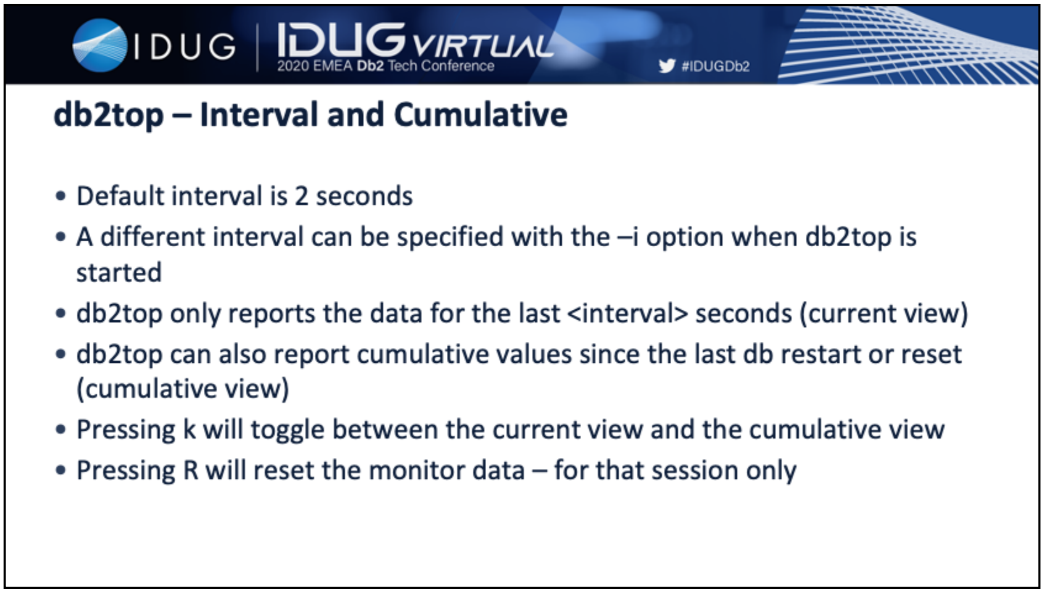
```
[/]15:24:36,refresh=2secs(0.002)          Database          Linux,part=[1/1],DB2INST1:WCOS2DO1
[d=Y,a=N,e=N,p=ALL]                                                                   [qp=off]

              +-----------+--------25%|-------50%|-------75%|------100%|
              |MaxActSess |                                            |
              |SortMemory |                                            |
              |LogUsed    | -                                          |
              |FCM BufLow |                                            |
              +-----------+--------------------------------------------+

  Start Date Start Time      Status     Shthres    Buffers     FCMBuf   OtherMem
  2009/12/07   16:04:45      Active     195.3M     510.0M     832.0K     102.7M

    Sessions    ActSess    LockUsed LockEscals  Deadlocks   LogReads  LogWrites
          13          0          0%          0          0          0          0

     L_Reads    P_Reads    HitRatio    A_Reads     Writes   A_Writes  Lock Wait
           0          0       0.00%      0.00%          0          0          0

    Sortheap    SortOvf  PctSortOvf AvgPRdTime AvgDRdTime AvgPWrTime AvgDWrTime
           0          0       0.00%       0.00       0.00       0.00       0.00

Quit: q, Help: h              svd52db00e.rosetta.com                         db2top 2.0
```

db2top – Help Screen

- Pressing the h key in db2top will get you the help screen, which has a list of all the nifty options



```
[Help]
  d - Database                         l - Sessions
  a - Details for agent <agentid>      t - Tablespaces
  b - Bufferpools                      T - Tables
  D - Dynamic SQL                      U - Locks
  m - Memory pools                     s - Statements
  u - Utilities                        p - Members
  A - HADR                             F - Federation
  B - Bottlenecks                      J - Skew detection
  C - Toggle collector on/off          W - Watch user/agent
  / - Set regexp                       g - Toggle gauge on/off
  i - Toggle idle objects on/off       G - Toggle local/global snapshot
  P - Select member                    X - Toggle extended mode on/off
  k - Toggle actual/delta values       z - Descending sort
  Z - Ascending sort                   + - Longer default sort
  - - Shorter default sort             I - Set new snapshot interval
  R - Reset snapshot monitor           S - Run native DB2 snapshot
  > - Move right                       < - Move left
  c - Change columns order             f - Freeze display
  ! - Goto to system prompt            V - Set default explain schema
  O - Display settings                 w - Write parms to .db2toprc
  h - Help                             q - Quit


DB2 Interactive Snapshot Monitor V2.0
Licensed Materials - Property of IBM
Copyright IBM Corp. 2005, 2006 All Rights Reserved.
```

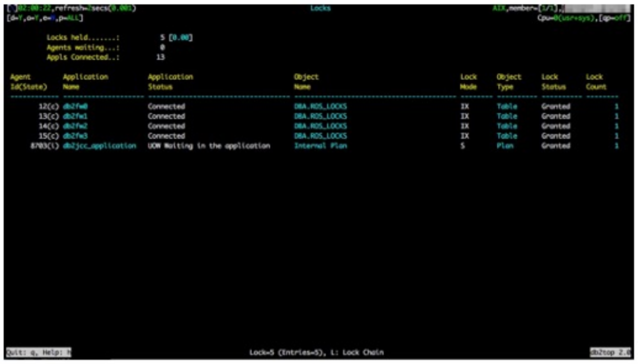


db2top – Interval and Cumulative

- Default interval is 2 seconds
- A different interval can be specified with the –i option when db2top is started
- db2top only reports the data for the last <interval> seconds (current view)
- db2top can also report cumulative values since the last db restart or reset (cumulative view)
- Pressing k will toggle between the current view and the cumulative view
- Pressing R will reset the monitor data – for that session only

db2top – Lock Chaining

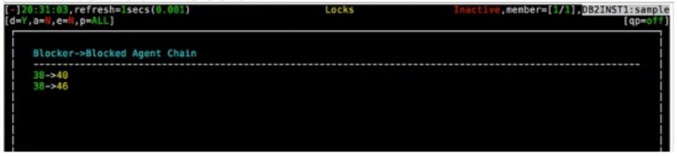
- Use U to get to the lock screen:



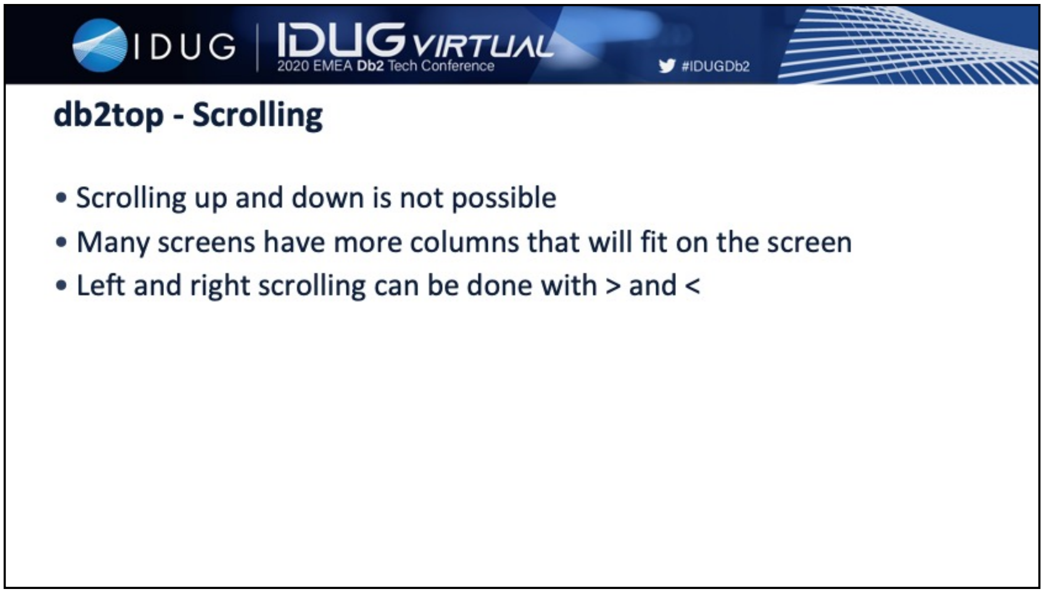
```
[-]02:00:22,refresh=2secs(0.005)                          Locks                          AIX,member=[1/1],[redacted]
[d=Y,a=Y,e=N,p=ALL]                                                                          Cpu=0(usr+sys),[qp=off]

        Locks held.......:          5 [0.00]
        Agents waiting...:          0
        Appls Connected..:         13

 Agent        Application          Application                       Object                             Lock     Object    Lock      Lock
 Id(State)    Name                 Status                            Name                               Mode     Type      Status    Count
 -----------  -------------------  --------------------------------  ---------------------------------  -------  --------  --------  -------
      12(c)   db2fw0               Connected                         DBA.RDS_LOCKS                      IX       Table     Granted         1
      13(c)   db2fw1               Connected                         DBA.RDS_LOCKS                      IX       Table     Granted         1
      14(c)   db2fw2               Connected                         DBA.RDS_LOCKS                      IX       Table     Granted         1
      15(c)   db2fw3               Connected                         DBA.RDS_LOCKS                      IX       Table     Granted         1
    8703(i)   db2jcc_application   UOW Waiting in the application    Internal Plan                      S        Plan      Granted         1









Quit: q, Help: h                                  Lock=5 (Entries=5), L: Lock Chain                                          db2top 2.0
```

db2top – Lock Chaining

- If locks are waiting, db2top can quickly tell which connection is waiting on which connection

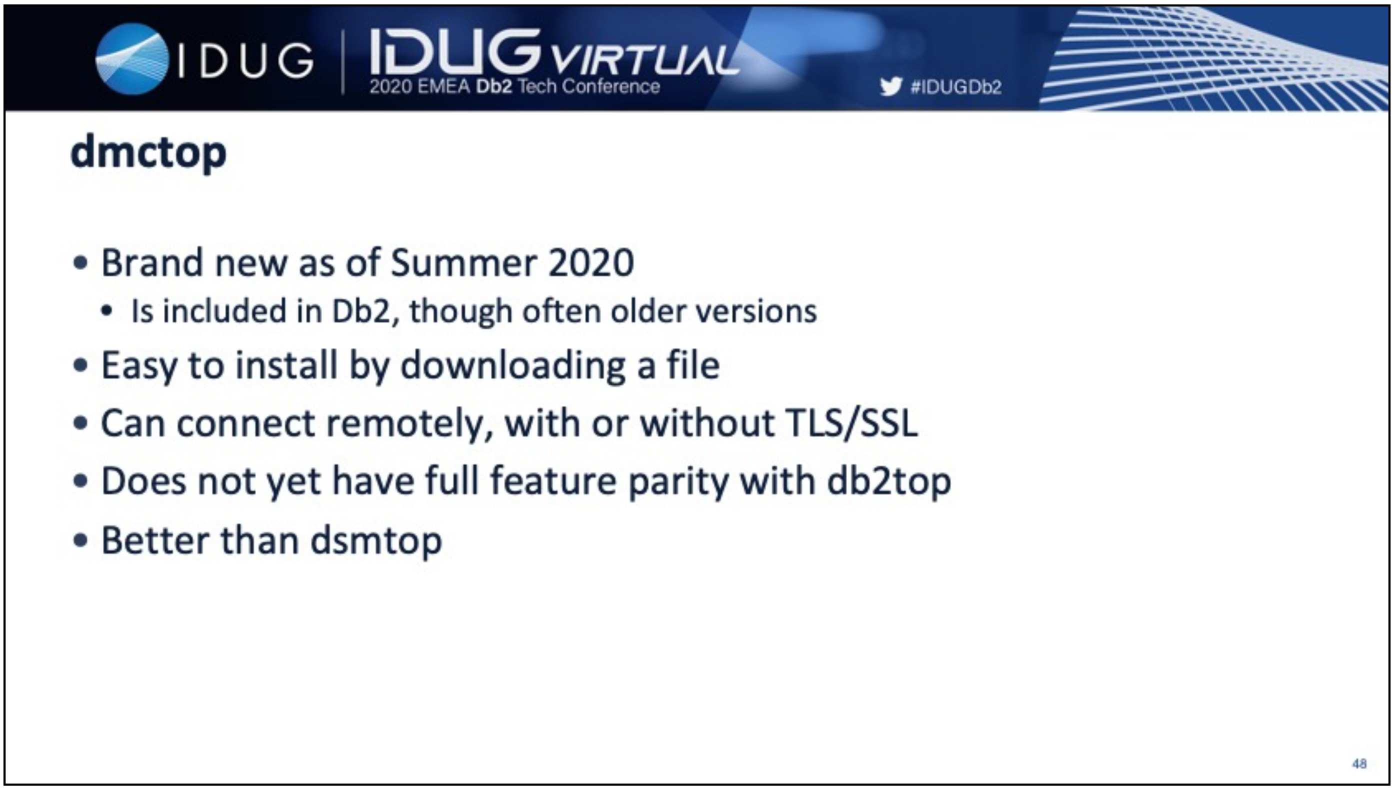


```
[-]20:31:03,refresh=1secs(0.001)                    Locks              Inactive,member=[1/1],DB2INST1:sample
[d=Y,a=N,e=N,p=ALL]                                                                                [qp=off]
 ---------------------------------------------------------------------------------------------------------
|                                                                                                         |
|   Blocker->Blocked Agent Chain                                                                          |
|   ---------------------------------------------------------------------------------------------------   |
|   38->40                                                                                                |
|   38->46                                                                                                |
|                                                                                                         |
|                                                                                                         |
|                                                                                                         |
|                                                                                                         |
```



db2top - Scrolling

- Scrolling up and down is not possible
- Many screens have more columns that will fit on the screen
- Left and right scrolling can be done with > and <

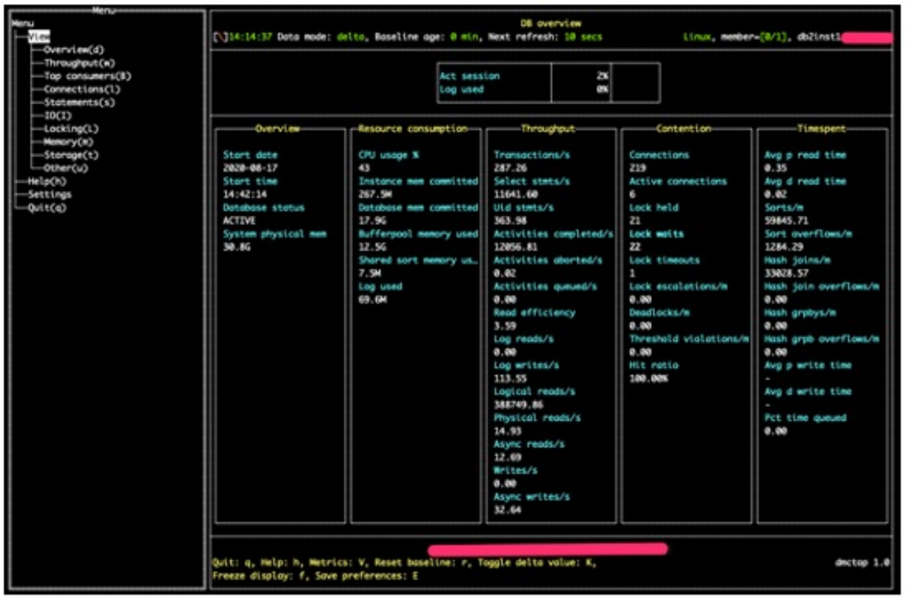


dmctop

- Brand new as of Summer 2020
  - Is included in Db2, though often older versions
- Easy to install by downloading a file
- Can connect remotely, with or without TLS/SSL
- Does not yet have full feature parity with db2top
- Better than dsmtop

dmctop

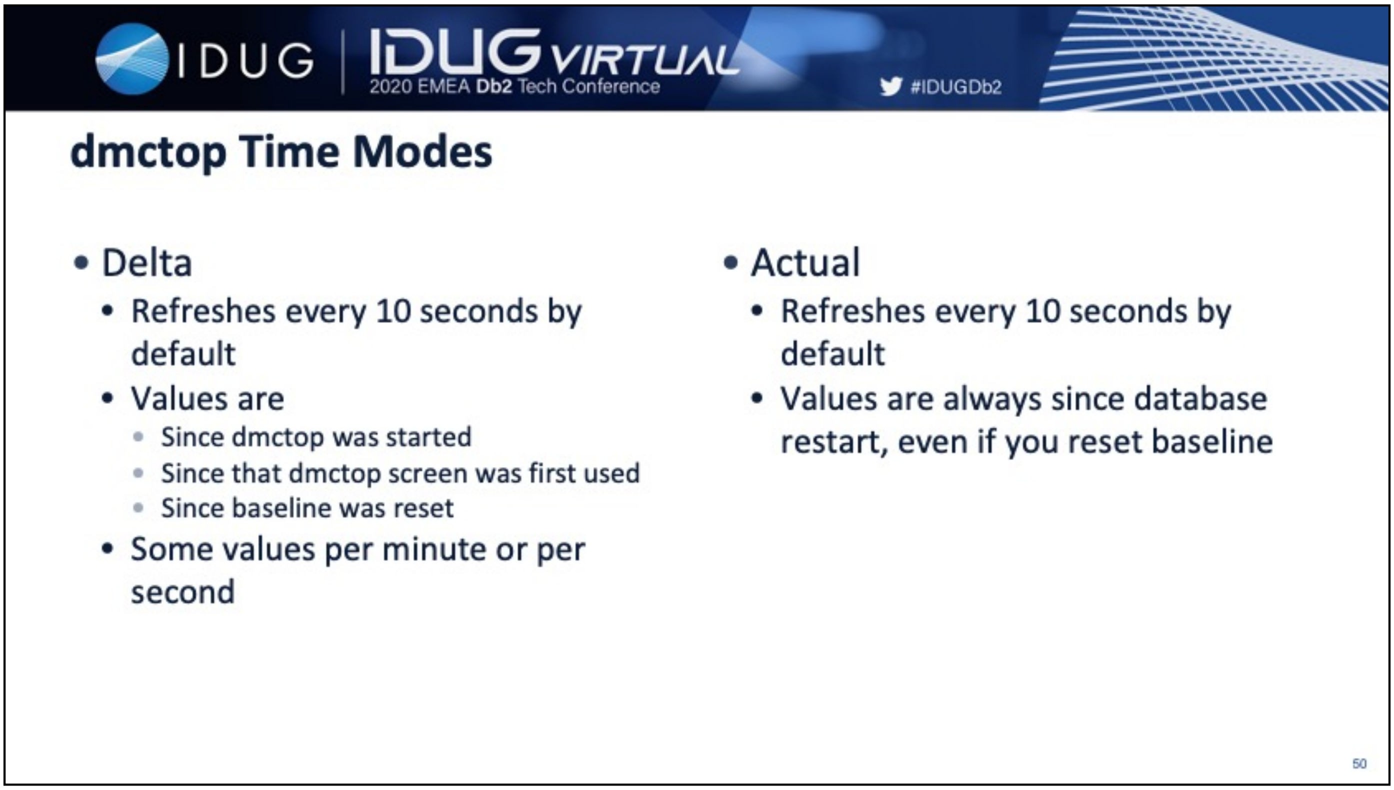
- Start with `dmctop -d dbname`
- CPU usage not always accurate
- Read efficiency!
- db2top shortcuts work when the Menu pane is active



```
Menu
Menu
├─View
│ ├─Overview(d)
│ ├─Throughput(w)
│ ├─Top consumers(B)
│ ├─Connections(l)
│ ├─Statements(s)
│ ├─IO(I)
│ ├─Locking(L)
│ ├─Memory(m)
│ ├─Storage(t)
│ └─Other(u)
├─Help(h)
├─Settings
└─Quit(q)
```

```
                                        DB overview
[\]14:14:37 Data mode: delta, Baseline age: 0 min, Next refresh: 10 secs          Linux, member=[0/1], db2inst1[redacted]

                          Act session          2%
                          Log used             0%

Overview              Resource consumption     Throughput               Contention               Timespent

Start date            CPU usage %              Transactions/s           Connections              Avg p read time
2020-08-17            43                       287.26                   219                      0.35
Start time            Instance mem committed   Select stmts/s           Active connections       Avg d read time
14:42:14              267.5M                   11641.60                 6                        0.02
Database status       Database mem committed   Uid stmts/s              Lock held                Sorts/m
ACTIVE                17.9G                    363.98                   21                       59845.71
System physical mem   Bufferpool memory used   Activities completed/s   Lock waits               Sort overflows/m
30.8G                 12.5G                    12056.81                 22                       1284.29
                      Shared sort memory us…   Activities aborted/s     Lock timeouts            Hash joins/m
                      7.5M                     0.02                     1                        33028.57
                      Log used                 Activities queued/s      Lock escalations/m       Hash join overflows/m
                      69.6M                    0.00                     0.00                     0.00
                                               Read efficiency          Deadlocks/m              Hash grpbys/m
                                               3.59                     0.00                     0.00
                                               Log reads/s              Threshold violations/m   Hash grpb overflows/m
                                               0.00                     0.00                     0.00
                                               Log writes/s             Hit ratio                Avg p write time
                                               113.55                   100.00%                  -
                                               Logical reads/s                                   Avg d write time
                                               388749.86                                         -
                                               Physical reads/s                                  Pct time queued
                                               14.93                                             0.00
                                               Async reads/s
                                               12.69
                                               Writes/s
                                               0.00
                                               Async writes/s
                                               32.64

                    [redacted]
Quit: q, Help: h, Metrics: V, Reset baseline: r, Toggle delta value: K,                          dmctop 1.0
Freeze display: f, Save preferences: E
```

dmctop has a lot of great information on the database overview screen. A few things to note are that I and some friends have seen the CPU usage metric not be very accurate. Personally, I love the read efficiency metric, which was not available in db2top. On this screen, there are two panes – the left menu pane and the right output pane. Keys pressed currently have a different meaning depending on which pane is active, so if keys aren't doing what you think they should, try hitting ESC to put the focus back on the menu pane.



IDUG | IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

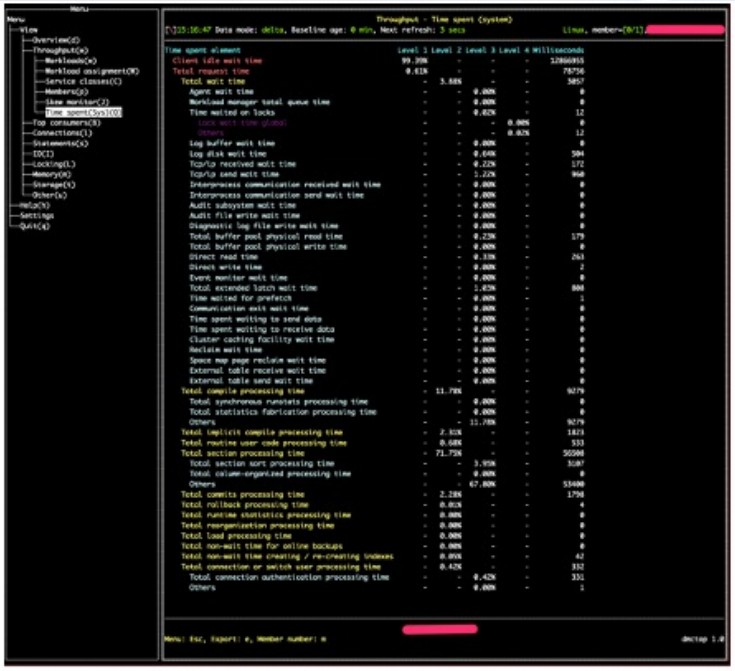
dmctop Time Modes

- Delta
  - Refreshes every 10 seconds by default
  - Values are
    - Since dmctop was started
    - Since that dmctop screen was first used
    - Since baseline was reset
  - Some values per minute or per second
- Actual
  - Refreshes every 10 seconds by default
  - Values are always since database restart, even if you reset baseline

50

There are two different time modes for dmctop. The Delta mode allows values to be reset.

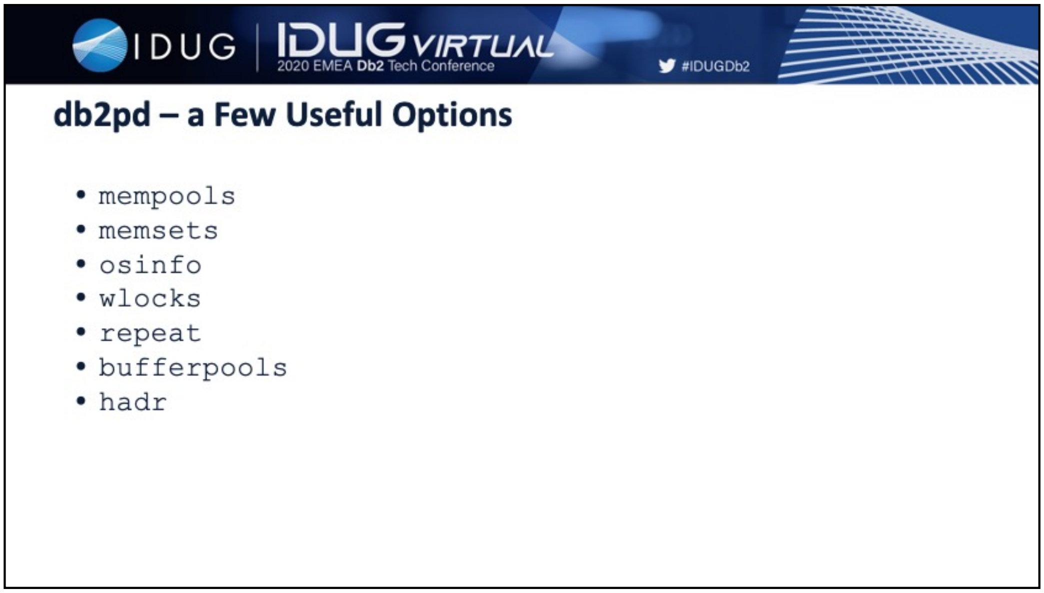
dmctop: Time Spent



```
Menu
├─View
│  ├─Overview(d)
│  ├─Throughput(e)
│  │  ├─Workloads(w)
│  │  ├─Workload assignment(W)
│  │  ├─Service classes(C)
│  │  ├─Members(p)
│  │  ├─Skew monitor(J)
│  │  └─Time spent(Sys)(O)
│  ├─Top consumers(B)
│  ├─Connections(l)
│  ├─Statements(s)
│  ├─IO(I)
│  ├─Locking(L)
│  ├─Memory(m)
│  ├─Storage(t)
│  └─Other(x)
├─Help(h)
├─Settings
└─Quit(q)
```

```
                              Throughput - Time spent (system)
[\]15:16:47 Data mode: delta, Baseline age: 0 min, Next refresh: 3 secs          Linux, member=[0/1],[redacted]

Time spent element                                   Level 1 Level 2 Level 3 Level 4 Milliseconds
  Client idle wait time                               99.39%       -       -       -     12866055
  Total request time                                   0.61%       -       -       -        78756
    Total wait time                                        -   3.88%       -       -         3057
      Agent wait time                                      -       -   0.00%       -            0
      Workload manager total queue time                    -       -   0.00%       -            0
      Time waited on locks                                 -       -   0.02%       -           12
        Lock wait time global                              -       -       -   0.00%            0
        Others                                             -       -       -   0.02%           12
      Log buffer wait time                                 -       -   0.00%       -            0
      Log disk wait time                                   -       -   0.64%       -          504
      Tcp/ip received wait time                            -       -   0.22%       -          172
      Tcp/ip send wait time                                -       -   1.22%       -          960
      Interprocess communication received wait time        -       -   0.00%       -            0
      Interprocess communication send wait time            -       -   0.00%       -            0
      Audit subsystem wait time                            -       -   0.00%       -            0
      Audit file write wait time                           -       -   0.00%       -            0
      Diagnostic log file write wait time                  -       -   0.00%       -            0
      Total buffer pool physical read time                 -       -   0.23%       -          179
      Total buffer pool physical write time                -       -   0.00%       -            0
      Direct read time                                     -       -   0.33%       -          263
      Direct write time                                    -       -   0.00%       -            2
      Event monitor wait time                              -       -   0.00%       -            0
      Total extended latch wait time                       -       -   1.03%       -          808
      Time waited for prefetch                             -       -   0.00%       -            1
      Communication exit wait time                         -       -   0.00%       -            0
      Time spent waiting to send data                      -       -   0.00%       -            0
      Time spent waiting to receive data                   -       -   0.00%       -            0
      Cluster caching facility wait time                   -       -   0.00%       -            0
      Reclaim wait time                                    -       -   0.00%       -            0
      Space map page reclaim wait time                     -       -   0.00%       -            0
      External table receive wait time                     -       -   0.00%       -            0
      External table send wait time                        -       -   0.00%       -            0
    Total compile processing time                          -  11.78%       -       -         9279
      Total synchronous runstats processing time           -       -   0.00%       -            0
      Total statistics fabrication processing time         -       -   0.00%       -            0
      Others                                               -       -  11.78%       -         9279
    Total implicit compile processing time                 -   2.31%       -       -         1823
    Total routine user code processing time                -   0.68%       -       -          533
    Total section processing time                          -  71.75%       -       -        56508
      Total section sort processing time                   -       -   3.95%       -         3107
      Total column-organized processing time               -       -   0.00%       -            0
      Others                                               -       -  67.80%       -        53400
    Total commits processing time                          -   2.28%       -       -         1798
    Total rollback processing time                         -   0.01%       -       -            4
    Total runtime statistics processing time               -   0.00%       -       -            0
    Total reorganization processing time                   -   0.00%       -       -            0
    Total load processing time                             -   0.00%       -       -            0
    Total non-wait time for online backups                 -   0.00%       -       -            0
    Total non-wait time creating / re-creating indexes     -   0.05%       -       -           42
    Total connection or switch user processing time        -   0.42%       -       -          332
      Total connection authentication processing time      -       -   0.42%       -          331
      Others                                               -       -   0.00%       -            1

                                   [redacted]
Menu: Esc, Export: e, Member number: m                                              dmctop 1.0
```

My favorite screen in dmctop is the time spent screen. It provides a nice overview of where Db2 is spending time that allows you to drill down into problem areas or quickly see where a problem might be lurking.



IDUG | IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

db2pd – a Few Useful Options

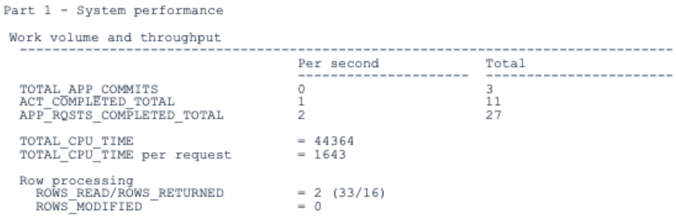
- `mempools`
- `memsets`
- `osinfo`
- `wlocks`
- `repeat`
- `bufferpools`
- `hadr`

Db2pd has a few advantages. First, it can be used without a database connection, meaning that if the database is hung, you can still get information. It is also a very lightweight interface, so there is very little risk of it adding load to a busy server. I find it most useful in a few areas, listed here. I often also find the –repeat option useful to have it repeat the command every minute or so – allowing me to track the progress of something like HADR.

IDUG | IDUG VIRTUAL 2020 EMEA **Db2** Tech Conference #IDUGDb2

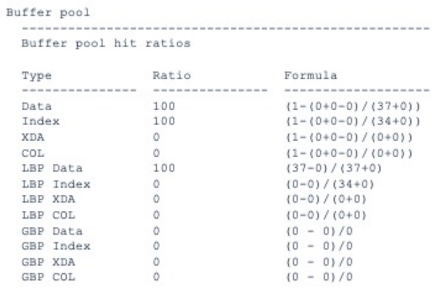
MONREPORT.DBSUMMARY

- Good summary of data
- Snapshot-like format that uses more recent monitoring infrastructure
- Snipets…

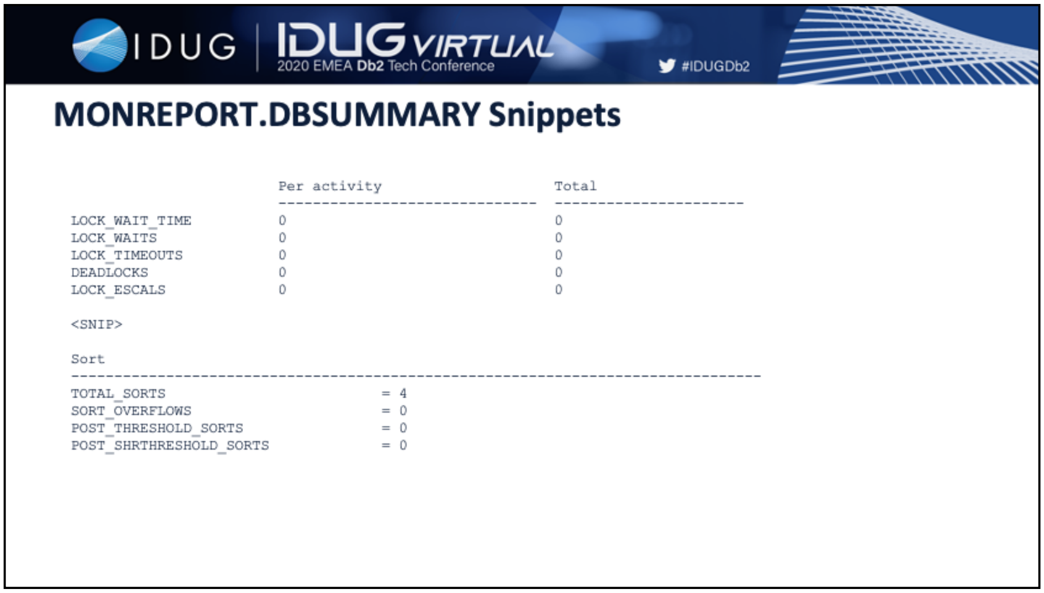


```
Part 1 - System performance

 Work volume and throughput
  --------------------------------------------------------------------------------
                                    Per second             Total
                                    ---------------------  -----------------------
  TOTAL_APP_COMMITS                 0                      3
  ACT_COMPLETED_TOTAL               1                      11
  APP_RQSTS_COMPLETED_TOTAL         2                      27

  TOTAL_CPU_TIME                    = 44364
  TOTAL_CPU_TIME per request        = 1643

  Row processing
    ROWS_READ/ROWS_RETURNED         = 2 (33/16)
    ROWS_MODIFIED                   = 0
```

MONREPORT.DBSUMMARY Snippets

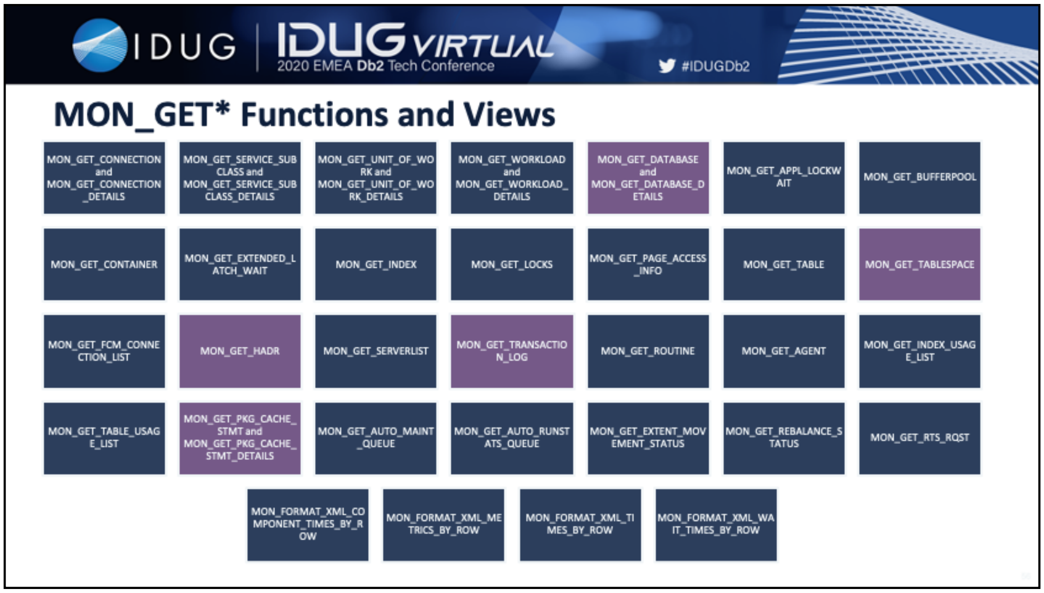


```
Buffer pool
  ----------------------------------------------------
  Buffer pool hit ratios

  Type             Ratio            Formula
  ---------------  ---------------  -------------------
  Data             100              (1-(0+0-0)/(37+0))
  Index            100              (1-(0+0-0)/(34+0))
  XDA              0                (1-(0+0-0)/(0+0))
  COL              0                (1-(0+0-0)/(0+0))
  LBP Data         100              (37-0)/(37+0)
  LBP Index        0                (0-0)/(34+0)
  LBP XDA          0                (0-0)/(0+0)
  LBP COL          0                (0-0)/(0+0)
  GBP Data         0                (0 - 0)/0
  GBP Index        0                (0 - 0)/0
  GBP XDA          0                (0 - 0)/0
  GBP COL          0                (0 - 0)/0
```



MONREPORT.DBSUMMARY Snippets

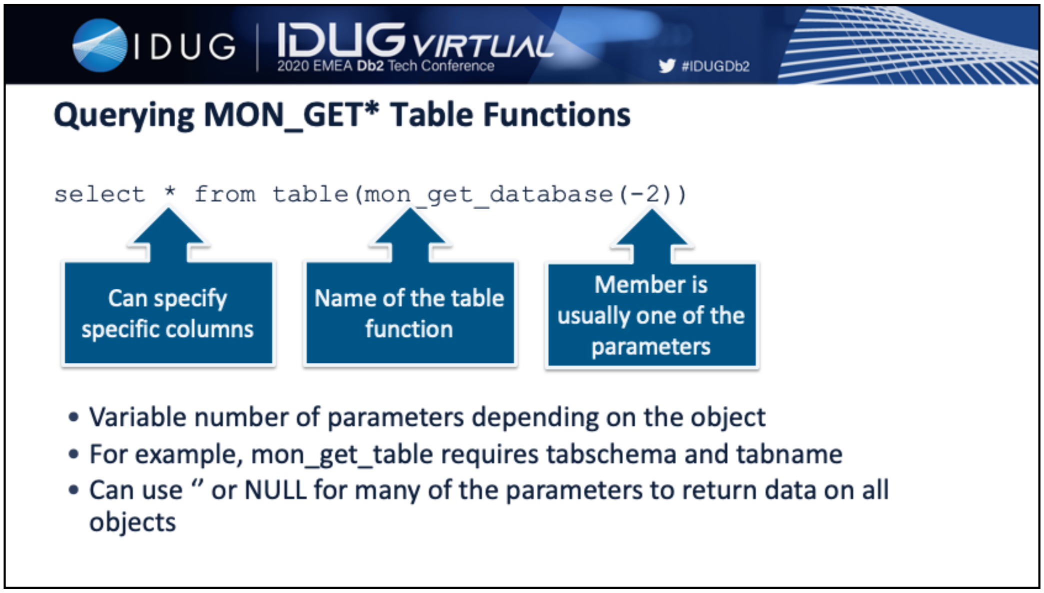
```
                        Per activity                    Total
                        ------------------------------  ----------------------
LOCK_WAIT_TIME          0                               0
LOCK_WAITS              0                               0
LOCK_TIMEOUTS           0                               0
DEADLOCKS               0                               0
LOCK_ESCALS             0                               0

<SNIP>

Sort
--------------------------------------------------------------------------------
TOTAL_SORTS                         = 4
SORT_OVERFLOWS                      = 0
POST_THRESHOLD_SORTS                = 0
POST_SHRTHRESHOLD_SORTS             = 0
```



MON_GET* Functions and Views

MON_GET_CONNECTION and MON_GET_CONNECTION_DETAILS	MON_GET_SERVICE_SUBCLASS and MON_GET_SERVICE_SUBCLASS_DETAILS	MON_GET_UNIT_OF_WORK and MON_GET_UNIT_OF_WORK_DETAILS	MON_GET_WORKLOAD and MON_GET_WORKLOAD_DETAILS	MON_GET_DATABASE and MON_GET_DATABASE_DETAILS	MON_GET_APPL_LOCKWAIT	MON_GET_BUFFERPOOL
MON_GET_CONTAINER	MON_GET_EXTENDED_LATCH_WAIT	MON_GET_INDEX	MON_GET_LOCKS	MON_GET_PAGE_ACCESS_INFO	MON_GET_TABLE	MON_GET_TABLESPACE
MON_GET_FCM_CONNECTION_LIST	MON_GET_HADR	MON_GET_SERVERLIST	MON_GET_TRANSACTION_LOG	MON_GET_ROUTINE	MON_GET_AGENT	MON_GET_INDEX_USAGE_LIST
MON_GET_TABLE_USAGE_LIST	MON_GET_PKG_CACHE_STMT and MON_GET_PKG_CACHE_STMT_DETAILS	MON_GET_AUTO_MAINT_QUEUE	MON_GET_AUTO_RUNSTATS_QUEUE	MON_GET_EXTENT_MOVEMENT_STATUS	MON_GET_REBALANCE_STATUS	MON_GET_RTS_RQST
		MON_FORMAT_XML_COMPONENT_TIMES_BY_ROW	MON_FORMAT_XML_METRICS_BY_ROW	MON_FORMAT_XML_TIMES_BY_ROW	MON_FORMAT_XML_WAIT_TIMES_BY_ROW	

SQL interface to lightweight in-memory monitoring



Querying MON_GET* Table Functions

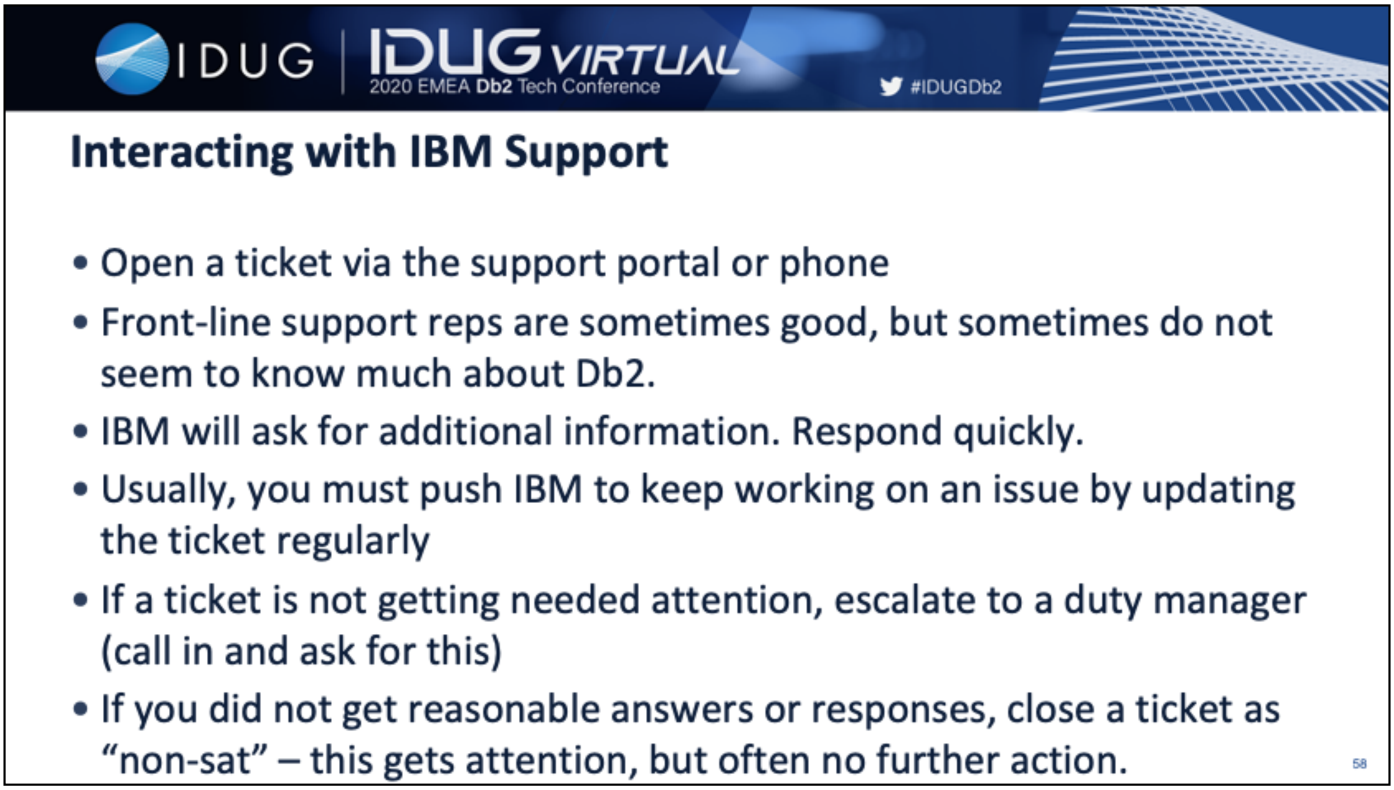
```
select * from table(mon_get_database(-2))
```

Can specify specific columns

Name of the table function

Member is usually one of the parameters

- Variable number of parameters depending on the object
- For example, mon_get_table requires tabschema and tabname
- Can use '' or NULL for many of the parameters to return data on all objects

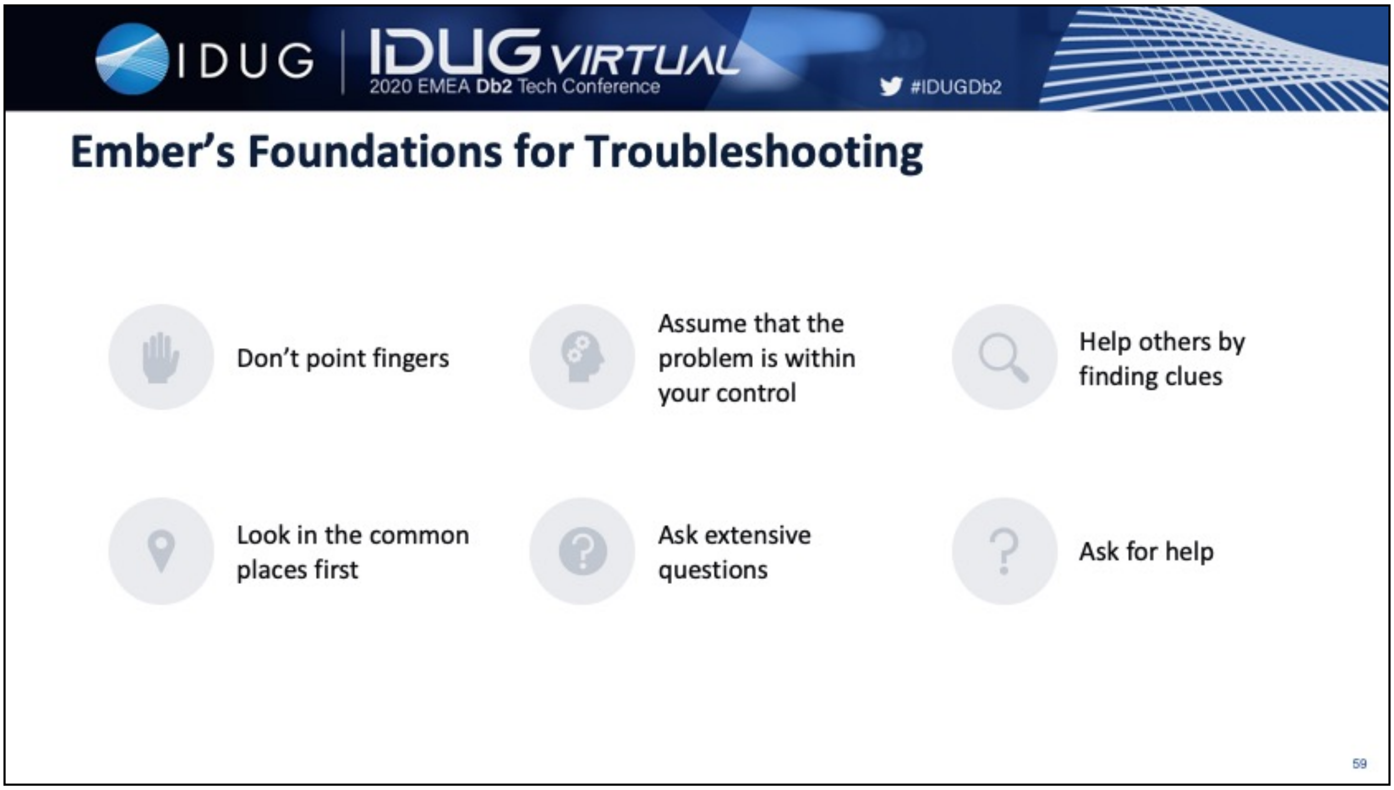

IDUG
IDUG VIRTUAL
2020 EMEA Db2 Tech Conference
#IDUGDb2

Interacting with IBM Support

- Open a ticket via the support portal or phone
- Front-line support reps are sometimes good, but sometimes do not seem to know much about Db2.
- IBM will ask for additional information. Respond quickly.
- Usually, you must push IBM to keep working on an issue by updating the ticket regularly
- If a ticket is not getting needed attention, escalate to a duty manager (call in and ask for this)
- If you did not get reasonable answers or responses, close a ticket as “non-sat” – this gets attention, but often no further action.

58

https://datageek.blog/en/2012/12/04/getting-the-most-out-of-db2-support/



IDUG | IDUG VIRTUAL 2020 EMEA Db2 Tech Conference #IDUGDb2

Ember’s Foundations for Troubleshooting

- ✋ Don’t point fingers
- Assume that the problem is within your control
- 🔍 Help others by finding clues
- Look in the common places first
- ? Ask extensive questions
- ? Ask for help

59

When troubleshooting you’re rarely able to immediately say “this is a database problem”, ”this is an application problem”, or “this is a network problem”. The key is that if you’re working with experts in other areas or even if you’re trying to cover all the areas yourself, don’t rule things out prematurely. If someone calls me in for a problem, I do my best not to just say “That’s not the database.”, even if it clearly isn’t. Even if it isn’t the database, often there are clues that you can find within the database that may help others locate the problem in their areas. My approach is to start from “This is in the database, but where?”, and then once I have significant proof it isn’t, provide others with the details that will help them in their own troubleshooting efforts.

IDUG

Leading the Db2 User Community since 1988

Ember Crooks

ember.crooks@gmail.com

#IDUGDb2

Ember has 19+ years of experience with Db2 on Linux, Unix, and Windows platforms. She is the founder and principal author of the popular datageek.blog technical blog where she educates herself and others through example and case study. Ember is an IBM Gold Consultant and IBM Champion in Information Management.