



DB2 Information Management
Software

Best Practices Upgrade to DB2 LUW 10.5 with BLU Acceleration

Melanie Stopfer
IBM SWG

Email: mstopfer@us.ibm.com
Twitter: [mstopfer1](#)
LinkedIn: [Melanie Stopfer](#)



5.4

Melanie Stopfer is a Consulting Learning Specialist and Developer for IBM Software Group. She is recognized worldwide as an advanced database specialist. As a Certified DB2 LUW 10.5 Technical Expert and Learning Facilitation Specialist, she has provided in-depth technical support to IM customers specializing recovery, performance and database upgrade and migration best practices since 1988. In 2009, Melanie was the first DB2 LUW speaker to be inducted into the IDUG Speaker Hall of Fame and was again selected Best Overall Speaker at IDUG EMEA 2011 and 2012 and IDUG NA 2012 and 2014. In 2013, IBM IOD rated her presentations in top two for customer satisfaction.

Objectives

- Describe changes to DB2 10.5 packaging and software and operating system requirements.
- Understand DB2 10.5 changes to DDL, utilities, tools and monitoring.
- Configure a database and instance to implement DB2 10.5 new features.
- Analyze DB2 10.5 changes that affect applications and database runtime behavior.
- Develop a DB2 10.5 upgrade strategy



Abstract:

Are you responsible for upgrading your database to DB2 10.5? Take advantage of columnar data store integrated into the core DB2 engine which has huge benefits for Business Intelligence workloads. DB2 10.5 is not just BLU. There is additional new great functionality. Learn the entire story! If you are looking for a detailed step by step approach to implement DB2 10.5, attend this presentation to learn about often overlooked but very important tips and tricks to understand before upgrading. Planning your upgrade is much easier if you know these best practices.

Planning Your Upgrade to DB2 10.5 LUW

DB2 10.5 LUW Upgrade from *pre-DB2 Version 10.5* copy:

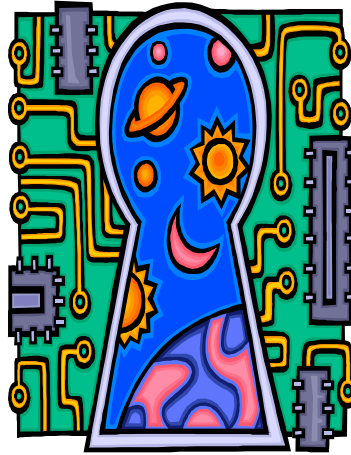
- DB2 10.1-> DB2 10.5
- DB2 9.8 -> DB2 10.5
- DB2 9.7 -> DB2 10.5

- If DB2 servers are running on release prior DB2 9.7:

- Upgrade to DB2 9.7 or 10.1 latest fixpack
- Then upgrade to DB2 10.5

See notes for more details/URLs:

- **Upgrading to DB2 V10.5 Roadmap**
- **DB2 Upgrade Portal**
- **DB2 V10.5 English Manuals**
- **DB2 V10.5 Translated Manuals**



SC27-5519-00 What's New for DB2 V10.5
SC27-5513-00 Upgrading to DB2 V10.5 Guide

3

DB2 10.5 for Linux, UNIX and Windows became generally available on June 14, 2013. In the upgrade tasks, the term *pre-DB2 Version 10.5 releases* refers to DB2 Version 10.1, DB2 Version 9.8 or DB2 Version 9.7.

Refer to the DB2 Information Center for details about the installation and upgrade process. It is highly recommended that you consult the DB2 IC while planning and executing an upgrade. You should also test the upgrade process on non-production servers first. Your first point of reference to learn how to upgrade each component of your DB2 environment and how to create a upgrade plan is the **SC27-5513-00 Upgrading to DB2 Version 10.5 Guide**. The content of this guide is available in the *DB2 Information Center*. You may also download the DB2 Version 10.5 for Linux, UNIX, and Windows English manuals from the following website:

<http://www-01.ibm.com/support/docview.wss?uid=swg27038855>

The DB2 Version 10.5 for Linux, UNIX, and Windows translated manuals can be downloaded from <http://www-01.ibm.com/support/docview.wss?rs=71&uid=swg27038856>

Checking Whether Upgrade is Possible

Direct upgrades are supported from Version 9.7 and higher systems. Upgrading your DB2 server requires that you install a DB2 10.5 copy and then upgrade all the instances and databases to be able to run them under the DB2 10.5 copy. You can directly upgrade existing DB2 9.8, 9.7 or DB2 10.1 instances and databases to DB2 10.5. On Windows operating systems, you have an option to automatically upgrade an existing pre-V10.5 DB2 copy. If you choose to upgrade your existing DB2 copy during installation, you only need to upgrade your databases after installation.

Review also the DB2 upgrade portal at www.ibm.com/support (formerly known as DB2 migration portal) that provides access to additional resources and up-to-date information about the upgrade process as they become available. These resources include educational material, white papers, and webcasts for upgrade. **The DB2 Upgrade Portal is located at <http://www-01.ibm.com/support/docview.wss?uid=swg21200005>**

The Upgrading to DB2 Version 10.5 Roadmap is located at <http://www-01.ibm.com/support/docview.wss?uid=swg21633449>

REF: Upgrading DB2 pureScale environments

- If upgrading from DB2 Version 9.8, follow upgrade steps detailed in Upgrading a DB2 pureScale server.

http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0060571.html?lang=en

- Taking DB2 server offline for upgrade or converting to DB2 pureScale environment

http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007197.html?cp=SSEPGG_10.5.0%2F2-0-7-1-5&lang=en

If you are upgrading from DB2 9.8 follow the upgrade steps detailed in Upgrading a DB2 pureScale server.

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.qb.upgrade.doc/doc/t0060571.html>

Before you can continue with the upgrade process, or the conversion of your environment for DB2 pureScale, you must take your DB2 server offline by stopping the DB2 license service, stopping all command line processor sessions, disconnecting applications and users, and stopping the database manager.

REF: Upgrading partitioned database environments

- Install DB2 10.5 on all database partition servers, upgrade the instances and then upgrade the databases.
- Use only Install New option in Install a Product panel to install DB2 10.5.
 - If choose to upgrade action when select Work with Existing option on Install a Product panel, installation fails.
- Perform a full offline backup for all database partitions. Verify databases are ready for upgrade, and perform any other pre-upgrade tasks that apply.
- Upgrade each instance on database partition server that owns the instance. Run UPGRADE DATABASE on catalog partition.
- Catalog partition must be available when issue UPGRADE DATABASE regardless on what database partition issue command from.
- If any database partitions are not available, these database partitions are not upgraded.
- For Details:
http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0011548.html?lang=en

5

Upgrading partitioned database environments requires that you install DB2 10.5 as a new copy in all database partition servers, upgrade the instances and then upgrade the databases.

Ensure that you have root access on Linux and UNIX operating systems or Local Administrator authority on Windows.

Ensure that you have SYSADM authority.

REF: How to Upgrade or Downgrade DB2 Instances

Upgrading Fixpak

- Update db2inst1 from V10.5 FP1 to FP2. To do this, you should:
 - Stop the DB2 instance if necessary (db2stop)
 - As root install V10.5 FP2 in location of your choosing
 - Still as root cd to install dir then cd to the "instance" dir
 - Run the 'db2iupdt' script from this location
./db2iupdt db2inst1

Downgrading Fixpak

- If wish to downgrade instance back to V10.5 FP1 code base:
 - Stop the db2 instance if necessary (db2stop)
 - cd to the V10.5 FP1 dir then cd to the "instance" dir
 - As root, run the 'db2iupdt' script from this location
./db2iupdt -D db2inst

6

How to Upgrade or Downgrade select DB2 Instances between various Product Offerings

There may be circumstances where you have multiple instances on your system, where you wish to selectively update a particular instance to a higher (or lower) fixpak level, between DB2 product offerings (WSE to ESE for example). Make sure you take backups of your database(s) before you pursue any sort of upgrade, not only with DB2 itself, but even when upgrading other software that interacts with DB2, or upgrading the OS that DB2 lives on. For details see: <http://www-01.ibm.com/support/docview.wss?uid=swg21399113&myns=swgimgmt&mynp=OCSSEPGG&mync=E>

NOTE: It is not necessary to install the GA (General Availability) of DB2 first, in order to apply a fixpak. It will have the option to install the full DB2 Version and fixpak at once, or you can choose to apply just the fixpak to an existing install. You can then simply apply the proper DB2 License file after installing with this method.

Moving Between Editions

Just as you were able to move up to ESE from WSE, you can move the opposite direction as well (ESE to WSE). It is even possible to move to different fixpak levels as well. For example you can move from DB2 V9.1 FP4 ESE to DB2 V9.1 FP2 WSE using the same general instructions shown above. Moving from ESE to WSE will likely produce a warning message (DBI1282W), indicating the instance was updated from a higher instance type to a lower instance type. The new instance will use the default configuration settings for the new instance type. Check the values in the new database manager configuration file and update the parameters as required.

DBI1282W The database manager configuration files could not be merged. The original configuration file was saved as ~/sqllib/backup/db2system.old (The original instance type is ese. The instance type to be migrated or updated is wse.)

```
>><< DBI1070I Program db2iupdt completed successfully.
```

If you are concerned about your instance settings, it may be a good idea to back up your configurations before any instance updating. You can export (and import) your configuration settings using the db2cfexp and db2cfimp tools. Export your instance configuration settings for the one instance you wish to upgrade:

```
db2cfexp <filename> BACKUP
```

The BACKUP option creates a configuration profile of the DB2 instance. This profile contains all of the instance configuration information. You can import settings from the profile we created.

- Log in as the instance user
- Run: db2cfimp <filename>

Supported Client Access and Client Upgrade

- **ALWAYS UPGRADE SERVERS before upgrading CLIENTS!**
- **DB2 10.5, 10.1 and DB2 9.7 IBM clients and drivers can access remote DB2 10.5 Server:**
 - IBM Data Server Driver Package
 - IBM Data Server Driver for JDBC and SQLJ
 - IBM Data Server Driver for ODBC and CLI
 - IBM Data Server Runtime Client
 - IBM Data Server Client
- **If use pre-10.5 clients to access DB2 10.5 servers, applications are restricted to functionality available for that client.**
- If different versions of client and DB2 server located on same system, local client-to-server connections using IPC are not supported. Establish connection as remote connection (loopback connection) using TCP/IP.
- When later-level client accesses earlier-level server, later level client functionality not available to server.
- Upgrading client involves installing DB2 10.5 client copy and then upgrading client instance.
- Current level of client installed determines way to proceed with upgrade to DB2 10.5.
 - Can directly upgrade to DB2 10.5 clients from DB2 10.1 or DB2 9.7.
 - **If have DB2 9.5 or earlier clients, migrate to any DB2 10.1 or DB2 9.7 client first.**

Which JDBC driver (JCC) version corresponds with each DB2 release and Fix Pack level?
<http://www-01.ibm.com/support/docview.wss?uid=swg21363866>

For JDBC or SQLJ applications, if you are using the IBM DB2 Driver for SQLJ and JDBC, you can determine the level of the driver by running
db2jcc -version

7

A client instance allows connecting application to database and keeps information about client configuration, cataloged nodes, and cataloged databases. The upgrade process consists of all the tasks that you must perform to have your DB2 environment running successfully after upgrading. These components are DB2 servers, DB2 clients, database applications, and routines. Upgrading DB2 servers to DB2 10.5 requires an understanding of upgrade concepts, upgrade restrictions, upgrade recommendations, and your DB2 server. When you have a complete understanding of what upgrading your DB2 server involves, you can create your own upgrade plan. The upgrade of each of the components in your DB2 environment requires that you perform different tasks:

- *Upgrading DB2 servers* involves upgrading your existing instances and databases so that they can run in the new release.
- *Upgrading clients* involves upgrading your client instances to keep the configuration of your existing clients.
- *Upgrading database applications and routines* involves testing them in the new release and modifying them only when you must support changes in this new release.

The IBM clients and drivers are as follows:

IBM Data Server Driver Package
IBM Data Server Driver for JDBC and SQLJ
IBM Data Server Driver for ODBC and CLI
IBM Data Server Runtime Client
IBM Data Server Client

Product packaging enhancements for DB2 10.5

- IBM updated product packaging to provide simplicity
 - All functionality offered in prior generation of DB2 LUW and IBM InfoSphere Warehouse editions and features now included in package of six editions and one feature.
- New DB2 Advanced Workgroup Server Edition
 - For deployment in departmental, workgroup, or medium-sized business
 - Supports partitioned database environments and DB2 pureScale environments.
- New IBM DB2 Advanced Recovery Feature - tools to protect mission-critical data
 - DB2 Recovery Expert
 - DB2 Merge Backup
 - IBM InfoSphere Optim High Performance Unload
 - Only separately-priced feature available in Version 10.5.
- Can update some DB2 editions to another by just updating the license file.
 - Do not have to reinstall DB2 database product.

8

As IBM data servers continue to evolve, the names and the packaging of the DB2 components change to respond to market needs.

IBM has updated product packaging to provide simplicity through fewer packages, and greater value through more function and features included in the base DB2 editions such as warehouse functionality. The new DB2 Advanced Workgroup Server Edition is the ideal choice for deployment in a departmental, workgroup, or medium-sized business environment. It supports partitioned database environments and DB2 pureScale environments.

The new IBM DB2 Advanced Recovery Feature offers additional tools to protect the mission-critical data that runs your business such as DB2 Recovery Expert, DB2 Merge Backup, and IBM InfoSphere Optim High Performance Unload. The IBM DB2 Advanced Recovery Feature is the only separately-priced feature available in Version 10.5. The functionality of DB2 features available in earlier releases is available in DB2 database product editions.

You can update some DB2 editions to another one by just updating the license file. You do not have to reinstall the DB2 database product. You can also update some DB2 editions to enable DB2 Connect functionality by adding your DB2 Connect license file.

Product packaging enhancements & licensing

- Functionality in DB2 features and DB2 product editions details:
http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.licensing.doc/doc/r0053238.html?lang=en
- Which DB2 10.5 product edition is right for you?
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1311whichdb2edition/>
- Compare the distributed DB2 10.5 database servers
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1311db2compare/index.html?ca=drs>
- IBM DB2 Recommendation Tool
www-01.ibm.com/software/data/db2/editiontool/#/?S_CMP=fromsocialmedia&utm_content=buffercf556&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer?S_CMP=fromsocialmedia
- Read about products and view related licensing and marketing information:
<http://www-01.ibm.com/software/data/db2/linux-unix-windows/>
- Licensing distributed DB2 10.5 servers in HA environment
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1311db2halicensing/>

9

The functionality in DB2 features and DB2 product editions is explained at

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.licensing.doc/doc/r0053238.html>.

IBM has updated product packaging to provide simplicity through fewer packages, and greater value through more function and features included in the base DB2 editions. To read about these products and to view the related licensing and marketing information, see the DB2 product page at <http://www-01.ibm.com/software/data/db2/linux-unix-windows/>.

Program charges:

- **DB2 Express Edition** can be deployed on pervasive SMB operating systems, such as Linux, Windows, or Solaris and includes pureXML, web services federation, DB2 Homogeneous Federation, Homogeneous SQL Replication, and backup compression. If licensed as a yearly subscription (DB2 Express Edition FTL), it also includes High Availability feature provided both primary and secondary servers in the high availability cluster are licensed. If licensed under the Limited Use Virtual Server metric, DB2 Express Edition uses up to eight cores on the server. The DB2 data server cannot use more than 64 GB of memory per server. You must acquire a separate user license for each authorized user of this product with a minimum purchase of five users per server.
- **DB2 Express-C** can be used for development and deployment at no charge, and can also be distributed with third-party solutions without any royalties to IBM. It can be installed on physical or virtual systems with any amount of CPU and RAM, and is optimized to use up to a maximum of 2 cores and 16 GB of memory. DB2 Express-C is refreshed at major release milestones and comes with online community-based assistance. Users requiring more formal support, access to fix packs, or additional capabilities such as high availability clustering and replication features, can purchase optional yearly subscription for DB2 Express Edition (FTL) or upgrade to other DB2 editions.
- **DB2 Developer Edition** - The software in this package cannot be used for production systems. You must acquire a separate user license for each Authorized User of this product.

Features and Functions by DB2 LUW Editions

	EXPRESS c	EXPRESS	Workgroup	Advanced Workgroup	Enterprise Server	Advanced Enterprise
Storage Optimization	No	No	No	Yes	No	Yes
PureScale Cluster	No	No	No	Yes	No	Yes
HADR	No	Yes	Yes	Yes	Yes	Yes
Column-organized Table	No	No	No	Yes	No	Yes
Multi-Temperature Storage	No	No	No	Yes	Yes	Yes
Temporal Tables	Yes	Yes	Yes	Yes	Yes	Yes
Range Partitioned Tables	No	No	Yes	Yes	Yes	Yes
Workload Management	No	No	No	Yes	No	Yes
Data Studio	Yes	Yes	Yes	Yes	Yes	Yes

10

To read about these products and to view the related licensing and marketing information, see the DB2 product page at

<http://www.ibm.com/software/data/db2/linux-unix-windows>.

REF: Suggestions for a good *upgrade strategy*

1. Review new, deprecated, discontinued functionality for DB2 10.5 and for any releases between release are upgrading from and DB2 10.5.
2. Plan how to and modify database applications and routines. Ensure they run successfully in DB2 10.5.
3. Set up DB2 10.5 test server and create test databases.
4. Test database applications and routines on DB2 10.5 test database to determine whether run successfully. If application requires a client, use DB2 10.5 client.
5. Upgrade DB2 servers and clients in test environment. Determine what issues are and how to resolve. Use info to adjust upgrade plan.
6. Upgrade DB2 servers to DB2 10.5 in production environment. Ensure they operate as expected.
7. Upgrade clients to DB2 10.5 in production environment. Ensure clients operate as expected.
8. Test database applications and routines in DB2 10.5 upgraded environment to determine whether run as expected.
9. Make upgraded environment available to users.
10. Plan to remove use of deprecated functionality and incorporate new functionality from DB2 10.5.



11

Your environment has several components such as DB2 servers, DB2 clients, database applications, scripts, routines and tools. Planning your upgrade requires a thorough understanding of the upgrade process of each component in your environment.

First, devise a strategy on how to approach your environment upgrade. You must determine the order in which you are going to upgrade each component. The characteristics of your environment and the information in upgrade essentials, especially the best practices and restrictions, can help you determine your strategy.

The above list is an example of a good *upgrade strategy* in which you test your database applications and routines and determine that they run successfully in DB2 10.5.

After you have a strategy that will give you the outline for your upgrade plan, you can define the upgrade plan details for each component in your environment. An *upgrade plan* should include for each component:

- Upgrade prerequisites
- Pre-upgrade tasks
- Upgrade tasks
- Post-upgrade tasks

If you have previous upgrade plans, review them and compare them with the upgrade plan for DB2 10.5. Include in your new plan any steps related to internal procedures to request access, software installation or other system services within your organization.

Finally, plan to remove the use of deprecated functionality and incorporate new functionality from DB2 10.5. Although you are only required to remove the use of discontinued functionality, you should also plan to remove the use of deprecated functionality after upgrade because they will become unsupported in a future release. Also, you should take advantage of new functionality for your database products, applications, and routines to enhance functionality and improve performance.

Benchmark DB2 server performance – before and after

- Run performance tests before upgrading your DB2 server.
- Use db2batch benchmark tool to collect elapsed and CPU times for running queries.

db2batch -d DBX -f input.sql -r output.fil -i complete -o e yes -isol CS

- Record the exact environment conditions where you run your tests.
- Keep record of **db2exfmt** command output for each test query.
- Compare the results before and after upgrade.
- Helps identify and correct any performance degradation that might occur.

12

Run a number of performance tests before upgrading your DB2 server. The db2batch benchmark tool helps you to collect elapsed and CPU times for running queries. You can use this tool to develop performance tests. Record the exact environment conditions where you run your tests.

Also, keep a record of the db2expln command output for each test query. Compare the results before and after upgrade. This practice can help to identify and correct any performance degradation that might occur.

REF: Upgrade restrictions for DB2 servers

- To move to DB2 pureScale environment, upgrade to DB2 9.7 and then upgrade to DB2 10.5. Upgrading to DB2 9.8 as intermediate release requires that upgrade to DB2 9.7 first.
- Upgrading to a DB2 10.5 non-root installation is supported from a DB2 10.1 and DB2 9.7 non-root installation. Upgrading to a DB2 10.5 non-root installation from a pre DB2 10.5 root installation is not supported.
- On Windows, upgrade action shows for existing DB2 copies that can be upgraded during installation of DB2 10.5. This action automatically installs DB2 10.5 and upgrades all instances running on DB2 copy. This action also uninstalls DB2 copy and any add-on products installed in this copy. If do not choose upgrade action, must manually upgrade instances after installation.
- On Linux/UNIX, upgrade action is not available and can only install a new copy of DB2 10.5. Then manually upgrade instances after installation.
- Instance bit size is determined by the operating system where DB2 10.5 is installed. See notes for support for 32-bit kernels and 64-bit kernels.
- Upgrading from system with multiple copies of DB2 10.1 and DB2 9.7 of all levels is supported.
- Upgrading from a partitioned database environment with multiple database partitions is supported.
- Restoring full database offline backups from pre-DB2 10.5 copies is supported. Rolling forward of logs from a previous level is not possible.

13

Before you start to upgrade your DB2 server, you must understand what the support for upgrade is and what restrictions exist.

If you have 32-bit instances and you upgrade to DB2 Version 10.5 Developer Edition on a 32-bit system. You can only upgrade to 32-bit instances on 32-bit Windows or 32-bit Linux on x86. For any other editions in DB2 Version 10.5, you must upgrade to 64-bit system. If you have 64-bit instances and you upgrade to DB2 Version 10.5 on a 64-bit system. You can only upgrade to a 64-bit instance on a 64-bit system. If you have 32-bit instances and you upgrade to DB2 Version 10.5 on a 64-bit system, you must manage incompatibilities so that your applications and routines can run successfully. Incompatibilities arise because of discontinued functionality or incorrect shared library path specification. For example, 32-bit unfenced stored procedures in any supported language except Java are not supported. By dropping and recreating these stored procedures as fenced you can resolve this issue.

For a summary of the details on the available 32-bit and 64-bit support refer to:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/index.jsp?topic=%2Fcom.ibm.db2.luw.qb.upgrade.doc%2Fdoc%2Fc0022266.html>

REF: What is unsupported?

- **DB2 10.5 installation fails if following situations exist:**

- Operating system not supported. Must upgrade to supported operating system before upgrade to DB2 10.5 or upgrade to new DB2 server that meets requirements. See URL in notes.
- A 32-bit kernel is running on Linux and UNIX operating systems except for Linux on x86. A 64-bit kernel must be installed before installing DB2 10.5.
- A version of DB2 before Version 9.7 is installed on Windows operating systems.
- If have DB2 9.7 copy on SUSE Linux Enterprise Server 10, first apply DB2 9.7 Fix Pack 2 or later, before upgrade operating system to SUSE Linux Enterprise Server 11.
- If upgrading pre-DB2 10.5 on POWER4 processor-based systems, first upgrade to supported POWER processor-based systems before upgrading to DB2 10.5. POWER4 and older processor-based systems are not supported in DB2 Version 10.5.



- **db2iupgrade command fails if:**

- Do not have authorization to upgrade instance.
- Instance trying to upgrade is active. Run db2stop command to stop instance.
- Instance already at DB2 10.5 or later. Run db2iupdt command to update to different fix pack levels or copies of DB2 10.5.
- Try to upgrade from DB2 10.5 back to prior version.
- db2ckupgrade fails and causes db2iupgrade command to fail. db2iupgrade command calls db2ckupgrade command to verify whether cataloged local databases are ready for upgrade to DB2 10.5.

14

You must upgrade to supported version of operating system before you upgrade to DB2 10.1 or upgrade to new DB2 server that meets operating system requirements.

If you have a DB2 Version 9.7 copy on SUSE Linux Enterprise Server 10, first apply DB2 Version 9.7 Fix Pack 2 or later, before you upgrade the operating system to SUSE Linux Enterprise Server 11. If you are upgrading a pre-DB2 Version 10.5 copy on POWER4 processor-based systems, first upgrade to supported POWER processor-based systems before upgrading to DB2 Version 10.5. POWER4 and older processor-based systems are not supported in DB2 Version 10.5.

There is a list of operating system, software, and hardware prerequisites that must be met for each operating system at:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.qb.server.doc/doc/r0025127.html>

Operating systems are no longer supported

- Starting in DB2 10.5, the following operating systems and Linux distributions are no longer supported:
 - Red Hat Enterprise Linux 5.6
 - Ubuntu 10.4 LTS
 - Windows 2003
 - Windows 2003 R2
 - Windows XP
 - Windows Vista
- Starting in DB2 10.5, the following architecture is no longer supported:
 - POWER4 processor-based systems (DB2 servers and data clients)
- POWER5, POWER6, POWER7 and POWER7+ are supported in DB2 Version 10.5. For BLU Acceleration, POWER7 is recommended.
- List of operating system, software, and hardware prerequisites
http://www-01.ibm.com/support/knowledgecenter/?lang=en#!/SSEPGG_10.5.0/com.ibm.db2.luw.qb.server.doc/doc/r0025127.html

15

Starting in DB2 10.5, some operating systems, Linux distributions and architecture are no longer supported. POWER5, POWER6, POWER7 and POWER7+ are supported in DB2 Version 10.5. For BLU Acceleration, POWER7 is recommended.

db2prereqcheck command

- Check if system meets DB2 10.5.0.0 prerequisites:

- **db2prereqcheck -v 10.5.0.0 -s**

-v <version>

-s Print prerequisite validation summary on screen

-c Checks prerequisites for thin client

-p Checks prerequisites for DB2 pureScale environment

-i Checks the non-DB2 pureScale prerequisites for the latest DB2 version that is defined in the resource XML file.

If do not specify any options, both DB2 pureScale and non-DB2 pureScale server prerequisites checked for all DB2 versions listed in DB2prereqs.xml.

The -i and -v parameters are mutually exclusive.

-o <report-file>

- Uses resource XML file that contains prerequisites. Default path of XML file located in: <DB2 installation>/cfg/DB2prereqs.xml



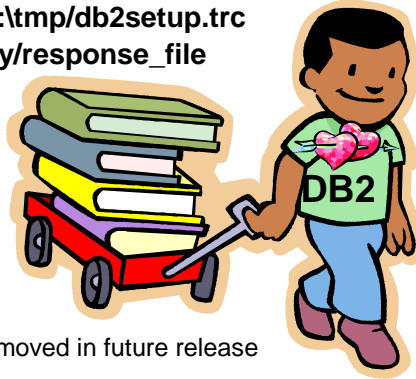
16

The supported UNIX, Linux and Windows operating systems have changed in DB2 Version 10.5. Review the installation requirements for DB2 servers and IBM data server clients to determine whether your operating system version is supported and if you need to upgrade your operating system before installing DB2 10.1. Newer versions of operating systems can also bring new hardware requirements. Performing hardware and operating system upgrades separately from DB2 database product upgrade simplifies problem determination if you encounter upgrade difficulties. If you upgrade your software or hardware before a DB2 database product upgrade, ensure that your system is operating as expected before attempting to upgrade your DB2 database product. Checks whether your system meets the prerequisites for the installation of a specific version of DB2 Database for Linux, UNIX, and Windows. By using the db2prereqcheck command, you can determine whether your system satisfies the prerequisites before you download DB2 Database for Linux, UNIX, and Windows and start the installation process. The db2prereqcheck command uses a resource XML file that contains the prerequisites. The default path of the XML file is located in DB2 installation/cfg/DB2prereqs.xml. You must have read or write permissions on the XML file. Do not modify the contents of the XML file.

- -c Checks the prerequisites for thin client. The -c, -u, -g, and -p parameters are mutually exclusive.
- -u Checks the uDAPL requirement for the DB2 pureScale environment (Linux and AIX only). The -c, -u, -g, and -p parameters are mutually exclusive. On RoCE networks, you must ensure the AIX and uDAPL software prerequisites are satisfied. The db2prereqcheck command does not automatically validate these levels.
- -g Checks the GPL compilation requirement (Linux only). The -c, -u, -g, and -p parameters are mutually exclusive.
- -p Checks the prerequisites for the DB2 pureScale environment (Linux and AIX only). The -c, -u, -g, and -p parameters are mutually exclusive.
- -t network configuration type. The -t parameter validates prerequisites requirement for a specific type of network configuration (Linux only). The network configuration type must be SINGLE_IB_PORT_CLUSTER, MULT_IB_PORT_CLUSTER, SINGLE_ROCE_PORT_CLUSTER or MULTI_ROCE_PORT_CLUSTER.
- -s Print prerequisite validation summary on screen.
- -i Checks the prerequisites for the latest DB2 version that is defined in the resource XML file.
- To display the latest version number in the XML file, run -i with -s. The -i and -v parameters are mutually exclusive.
- -v version - Checks the prerequisites for the specific DB2 version. You can use this parameter with other db2prereqcheck command parameters. The -i and -v parameters are mutually exclusive.
- -f xml-file - Specifies the name of the XML resource file. The default XML file (with file path : DB2 installation/cfg/DB2prereqs.xml) will be used if the -f parameter is not specified.
- -o report-file - Specifies a name for the report file. For example, db2prereqcheck.rpt.
- hl list_of_hostnames - Specifies a list of hosts that are checked for passwordless root SSH access between all hosts.
- -dl list_of_shared_devices - Specifies one or more PVIDs (on AIX operating systems) or WWNs (on Linux operating systems) of the shared disks that are verified to make sure that they are accessible by all hosts. If you specify this parameter, you must also specify the -hl parameter.
- -iu instance_user - Specifies the instance user name. The UID and GID of the instance user name are verified to make sure they are the same across all hosts. If you specify this parameter, you must also specify the -hl parameter.
- -nl list_of_netnames - Specifies the list of netnames. These netnames are used to ping the RoCE & IB networks to verify that they are pingable between all hosts.
- -upgrade - Checks whether the required entry in the netmon.cf file is correct when you are upgrading from the GA level of Version 9.8 or Version 9.8 Fix Pack 1.

INSTALLATION METHODS

- On Linux/UNIX, use **db2setup** command or **response file** installation method.
 - **./db2setup -l /temp/db2setup.log -t /tmp/db2setup.trc**
 - **./db2setup -r responsefile_directory/response_file**
 - **-c validates contents of response file before install fix pack. Ensures contents of response file correct before installation.**
- On Windows, use **setup** command or **response file** installation method.
 - **setup -l c:\temp\db2setup.log -t c:\tmp\db2setup.trc**
 - **setup -u c:\responsefile_directory/response_file**



- **db2_install** is deprecated and might be removed in future release
- **GC27-5514-00 Installing DB2 Servers**
- **GC27-5515-00 Installing IBM Data Server Clients**

17

The **db2setup** command queries the operating system to determine the existing language settings. If the language setting of your operating system is supported by **db2setup**, then that language will be used when displaying the DB2 Setup wizard. If your system uses the same code pages but different locale names than those supported by the DB2 interface, you can still see the translated **db2setup** by setting your **LANG** environment variable to the appropriate value. The **db2setup** command is only available on Linux and UNIX operating systems. The command for Windows operating systems is **setup**.

To start the IBM DB2 Setup Launchpad, run the **db2setup** command from the directory where the DB2 installation image resides:

./db2setup -l /temp/db2setup.log -t /tmp/db2setup.trc

- The **-l** and **-t** recommended parameters enable DB2 logging and tracing.

To create a response file use the DB2 Setup wizard:

- Type the **db2setup** command to launch the DB2 Instance Setup Wizard.
- On the Select installation, response file creation or both panels, select the **Save my installation settings in a response file** option.

After creating a response file, you can make changes to the response file to activate or deactivate keywords.

./db2setup -r responsefile_directory/response_file

Student Notes: Reference Only

- In V10.5 you might run into the following error when running db2_install with '-p' option.

```
$.db2_install -b /opt/IBM/db2/V10.5 -p ESE
```

DBI1324W Support of the db2_install command is deprecated. For more information, see the DB2 Information Center.

DBI1060E Invalid package name ESE.

Explanation:

An incorrect name has been entered. The package either does not exist or the name has been entered incorrectly.

The db2_install Information Center says to look into the provided ComponentList.htm file to check for the 'Short Name' when using -p option. However, looking into this file you will see the following:

```
DB2 Server Edition
Product Short Name: ESE
Product Response File ID: DB2_SERVER_EDITION
```

You would see the same error for any other editions.
- **Cause**
 - Starting from V10.5 there has been a change to only use short name 'SERVER' when specifying for -p option.
 - **Answer**
 - The ComponentList.htm file has not been modified to show SERVER as a short name. This will be addressed in future V10.5 fixpack releases.

As shown below, use SERVER as -p input:

```
$.db2_install -b /opt/IBM/db2/V10.5 -p SERVER
```

 - See: <http://www-01.ibm.com/support/docview.wss?uid=swg21662600&myns=swgimgmt&mynp=OCSSEPPG&mync=E>

18

If you are planning on deep-embedding DB2 software within your solutions, check out **DB2 10 Silent Installation and Uninstallation Best Practices** whitepaper. Deep embedding means that no database administrator is involved and that all administration is done programmatically. This paper describes the processes of DB2 silent installation and uninstallation, silent DB2 fix pack updates, and silent upgrades to later DB2 versions and releases.

- <https://ibm.biz/Bdx8Hr>

REF: DB2 Setup wizard: Installing Windows servers

1. Log on to system with local Administrator account defined for DB2 installation.
2. Stop all DB2 instances, services and applications.
3. If have DB2 database product DVD, insert it into drive. If enabled, autorun feature automatically starts DB2 Setup Launchpad. If autorun does not work, use Windows Explorer to browse DB2 database product DVD and double-click setup icon to start DB2 Setup Launchpad. Can also run **setup.exe** to launch DB2 wizard.
4. If downloaded DB2 database product from Passport Advantage, run executable file to extract DB2 database product installation files. Use Windows Explorer to browse DB2 installation files and double-click setup icon to start DB2 Setup Launchpad.
5. From DB2 Setup Launchpad, view installation prerequisites and release notes, or proceed directly to installation with following choices:
 - To update existing DB2 copy and update all instances running on this DB2 copy, select Work with Existing in Install a Product panel. Then select DB2 copy want to update with update action. You cannot install additional functionality with this action.
 - To install new DB2 copy and selectively update instances running on existing DB2 copy to new copy after installation, select Install New in Install a Product panel. Allows you to also install additional functionality.
 - To add functionality to existing DB2 copy, select Work with Existing in Install a Product panel. Then select DB2 copy that want to update with Add new function action. Action only available when DB2 copy is at same release level as install image.
6. DB2 Setup wizard determines system language, and launches setup program for that language. Online help is available to guide you through remaining steps. To invoke online help, click Help or press F1. Can click Cancel at any time to end installation.
7. Sample panels when using DB2 setup wizard lead you to installation process.
8. IBM Data Studio can be installed by running the DB2 Setup wizard.
9. You will not be able to update more than one DB2 copy at the same time. In order to update other DB2 copies that might be installed on the same computer, you must rerun the installation.
10. Once installed or updated a DB2 copy, can always update instances that run in other DB2 copies of same release, to run on this new DB2 copy by issuing **db2iupdt** command.

Your DB2 database product is installed, by default, in the *Program_Files\IBM\sqllib* directory, where *Program_Files* represents the location of the Program Files directory.

If you are installing on a system where this directory is already being used, the DB2 database product installation path has *_xx* added to it, where *xx* are digits, starting at 01 and increasing depending on how many DB2 copies you have installed.

You can also specify your own DB2 database product installation path.

For information about errors encountered during installation, review the installation log file located in the *My Documents\DB2LOG* directory. The log file uses the following format: *DB2-ProductAbbrrev-DateTime.log*.

If this is a new DB2 product installation on Vista 64-bit, and you use a 32-bit OLE DB provider, you must manually register the IBMDADB2 DLL. To register this DLL, run the following command:

```
c:\windows\SysWOW64\regsvr32 /s c:\Program_Files\IBM\SQLLIB\bin\ibmdadb2.dll
```

where *Program_Files* represents the location of the Program Files directory.

If you want your DB2 database product to have access to DB2 documentation either on your local computer or on another computer on your network, then you must install the *DB2 Information Center*. The *DB2 Information Center* contains documentation for the DB2 database system and DB2 related products. By default, DB2 information is accessed from the web if the *DB2 Information Center* is not locally installed.

REF: Windows Installation Considerations

- During installation on Windows, if select **DB2 copy with upgrade action** in **Work with Existing** window and have local databases cataloged on instances, a message box warns that you must run **db2ckupgrade** from DB2 database product CD.
- Then can choose one of following actions:
 - Ignore message and continue the installation process.
 - Run **db2ckupgrade**. If command runs successfully, continue installation process. If find errors, quit installation process, fix any errors, and then rerun installation process.
 - Quit installation process.

Before you upgrade your DB2 server, review the upgrade essentials for DB2 servers, including recommendations, restrictions, and disk space requirements to identify the changes or restrictions that can affect your upgrade. You must be ready to address any issues before upgrade in order to have a successful upgrade.

If index root page does not have enough free space during upgrade, index will grow by 1 page. If free page cannot be found in index object, page will be requested from table space. If table space is full, entire index object marked invalid and rebuilt when the underlying table is accessed for first time after upgrade.

REF: DB2 Setup wizard: Installing Linux/UNIX servers

- If have physical DB2 database product DVD, change to directory where DB2 database product DVD is mounted by entering: `cd /dvdrom` where `/dvdrom` represents mount point of DB2 database product DVD.
- If downloaded DB2 database product image, extract product file.
 - **gzip** -d `product.tar.gz` where `product` name of the downloaded product.
- Untar on Linux:
 - **tar** -xvf `product.tar`
- Untar on AIX, HP-UX, and Solaris:
 - **guntar** -xvf `product.tar` where `product` name of the downloaded product.
- Change directory: `cd ./product` where `product` name of the downloaded product.
- **Note:** If downloaded National Language Package, untar into same directory which creates subdirectories (for example `./nlpack`) in same directory, and allows installer to automatically find installation images without prompting.
- DB2 Setup wizard is graphical installer. Must have X windows software capable of rendering graphical user interface. Export your display: **export DISPLAY=9.26.163.144:0**
- Enter **./db2setup** from directory where database product image resides to start DB2 Setup wizard.
- IBM DB2 Setup Launchpad opens. From window, can view installation prerequisites and release notes, or can proceed directly to installation. Can also review installation prerequisites and release notes for late-breaking information.
- Click Install a Product and Install a Product window will display products available for installation.
- Launch installation by clicking Install New. Proceed through installation following DB2 Setup wizard's prompts.
- Sample panels when using DB2 setup wizard will lead you to installation process.
- After have initiated installation, proceed through DB2 Setup wizard installation panels and make selections. Installation help is available to guide through remaining steps. To invoke installation help, click Help or press F1. Can click Cancel at any time to end installation.

21

For non-root installations, DB2 database products are always installed in the `$HOME/sqllib` directory, where `$HOME` represents the non-root user's home directory. For root installations, DB2 database products are installed, by default, in one of the following directories:

- **AIX, HP-UX, and Solaris:** `/opt/IBM/db2/V10.1`
- **Linux:** `/opt/ibm/db2/V10.1`

If you are installing on a system where this directory is already being used, the DB2 database product installation path will have `_xx` added to it, where `_xx` are digits, starting at 01 and increasing depending on how many DB2 copies you have installed.

You can also specify your own DB2 database product installation path.

DB2 installation paths have the following rules:

- Can include lowercase letters (a-z), uppercase letters (A-Z), and the underscore character (_)
- Cannot exceed 128 characters
- Cannot contain spaces
- Cannot contain non-English characters

The installation log files are:

- The DB2 setup log file. This file captures all DB2 installation information including errors.
 - For root installations, the DB2 setup log file name is `db2setup.log`.
 - For non-root installations, the DB2 setup log file name is `db2setup_username.log`, where `username` is the non-root user ID under which the installation was performed.
- The DB2 error log file. This file captures any error output that is returned by Java (for example, exceptions and trap information).
 - For root installations, the DB2 error log file name is `db2setup.err`.
 - For non-root installations, the DB2 error log file name is `db2setup_username.err`, where `username` is the non-root user ID under which the installation was performed.

By default, these log files are located in the `/tmp` directory. You can specify the location of the log files.

There is no longer a `db2setup.his` file. Instead, the DB2 installer saves a copy of the DB2 setup log file in the `DB2_DIR/install/logs/` directory, and renames it `db2install.history`. If the name already exists, then the DB2 installer renames it `db2install.history.xxxx`, where `xxxx` is 0000-9999, depending on the number of installations you have on that machine.

Student Notes: Reference Only

Each installation copy has a separate list of history files. If an installation copy is removed, the history files under this install path will be removed as well. This copying action is done near the end of the installation and if the program is stopped or aborted before completion, then the history file will not be created.

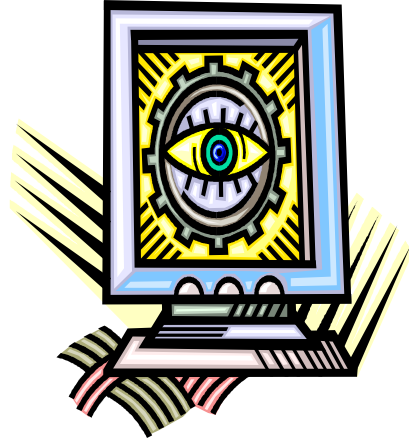
On Linux x86, if you want your DB2 database product to have access to DB2 documentation either on your local computer or on another computer on your network, then you must install the *DB2 Information Center*. The *DB2 Information Center* contains documentation for the DB2 database system and DB2 related products.

DB2 Express Edition and DB2 Workgroup Server Edition memory limits. If you are installing DB2 Express Edition, the maximum allowed memory for the instance is 4 GB. If you are installing DB2 Workgroup Server Edition, the maximum allowed memory for the instance is 64 GB. The amount of memory allocated to the instance is determined by the INSTANCE_MEMORY database manager configuration parameter.

Important notes when upgrading from V9.7, V9.8, or V10.1: If the memory configuration for your Important notes when upgrading from V9.7, V9.8, or V10.1 DB2 database product exceeds the allowed limit, the DB2 database product might not start after upgrading to the current version. The self tuning memory manager will not increase your overall instance memory limit beyond the license limits.

REF: Verify installation using CLP – SAMPLE Database

1. Create new 10.5 instance.
`#DB2DIR/instance/db2icrt -u db2fenc1 db2inst1`
2. Log on to system as user with SYSADM authority.
3. Start database manager
 - **db2start**
4. Create the SAMPLE database.
 - **db2sampl**
 - Can take a few minutes to process. No completion message; when command prompt returns, process is complete.
 - SAMPLE database is automatically cataloged with database alias SAMPLE when created.
 - On non-partitioned created with XML objects. On partitioned database can use **-xml** option to create XML objects.
5. Connect to SAMPLE database, retrieve a list of employees that work in department 20, and reset the database connection.
 - **db2 connect to sample**
 - **db2 "select * from staff where dept = 20"**
 - **db2 connect reset**
6. After verified installation, remove SAMPLE database to free disk space.
 - **db2 drop database sample**
 - `/opt/IBM/db2/copy1/instance/db2idrop db2inst1`



23

You can verify the installation by creating the SAMPLE database and running SQL commands to retrieve sample data. The SAMPLE database component, found in the features selection, must be installed on your system and is included in a typical installation. You require a user with SYSADM authority.

In non-partitioned database environments:

- Creates a database named SAMPLE with a Unicode (UTF-8) code set in the default database path.
- Creates relational database objects including tables, indexes, constraints, triggers, functions, procedures, multi-dimensional clustered tables and materialized query tables.
- Populates relational tables with data.
- Creates tables with XML data type columns.
- Creates indexes over XML data.
- Creates an XML schema repository that contains XML schema documents.
- All database object names are qualified with the value of the CURRENT_SCHEMA special register.

Student Notes: Reference Only

In partitioned database environments:

- Creates a database named `SAMPLE` with a Unicode (UTF-8) code set in the default database path.
- Creates relational database objects including tables, indexes, constraints, triggers, functions, procedures, multi-dimensional clustered tables and materialized query tables.
- Populates tables with data.
- All database object names are qualified with the value of the `CURRENT_SCHEMA` special register.
- The `-xml` option can be used to create tables with columns of data type XML, creates indexes on the XML columns, registers XML schemas, and populates these tables with data including XML document values. This option is only supported where XML is supported. If XML is not supported, this option is ignored.

Checklist: First steps after installation

1. [Deleting a Firefox browser profile](#)

After run DB2 Setup wizard or DB2 First Steps, may want to delete browser profile to prevent JavaScript from running automatically when visiting other Web pages with this profile.

2. [Applying DB2 licenses](#)

To license your product, register appropriate license key with DB2 database product.

3. [Updating DB2 licenses](#)

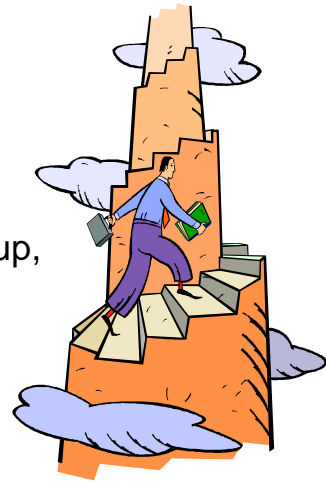
Can switch between editions in DB2 10.5 - Workgroup, Enterprise, and Advanced Editions common image.

4. [Post-installation tasks for DB2 database servers](#)

After installing DB2 database, you must complete a number of tasks.

5. **OPTIONAL:** [First Steps interface](#)

Launches First Steps interface, which contains links to functions that you need to begin learning about and using DB2 product.



After you install a DB2 database product, there are additional necessary and optional tasks available.

REF: 1. Deleting Firefox browser profile:

- If any Firefox browsers are open, close them.
- Open the Firefox Profile Manager.
 - On Windows
 - Open the Windows Start menu and select Run...
 - Type `firefox.exe -profilemanager` and click OK. The Firefox - Choose User Profile dialog opens.
 - On Linux and UNIX
 - Open a command prompt.
 - Switch to the Firefox directory.
 - Start Firefox with the `-profilemanager` switch. For example: `./firefox -profilemanager`
- Select the `DB2_FIRSTSTEPS` profile.
- Click Delete Profile...
- Click Exit to close the Firefox - Choose User Profile dialog.



26

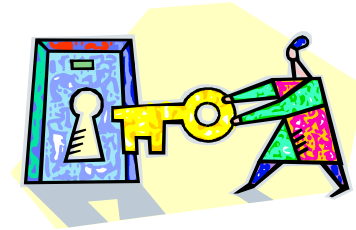
If you use this profile to view other Web pages that have JavaScript, the JavaScript will run automatically without warnings.

If you use Firefox as your Web browser, a browser profile is created when you run DB2 First Steps. If you agree, a browser profile named `DB2_FIRSTSTEPS` is created.

2. Applying DB2 licenses – db2licm

- If want license key added automatically during product installation, copy license key to **/db2/license** directory of installation image before launching DB2 Setup wizard.
- If do not add any license files to /db2/license directory, see "*License not registered*" message from **db2licm -l** command.

- Can apply licenses manually after installation by running **db2licm -a**.



- If register valid license key, **db2licm -l** command successfully lists all products with available license information.

27

To license your product, register the appropriate license key with your DB2 database product.

If you want the license key added automatically during product installation, copy the license key to the /db2/license directory of the installation image before launching the DB2 Setup wizard.

If you do not add any license files to the /db2/license directory, you will see a "License not registered" message in the output from the db2licm -l command. The following base licenses are available at the /db2/license directory of the server image and the product-specific base license is automatically applied during DB2 product installation.

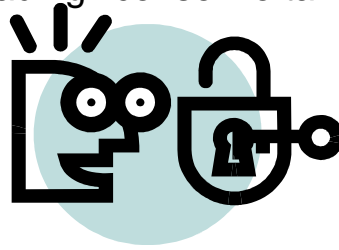
- db2ese.lic
- db2exp.lic
- db2consv.lic

You can apply the licenses manually after the installation by running the db2licm -a command. If you register a valid license key, the db2licm -l command will successfully list all the products with available license information

To use DB2 features or to be in compliance with your current product entitlement, you might be required to register additional license keys.

How to download DB2 license?

- **License files for DB2 Version 10.5 are shipped separately.**
 - Done for convenience so downloading license file takes less time due to small size.
 - Download license activation key from Passport Advantage.
 - Then install it.



How do I download my DB2 license from Passport Advantage?

[http://www-01.ibm.com/support/docview.wss?](http://www-01.ibm.com/support/docview.wss?uid=swg21305417&myns=swgimgmt&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E)

[uid=swg21305417&myns=swgimgmt&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E](http://www-01.ibm.com/support/docview.wss?uid=swg21305417&myns=swgimgmt&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E)

DB2 10.5 Activation Key Part Numbers

<http://www-01.ibm.com/support/docview.wss?uid=swg21641360>

28

How do I download my DB2 license from Passport Advantage? The license files for DB2 Version 10.5 are shipped separately. This is done for convenience so that downloading the license file takes less time due to its small size. You will need to download the license activation key from Passport Advantage and then install it.

A detailed example of how to obtain the activation key for your DB2 product is shown at:

[http://www-01.ibm.com/support/docview.wss?](http://www-01.ibm.com/support/docview.wss?uid=swg21305417&myns=swgimgmt&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E)

[uid=swg21305417&myns=swgimgmt&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E](http://www-01.ibm.com/support/docview.wss?uid=swg21305417&myns=swgimgmt&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E)

The Activation key part numbers for DB2 10.5 products are located at the following website: <http://www-01.ibm.com/support/docview.wss?uid=swg21641360>

3. Updating licenses

- To switch between DB2 10.5 Workgroup, Enterprise, and Advanced Edition licenses:
 - Cannot apply DB2 pureScale license over DB2 Advanced Enterprise Server Edition or DB2 Advanced Workgroup Server Edition terabyte license. If a DSF license exists then cannot apply DB2 Advanced Enterprise Server Edition or DB2 Advanced Workgroup Server Edition terabyte licenses. DB2 pureScale license cannot coexist with DB2 Advanced Enterprise Server Edition or DB2 Advanced Workgroup Server Edition terabyte licenses.
- Run **db2licm -l** to list all DB2 product licenses registered on system.
db2licm -l *product-identifier*
- Run **db2licm -r** to remove existing product license, feature license, and trial license.

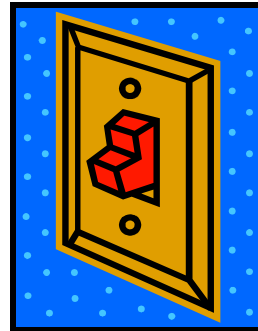
db2licm -r *product-identifier*
- Run **db2licm -a** to apply new DB2 product license on system. If existing product license was not removed by running db2licm -r, it is automatically removed.
db2licm -a *product-identifier*
- Run **db2licm -l** to view newly applied DB2 product license information.
db2licm -l *product-identifier*
- Run **db2start** to restart instance, and for license compliance report to take effect for new license.
db2start

29

You can switch between editions in the DB2 Version 10.5 - Workgroup, Enterprise, and Advanced Editions common image.

4. Post-installation tasks: DB2 Windows servers

- Add users to DB2ADMNS and DB2USERS user groups
- Update 32-bit DB2 instances to 64-bit instances
- Validating DB2 copy: **db2val -a**
- Set up notification and contact lists
- List DB2 copies and change default DB2 or default IBM database client interface copy after installation:
db2swtch -l
db2swtch -d DB2_copy_name
- Check for DB2 updates
- Installing SSH Server component after installing DB2 product
- IBM Secure Shell Server For Windows required for Optim Database Administrator usage



http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.qb.server.doc/doc/c0059310.html?lang=en

30

After installing DB2 database on Windows systems, you must complete a number of tasks.

- **Adding your user ID to the DB2ADMNS and DB2USERS user groups (Windows)**
 - After successfully completing a DB2 installation, you now have to add users to the DB2ADMNS or the DB2USERS groups for users that need to run local DB2 applications and tools on the machine. The DB2 installer creates two new groups. You can either use a new name or accept the default names. The default group names are DB2ADMNS and DB2USERS. This process is also applicable to instance based clients.
- **Updating your 32-bit DB2 instances to 64-bit instances (Windows)**
 - If you have more than one 32-bit DB2 copy on your system, you can convert them to 64-bit instances.
- **Validating your DB2 copy**
 - The **db2val** command ensures that your DB2 copy is functioning properly.
- **Setting up notification and contact lists**
 - For your DB2 database product to contact you or others regarding the status of your databases, set up notification and contact lists. If you did not do this during the DB2 database product installation, you can manually set up these lists.
- **Changing the default DB2 and default IBM database client interface copy after installation (Windows)**
 - The **db2swtch** command can be run from any DB2 copy or IBM data server driver copy. If no arguments, launches the utility in graphical mode. The **-l** displays a list of DB2 copies and IBM data server driver copies on the system. To switch the default DB2 copy to the name specified: **db2swtch -db2 -d DB2_copy_name-client**. The **-d** specifies the name of DB2 copy or IBM data server driver copy. To switch the default client interface copy to the name specified: **db2swtch -client -d name of DB2 copy or IBM data server driver copy-d DB2_copy_name**. To switch both default DB2 copy and client interface copy to the name specified:
db2swtch -d DB2_copy_name.
- **IBM data server client connectivity using multiple copies**
 - When using multiple DB2 copies of DB2 database products or multiple data server driver copies, various options are available for applications to access the DB2 databases. Existing applications will continue to function properly.
- **Working with existing DB2 copies**
 - You can install and run multiple DB2 copies on the same computer. Each DB2 copy can either be at the same DB2 database product level or at a different DB2 database product level.

Configuring IBM Data Studio web console for remote operations

To run SQL statements and commands on remote servers from Data Studio web console, must ensure Secure Shell (SSH) and Secure File Transfer (SFTP) protocols properly configured.

- For remote operations, Data Studio web console requires Secure Shell (SSH) and Secure File Transfer (SFTP) protocols.
- Computers that host remote databases must have Secure Shell Daemon (SSHD) configured with both SSH and SFTP protocols enabled.
- For windows, Data Studio web console supports IBM SSH Server that is bundled with IBM DB2 v10.1 and later database. OpenSSH server implementation of Cygwin environment is also supported.
- Data Studio web console uses SSH and SFTP for remote operations and also when Data Studio web console is running on same system as database server. Unlike Data Studio client, Data Studio web console does not use DB2 Administration Server (DAS), nor does it require that IBM Data Server Client software be installed on the same machine.

31

Check for product updates to ensure that you are aware of the available product updates and enhancements for DB2 products.

A new service called "IBM Secure Shell Server For Windows" will be installed together with the DB2 Server products. This service is a limited functionality port from the OpenSSH 5.4p1 package. This service provides a server service for the Optim Database Administrator, as well as other approved IBM products. This service is not for general usage outside this scope.

IBM Secure Shell Server For Windows uses the same configuration method as OpenSSH which is based on the `sshd_config` file. This configuration file can be found under `%ALLUSERSPROFILE%\Application Data\IBM\IBMSSH\etc` directory, where `%ALLUSERSPROFILE%` is the Windows provided environment variable which can be displayed using `"set ALLUSERSPROFILE"`. Not all server functionalities are provided, refer to the `sshd_config` file for the supported feature. Only the supported feature can be configured through this file.

Logs of this service can be viewed using the Windows Event Viewer, under the Application category. Search for Event Source - "IBM Secure Shell Server For Windows".

To start or stop the IBM Secure Shell Server For Windows, you can use the Windows Services Microsoft Management Console. Alternatively, you can use the command interpreter to issue `"net start IBM Secure Shell Server For Windows"` or `"net stop IBM Secure Server For Windows"`.

4. Post-installation tasks: DB2 Linux/UNIX servers

- List DB2 database products installed on system: **db2ls**
 - To list info about DB2 database products or features in particular installation path `q` parameter must be specified:
 - **db2ls -q -p -b baseInstallDirectory**
 - `q` specifies querying product or feature. Parameter is mandatory.
 - `p` specifies listing displays products rather than listing features.
 - `b` specifies installation directory of product or feature. Parameter is mandatory if not running command from installation directory.
 - Forgot which DB2 features you installed?
`/opt/IBM/db2/V10.5/install/db2ls -q -a -b /opt/IBM/db2/V10.5`
- Verify basic functions of DB2 copy by checking state of installation files, instance setup, and local database connections.
 - To validate all instances for DB2 copy: **db2val -a**
- Main menu entries for DB2 tools (Linux)
 - After installation, can add several DB2 tools to Main menu. See notes for details.

32

After installing DB2 database on Linux or UNIX systems, you must complete a number of tasks.

• Listing DB2 database products installed on your system (Linux and UNIX)

On supported Linux and UNIX operating systems, the `db2ls` command lists the DB2 database products and features installed on your system, including the DB2 Version 10.1 HTML documentation. With the ability to install multiple copies of DB2 database products on your system and the flexibility to install DB2 database products and features in the path of your choice, you need a tool to help you keep track of what is installed and where it is installed. On supported Linux and UNIX operating systems, the `db2ls` command lists the DB2 products and features installed on your system, including the DB2 HTML documentation. The `db2ls` command can be found both in the installation media and in a DB2 install copy on the system. The `db2ls` command can be run from either location. The `db2ls` command can be run from the installation media for all products except IBM® Data Server Driver Package. The `db2ls` command can be used to list:

- Where DB2 database products are installed on your system and list the DB2 database product level
- All or specific DB2 database products and features in a particular installation path.

The output that the `db2ls` command lists is different depending on the ID used:

- When the `db2ls` command is run with root authority, only root DB2 installations are queried.
- When the `db2ls` command is run with a non-root ID, root DB2 installations and the non-root installation owned by matching non-root ID are queried. DB2 installations owned by other non-root IDs are not queried.

The `db2ls` command is the only method to query a DB2 database product. You *cannot* query DB2 database products using Linux or UNIX operating system native utilities, such as `pkginfo`, `rpm`, `SMIT`, or `swlist`. Any existing scripts containing a native installation utility that you use to query and interface with DB2 installations must change. You *cannot* use the `db2ls` command on Windows operating systems.

• Validating your DB2 copy

- The `db2val` command ensures that your DB2 copy is functioning properly. The `db2val` tool verifies the core function of a DB2 copy by validating installation files, instances, database creation, connections to that database, and the state of partitioned database environments. This validation can be helpful if you have manually deployed a DB2 copy on Linux and UNIX operating systems using `tar.gz` files. The `db2val` command can quickly ensure that all the configuration has been correctly done and ensure that the DB2 copy is what you expect it to be. You can specify instances and databases or you can run `db2val` against all of the instances. The `db2val` command can be found in the `DB2-install-path\bin` and `sqlib\bin` directories

Student Notes: For Reference Only

Main menu entries for DB2 tools (Linux)

After installation, you can add several DB2 tools to the Main menu:

Check for DB2 Updates

Command Line Processor (CLP) and Command Line Processor Plus (CLPPlus)

First Steps

These DB2 tools can be added to main menu automatically or manually. Main menu entries are created automatically when any of the following DB2 commands are run. The `db2icrt` and `db2iupdt` commands must be run by root. The `db2nrcfg` and `db2nrupdt` are for non-root install and are run by the instance owner.

- `db2icrt`, `db2iupdt`, `db2nrcfg`, `db2nrupdt`.

To see the menu entries you might need to restart the desktop. The main menu entries are automatically removed when either of the following commands are run:

- `db2_deinstall` (only removes the menu entries for the DB2 non-root instance when the non-root installation is removed)
- `db2idrop`.

You can manually create or remove main menu entries by running the following commands:

- `db2addicons` – to add menu entries
- `db2rmicons` – to remove menu entries.

Before running the `db2addicons` command, the DB2 instance environment must be set for the current user. The instance environment can be setup with: `Instance_HOME/sql/lib/db2profile` (for Bourne shell and Korn shell users), or `Instance_HOME/sql/lib/db2chsrc` (for C shell users), where `Instance_HOME` is the instance owner's home directory.

REF: db2val - DB2 copy validation tool

- **-o** Specifies only installation files will be validated. If specified, -i, -a, -b, and -s are ignored. Validation of instance, database, and extended security not performed
- **-i instance_name** Specifies instance to validate. For multiple instances, specify parameter multiple times. On Windows, if not specified, current instance is default. On Linux/UNIX, parameter only be used by root users in a root installation of a DB2 copy.
- **-a** Validates all instances in DB2 copy. On Linux /UNIX, parameter only be used by root users in a root installation of a DB2 copy. Overrides parameter -i.
- **-b <db_name>** Validates database creation and connections to database specified. Only active DB2 instances will be validated. Parameter ignored for DB2 client and DB2 pureScale instances.
- **-t <trace_file>** Applies only to Linux/UNIX. Specifies full path and name of trace file.
- **-d** Deprecated and might be removed in a future release. Use the -t parameter instead.
- **-s** Starts DB2 database manager for specified instance that is part of partitioned database environment.
- **-l <log_file>** Writes log to file name specified. Default on Linux/UNIX is /tmp/db2valxx.log and on Windows My Documents\DB2LOG\db2valxx.log.
- **-? | -h** Displays usage information

34

Verifies basic functions of DB2 copy by checking state of installation files, instance setup, and local database connections.

You can use the db2val command only to validate installation files in a DB2 pureScale environment.

Instance validation requires one of the following authorities:

- On root copies, root authority is required on Linux and UNIX operating systems.
- SYSADM plus one of the following authorities:
 - Instance owner
 - Root access on Linux and UNIX operating systems, or Local Administrator authority on Windows operating systems

To validate the instance TEST1 and the database DATA1, run the following command:

```
db2val -i TEST1 -b DATA1
```

To validate all the instances for the DB2 copy, run the following command:

```
db2val -a
```

To validate only the DB2 installation files, run the following command:

```
db2val -o
```

REF: 5. OPTIONAL: First Steps interface - db2fs

- Manually start First Steps using **db2fs**:
 - **\$ db2fs**
- Create sample database.
- Check for product updates.
- Download and learn about Data Studio.
- Learn about licensing and support.



- To uninstall single feature:
\$ /opt/IBM/db2/V10.5/install/db2_deinstall -F FIRST_STEPS
- **Linux and UNIX**
 - db2fs command located in *INSTHOME*/sqllib/bin directory, where *INSTHOME* is instance home directory. SYSADM authority required.
 - Alternatively, on Linux operating systems can start First Steps from Main Menu by selecting IBM DB2 > First Steps.
- **Windows**
 - db2fs command located in *DB2DIR*/bin directory, where *DB2DIR* is set to location specified during DB2 installation.
 - Alternatively, can start First Steps from Start menu by selecting Programs > IBM DB2 > [*DB2 Copy Name*] > Set Up Tools > First Steps or from DB2 database product folder on desktop.

35

Launch the First Steps interface, which contains links to the functions that you need to begin learning about and using the DB2 product.

From First Steps you can:

- Create the sample database or your own database.
- Check for product updates.
- Learn more about IBM Data Studio components.
- Learn more about DB2 database product licensing and support.

Product updates

Ensure that you are aware of the available product updates and enhancements for DB2 database products. With the update service you can view a list of available DB2 database product updates, and learn details about DB2 database product updates.

db2ckupgrade - Check database for upgrade

db2ckupgrade <database-> -I filename

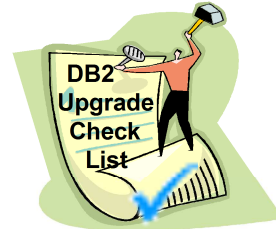
- In partitioned database, db2ckupgrade **will check each** database partition.
- On Linux/UNIX, install new DB2 copy and run from *DB2DIR/bin* directory where *DB2DIR* is location where DB2 copy is installed
- On Windows, insert DB2 product CD and run from db2\Windows\Utilities directory on CD
- SYSADM authority

db2ckupgrade -e -I filename

- e specifies all local cataloged databases are to be scanned

db2ckupgrade <database-> -I filename **-not1**

- I filename specifies log file to keep list of errors and warnings



- Confirm running correct level of db2ckupgrade command. Ensure db2ckupgrade log file contains following text:

Version of DB2CKUPGRADE being run: Version 10.5

26

The **db2ckupgrade** command must be issued prior to upgrading the instance to verify that your databases are ready for upgrade. Backup all databases prior to upgrade, and prior to the installation of the current version of DB2 database product on Windows operating systems. Once the codebase has been installed you should then run the db2ckupgrade before upgrading the instance or databases. The db2ckupgrade fails to run against databases which are catalogued as remote databases.

The **db2ckupgrade** command verifies that a database can be migrated. In a partitioned database environment, the db2ckupgrade command will check each database partition.

Ensure that the log file for db2ckupgrade command contains the following text:

Version of DB2CKUPGRADE being run: Version 10.5. This text confirms that you are running the correct level of the db2ckupgrade command.

SYSADM authority is required to execute. No database connection is required.

REF: db2ckupgrade verifies following conditions are true:

- A catalogued database actually exists.
- A database is not in inconsistent state.
- A database is not in backup pending state.
- A database is not in restore pending state.
- A database is not in rollforward pending state.
- Tables are not in load pending state.
- Tables are not in redistribute pending state.
- For V9.8 or later, that all table space container paths use the same mount point.
- For V9.8 FixPack 3 or later, that I/O write operations for database are not suspended
- No MQT's that depend on system views.
- Table spaces are in normal state.
- Database does not contain UDTs with name ARRAY, BINARY, CURSOR, DECFLOAT, ROW, VARBINARY, or XML.
- Database does not contain built-in DATALINK data type.
- Database does not have schema with name SYSPUBLIC.
- Database does not have orphan rows in system catalog tables that cause database upgrade to fail.
- Database enabled as HADR primary database allows successful connections.
- HADR database role is not standby.
- Database is not enabled for XML Extender.
- If SYSCATSPACE is DMS table space and AUTORESIZE not enabled, SYSCATSPACE has at least 50% free pages of total pages.

37

A local database must pass all of these checks to succeed at the upgrade process.

The **db2iupgrade** command implicitly calls the **db2ckupgrade** command with the **-not1** parameter, and specifies **update.log** as the log file for **db2ckupgrade**. The default log file created for **db2iupgrade** is **/tmp/db2ckupgrade.log.processID**. The **db2iupgrade** fails if the **db2ckupgrade** command finds that any of the above listed conditions are not true, and returns the **DBI1205E** error code. The user needs to resolve these errors before upgrading the instance.

REF: db2ckupgrade verification (continued)

- db2ckupgrade writes warning message to log file (-l parameter) for any of following conditions:
 - Column names, routine parameter names, or variable names are called NULL.
 - Workload connection attributes contain asterisks (*).
 - Database enabled for DB2 WebSphere MQ functions.



For partitioned database environments, run the db2ckupgrade command before you issue the db2iupgrade command. The db2ckupgrade command will check all partitions and will return errors found in any partition. If you do not check whether all database partitions are ready for upgrade, subsequent database upgrades could fail even though the instance upgrade was successful.

db2ckupgrade – SQL0430N error

SQL0430N – Someone has removed the fenced user id or fenced userid does not have adequate permissions on instance

[IBM] [CLI Driver][DB2/AIX64] SQL0430N User defined function "SYSPROC.REG_LIST_VARIABLES" (specific name "REG_LIST_VARIABLES") has abnormally terminated SQLSTATE=38503

1. Check if DB2 fenced user id exists:

\$ ls -al \$HOME/sqllib/.fenced

(where \$HOME is instance home directory)

If owner of .fenced file is numeric value, then user was dropped.

2. Root resolve problem by issuing db2iupdt from \$DB2DIR/instance with -u option and valid fenced userid (where \$DB2DIR is installation directory for DB2):

db2iupdt -u db2fenc1

When migrating a DB2 database, the db2ckupgrade command could produce a SQL0403N error which means that someone removed the fenced user id or the fenced user id does not have adequate permissions on the instance.

Check if DB2 fenced user id exists:

ls -al \$HOME/sqllib/.fenced (where \$HOME is instance home directory)

If owner of .fenced file is number then represents user that was dropped.

Resolve problem by issuing db2iupdt as root from \$DB2DIR/instance

(where \$DB2DIR is installation directory for DB2) with -u option and valid fenced userid.:

#db2iupdt -u db2fenc1

db2chkupgrade or db2iupgrade -b option

- > **PROBLEM:** db2chkupgrade or db2iupgrade take long time to complete.
 - > Additional check for tables in load pending state added to db2chkupgrade tool in the DB2 10.1 and 10.5.
 - > If large number of tables, indexes, data, check can take long time.
- > **SOLUTION:** Only use if 0 rows returned!
**SELECT tablename FROM SYSIBMADM.ADMINTABINFO
WHERE LOAD_STATUS IS NOT NULL**
0 Records returned
- > **Internal -b option added in DB2 10.5 Fixpack 3 and DB2 10.1 Fixpack 3 bypasses check for tables in load pending state:**
db2chkupgrade <dbname> -b -l <mylog.log>
db2iupgrade -b <myinstance>

(Remember: db2iupgrade calls db2chkupgrade.)

The db2chkupgrade or db2iupgrade tools may take a long time to complete when upgrading to the DB2 v10.1 or v10.5 products. An additional check for tables in load pending state was added to the db2chkupgrade tool in the DB2 v10.1 and v10.5 products. This check can take a long time if a database has a large number of tables, defined indexes and hosts a considerable amount of data. A query against SYSIBMADM.ADMINTABINFO, may be used to obtain this information from the database.

To diagnose the problem, looking at the db2diag.log file. It may show that the db2chkupgrade tool is spending a lot of time using the sqlmCkTableLoadPending() function.

The following example from db2diag.log file shows gap of 2 hours between beginning and end message for the sqlmCkTableLoadPending function:

```
2013-02-16-09.44.55.802740+000 I161183A335 LEVEL: Warning
PID : 5178038 TID : 1 PROC : db2chkupgrade64_exe
INSTANCE: db2inst NODE : 000
APPID : *N0.db2inst.130216094415
EDUID : 1
FUNCTION: <0>, <0>, <0>, probe:4138
DATA #1 : <preformatted>
DB2CKUPGRADE: Begin: sqlmCkTableLoadPending
```


Student Notes: For Reference Only

```
2013-02-18-09.46.20.491662+000 I64434131A309 LEVEL: Warning
PID : 5178038 TID : 1 PROC :
db2ckupgrade64_exe
INSTANCE: db2inst NODE : 000
APPID : *N0.db2inst.130216094415
EDUID : 1
DATA #1 : <preformatted>
DB2CKUPGRADE: End: sqlmCkTableLoadPending with rc = 0
```

While the issue actually occurs during the execution of the db2ckupgrade or db2iupgrade tools, only one initiate see the begin message for the sqlmCkTableLoadPending function.

Additional active SQL monitoring will show following statement running:

```
"select tabname from SYSIBMADM.ADMINTABINFO where load_status is not NULL"
```

You can use a temporary work around for the issue. Starting with the v10.1 Fix Pack 3 and v10.5 Fix Pack 3 versions of the DB2 product, the db2ckupgrade and db2iupgrade tools have an internal option , -b, which bypasses the check for tables in load pending state

Disk space requirements (1)

- **System catalog table space (SYSCATSPACE)**

- Increase to twice the total of used space.
- Free pages should be \geq used pages.
- System catalog table space is required for both old and new database catalogs during upgrade.



- **Temporary table space (TEMPSPACE1)**

- Increasing total size to twice total size of system catalog table space.

- db2 "SELECT SUBSTR(TBSP_NAME,1,15) NAME, TBSP_TYPE TYPE, TBSP_AUTO_RESIZE_ENABLED AUTO_RESIZE, TBSP_NUM_CONTAINERS CONTAINERS, TBSP_TOTAL_PAGES TOTAL_PGS, TBSP_USED_PAGES USED_PGS, TBSP_FREE_PAGES FREE_PGS, TBSP_MAX_SIZE MAX_SZ, TBSP_PAGE_SIZE PG_SZ FROM SYSIBMADM.TBSP_UTILIZATION WHERE TBSP_CONTENT_TYPE IN ('ANY','SYSTEMP')"

42

You must be aware that the upgrade process requires additional disk space. Ensure that you have enough free disk space to complete this process successfully. The above disk space recommendations are applicable for upgrading to DB2 10.5.

Ensure that you have sufficient free space on the system catalog and the system temporary table spaces for the databases that you are upgrading. System catalog table space is required for both old and new database catalogs during upgrade. The amount of free space required varies, depending on the complexity of the database, as well as on the number and size of database objects.

To increase the amount of free space on your automatic storage table spaces, you can increase the space on the current storage paths or add a new storage path.

To increase the amount of free space on your System Managed Space (SMS) table spaces, free sufficient disk space on the corresponding file systems or increase the size of your file systems if you are using a volume manager.

To increase the amount of free space on your Database Managed Space (DMS) table spaces, you can increase the size of existing containers. You can also add additional containers although this might trigger data rebalancing. You can reduce the size of the containers after upgrade.

Disk space requirements (2)

- **Log file space**
 - Upgrade changes to system catalog objects are single UOW.
 - Set LOGSECOND to twice current value of LOGPRIMARY and LOGSECOND if file system with log files has enough disk free space.
 - If already have large log file space, might not be necessary.
 - **Only need to increase log space on catalog partition.**
 - **Must update these values before upgrade the instance to DB2 10.5**
 - Optional: Consider infinite logging (LOGSECOND=-1)

db2 GET DB CFG FOR sample | grep '(LOG[FPS])' tee logsize.txt

db2 UPDATE DB CFG FOR sample using LOGSECOND
(current_value of LOGPRIMARY + current_value of LOGSECOND)

- **Index space**
 - Each index on every populated table requires 1 additional page per index to use following functionality:
 - Real-time statistics.
 - Deferred cleanup roll out for MDC tables.
 - Index rebuild on a populated table.
- **Automatic storage files**
 - If enable automatic storage on database with ALTER DATABASE ... ADD STORAGE, creates SQLSGF.1 and SQLSGF.2 files required for maintaining automatic storage.



43

You must update these LOGSECOND values before you upgrade the instance to DB2 10.5, because you will not be able to update these DB CFG parameters until issue the UPGRADE DATABASE command. If this command fails because there is insufficient log file space, then you can set these database configuration parameters to higher values and then re-issue the UPGRADE DATABASE command.

The new database configuration parameter settings for log space can be restored to their original value after the upgrade is complete.

If have limited amount of free disk space for indexes, can get error message SQL0289N that indicates the table space is full. Ensure that you have enough free pages in the corresponding index table space to account for one additional page per index on populated tables before:

- Populating tables in databases created in DB2 Version 9.7 or later, real-time statistics are enabled by default in these newly created databases.
- Enabling deferred cleanup roll out by setting DB2_MDC_ROLLOUT to DEFER, or when DB2_WORKLOAD is set to SAP.
- Reorganizing or recreating indexes on populated tables.

REF: Pre-upgrade tasks for DB2 servers

1. Ensure at least 1 free page of index space per object index to eliminate overhead of potential index rebuild.
2. If use distributed transactions, ensure databases do not contain any indoubt transactions by using LIST INDOUBT TRANSACTIONS to get list of indoubt transactions and interactively resolve any indoubt transactions.
3. Verify databases are ready for DB2 upgrade to identify any problems before actual upgrade. Must resolve them before proceed with upgrade.
4. Optional: **Stop HADR on primary and standby databases.**
5. **Upgrade from DB2 Query Patroller to WLM. Query Patroller is discontinued.**
6. **Back up databases to be able to upgrade them to a new upgraded system or restore them in the original pre-upgrade system.**
7. **Back up configuration and diagnostic information to have record of current configuration that can compare with configuration after upgrade. Use this information to create new instances or databases using same configuration that had before upgrade.**
8. Archive all of DB2 log files, either for SQL or Q replication if log files needed by Capture or Q Capture, or for HADR if needed to create standby database.
9. **Review disk space requirements to ensure have enough free disk space, system temporary table space and log space for the upgrade and increase table space and log file sizes if necessary.**
10. Windows only: If obtained customized code page conversion tables from the DB2 support service, need to backup all files in DB2OLD\conv directory where DB2OLD is the location of your existing pre-DB2 10.5 copy.
11. Linux only: Change raw devices to block devices.
12. Optional: Upgrade DB2 server in test environment to identify upgrade issues and verify applications, scripts, tools and routines work as expected before upgrading DB2 server in production environment.
13. **If diaglevel is 2 or less, set diagnostic error capture level to 3 or higher.**
14. **Take DB2 server offline for upgrade.**
15. Refresh data in existing MQTs. All MQTs that depend on system views are dropped during database upgrade. After upgrade, must refresh data in existing MQTs by using REFRESH TABLE. 44

Ensure that you have at least one free page of index space per object index to eliminate the overhead of a potential index rebuild. If an index root page does not have enough free space during upgrade, then the index will need to grow by one page. If a free page cannot be found in the index object, then a page will be requested from the table space. If the table space is full, then the entire index object will be marked invalid and will be rebuilt when the underlying table is accessed for the first time after upgrade.

If you use distributed transactions involving DB2 databases, ensure that the databases to be upgraded do not contain any indoubt transactions by using the LIST INDOUBT TRANSACTIONS command to get a list of indoubt transactions and to interactively resolve any indoubt transactions.

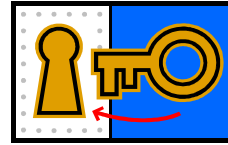
In DB2 10.5, all significant upgrade events are logged in the **db2diag** log files when the **diaglevel** database manager configuration parameter is set to 3 (default value) or higher. If this parameter is set to 2 or less, set this parameter to 3 or higher before upgrade.

Back up databases before AND after upgrade

After upgrade instances to DB2 10.5, cannot backup databases until upgrade them!

- Perform a full offline back up for each of your local databases:

1. To disconnect all applications and users:
db2 FORCE APPLICATION ALL



2. Backup database using BACKUP DATABASE.
db2 BACKUP DATABASE database_alias TO backup-dir

- In **partitioned database, back up all database partitions.**
- If performed full online or offline database backup recently and cannot perform another one before upgrading, can perform incremental offline database backup instead if **TRACKMOD=ON**.
- If activated and configured DB2 Advanced Copy Services (ACS) on DB2 9.7 or later, use the USE SNAPSHOT parameter to perform snapshot backup.
 - **Can only restore snapshot backup to instance of same version.**
 - **Cannot use snapshot backup to upgrade to new version.**



45

Before and after the upgrade process to DB2 10.5, it is strongly recommended that you perform a full *offline* database backup. If an error occurs during the upgrade process, you need full database backups to recover and upgrade your databases.

After you upgrade your instances to DB2 10.5, you cannot backup databases until you upgrade them.

Backing up DB2 server configuration and diagnostic info

1. Collect information from DB2 servers by running db2support command for all databases going to upgrade or convert to DB2 pureScale, in all your instances. Collect information about database system catalog, DB CFG and DBM CFG settings, DB2 registry variables settings, explain table data, and diagnostic information required by DB2 support in case of problems.

```
db2support output-directory -d database-name -cl 0
```

- Keep zip for several months after complete upgrade or conversion to DB2 pureScale to help resolve performance issues with new release.

2. Back up information about all packages for applications associated with each database.

```
db2 LIST PACKAGES FOR ALL SHOW DETAIL > /upgrade/sample_pckg.txt
```

3. If enabled audit facility, back up audit cfg for each instances:

```
db2audit describe > audit_instance-name.cfg
```

4. Back up all external routines.

```
cp -R $INSTHOME/sqlib/function $INSTHOME/routine_backup
```

Optional Steps: db2support HTML report includes following info.

5.** Use GET DBM CFG to back up settings for each instance:

```
db2 GET DBM CFG > dbm_instname.cfg
```

6.** Use GET DB CFG to back up settings for each database.

If not same settings, back up DB CFG for each database partition.

```
db2 CONNECT TO database_alias
```

```
db2 GET DB CFG FOR database_alias SHOW DETAIL > db_database_alias.cfg
```

7.** Use db2look to save additional information.

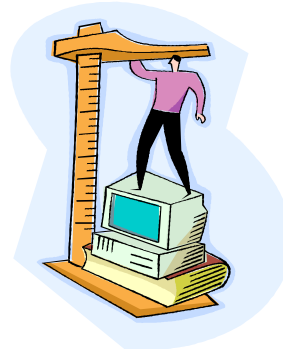
```
db2look -d sample -e -o sample_tbs.db2 -l -x
```

8.** Use db2set command to back up DB2 profile registry variables settings:

```
db2set -all > reg_instname.txt
```

9. If set DB2 environment variables, use appropriate system command to list environment variables values.

```
set | grep DB2 > env_instname.txt (UNIX example)
```



** Optional: Contained in db2support zip file

46

Backing up your settings for database and database manager configuration parameters before DB2 server upgrade, or conversion to DB2 pureScale, allows you to verify DB2 server behavior after upgrade, or converting to DB2 pureScale, and to re-create instances and databases. In addition, you can collect information from your DB2 servers about the database system catalogs, DB2 registry variables settings, explain table data, and diagnostic information that can help in problem determination if you encounter any post-upgrade differences in the database manager behavior or performance.

The db2support -cl 0 parameter collects the database system catalog, database and database manager configuration parameters settings, DB2 registry variables settings. The information collected is stored in the db2support.zip compressed zip file under the output directory. A summary report in HTML format is included. In the db2supp_opt.zip file that is also included, you should check the optimizer.log file to verify that the collection of information was performed successfully. Keep this zip file for several months after you complete the upgrade, or conversion to DB2 pureScale. The information in the zip file can help in quickly resolving any performance issues with the new release.


The LIST PACKAGES ... FOR SCHEMA clause allows you to list all packages for a specific schema, if your application has several schemas you need to repeat this command for each schema name or use the FOR ALL clause.

When backing up external routines, INSTHOME is set to home directory of instance owner. If you have specified a full path that is not under the default routines path when you created your external routines in the database, you must ensure the existing libraries remain on their original location.

The SHOW DETAIL clause on GET DB CFG displays the values calculated by the database manager when configuration parameters are set to AUTOMATIC.

When possible, use the output from the set command and run the db2set command to set these environment variables as registry variables in the DB2 profile registry.

REF: Instance upgrade - db2iupgrade

- Call instance upgrade explicitly with **db2iupgrade** command
- Call implicitly when install DB2 10.5 on Windows and select **Work with Existing option** and choose a pre-10.5 copy with upgrade action
- db2iupgrade does following things:
 - Upgrades existing instance to new instance under DB2 10.5 copy.
 - Upgrades instance profile registry variables.
Global profile registry variables are not upgraded. 
 - Upgrades dbm cfg file.
 - Sets jdk_path dbm cfg parameter.
 - Upgrades db2audit.cfg audit cfg file when audit facility enabled.
 - Uses SSLconfig.ini SSL configuration file to set new dbm cfg parameters to corresponding SSL parameter value in this file and upgrades instance profile registry setting DB2COMM=SSL.
 - For successful instance upgrade, all files must exist for all instances and all files must have write access granted.

47

Only DB2 Enterprise Server Edition instances (instance type *ese*) can be upgraded using the db2iupgrade command. The db2iupgrade command calls the db2ckupgrade command and specifies update.log as the log file for db2ckupgrade. The default log file created for db2iupgrade is /tmp/db2ckupgrade.log.processID. Verify that local databases are ready for upgrade before upgrading the instance. The log file is created in the instance home directory for Linux and UNIX operating systems or in the current directory for Windows operating systems. The instance upgrade will not continue if the db2ckupgrade command returns any errors.

If you use the db2iupgrade command to upgrade a DB2 instance from a previous version to the current version of a Linux/UNIX DB2 database system, the DB2 Global Profile Variables defined in an old DB2 database installation path will not be upgraded over to the new installation location. The DB2 Instance Profile Variables specific to the instance to be upgraded will be carried over after the instance is upgraded. If you are using the su command instead of the login command to become the root user, you must issue the su command with the - option to indicate that the process environment is to be set as if you had logged in to the system using the login command. You must not source the DB2 instance environment for the root user. Running the db2iupgrade command when you sourced the DB2 instance environment is not supported. On AIX 6.1 (or higher), when running this command from a shared DB2 copy in a system workload partition (WPAR) global environment, this command must be run as the root user. WPAR is not supported in a DB2 pureScale environment.

On Windows operating systems, this command is located in the DB2PATH\bin directory, where DB2PATH is the location where the DB2 copy is installed. To move your instance profile from its current location to another location, use the /p option and specify the instance profile path. Otherwise, the instance profile will stay in its original location after the upgrade.

Validating Instance Upgrade

- Review **db2iupgrade** log file written to Windows current path or UNIX/Linux instance home directory
- Switch to the instance owner and issue:

db2start



db2level



- **DB21085I** This instance or install (instance name, where applicable: "inst461") uses "64" bits and DB2 code release "SQL10050" with level identifier "0601010E". Informational tokens are "DB2 v10.5.0.0", "s130528", "LINUXAMD64105", and Fix Pack "0". Product is installed at "/opt/ibm/db2/V10.5".
- On Windows, db2level shows DB2 copy name.
- To display the same information for DB2 pureScale member, use **db2level -localMember**

46

The db2level output shown above was ran on a Linux 64-bit operating system with DB2 10.5 and no fixpacks installed.

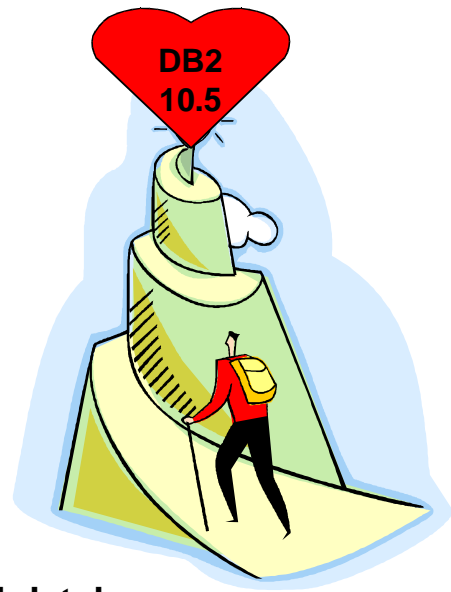
The db2iupgrade log file is created in the instance home directory for Linux and UNIX operating systems or in the current directory for Windows operating systems. The instance upgrade will not continue if the db2ckupgrade command returns any errors.

If the db2iupgrade command completes successfully, switch to the instance owner and issue the **db2start** to restart the instance. To confirm you are now running DB2 10.5 issue the **db2level** command.

The **-localMember** parameter specifies the current version and service level for the member where the db2level command is being issued.

Database directory upgrade

- During first time access, database directory is implicitly upgraded if necessary.
- Database directory is accessed when issue:
 - **LIST DATABASE DIRECTORY**
 - **UPGRADE DATABASE**
- Obtain list of all local databases that must be upgraded:
db2 list db directory > db-dir-before.txt



DBY
DBX

**Must upgrade all databases
in upgraded instance**

50

The database directory is implicitly upgraded if necessary during first time access.

After the instance upgrade has been completed, all databases need to be upgraded to the latest version. Obtain a list of all local databases using the **list database directory** command.

REF: Database upgrade

- Call explicitly using **UPGRADE DATABASE** command
- Following database entities converted during database upgrade:
 - Database configuration file
 - Log file header
 - Global log file header file
 - Table root page for all tables
 - Index root page for all tables
 - Catalog tables
 - Storage group files
 - Buffer pool files
 - Table space files
 - History file
- For recoverable databases, UPGRADE DATABASE renames all log files in active log path with the extension .MIG. After upgrade databases successfully, can delete all S*.MIG files.
- UPGRADE DATABASE upgrades files SQLSPCS.1, SQLSPCS.2 , SQLSGF.1, and SQLSGF.2 to support new functionality on automatic storage table spaces such as removing storage paths from database and rebalancing automatic storage table spaces after add or drop storage paths from database.
- COBOL and FORTRAN language support for db2DatabaseUpgrade API is deprecated. The db2DatabaseUpgrade API still supported in C and Java programming languages.

51

For details about log migration, see [Post-upgrade tasks for DB2 servers](#) discussed later in this topic.

The COBOL and FORTRAN language support for the db2DatabaseUpgrade API is deprecated and might be removed in a future release. The db2DatabaseUpgrade API is still supported in C and Java programming languages. Start using a supported programming language to continue to call this API before it becomes discontinued. Alternatively, use a DB2 command script that issues the UPGRADE DATABASE command.

Smart Idea - Before upgrade, clean up diagpath

- Rename or delete db2diag log files so new files are created.
- Remove or move to another directory any existing dump files, trap files, and alert log files in diagnostic path indicated by diagpath.
- Now files only contain information about database upgrade process.
- Helps to isolate and understand any problem that might occur during database upgrade.



52

Rename or delete the db2diag log files so that new files are created. Also, remove or move to another directory any existing dump files, trap files, and alert log files in the directory indicated by the diagpath parameter. By doing this, the files only contain information about the upgrade process that helps you to isolate and understand any problem that might occur during database upgrade.

Upgrading Local Databases to DB2 10.5

• **db2 UPGRADE DATABASE** *database_alias* -REBINDALL

- Instance owner authority required
- **REBINDALL** rebinds all packages during upgrade and ensures step not missed and that applications not started before completed
 - Alternately, rebind all valid and invalid packages after database upgrade:
 - **db2rbind** *database-name* -l *logfile* all

db2 UPGRADE DATABASE DBY -REBINDALL

db2 UPGRADE DATABASE DBX -REBINDALL



If error occurs during upgrade, may need to issue **TERMINATE**

```
db2 CONNECT to <dbname>
db2 GET DB CFG FOR <dbname> SHOW DETAIL
> <dbname>-dbcfg-after.txt
db2 LIST TABLESPACES SHOW DETAIL
> <dbname>-tsp-after.txt
db2 LIST PACKAGES FOR ALL SHOW DETAIL
> <dbname>-pkg-after.txt
db2 CONNECT RESET
```

...

Compare before and after txt files.

50

The **db2ckupgrade** command must be issued prior to upgrading the instance to verify that your databases are ready for upgrade. The **db2iupgrade** command implicitly calls the **db2ckupgrade**. Backup all databases prior to upgrade, and prior to the installation of the current version of DB2 database product on Windows operating systems.

The **upgrade database** command will only upgrade a database to a newer version, and cannot be used to convert an upgraded database to its previous version. The database must be cataloged before upgrade.

If an error occurs during upgrade, it might be necessary to issue the **TERMINATE** command before attempting the suggested user response. For example, if a log full error occurs during upgrade (SQL1704: Database upgrade failed. Reason code "3"), it will be necessary to issue the **TERMINATE** command before increasing the values of the database configuration parameters **LOGPRIMARY** and **LOGFILSIZ**. The CLP must refresh its database directory cache if the upgrade failure occurs after the database has already been relocated (which is likely to be the case when a "log full" error returns).

After each database has been upgraded repeat the statistics gathering as above and compare the two sets of information. With the exception of the system packages, the two sets of information should be identical.

The **REBINDALL** option specifies that a **REBIND** of all packages is performed during upgrade. Performing the **REBINDs** automatically during the upgrade ensures that the step is not missed and will help ensure that other applications are not started before the **REBINDs** are complete.

Upgrade Database Warning and Error Messages

SQL1704N Database upgrade failed. Reason code "3".

Fix: Increase LOGFILSIZ

SQL1243W Database upgrade warning

Fix: Drop or rename SYSTOOLS.DB2LOOK_INFO table. Otherwise, ALTER TABLE and COPY SCHEMA statements fail to run.

- db2 "SELECT tabname, tabschema, definer FROM syscat.tables WHERE tabschema = 'SYSTOOLS' AND tabname ='DB2LOOK_INFO'"
- If did create: db2 RENAME SYSTOOLS.DB2LOOK_INFO TO new-table-name
- If you did not create: db2 DROP TABLE SYSTOOLS.DB2LOOK_INFO

For other upgrade database warning and error messages:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007193.html>

If the UPGRADE DATABASE command fails and returns the [SQL1704N](#) error message with a reason code that describes the cause of the failure, find this SQL error code and determine the action to take from the list of the possible solutions for each reason code. One of the most common causes of upgrade failure is that the log file space is not large enough.

If the UPGRADE DATABASE command returns the [SQL1243W](#) warning message, you must drop or rename the SYSTOOLS.DB2LOOK_INFO table. Otherwise, the ALTER TABLE and COPY SCHEMA statements will fail to run.

REF: UPGRADE DATABASE fails if following exist:

- Do not have authorization to upgrade the database.
- Cataloged database does not exist.
- Database upgrade encounters any of problems described in reason codes of error message SQL1704N.
- User-defined distinct types (UDTs) encountered with names ARRAY, BINARY, CURSOR, DECFLOAT, ROW, VARBINARY, or XML. Must drop these UDTs and re-create with different names before database upgrade.
- Database objects were created using restricted schema names described in the error message SQL0553N. **List of restricted schema names now includes SYSPUBLIC.**
- Database is enabled as a high availability disaster recovery (HADR) standby database.

55

Changes in existing functionality introduced in DB2 10.5 can potentially impact your applications, scripts, maintenance processes, and any other aspects related your DB2 server upgrade process.

Changes in existing functionality introduced in pre-DB2 10.5 releases can also have an impact.

Upgrading in a test environment allows you to learn about possible issues, evaluate the impact on your environment and find a resolution.

ENV_GET_INSTANCE_CODE_LEVELS

- **db2 GET DBM CFG | grep -i CUR_EFF**
- **SELECT NAME, DEFERRED_VALUE FROM SYSIBMADM.DBMCFG**
...
cur_eff_arch_lvl V:10 R:5 M:0 F:1 I:0 SB:0
cur_eff_code_lvl V:10 R:5 M:0 F:1 I:0 SB:0
 - Informational configuration parameters
- Invoke **ENV_GET_INSTANCE_CODE_LEVELS() function**
 - can use to facilitate rolling fixpack upgrades
 - **SELECT ... FROM TABLE (SYSPROC.ENV_GET_INSTANCE_CODE_LEVELS())**

V:10 R:5 M:0 F:0 I:0 SB:0

56

You can see the output of instance code levels either when doing a "get dbm cfg" for the informational configuration parameters **CUR_EFF_ARCH_LVL** or **CUR_EFF_CODE_LVL**, or by invoking the **ENV_GET_INSTANCE_CODE_LEVELS() function**. The function can be used to facilitate rolling fixpack upgrades.

- **cur_eff_arch_level** - current effective architecture level configuration parameter. This parameter displays the current effective architecture level (CEAL) at what the instance is operating.
- **cur_eff_code_level** is current effective code level configuration parameter. This parameter displays the current effective code level (CECL) at what the instance is operating.

The ENV_GET_INSTANCE_CODE_LEVELS table function returns the code level for the instance, members, and cluster facilitators.

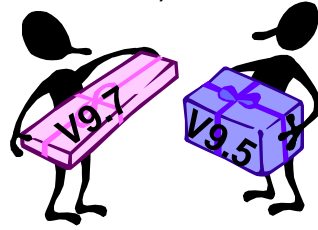
SELECT ... FROM TABLE (SYSPROC.ENV_GET_INSTANCE_CODE_LEVELS())
V:10 R:5 M:0 F:0 I:0 SB:0

See following URL for more details:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.sql.rtn.doc/doc/r0060901.html>

REF: Download DB2 Bind files – sqllib/bnd/*.bnd

- Included in DB2 install image are bind files (sqllib/bnd/*.bnd) containing packages for various database and CLI utilities.
- When remote client executes utilities, it's necessary for package associated with remote client's version and release be bound to database first.
- Often difficult to track down client's image to bind .bnd files.
- Application may execute on DB2 Runtime client of different version than DB2 10.5 server. If client attempts to execute utility (such as CLP, Import, Export, ...) may generate SQL0805N because package associated with version of utility not bound to new DB2 10.5 database.
- Bind packages using bind files from client version.
Download tar of pre-10.5 client bind files from following website:



<http://www-01.ibm.com/support/docview.wss?uid=swg21501302&myns=swgimgmt&mynp=OCSSSNY3&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E>

57

Included in every DB2 install image are bind files (sqllib/bnd/*.bnd) containing the packages for various database and CLI utilities. Many of these bind files correspond to utilities that reside in the Runtime client libraries. When a remote client executes one of these utilities, it is necessary for the package associated with that remote client's version and release to be bound to the database first.

When creating a new database that may accept connections from various remote clients which may run these utilities, it is often difficult to track down each client's image to bind the .bnd files. If you deploy a new database created on DB2 10.5, an application that is executing on a remotely connected client may be running a DB2 Runtime client of a different version. If the application attempts to execute a utility (such as CLP, Import, Export, etc...) it could generate an SQL0805N because the package associated with this version of the utility was not bound to the new DB2 10.5 database. You will have to bind the packages using the bind files from client version. You can download the tar file from the following website and use the bind files from it.

<http://www-01.ibm.com/support/docview.wss?uid=swg21501302&myns=swgimgmt&mynp=OCSSSNY3&mynp=OCSSEPGG&mynp=OCSSEPDU&mync=E>

REF: Upgrade database - runstats on system tables

- UPGRADE DATABASE automatically collects statistics for all system catalog tables during database upgrade.
- If manually modified SYSTATS views, reissue updates.
- Which RUNSTATS is called for automatic statistics collection?

auto_runstats	User profile	RUNSTATS command
Enabled	Exists	RUNSTATS command with SET PROFILE parameter using information in STATISTICS_PROFILE column in SYSCAT.TABLES.
Enabled	Does not exist	RUNSTATS command with default parameters¹
Disabled	N/A	auto_runstats¹

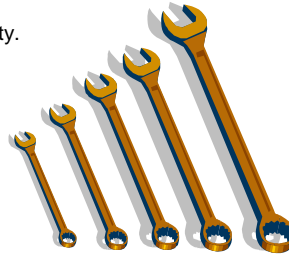
¹ If statistics previously collected for table, RUNSTATS is issued as indicated in table. If there are no statistics collected for table, RUNSTATS is not issued.

The UPGRADE DATABASE command automatically collects statistics for all system catalog tables during database upgrade.

If you have manually modified your system catalog table statistics via updates to SYSTATS views, manually reissue these updates to the SYSTATS views.

REF: Runstats on objects to use new functionality

- **Runstats** on objects so optimizer uses new functionality.
 - DB CFG **AUTORUNSTATS=ON**
 - **db2 RUNSTATS ON TABLE mel.t1 WITH DISTRIBUTION AND DETAILED INDEXES ALL**
 - **or update statistics on multiple objects:**
db2 REORGCHK UPDATE STATISTICS ON TABLE [ALL | SCHEMA <schema-name> | SYSTEM]
- Complete Runstats first before rebind static embedded SQL packages!
- Then rebind all valid and invalid packages in each database.
 - **Could specify REBINDALL on UPGRADE DATABASE but before runstats on objects.**
 - **db2rbind database-name -l logfile all**
 - From IBM Data Studio, can open task assistant to rebind packages.
 - Applies only to C, C++, COBOL, FORTRAN, REXX embedded static SQL applications.
 - Can request access plans for static SQL be preserved across binds or rebinds because don't want existing access plan to change.
ALTER PACKAGE/BIND/REBIND/PRECOMPILE ... APREUSE YES
SYSCAT.PACKAGES.APREUSE = Y when completed.
- Test your applications and tools to ensure that the server is working as expected.
- After rebound all packages, can automatically take advantage of optimizer enhancements. 59



Runstats on objects so optimizer uses new functionality. You can use reorgchk to update statistics on multiple objects.

During database upgrade, all packages for user applications and routines are marked as invalid.

Packages implicitly rebound first time used after upgrade database. To eliminate overhead, explicitly rebind invalid packages. Must explicitly rebind inoperative packages. Alternatively, specify REBINDALL on UPGRADE DATABASE. Applies only to C, C++, COBOL, FORTRAN, and REXX embedded SQL applications.

The REBINDALL option specifies that a REBIND of all packages is performed during upgrade. Performing the REBINDs automatically during the upgrade ensures that the step is not missed and will help ensure that other applications are not started before the REBINDs are complete.

When a new DB2 data server release is installed, existing packages must be rebound in order for access sections to be rebuilt, so that they are compatible with runtime component of new data server release. However, you may not want the existing access plan to change. You can enable access plan reuse through ALTER PACKAGE statement or by using the APREUSE option on BIND, REBIND or PRECOMPILE. Packages that are subject to access plan reuse have the value "y" in APREUSE columns of the SYSCAT.PACKAGES catalog view.

Upgrade explain tables & rerun benchmarks

- To maintain explain table info gathered from previous releases, upgrade explain tables to DB2 10.5.

db2exmig -d dbname -e explain_schema

OR

CALL SYSPROC.SYSINSTALLOBJECTS ('EXPLAIN', 'M', CAST (NULL AS VARCHAR(128)), CAST (NULL AS VARCHAR(128)))

- Renames original explain tables, creates new set of tables by using EXPLAIN.DDL file, and copies contents of original explain tables to new tables. Finally, the tool drops original explain tables and preserves any user-added columns in explain tables.
- After upgrade database, update statistics, rerun benchmarks, compare and examine explain output if necessary.
 - Use **db2batch** benchmark tool to collect elapsed and CPU times
db2batch -d DBX -f input.sql -r output.fil -i complete -o e yes -isol CS
 - Keep **db2exfmt** output for each test query

60

Run a number of performance tests before upgrading your DB2 server. The **db2batch** benchmark tool helps you to collect elapsed and CPU times for running queries. You can use this tool to develop performance tests. Record the exact environment conditions where you run your tests.

Also, keep a record of the **db2exfmt** command output for each test query. Compare the results before and after upgrade. This practice can help to identify and correct any performance degradation that might occur.

The SYSINSTALLOBJECTS procedure can migrate explain tables. The M option is only valid when used with the tool name EXPLAIN. This option migrates explain tables to be compatible with the current version.

The db2batch tool reads SQL statements and XQuery statements from a flat file, dynamically prepares and describes the statements, returns an answer set, populates explain tables, and measure the elapsed time to run each statement where the prepare, execute, and fetch times are reported separately. The default isolation level is RR.

The db2exfmt tool can be used to analyze explain output stored in explain tables.

Upgrade event monitor output tables

- As event monitors enhanced, tables require new columns.
- **EVMON_UPGRADE_TABLES** stored procedure
 - upgrades definitions of existing tables along with all data
 - automatically deactivate event monitor before upgrade
 - automatically reactivates after upgrade completed

call **EVMON_UPGRADE_TABLES** (*evmon_name*,
evmon_type, *options*, *num_evmons_evaluated*,
num_evmons_to_upgrade, *num_evmons_upgraded*)

or

call **evmon_upgrade_tables** (null, null, null, ?, ?, ?)

- Can also use ALTER EVENT MONITOR statement to add new logical groups to event monitor as alternative

61

Beginning in DB2 10.5, you can upgrade event monitor output tables after you upgrade the DB2 product. This capability lets you retain any data that might exist in event monitor tables that you had before you upgraded.

As event monitors are enhanced in the DB2 product, the tables they produce might change. For example, new columns might be added to a table for reporting new monitor elements. Before 10.5, if you had existing event monitors that wrote to tables that contained data that you wanted to retain, and you wanted to collect the data in the newly-added columns, you were required to manually alter them after upgrading to the new release. This alteration involved adding any of the new columns that you might want to use. If you did not add the new columns, the event monitor would work as it had in the previous release, capturing only the data supported by that the event monitor in that release. Unformatted event tables that had changed could not be upgraded at all; you were required to drop them and then re-create them.

The EVMON_UPGRADE_TABLES stored procedure upgrades the definitions of existing event monitor tables to match those produced by the current level of the DB2 product. This feature lets you keep any existing tables that you might have, along with all the data they contain, eliminating the need to manually alter, or to drop and re-create tables.

The EVMON_UPGRADE_TABLES procedure works with both regular and UE tables. For regular tables, the procedure adds any new columns needed, drops old columns that are no longer required, and alters any columns as needed. For UE tables, the procedure adds new columns and modifies existing columns as needed to allow the UE table to be processed by the db2evmonfmt tool, or the EVMON_FORMAT_UE_TO_TABLES or EVMON_FORMAT_UE_TO_XML routines.

Important: Any active event monitors must be deactivated for the upgrade process to work properly. The EVMON_UPGRADE_TABLES procedure automatically deactivates any active event monitors before it begins upgrading tables. Do not reactivate any event monitors with tables being processed by EVMON_UPGRADE_TABLES, or the upgrade process will fail. Any event monitors that were active before the upgrade are activated again after the upgrade has completed.

Implications of not upgrading event monitor tables

As in past releases, you can choose to not upgrade your event monitor tables. However, any new columns that have been added to the event monitor in the new release will not be populated with data, and will not be available for queries. Also, the values for any monitor elements that previously existed in the old release and that increased in size in the new release might be truncated. For example, if a monitor element increases in size from VARCHAR(20) to VARCHAR(128) in the new release, and you do not upgrade the previously-existing tables, the column that contains the monitor element values will continue to store only 20 characters of data, even though the system may be sending 128-bytes of data for that monitor element to the event monitor.

Student Notes: For Reference Only

Upgrading tables produced by EVMON_FORMAT_UE_TO_TABLES

When used with UE tables, the EVMON_UPGRADE_TABLES procedure upgrades the UE table itself; it has no effect on any regular tables that you might have created using the EVMON_FORMAT_UE_TO_TABLES procedure. After you use EVMON_UPGRADE_TABLES to upgrade a UE table, you can also upgrade the output tables produced by EVMON_FORMAT_UE_TO_TABLES. As of DB2 Version 10.1 EVMON_FORMAT_UE_TO_TABLES procedure supports a new option: UPGRADE_TABLES. When you run the EVMON_FORMAT_UE_TO_TABLES procedure with this option, any existing tables produced by the procedure are altered so that the table columns match the output produced by the new version of EVMON_FORMAT_UE_TO_TABLES procedure.

REF: Preparation Steps: Reversing DB2 server upgrade

- Ensure have SYSADM, as well as root on Linux/UNIX or Local Administrator authority on Windows.
- Perform following steps before upgrading your DB2 server:
 - Review disk space requirements.
 - Take offline full backup of all databases that upgrading.
 - Back up all dbm cfg values for each instance and all db cfg values for each database.
- Keep existing pre-DB2 10.5 copy during upgrade of your DB2 server. To do this:
 - Select Install New option to create new copy when installing DB2 10.5.
 - Do not select Work with existing option
 - Then choose a pre-DB2 10.5 copy with upgrade action that is available on Windows.
- Keep all S*.MIG files in active log path in case want to rollforward through these log files after reversing upgrade. For recoverable databases, UPGRADE DATABASE renames log files in active log path with extension .MIG.
- In partitioned databases, must perform this procedure on all database partitions.



Reversing DB2 server upgrade involves creating a plan using the steps in this procedure to fall back to the DB2 release from which you upgraded your DB2 server. There is no utility to fall back to a previous release of DB2 database after upgrading your DB2 server.

Performing an upgrade in a test environment will help you identify any issues with the process and avoid having to reverse the upgrade.

REF: Reversing DB2 server upgrade

1. Log on to DB2 server as user with SYSADM authority.
2. Drop all databases in DB2 10.5 by running DROP DATABASE.
3. Log on to DB2 server as root on Linux/UNIX or user with Local Administrator authority on Windows.
4. Drop DB2 10.5 instances by running db2idrop. Does not remove database files. Need to drop databases before dropping instances.
5. If upgraded pre-DB2 10.5 instances to DB2 10.5, re-create instances in pre-DB2 10.5 by running db2icrt. Then restore DBM CFG values for each instance using UPDATE DBM CFG.
6. For each pre-DB2 10.5 instance, log on to DB2 server as instance owner and restore upgraded databases from pre-DB2 10.5 offline full backup by running RESTORE DATABASE. Cannot upgrade databases from DB2 10.5 to pre-DB2 10.5 release.
7. If recreated instances using same instance owner had before upgrade and did not upgrade database to DB2 10.5 instance, database is still pre-DB2 10.5 release and can access it by just re-cataloging it.
8. If have recoverable databases and want to rollforward through log files had before upgrade, rename all S*.MIG files in active log path using .LOG extension and issue ROLLFORWARD DATABASE.

To reverse a DB2 server upgrade, you need to perform the above steps.

Migrating Data to IBM DB2 with BLU Acceleration (1)

- **db2set WORKLOAD=ANALYTICS** automatically sets config parms when create new database unless disable Configuration Advisor.
- **Tip: Running AUTOCONFIGURE against existing database when DB2_WORKLOAD=ANALYTICS has the same result.**
 - `dft_table_org` (default table organization for user tables) db cfg parm set to COLUMN.
 - `dft_degree` (default degree) db cfg parm set to ANY.
 - `dft_extent_sz` (default extent size) db cfg parm set to 4.
 - `catalogcache_sz` (catalog cache) db cfg parm set to value > value for non-analytics workload.
 - `sortheap` (sort heap) and `sheapthres_shr` (sort heap threshold for shared sorts) db cfg parm are calculated specifically for analytics workload. Settings take into account additional memory requirements for processing column-organized data.
 - `util_heap_sz` (utility heap size) db cfg parm set to value that takes into account additional memory required to load data into column-organized tables.
 - `auto_reorg` (automatic reorganization) db cfg parm set to ON.
- The following extra choices are made automatically:
 - Default database page size for newly created database is set to 32 KB.
 - Intraquery parallelism enabled for any workload (including `SYSDEFAULTUSERWORKLOAD`) that inherits intrapartition parallelism setting from `intra_parallel` (enable intrapartition parallelism) dbm cfg parm, even if `intra_parallel` set to NO.
 - Concurrency control enabled on `SYSDEFAULTMANAGEDSUBCLASS`.
- Automatic table maintenance performs space reclamation for column-organized tables by default. For other considerations see
http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.admin.dboj.doc/doc/c0060592.html?lang=en
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1309db2bluaccel/index.html?ce=ism0070&ct=is&cmp=ibmsocial&cm=h&cr=crossbrand&ccy=us#Scenario2>

65

`DB2_WORKLOAD` registry variable is available to be set on all operating systems. The default is that it is not set. Each value for `DB2_WORKLOAD` represents a specific grouping of several registry variables with predefined settings. Only one value for `DB2_WORKLOAD` can be in effect at any given time. Use `ANALYTICS` setting before creating the database to establish an optimal default configuration for analytic workloads. The `ANALYTICS` option ensures that configuration parameters are automatically set as follows unless you disable the Configuration Advisor:

- The `dft_table_org` (default table organization for user tables) database configuration parameter is set to COLUMN.
- The `dft_degree` (default degree) database configuration parameter is set to ANY.
- The `dft_extent_sz` (default extent size) database configuration parameter is set to 4.
- The `catalogcache_sz` (catalog cache) database configuration parameter is set to a value that is higher than the value for a non-analytics workload.
- The values of the `sortheap` (sort heap) and `sheapthres_shr` (sort heap threshold for shared sorts) database configuration parameters are calculated specifically for an analytics workload. These settings take into account the additional memory requirements for processing column-organized data.
- The `util_heap_sz` (utility heap size) database configuration parameter is set to a value that takes into account the additional memory that is required to load the data into column-organized tables.
- The `auto_reorg` (automatic reorganization) database configuration parameter is set to ON.

Tip: Running the `AUTOCONFIGURE` command against an existing database when `DB2_WORKLOAD` is set to `ANALYTICS` has the same result.

The following extra choices are made automatically:

- The default database page size for a newly created database is set to 32 KB.
- A larger database shared sort heap is allocated.
- Intraquery parallelism is enabled for any workload (including `SYSDEFAULTUSERWORKLOAD`) that inherits the intrapartition parallelism setting from the `intra_parallel` (enable intrapartition parallelism) database manager configuration parameter, even if `intra_parallel` is set to NO.
- Concurrency control is enabled on `SYSDEFAULTMANAGEDSUBCLASS`.
- Automatic table maintenance performs space reclamation for column-organized tables by default.

Migrating Data to IBM DB2 with BLU Acceleration (2)

- **BACKUP YOUR DATABASE!!!**
- Set the workload to ANALYTICS and autoconfigure
 - **db2set DB2_WORKLOAD=ANALYTICS**
 - **db2stop force**
 - **db2start**
 - **db2 CONNECT TO <dbalias>**
 - **db2 AUTOCONFIGURE APPLY DB AND DBM**
or display changes and apply yourself with:
db2 AUTOCONFIGURE APPLY NONE
- If not already using create Automatic Storage:
 - **db2 CREATE STOGROUP <storage group name> ON <path(s)> SET AS DEFAULT**
- Convert DMS data and index table spaces to AUTOMATIC STORAGE.
 - **db2 ALTER TABLESPACE <tablespace name> MANAGED BY AUTOMATIC STORAGE**
 - **db2 ALTER TABLESPACE <tablespace name> REBALANCE**
 - **db2 CREATE TABLESPACE <tablespacename> PAGESIZE 32K EXTENTSIZE 4**
- Convert tables from row to column organization:
 - **db2convert -d mydatabase -z user -t table1 -ts coltbsp**
or convert all tables
db2convert -d mydatabase

66

The above steps can be used to perform the data migration task to BLU acceleration.

The db2convert utility converts one or all row-organized user tables in a specified database into column-organized tables. The row-organized tables remain online during command processing. For monitoring purposes, the command displays statistics about the conversion. You must have SQLADM or DBADM authority to invoke the ADMIN_MOVE_TABLE stored procedure, on which the db2convert command depends. You must also have the appropriate object creation authorities, including the authority to issue the SELECT statement on the source table.

Convert the tables from row organization to column organization using the db2convert utility:

db2convert -d mydatabase -z user -t table1

or to convert all tables

db2convert -d mydatabase

Also if creating a new database, see *5 Steps for Migrating Data to IBM DB2 with BLU Acceleration* by Mohankumar Saraswatipura at: http://ibmdatamag.com/2013/06/5-steps-for-migrating-data-to-ibm-db2-with-blu-acceleration/?utm_content=buffer430d1&utm_source=buffer&utm_medium=facebook&utm_campaign=Buffer

After setting DB2_WORKLOAD=ANALYTICS, run the AUTOCONFIGURE command to get most of the settings configured. During table space creation, explicitly specify 32K page size and 4 extent size since the database level settings are likely different. To avoid having to specify the extent size on every table space creation, you could update DFT_EXTENT_SZ to 4 instead.

A reasonable starting point is to set SHEAPTHRES_SHR to the size of the buffer pool and SORTHEAP to 1/20th of SHEAPTHRES_SHR.

Migrating Data to IBM DB2 with BLU Acceleration (3)

- **BEST PRACTICE: Perform backup before convert since no reverse convert option**

- **Cannot convert following into column-organized tables:**
 - Range partitioned tables
 - Partition Database environments
 - Multidimensional clustering tables
 - Range clustered tables
 - Insert time clustering tables
 - Typed tables
 - Materialized query tables
 - Declared global temporary tables
 - Created global temporary tables
 - Tables in non-automatic storage table spaces
 - Tables that have generated columns
 - Tables with columns of type BLOB, DBCLOB, CLOB or XML
 - Drop any dependent objects that cannot be transferred to COT
 - Triggers
 - Secondary indexes
 - Foreign keys and Check constraints table attributes used as NOT ENFORCED during conversion

67

You cannot convert the following table types into column-organized tables:

- Range partitioned tables
- Multidimensional clustering tables
- Range clustered tables
- Insert time clustering tables
- Typed tables
- Materialized query tables
- Declared global temporary tables
- Created global temporary tables

Tables in partitioned database environments, tables in nonautomatic storage table spaces, tables that have generated columns, and tables with columns of type BLOB, DBCLOB, CLOB, or XML cannot be converted into column-organized tables.

The following table attributes are not used during conversion to column-organized tables:

- Triggers
- Secondary indexes

If they are not required, drop any dependent objects that cannot be transferred to column-organized tables before invoking the db2convert command.

The following table attributes are used as NOT ENFORCED during conversion to column-organized tables:

- Foreign keys
- Check constraints

The table conversion process temporarily requires space for both the source and the target tables.

Because there is no online process to convert column-organized tables back to row-organized tables, the best practice is to perform a backup before you convert the tables to column organization.

Workload Table Organization Advisor (OQWT)

- OQWT can identify tables that are candidates for being column-organized tables.
 - Bases advice on table statistics and query workload characteristics.
 - Indicates expected performance improvement for queries in workload.
 - Can decide if want to execute script generated by tool to implement recommendations.
 - Make sure to back up database prior to making these changes.
- Should be able to measure compression savings and start running queries without any change to SQL.
 - No need for indexes or MQTs to be created.
 - Only indexes for column-organized tables are ones used for enforced primary keys or unique constraints.
 - If analytic data is already cleansed during ETL processing, create non-enforced constraints to allow DB2 Optimizer to leverage this information.

68

For additional information about OQWT see:

<http://pic.dhe.ibm.com/infocenter/dstudio/v4r1/index.jsp?topic=%2Fcom.ibm.datatools.crytune.workloadtunedb2luw.doc%2Ftopics%2Fgenrecswtoa.html>

IBM InfoSphere Optim Query Workload Tuner for DB2 for LUW estimates benefits for column-organized tables

Advisor identifies candidate tables for conversion to columnar format.

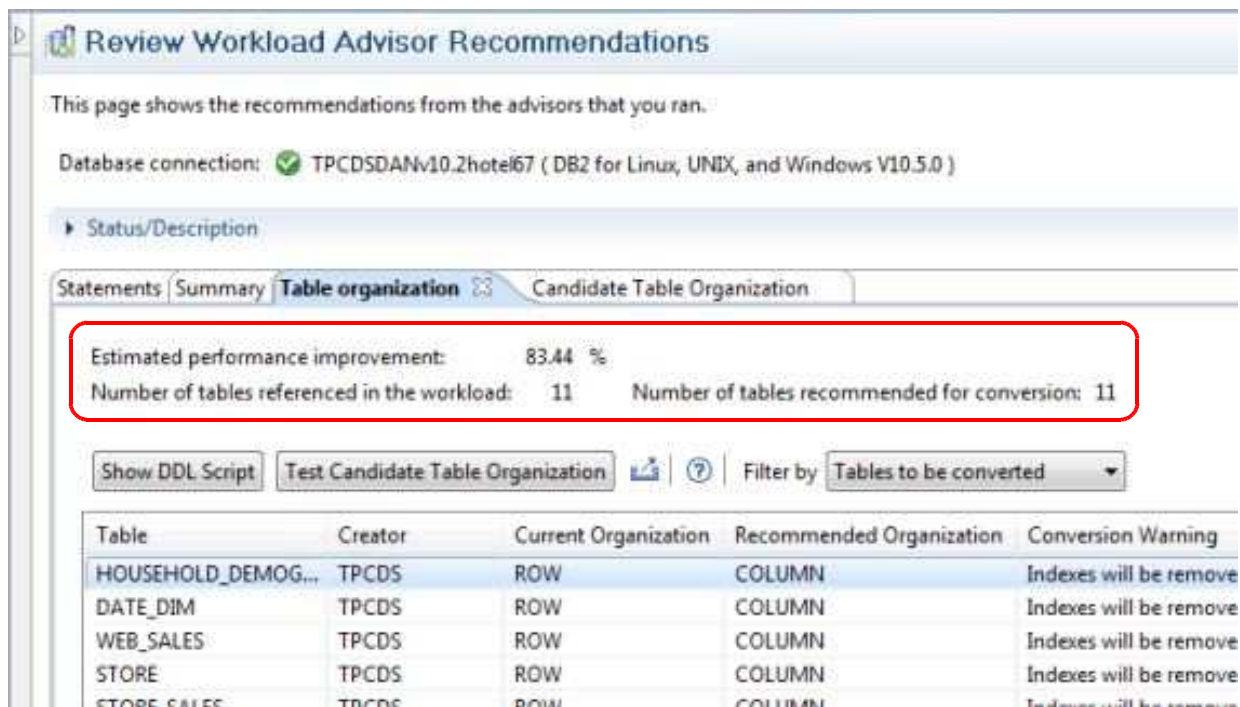
Analyzes SQL workload and estimates execution cost on row- and column-organized tables.

Table	Creator	Current Org	Recommended Organization	Conversion Warning	References to Table	Cumulative Total Cost
HOUSEHOLD_DEMOG...	TPCDS	ROW	COLUMN	Indexes will be remove...	19	14,261,145,494.00
DATE_DIM	TPCDS	ROW	COLUMN	Indexes will be remove...	3	1,022,691,606.00
WEB_SALES	TPCDS	ROW	COLUMN	Indexes will be remove...	1	973,711,296.00
STORE	TPCDS	ROW	COLUMN	Indexes will be remove...	25	11,926,096,848.00
STORE_SALES	TPCDS	ROW	COLUMN	Indexes will be remove...	4	2,635,452,752.00

Execution Count	Weight	Estimated Performance Gain(%)	Cost Before	Cost After	Statement Text
1	0.00	-202.20	22,203,01...	67,097.3...	SELECT Q5."SS_ADDR_SK" AS Q5C6, Q5."SS_CDEMO_SK" AS Q5C4, Q5."SS_CUSTOMER_SK" AS Q5C3, Q5."SS...
1	0.00	78.99	438,398,0...	92,125.0...	select iss.i_brand_id as brand_id , iss.i_class_id class_id , iss.i_category_id category_id from tpcds.store_sal...
1	0.00	81.84	824,558,8...	149,764,...	select avg(ss_quantity), avg(ss_ext_sales_price), avg(ss_ext_wholesale_cost), sum(ss_ext_wholesale_cost) from ...
1	0.00	77.40	2,085,428,...	471,247,...	with cross_items as (select i_item_sk ss_item_sk from tpcds.item, (select iss.i_brand_id brand_id , iss.i_clas...
1	0.00	48.12	175,929,7...	91,275.6...	select cd_gender, cd_marital_status, cd_education_status, count(*) , cd_purchase_estimate, count(*) , c...

IBM® InfoSphere® Optim™ Query Workload Tuner Version 4.1 includes the Workload Table Organization Advisor, which examines all of the tables that are referenced by the statements in a query workload. Its recommendations lead to the best estimated performance improvement for the query workload as a whole. The advisor presents its analysis and rationales so that you can see the tables that are recommended for conversion from row to column organization.

Sample recommendations for workload using Optim Query Workload Tuner



This page shows the recommendations from the advisors that you ran.

Database connection: ✔ TPCDSDANv10.2hotel67 (DB2 for Linux, UNIX, and Windows V10.5.0)

► Status/Description

Statements | Summary | **Table organization** | Candidate Table Organization

Estimated performance improvement: 83.44 %
Number of tables referenced in the workload: 11 Number of tables recommended for conversion: 11

Show DDL Script Test Candidate Table Organization Filter by: Tables to be converted

Table	Creator	Current Organization	Recommended Organization	Conversion Warning
HOUSEHOLD_DEMOG...	TPCDS	ROW	COLUMN	Indexes will be remove
DATE_DIM	TPCDS	ROW	COLUMN	Indexes will be remove
WEB_SALES	TPCDS	ROW	COLUMN	Indexes will be remove
STORE	TPCDS	ROW	COLUMN	Indexes will be remove
STORE SALES	TPCDS	ROW	COLUMN	Indexes will be remove

70

For additional information about OQWT see:

<http://pic.dhe.ibm.com/infocenter/dstudio/v4r1/index.jsp?topic=%2Fcom.ibm.datatools.qrytune.workloadtunedb2luw.doc%2Ftopics%2Fgenrecswtoa.html>

Database Configuration Parameter Changes

- **SOFTMAX** db cfg is depreciated. Replaced with **page_age_trgt_mcr** and **page_age_trgt_gcr** which control saving changed buffer pool pages.
 - **page_age_trgt_mcr** - Page age target member crash recovery
 - Specifies target duration (in seconds) for changed pages kept in local buffer pool
 - Default 120 for pureScale, 240 for ESE and partitioned databases
 - **page_age_trgt_gcr** - Page age target group crash recovery
 - Specifies target duration (in seconds) for changed pages kept in group buffer pool before pages are persisted to disk
 - PureScale only, default is 240
 - `page_age_trgt_gcr` parameter must be \geq `page_age_trgt_mcr`
 - Existing upgraded databases continue to use softmax parameter.
 - Switch from using softmax to these new parameters by setting softmax to 0.
 - New DB2 10.5 databases created with value of softmax set to 0 by default.
- **buffpage** db cfg is depreciated. Use SIZE parameter on CREATE/ALTER BUFFERPOOL for explicit value or AUTOMATIC with self tuning memory.
- **auto_stats_prof** and **auto_prof_upd** db cfgs are discontinued because automatic statistics profiling is discontinued.

71

The `page_age_trgt_mcr` parameter is used when the softmax parameter is configured to a value of 0. Migrated databases retain the previous value for softmax and ignore the `page_age_trgt_mcr` parameter if this value is not 0. The value of softmax is set to 0 in new databases. Increasing the value of this configuration parameter keeps changed pages in memory for a longer time, allowing more page updates to be buffered before the pages are persisted to disk. This behavior can help to improve performance but also increases recovery time. The `page_age_trgt_gcr` parameter must be greater than or equal to `page_age_trgt_mcr` database configuration parameter. The `page_age_trgt_gcr` parameter is used when the softmax parameter is configured to a value of 0. Migrated databases retain the previous value for softmax and ignore the `page_age_trgt_gcr` parameter if this value is not 0. The value of softmax is set to 0 in new databases. Increasing the value of this configuration parameter keeps changed pages in memory for a longer time, allowing more page updates to be buffered before the pages are persisted to disk. This behavior can help to improve performance but also increases recovery time.

The **buffpage** is depreciated. This parameter indicated the default buffer pool size for the ALTER BUFFERPOOL or CREATE BUFFERPOOL statements that indicated the value -1 as the size. Use the SIZE parameter for both statements to indicate an explicit value or use the AUTOMATIC value to enable self tuning of the buffer pool.

The `auto_stats_prof` is discontinued because automatic statistics profiling is discontinued. The `auto_prof_upd` is discontinued because automatic statistics profiling is discontinued. Automatic statistics profiling is discontinued because of its noticeable performance overhead and restrictions.

Student Notes: For Reference Only

It is not supported in partitioned database environments, federated systems, or databases with intrapartition parallelism enabled or enables collection of section actuals. The ASP value as a tool name parameter for the SYSINSTALLOBJECTS procedure is also discontinued. The following tables are also discontinued as they are related to the automatic statistics profiler functionality:

- SYSTOOLS.OPT_FEEDBACK_QUERY
- SYSTOOLS.OPT_FEEDBACK_PREDICATE
- SYSTOOLS.OPT_FEEDBACK_PREDICATE_COLUMN
- SYSTOOLS.OPT_FEEDBACK_RANKING
- SYSTOOLS.OPT_FEEDBACK_RANKING_COLUMN

In earlier releases, you could use automatic statistics profiling to determine recommended parameters for the RUNSTATS command. Also, automatic statistics profiling could detect whether table statistics were outdated. Statistics profiles for the RUNSTATS command are not affected by this discontinuation and are still fully supported. You can use the IBM Data Studio tool to help you determine the best parameters for the RUNSTATS command to continue collecting statistics by using statistics profiles or by running the RUNSTATS command.

DB & DBM CFG Parameter Changes

- **opt_direct_wrkld** - Enables or disables explicit hierarchical locking (EHL).
 - Affects database across entire DB2 pureScale instance. DB CFG.
- **string_units** - Specifies default string units used when defining character data types and graphic data types. DB CFG.
- **db_mem_thresh** - In 10.5, default value for Database Memory Threshold changed from 10 to 100 for new (not upgraded) databases. DB CFG.
 - Max percentage of committed but current unused database shared memory allow before starting to release committed pages of memory back to operating system
 - 100 means never release unused database shared memory-avoid memory thrashing
 - 0 release immediately
- **sortheap** - In 10.5, default value for Sort Heap Size changed from 4 194 303 to 4 294 967 295 for 64-bit platforms. DB CFG.
- **cf_transport_method** - Network transport method. In DB2 pureScale controls what method used for communication between DB2 members and cluster caching facility (CF). DBM CFG.

73

opt_direct_wrkld - Optimize directed workload enables or disables explicit hierarchical locking (EHL). It affects the database across the entire DB2 pureScale instance.

string_units - Default string units specifies the default string units that are used when you are defining character data types and graphic data types.

db_mem_thresh - In 10.5, the default value for Database Memory Threshold is changed from 10 to 100 for new (not upgraded) databases.

sortheap - In 10.5, the default value for Sort Heap Size is changed from 4 194 303 to 4 294 967 295 for 64-bit platforms.

The **cf_transport_method** is network transport method DBM configuration parameter. In DB2 pureScale environments, controls what method is used to communication between DB2 members and the cluster caching facility (CF).

REF: Registry Variable Changes

- **DB2DSDRIVER_CFG_PATH** - Use to specify multiple configuration files with different names at same or different locations. If do not specify file name in path and name pair, defaults to value of db2dsdriver.cfg. If do not specify path name, defaults to directory structure for operating system.
- **DB2_COLLECT_TS_REC_INFO** - Starting in Fix Pack 1, default setting is OFF for HADR databases. Table space recovery is not supported with HADR, so no need for overhead of keeping track of which table spaces changed in each log file.
- **DB2_NCHAR_SUPPORT** - Enables use of user-defined types called NCHAR, NVARCHAR or NCLOB. Data types no longer required because all character length sensitive functions now return number of characters.
 - Start using nchar_codeset DB CFG parm to specify data type mapping for national character string data types in Unicode databases before discontinued.
- **DB2_EVMON_EVENT_LIST_SIZE** - Setting has no effect because of changes made to underlying event monitor infrastructure in DB2 9.7.
- **DB2_INDEX_FREE** is discontinued. Use CREATE INDEX ...PCTFREE and LEVEL2 PCTFREE options.
 - Setting results in error because associated functionality no longer available.
- **DB2_TRUSTED_BINDIN** is discontinued. No longer supported for improving performance of query statements containing host variables within embedded unfenced stored procedure.

74

There are a number of registry variables that are deprecated in DB2 10.5. Deprecated variables are still available, but you should not use them because they will likely be removed in a future version. Start using the replacement feature or function related to the registry variable that is deprecated. Remove the use of registry variables that are deprecated because the functionality associated with the configuration parameter is obsolete or has been replaced by new functionality

DB2DSDRIVER_CFG_PATH - You can now use this variable to specify multiple configuration files with different names at the same or different locations. If you do not specify a file name in a path and name pair, the file name defaults to a value of db2dsdriver.cfg. If you do not specify a path name, it defaults to the directory structure for your operating system.

DB2_COLLECT_TS_REC_INFO - Starting in Fix Pack 1, the default setting for this registry variable is now OFF for high availability disaster recovery (HADR) databases. Table space recovery is not supported with HADR, so there is no need for the overhead of keeping track of which table spaces changed in each log file.

DB2_INDEX_FREE is discontinued. It provided the same functionality as the PCTFREE clause in a CREATE INDEX statement. The difference is that **DB2_INDEX_FREE** applies to all indexes that you create and the PCTFREE clause applies to one index. For more details, see CREATE INDEX statement. Setting this registry variable results in an error because the associated functionality is no longer available.

DB2_TRUSTED_BINDIN - This variable is discontinued because it is no longer supported for improving the performance of query statements containing host variables within an embedded unfenced stored procedure.

REF: New Global Variable - NLS_STRING_UNITS

- Built-in global variable specifies default string units used when defining character and graphic data types in Unicode database.
 - It is read/write variable, with values maintained by user.
 - Type is VARCHAR(11 OCTETS).
 - Schema is SYSIBM.
 - Scope of this global variable is session.
 - Has default value of NULL.
- Must have value of NULL, 'SYSTEM', or 'CODEUNITS32' (SQLSTATE 42815).
 - NULL - STRING_UNITS db cfg parm used to determine default string units.
 - SYSTEM - CHAR, VARCHAR, and CLOB data types defined without specifying CODEUNITS32 keyword default to OCTETS. GRAPHIC, VARGRAPHIC, and DBCLOB data types defined without specifying tCODEUNITS32 keyword default to CODEUNITS16.
 - CODEUNITS32 - CHAR, VARCHAR, and CLOB data types defined without specifying OCTETS keyword default to CODEUNITS32. GRAPHIC, VARGRAPHIC, and DBCLOB data types defined without specifying CODEUNITS16 keyword default to CODEUNITS32.

75

This built-in global variable specifies the default string units that are used when defining character and graphic data types in a Unicode database.

This global variable has the following characteristics:

1. It is a read/write variable, with values maintained by the user.
2. The type is VARCHAR(11 OCTETS).
3. The schema is SYSIBM.
4. The scope of this global variable is session.
5. It has a default value of NULL.

This global variable is only applicable to a Unicode database and must have a value of NULL, 'SYSTEM', or 'CODEUNITS32' (SQLSTATE 42815).

- NULL - The STRING_UNITS database configuration parameter is used to determine the default string units.
- SYSTEM - CHAR, VARCHAR, and CLOB data types defined without specifying the CODEUNITS32 keyword will default to OCTETS. GRAPHIC, VARGRAPHIC, and DBCLOB data types defined without specifying the CODEUNITS32 keyword will default to CODEUNITS16.
- CODEUNITS32 - CHAR, VARCHAR, and CLOB data types defined without specifying the OCTETS keyword will default to CODEUNITS32. GRAPHIC, VARGRAPHIC, and DBCLOB data types defined without specifying the CODEUNITS16 keyword will default to CODEUNITS32.

DB2 command changes for DB2 10.5 (1)

DB2 Command	Command Option
db2pd	<ul style="list-style-type: none">- apinfo displays more info about current/past UOW activities- edus displays agents processing column-organized data- locks showlocks displays table name (TableNm) and schema name (SchemaNm) of locks being held- tablespace displays if reclaimable space enabled (RSE)- transactions displays total number of application commits and rollbacks- ruStatus displays status of online fix pack update in DB2 pureScale instance- wlocks ub option displays TableNm, SchemaNm, and AppNode columns for application locks being waited on- extentmovement displays extent movement status of database- membersubsetstatus displays state of member subsets- subsetid identifies subsets

76

The changes in DB2 10.5 to CLP and system commands can impact your existing applications and scripts after you upgrade to DB2 10.5. Modify any existing scripts and applications to discontinue using deprecated or discontinued options and parameters.

db2pd

- The **-apinfo** parameter now displays more information about current and past activities of the unit of work (UOW).
- The **-edus** parameter now also displays the agents that are processing column-organized data.
- The **showlocks** parameter now displays the TableNm and SchemaNm columns, which indicates the table name and schema name of locks that are being held by applications. To display this information, you must use the showlocks parameter with the **-locks** parameter.
- The **-tablespace** parameter now displays the RSE column to indicate whether the reclaimable space feature is enabled.
- The **-transactions** parameter now displays the total number of application commits and the total number of application rollbacks.
- The new **-ruStatus** parameter displays the status of an online fix pack update in a DB2 pureScale instance.
- The new details **uboption** for the **-wlocks** parameter displays the TableNm, SchemaNm, and AppNode columns for application locks that are being waited on.
- The new **-extentmovement** parameter displays the extent movement status of a database.
- The new **-membersubsetstatus** parameter displays the state of member subsets.
- The new **-subsetid** parameter identifies subsets.

REF: DB2 command changes for DB2 10.5 (2)

DB2 command	Changes
db2exfmt db2expln	<ul style="list-style-type: none"> • Support Random indexes and Column-organized tables. • dynexpln is discontinued. Use db2expln command to process dynamic or static statements directly.
db2IdentifyType1	Command is discontinued. Type 1 indexes discontinued in DB2 9.7.
db2cat	Displays information for random ordering of index keys
db2iupdt	<ul style="list-style-type: none"> • New -commit_level parameter completes update of DB2 pureScale instance to a higher code level. • New -check_commit parameter determines whether update of DB2 pureScale instance to higher code level can be completed successfully.
db2level	Now displays current version and service level for DB2 pureScale instance. To display same information for member, use -localMember .
db2look	Generates DDL statements to create column-organized tables

74

The changes to commands include new parameters, modifications to existing parameters, deprecated or discontinued parameters, and modifications to command output. The above table lists the changes that impact applications and scripts.

- **db2cat** The **db2cat** output now displays information for random ordering of index keys. Since random ordered indexes are extended indexes, Extend Index Info is also displayed for any output that is returned by the command.
- **db2exfmt** The **db2exfmt** output now displays information for random ordering of index keys. R indicates a column with a RANDOM order. A indicates a column with an ASCENDING order. D indicates a column with a DESCENDING order. The **db2exfmt** command now also generates output for queries that access column-organized tables.
- **db2expln** The **db2expln** output now displays information for random ordering of index keys. Random indicates a column with a RANDOM order. The **db2expln** command now also generates output for queries that access column-organized tables.
- **dynexpln** The **dynexpln** command is discontinued. Use the **db2expln** command instead of the **dynexpln** command. The **db2expln** command provides equivalent functionality because it can process dynamic statements directly.
- **db2IdentifyType1** The **db2IdentifyType1** command is discontinued. The **db2IdentifyType1** command was provided to help you convert type-1 indexes to type-2 before upgrading to Version 9.7 because type-1 indexes were discontinued in Version 9.7. In Version 10.5, this command is no longer required because you can only upgrade from Version 9.7 or later releases. Databases in these releases no longer have type-1 indexes.
- **db2iupdt** The new **-commit_level** parameter completes update of DB2 pureScale instance to a higher code level. The new **-check_commit** parameter determines whether update of DB2 pureScale instance to higher code level can be completed successfully.
- **db2look** This command now generates DDL statements to create column-organized tables in addition to row-organized tables.
- **db2level** This command now displays the current version and service level for a IBM DB2 pureScale instance. To display the same information for a member, use the **db2level** command with the **-localMember** parameter. In earlier releases, the **db2level** command displayed the current version and service level for the member where the command was issued

DB2 command changes for DB2 10.5 (3)

DB2 Command	Changes
db2support	<ul style="list-style-type: none">-d now supports collection of information from multiple databases. Separate database names with comma.-alldatabases specifies that information about all databases in database directory is collected.-system_group collects system-related information about specified system group.-system_user collects system-related information about specified system user.-wlm collects information related to WLM issues as part of optimizer mode with collection level 0 and above.

78

db2support

The **-d** parameter now supports collection of information from multiple databases. To specify multiple databases, separate the database names with a comma.

The new **-alldatabases** parameter specifies that information about all databases in the database directory is collected.

The new **-system_group** parameter collects system-related information about the specified system group.

The new **-system_user** parameter collects system-related information about the specified system user.

The new **-wlm** parameter collects information that is related to WLM issues as part of the optimizer mode with collection level 0 and above

DB2 command changes for DB2 10.5 (4)

DB2 Command	Changes
installFixPack	<ul style="list-style-type: none">-check_commit determines whether update of DB2 pureScale instance to higher code level can be completed successfully.-commit_level updates DB2 pureScale instance to new level of code.-I specifies DB2 pureScale instance name to which fix pack update is applied.-online starts or continues with fix pack online update.-offline starts or continues with offline fix pack update.-quiesce_time_out specifies how long command waits before disconnecting applications from given member. Once timeout is reached, any active units of work remaining at the time are interrupted.-show_level_info parameter displays attribute status of a fix pack level.

79

installFixPack

The new **-check_commit** parameter determines whether the update of a DB2 pureScale instance to a higher code level can be completed successfully.

The new **-commit_level** parameter updates the DB2 pureScale instance to a new level of code.

The new **-I** parameter specifies the DB2 pureScale instance name to which the fix pack update is applied.

The new **-online** parameter starts or continues with a fix pack online update.

The new **-offline** parameter starts or continues with an offline fix pack update.

The new **-quiesce_time_out** parameter specifies how long the command waits before disconnecting applications from the given member. Once this timeout is reached, any active units of work remaining at the time are interrupted.

The new **-show_level_info** parameter displays the attribute status of a fix pack level.

LOAD command STATISTICS parameter changes

- **STATISTICS YES** is discontinued.
- **LOAD ... STATISTICS USE PROFILE**
 - Instructs load to collect statistics during load according to profile defined for table.
 - Default for column-organized tables during LOAD REPLACE.
 - Also during LOAD REPLACE against column-organized table, column compression dictionaries replaced by default.
 - To explicitly disable automatic statistics collection, specify STATISTICS NO parameter.
 - Profile must be created and stored in SYSCAT.TABLES.STATISTICS_PROFILE before issuing LOAD command.
**RUNSTATS ON TABLE mel.t1
WITH DISTRIBUTION AND DETAILED INDEXES ALL
SET PROFILE [ONLY]**
 - For row-organized tables, if profile does not exist and STATISTICS USE PROFILE specified, warning returned and no statistics collected.
 - For column-organized tables, if profile does not exist and STATISTICS USE PROFILE specified, load utility uses same default RUNSTATS options as those used during automatic runstats.
 - During load, distribution statistics are not collected for XML column type.
- **LOAD ... STATISTICS NO**
 - Specifies that no statistics are to be collected and statistics in catalogs are not to be altered.
 - Default for row-organized tables.

80

The **STATISTICS YES** parameter of the LOAD command is discontinued.

The **STATISTICS USE PROFILE** instructs load to collect statistics during the load according to the profile defined for this table. This is the default for column-organized tables. The profile must be created before issuing the LOAD command. The profile is created by the RUNSTATS command. For row-organized tables, if the profile does not exist and this parameter is specified, a warning is returned and no statistics are collected. For column-organized tables, if the profile does not exist and this parameter is specified, the load utility uses the same default RUNSTATS command options as those that are used during an automatic runstats operation. During load, distribution statistics are not collected for columns of type XML.

The **STATISTICS NO** specifies that no statistics are to be collected, and that the statistics in the catalogs are not to be altered. This option is the default for row-organized tables. **LOAD** For column-organized tables, automatic statistics collection occurs by default during the LOAD REPLACE command. To explicitly disable automatic statistics collection, specify the STATISTICS NO parameter. Also, during a LOAD REPLACE operation against a column-organized table, the column compression dictionaries are replaced by default.

NOT ENFORCED primary key & unique constraints support

- Informational constraints help query optimizer to select optimal data access plans when index access to the data provides no additional benefit.
- If know that data conforms to these constraints, can use NOT ENFORCED capability to help achieve two goals:
 - 1.Improve performance, primarily in insert, update, and delete operations on table
 - 2.Reduce space requirements associated with enforcing primary key or unique constraint
- Cannot reference not-enforced primary key constraints in any enforced referential integrity definitions.
- **Prior DB2 versions supported not-enforced foreign keys and not-enforced check constraints.**

82

You can create a not-enforced primary key or unique constraint for either a column-organized or row-organized table. Unlike an enforced primary key or unique constraint, a not-enforced primary key or unique constraint does not create an index on the data.

Specify an informational constraint only if the table data is independently known to conform to the constraint. Because the DB2 database manager does not enforce uniqueness for these constraints, if the table data violates the not-enforced constraint, incorrect results can occur. You cannot reference not-enforced primary key constraints in any enforced referential integrity definitions.

Expression-based indexes improve query performance

- CREATE INDEX statement extension to create index that contains expression-based key.
- Can create index on data that is not stored in table.
- When create index with expression-based key, results of expression are stored in index.
- SQL statement that contains predicate that includes expression can benefit from index whose definition includes expression-based keys. .
- Expression in index useful if query references expression in other ways:
 - query might return that expression in its results
 - query might order result set by that expression
- If index created with UNIQUE option, uniqueness enforced against values stored in index.
 - Uniqueness not enforced against original column values
- Create index with expression-based key using upper case of employee's name and ID:

```
CREATE INDEX EMP_UPPERNAME ON EMPLOYEE (UPPER(NAME), ID)
```

83

Starting with DB2 10.5, you can use an extension to the CREATE INDEX statement to create an index that contains an expression-based key. Using this expression-based key capability, you can create an index on data that is not stored in a table. When you create an index with an expression-based key, the results of the expression are stored in the index. An SQL statement that contains a predicate that includes an expression can benefit from an index whose definition includes expression-based keys. With this feature, the database manager can choose from many highly efficient index-based methods to access the data and evaluate the predicate. As a result, performance is improved for queries that contain expressions. An expression in an index is also useful if the query references the expression in other ways. For example, the query might return that expression in its results, or it might order the result set by that expression. To benefit from efficient evaluation of expressions that are specified at run time, use an expression-based key, rather than other types of indexes whose keys only consist of one or more table columns.

Expression-based indexes are best suited when you want an efficient evaluation of queries that involve a column expression. Simple index keys consist of a concatenation of one or more specified table columns. Compared to simple indexes, the index key values of expression-based indexes are not the same as values in the table columns. The values are transformed by the expressions that you specify. You can create the index with the CREATE INDEX statement. If an index is created with the UNIQUE option, the uniqueness is enforced against the values that are stored in the index. The uniqueness is not enforced against the original column values.

EXCLUDE NULL KEYS from Indexes

- **EXCLUDE NULL KEYS** on **CREATE INDEX** reduces size of sparse indexes.
 - does not insert key into index object when all columns in key are null.
 - Can result in improved storage and performance optimization when do not want queries to access data associated with null keys.
 - For unique indexes, enforcement of uniqueness of table data ignores rows where the index key is null.
- **INCLUDE NULL KEYS** specifies that index entry is created when all parts of index key contain null value. Default setting.
- Cannot specify **EXCLUDE NULL KEYS** with following:
 - Nickname
 - **GENERATE KEY USING XMLPATTERN** clause
 - **EXTEND USING** clause.

84

Starting with DB2 10.5 the new **EXCLUDE NULL KEYS** clause can be used on the **CREATE INDEX** statement to reduce the size of sparse indexes.

An index created with the **EXCLUDE NULL KEYS** clause does not insert a key into the index object when all the columns in the key are null. Excluding null keys can result in improved storage and performance optimization for cases where you do not want queries to access data associated with null keys. For unique indexes, the enforcement of the uniqueness of table data ignores rows where the index key is null.

Unique indexes are indexes that help maintain data integrity by ensuring that no two rows of data in a table have identical key values.

When you create a unique index for an existing table with data, values in the columns or expressions that comprise the index key are checked for uniqueness. If the table contains rows with duplicate key values, the index creation process fails. When a unique index is defined for a table, uniqueness is enforced whenever keys are added or changed within the index. This enforcement includes insert, update, load, import, and set integrity, to name a few. In addition to enforcing the uniqueness of data values, a unique index can also be used to improve data retrieval performance during query processing.

Non-unique indexes are not used to enforce constraints on the tables with which they are associated. Instead, non-unique indexes are used solely to improve query performance by maintaining a sorted order of data values that are used frequently.

It is important to understand that there is no significant difference between a primary key or unique key constraint and a unique index. To implement the concept of primary and unique key constraints, the database manager uses a combination of a unique index and the **NOT NULL** constraint. Therefore, unique indexes do not enforce primary key constraints by themselves because they allow null values. Although null values represent unknown values, when it comes to indexing, a null value is treated as being equal to other null values.

Therefore, if a unique index consists of a single column, only one null value is allowed—more than one null value would violate the unique constraint. Similarly, if a unique index consists of multiple columns, a specific combination of values and nulls can be used only one time.

INCLUDE NULL KEYS - Specifies that an index entry is created when all parts of the index key contain the null value. This is the default setting.

EXCLUDE NULL KEYS - Specifies that an index entry is not created when all parts of the index key contain the null value. When any part of the index key is not a null value, an index entry is created. You cannot specify **EXCLUDE NULL KEYS** with the following syntax elements:

- A nickname
- The **GENERATE KEY USING XMLPATTERN** clause
- The **EXTEND USING** clause.

REF: Random Key Indexes

- Some workloads may experience page contention on frequently accessed index leaf pages
 - For example, index key on monotonically incrementing value
 - Such as timestamp or identity column
 - Results in "hot" index leaf page, which is insert point for all new keys being generated
 - Typical example is "order number" column in retail industry schema
- In pureScale may be exacerbated where pages shared across members
 - Hot index page gets reclaimed/negotiated over and over again between those members (routed through CF)
- Random key indexes address this issue

```
CREATE INDEX IX1 ON TAB1 (ORDER_NUM RANDOM)
```

85

With some workloads, it is possible that high index key page contention can impact the performance of the workload being done. This can happen when an index is ordered on a monotonically incrementing value, such as a sequence number, a timestamp, or some other user-specified monotonically increasing value. An example of this is a schema associated with order processing, where there is a unique order number that is constantly being generated and assigned.

With such an index defined, every time that a transaction inserts a row into the table, it is generating the highest value seen for that column so far and it must be placed in the last index leaf page. For DB2 in general (pureScale or non-pureScale) this can result in page contention if multiple inserts are occurring concurrently. As well, for pureScale specifically, if the application is being executed on multiple members concurrently then that means that everybody is going to want to put their index key value into the same high key index leaf page. The result of this is that the index leaf page is going to bounce back and forth between members using the process of page reclaim/negotiation. This can be quite an expensive operation as a page reclaim involves flushing the log buffer to disk on the member where the latest version of the page currently resides and then the page is sent from that member to the GBP on the CF. The page is then pulled from the GBP to the member that is requesting it. If this is done repeatedly then it can have a serious impact on performance.

To resolve this, a new index order of `RANDOM` can be specified. This means that instead of index key values being sorted in a strictly ascending or descending order, you can have them sorted in a random fashion. That means that part of the key that gets generated is a random value and so the placement of the key can occur randomly across all of the leaf pages in the index. By doing this, there is no longer that page contention taking place on the last leaf page in the index. The next slide will show an example of this.

There were previously some methods that could be employed to work around this, such as caching sequence numbers or inserting a member number into the index key, but this should be an easier solution for most customers and it all gets handled automatically by DB2.

REF: Random Key Indexes reduce index leaf page contention

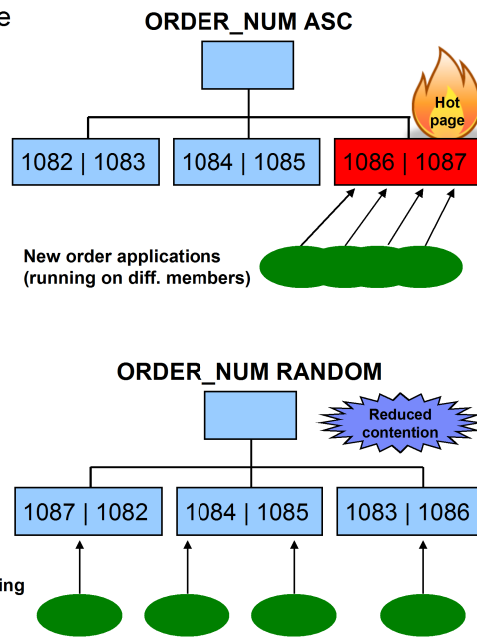
- Random key indexes allow you to randomize placement of index key values
- Spreads out contention of index high key
- Loss of order, so range queries become full index scans
 - Extra sort may be required for ORDER BY

ORDER_NUM BETWEEN 1080 AND 1090

- Index supports equality lookups (ORDER_NUM = 1083)

```
CREATE INDEX IX1 ON TAB1  
(INT ORDER_NUM RANDOM)
```

New order applications (running on different members)



92

As mentioned on the previous slide, by creating a RANDOM index you end up randomizing the index keys and therefore the placement of the keys within the index leaf pages. By doing this, there is no longer that contention on the last leaf page in the index.

Note, however, that this does not come without any cost. Because the key values are located in random locations within the index leaf pages, it is not possible to perform range queries (for example: ORDER_NUM BETWEEN 1080 AND 1090). Therefore, what could have been done previously with a simple scan of a portion of the index results in full index scans. Therefore, random indexes are best suited for situations where point queries are being made (for example: ORDER_NUM = 1082).

A similar solution to this exists for DB2 for z/OS

Note: A range predicate or IS NOT NULL predicate on the random column of an index cannot be used as a start-stop predicate for that index. Only equality predicates on the random column of an index can be used in a start-stop predicate. However, the random index can still be chosen to satisfy the query by doing a full index scan. The random index can also be chosen to satisfy the query by using jumpscan with the random column treated as a gap.

REF: Random Key Indexes: Implementation Details

- Extra hidden key part added into internal index definition
 - 2 bytes in size
 - Contains hash value of real index key column

- Example

```
CREATE TABLE T1(C1 INT NOT NULL, C2 INT NOT NULL,  
                C3 CHAR(10) NOT NULL)
```

```
CREATE INDEX I1(C1 RANDOM, C2 ASC, C3 ASC)
```

```
INSERT INTO T1 VALUES (100, 123, 'Holly')
```

- Index key will be created that looks like this

	Hidden column: Hashed value of C1	Column C1	Column C2	Column C3
Size	2 bytes	4 bytes	4 bytes	10 bytes
Value	0xB3D1	100	123	'Holly'

94

The approach taken to implement these indexes does not simply randomize the key part itself (e.g. the order number column) and insert the randomized value into the index. Instead, the approach taken creates a new, internal, hash key part in the index for each column defined with RANDOM ordering. This hash key part is defined internally as a NOT NULL SMALLINT (2 bytes) ASCENDING and precedes each random column key part in the index. The value of the random column is hashed into a 2-byte value which is stored in this internal key part.

In the example shown, an index is created with three key parts, the first column of which (C1) is defined to have random ordering. In the internal index definition we place the hidden key part column prior to C1. Contained within the hidden key value is the hashed value of the corresponding column's key value.

Large row size support

- Row size support extended allowing to create table where row length can exceed maximum record length for page size of table space
- **EXTENDED_ROW_SZ=ENABLE** db cfg parameter must be set
 - Migrate tables that have row sizes exceeding 32K
 - Improve performance where majority of data rows can fit on smaller page
 - Create tables with more VARCHAR or VARGRAPHIC columns
 - Maximum number of columns is not changing
 - Ability to exceed maximum record length for page size allows for more columns
- After upgrade, alter existing tables to take advantage of extended row size support.
CREATE TABLE T1 (C1 INT, C2 VARCHAR(3995)) in TS1
 - Table space TS1 has 4K page size and T1 created without need to store any data out of row because its byte count is 4005.
ALTER TABLE T1 ADD C3 CLOB(1M)
 - Byte count for T1 now exceeds maximum record length, but ALTER TABLE successful if **EXTENDED_ROW_SZ=ENABLE**.
- Row data that exceeds maximum row length moves subset of VARCHAR or VARGRAPHIC data out of row and stores as LOB data
 - ***data access might be slower because additional I/O resources might be required to fetch, insert, or update the LOB data***

88

Starting with DB2 10.5 row size support is extended allowing you to create a table where its row length can exceed the maximum record length for the page size of the table space.

Rows of table data are organized into blocks called pages which can be four sizes: 4, 8, 16, and 32 kilobytes. All tables created within a table space of a particular size have a matching page size. In previous releases, the maximum number of bytes allowed in a table row was dependant on the page size of the table space. Any attempt to create a table whose row length exceeded the maximum record length for the page size would result in an error (SQLSTATE 54010). For example, in previous releases the following table could not be created in a 4K page size table space because of its large row size. **CREATE TABLE T1 (C1 INTEGER, C2 VARCHAR(5000))** By extending row size support to allow for the creation of tables containing large rows that exceed the maximum record length for the page size of the table space, the table T1 can be created in a 4K page size table space.

When extended_row_sz is set to ENABLE, extended row size is enabled. During an INSERT or DELETE table operation, row data that exceeds the maximum row length moves a subset of VARCHAR or VARGRAPHIC data out of the row and stores it as LOB data. While the data is in LOB format, data access might be slower because additional I/O resources might be required to fetch, insert, or update the LOB data.

For further details see:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.admin.dboobj.doc/doc/c0061057.html>

New Database and Instance monitoring functions

- **MON_GET_DATABASE, MON_GET_DATABASE_DETAILS**

- Returns database statistics previously only available using DATABASE snapshot monitoring
- Determine activation time, activation state, total connections, and connection high water mark for current database:
SELECT DB_CONN_TIME, DB_ACTIVATION_STATE, TOTAL_CONS, CONNECTIONS_TOP FROM TABLE (MON_GET_DATABASE(-2));

- **MON_GET_INSTANCE**

- Returns DB2 instance level information previously only available using DATABASE MANAGER snapshot monitoring
- Determine start time and total number of current connections:
SELECT DB2START_TIME, TOTAL_CONNECTIONS FROM TABLE (MON_GET_INSTANCE(-2));

The MON_GET_DATABASE table function returns database level information within the monitor infrastructure.

The MON_GET_DATABASE_DETAILS table function retrieves database information metrics in an XML document. The schema for the XML document that is returned in the DETAILS column is available in the file sqllib/misc/DB2MonRoutines.xsd. Further details can be found in the file sqllib/misc/DB2MonCommon.xsd.

The MON_GET_INSTANCE table function returns DB2 instance level information within the monitor infrastructure.

New active utility monitoring - MON_GET_UTILITY

```
SELECT UTILITY_ID, UTILITY_START_TIME, UTILITY_TYPE,  
VARCHAR(OBJECT_SCHEMA,12) AS SCHEMA,  
VARCHAR(OBJECT_NAME,20) AS OBJECT_NAME,  
SUBSTR(UTILITY_DETAIL,1,200) AS CMD  
FROM TABLE(MON_GET_UTILITY(-1)) AS T1
```

```
UTILITY_ID  UTILITY_START_TIME      UTILITY_TYPE  SCHEMA  OBJECT_NAME  
-----  
          2  2013-06-25-14.02.22.325849  LOAD          TEST2      ACCT  
CMD  
-----  
[LOADID: 21.2013-06-25-14.02.22.266904.0 (6;8)] [*LOCAL.inst20.130625175129] OFFLINE LOAD DEL  
AUTOMATIC INDEXING REPLACE COPY NO TEST2      .ACCT  
  
1 record(s) selected.
```

- SNAPUTIL administrative view and SNAP_GET_UTIL table function deprecated.
- Alternative to db2 list utilities show detail and db2pd -util

90

The MON_GET_UTILITY table function returns information about utilities currently running on the database.

One of the following authorizations is required:

- EXECUTE privilege on the routine
- DATAACCESS authority
- DBADM authority
- SQLADM authority

Important: The SNAPUTIL administrative view and the SNAP_GET_UTIL table function are deprecated and have been replaced by the MON_GET_UTILITY table function.

DISCONTINUED SQL administrative routines and views (1)

UPGRADE DATABASE command drops discontinued administrative routines and marks dependent objects inoperative or invalid.

Discontinued in V10.5	Replacement View/Routine
SNAPSHOT_AGENT	MON_GET_AGENT, MON_GET_CONNECTION
SNAPSHOT_APPL	MON_GET_CONNECTION
SNAPSHOT_APPL_INFO	MON_GET_CONNECTION
SNAPSHOT_BP	MON_GET_BUFFERPOOL
SNAPSHOT_CONTAINER	MON_GET_CONTAINER
SNAPSHOT_DATABASE	MON_GET_DATABASE, MON_GET_TRANSACTION_LOG
SNAPSHOT_DBM	MON_GET_INSTANCE
SNAPSHOT_DYN_SQL	MON_GET_PKG_CACHE_STMT
SNAPSHOT_FCMNODE	MON_GET_FCM
SNAPSHOT_FILEW	SNAP_WRITE_FILE

91

A number of SQL administrative routines are discontinued in Version 10.5. You must use the replacement routines. The following shows the SQL administrative routines that are discontinued, their replacement routines or views, and the version when the replacement views or routines became available:

- Discontinued SNAPSHOT_AGENT table function replaced with MON_GET_AGENT table function and MON_GET_CONNECTION table function in Version 10.5 and 9.7.
- Discontinued SNAPSHOT_APPL table function replaced with MON_GET_CONNECTION table function and MON_GET_UNIT_OF_WORK table function in Version 9.7.
- Discontinued SNAPSHOT_APPL_INFO table function replaced with MON_GET_CONNECTION table function and MON_GET_UNIT_OF_WORK table function in Version 9.7.
- Discontinued SNAPSHOT_BP table function replaced with MON_GET_BUFFERPOOL table function in Version 9.7.
- Discontinued SNAPSHOT_CONTAINER table function replaced with MON_GET_CONTAINER table function in Version 9.7.
- Discontinued SNAPSHOT_DATABASE table function replaced with MON_GET_DATABASE table function and MON_GET_TRANSACTION_LOG table function in Version 10.5.
- Discontinued SNAPSHOT_DBM table function replaced with MON_GET_INSTANCE table function in Version 10.5.
- Discontinued SNAPSHOT_DYN_SQL table function replaced with MON_GET_PKG_CACHE_STMT table function in Version 9.7.
- Discontinued SNAPSHOT_FCM table function replaced with MON_GET_FCM table function in Version 9.7 Fix Pack 2.
- Discontinued SNAPSHOT_FCMNODE table function replaced with MON_GET_FCM_CONNECTION_LIST table function in Version 9.7 Fix Pack 2.
- Discontinued SNAPSHOT_FILEW procedure replaced with SNAP_WRITE_FILE procedure in Version 9.1.
- Discontinued SNAPSHOT_LOCK table function replaced with MON_GET_APPL_LOCKWAIT table function, MON_GET_LOCKS table function, MON_FORMAT_LOCK_NAME table function in Version 9.7 Fix Pack 1.
- Discontinued SNAPSHOT_LOCKWAIT table function replaced with MON_GET_APPL_LOCKWAIT table function, MON_GET_LOCKS table function, and MON_FORMAT_LOCK_NAME table function in Version 9.7 Fix Pack 1.
- Discontinued SNAPSHOT QUIESCERS table function replaced with SNAP_GET_TBSP QUIESCER table function in Version 9.1.
- Discontinued SNAPSHOT_RANGES table function replaced with SNAP_GET_TBSP_RANGE table function in Version 9.1.

Student Notes: For Reference Only

- Discontinued SNAPSHOT_STATEMENT table function replaced with MON_GET_ACTIVITY table function and MON_CURRENT_SQL administrative view in Version 10.5 and Version 9.7 Fix Pack 1.
- Discontinued SNAPSHOT_SUBSECT table function replaced with SNAP_GET_SUBSECTION table function in Version 9.1.
- Discontinued SNAPSHOT_SWITCHES table function replaced with SNAP_GET_SWITCHES table function in Version 9.1.
- Discontinued SNAPSHOT_TABLE table function replaced with MON_GET_TABLE table function in Version 9.7.
- Discontinued SNAPSHOT_TBREORG table function replaced with MON_GET_UTILITY table function and SNAP_GET_TAB_REORG table function in Version 10.5 and Version 9.1.
- Discontinued SNAPSHOT_TBS table function replaced with MON_GET_TABLESPACE table function in Version 9.7.
- Discontinued SNAPSHOT_TBS_CFG table function replaced with MON_GET_TABLESPACE table function and MON_GET_REBALANCE_STATUS table function in Version 9.7 and Version 10.1.
- Discontinued SNAPSHOT_UTIL table function replaced with MON_GET_UTILITY table function in Version 10.5.
- Discontinued SNAPSHOT_UTIL_PROG table function replaced with SNAPUTIL_PROGRESS administrative view and SNAP_GET_UTIL_PROGRESS table function in Version 9.1.

DISCONTINUED SQL administrative routines and views (2)

Discontinued in V10.5	Replacement View/Routine
SNAPSHOT_LOCK SNAPSHOT_LOCKWAIT	MON_GET_APPL_LOCKWAIT, MON_GET_LOCKS, MON_FORMAT_LOCK_NAME
SNAPSHOT QUIESCERS	SNAP_GET_TBSP QUIESCER
SNAPSHOT_STATEMENT	MON_GET_ACTIVITY, MON_CURRENT_SQL
SNAPSHOT_SUBSECT	SNAP_GET_SUBSECTION
SNAPSHOT_SWITCHES	SNAP_GET_SWITCHES
SNAPSHOT_TABLE	MON_GET_TABLE
SNAPSHOT_TABREORG	MON_GET_UTILITY
TAB_REORG	MON_GET_TABLESPACE
SNAPSHOT_UTIL	SNAPSHOT_UTIL_PROGRESS
SNAPSHOT_UTIL_PROGRESS	SNAPSHOT_UTIL_PROGRESS, SNAP_GET_UTIL_PROGRESS

93

Modify all user-defined objects that are dependent on the discontinued routines. Re-create these objects using the replacement routines or views indicated. If you upgrade a database that has dependent objects, the UPGRADE DATABASE command drops the discontinued administrative routines and marks the dependent objects inoperative or invalid.

Modify all your applications and scripts and remove all references to these routines or use the replacement routines or views indicated.

- Discontinued SNAPSHOT_LOCK table function replaced with MON_GET_APPL_LOCKWAIT table function, MON_GET_LOCKS table function, MON_FORMAT_LOCK_NAME table function in Version 9.7 Fix Pack 1.
- Discontinued SNAPSHOT_LOCKWAIT table function replaced with MON_GET_APPL_LOCKWAIT table function, MON_GET_LOCKS table function, and MON_FORMAT_LOCK_NAME table function in Version 9.7 Fix Pack 1.
- Discontinued SNAPSHOT QUIESCERS table function replaced with SNAP_GET_TBSP QUIESCER table function in Version 9.1.
- Discontinued SNAPSHOT_RANGES table function replaced with SNAP_GET_TBSP_RANGE table function in Version 9.1.
- Discontinued SNAPSHOT_STATEMENT table function replaced with MON_GET_ACTIVITY table function and MON_CURRENT_SQL administrative view in Version 10.5 and Version 9.7 Fix Pack 1.
- Discontinued SNAPSHOT_SUBSECT table function replaced with SNAP_GET_SUBSECTION table function in Version 9.1.
- Discontinued SNAPSHOT_SWITCHES table function replaced with SNAP_GET_SWITCHES table function in Version 9.1.

Student Notes: For Reference Only

- Discontinued SNAPSHOT_TABLE table function replaced with MON_GET_TABLE table function in Version 9.7.
- Discontinued SNAPSHOT_TBREORG table function replaced with MON_GET_UTILITY table function and SNAP_GET_TAB_REORG table function in Version 10.5 and Version 9.1.
- Discontinued SNAPSHOT_TBS table function replaced with MON_GET_TABLESPACE table function in Version 9.7.
- Discontinued SNAPSHOT_TBS_CFG table function replaced with MON_GET_TABLESPACE table function and MON_GET_REBALANCE_STATUS table function in Version 9.7 and Version 10.1.
- Discontinued SNAPSHOT_UTIL table function replaced with MON_GET_UTILITY table function in Version 10.5.
- Discontinued SNAPSHOT_UTIL_PROG table function replaced with SNAPUTIL_PROGRESS administrative view and SNAP_GET_UTIL_PROGRESS table function in Version 9.1.

Deprecated SQL administrative routines and views (1)

Discontinued in V10.5	Replacement View/Routine
APPL_PERFORMANCE	MON_CONNECTION_SUMMARY
APPLICATIONS	MON_GET_CONNECTION
BP_HITRATIO, BP_READ_IO, BP_WRITE_IO	MON_BP_UTILIZATION
CONTAINER_UTILIZATION	MON_GET_CONTAINER
LOG_UTILIZATION	MON_TRANSACTION_LOG_UTILIZATION
SNAPAGENT, SNAP_GET_AGENT	MON_GET_CONNECTION, MON_GET_AGENT
SNAPAPPL, SNAP_GET_AGENT, SNAP_GET_APPL	MON_GET_CONNECTION
SNAPBP, SNAP_GET_BP, SNAPBP_PART, SNAP_GET_BP_PART	MON_GET_BUFFERPOOL
SNAP_CONTAINER, SNAP_GET_CONTAINER	MON_GET_CONTAINER
SNAPDB, SNAP_GET_DB	MON_GET_DATABASE
SNAPDBM, SNAP_GET_DBM	MON_GET_INSTANCE

95

Expanded support is provided for the existing administrative routines and views by replacing them with new, more comprehensive routines or views. Starting with 10.1, routines with version-suffixed names were deprecated. The names of the routines do not have a version suffix so that the names remain consistent across releases. The replacement routines without a version-suffix might include modifications such as: columns added or removed, new data types for existing columns, or new values for existing columns. Use good practices when issuing queries on SQL administrative routines to minimize the impact from changes to these routines.

- Deprecated APPL_PERFORMANCE administrative view - Retrieve percentage of rows selected for an application replaced with MON_CONNECTION_SUMMARY - Retrieve metrics for all connections.
- Deprecated APPLICATIONS administrative view - Retrieve connected database application information replaced with MON_GET_CONNECTION table function - Get connection metrics.
- Deprecated BP_HITRATIO administrative view - Retrieve bufferpool hit ratio information replaced with MON_BP_UTILIZATION administrative view - Retrieve metrics for bufferpools.
- Deprecated BP_READ_IO administrative view - Retrieve bufferpool read performance information replaced with MON_BP_UTILIZATION administrative view - Retrieve metrics for bufferpools.
- Deprecated BP_WRITE_IO administrative view - Retrieve bufferpool write performance information replaced with MON_BP_UTILIZATION administrative view - Retrieve metrics for bufferpools.
- Deprecated CONTAINER_UTILIZATION administrative view - Retrieve table space container and utilization information replaced with MON_GET_CONTAINER table function - Get table space container metrics.
- Deprecated LOG_UTILIZATION administrative view - Retrieve log utilization information replaced with MON_TRANSACTION_LOG_UTILIZATION administrative view - Retrieve log utilization information.
- Deprecated LONG_RUNNING_SQL administrative view replaced with MON_CURRENT_SQL - Retrieve key metrics for all activities on all members.
- Deprecated SNAP_WRITE_FILE procedure - not available.
- Deprecated SNAPAGENT administrative view and SNAP_GET_AGENT table function - Retrieve agent logical data group application snapshot information replaced with MON_GET_CONNECTION table function - Get connection metrics and MON_GET_AGENT table function - List agents running in a service class.
- Deprecated SNAPAPPL administrative view and SNAP_GET_APPL table function - Retrieve appl logical data group snapshot information replaced with MON_GET_CONNECTION table function - Get connection metrics.
- Deprecated SNAPAPPL_INFO administrative view and SNAP_GET_APPL_INFO table function - Retrieve appl_info logical data group snapshot information replaced with MON_GET_CONNECTION table function - Get connection metrics.
- Deprecated SNAPBP administrative view and SNAP_GET_BP table function - Retrieve bufferpool logical group snapshot information replaced with MON_GET_BUFFERPOOL table function - Get buffer pool metrics.
- Deprecated SNAPBP_PART administrative view and SNAP_GET_BP_PART table function - Retrieve bufferpool_nodeinfo logical data group snapshot information replaced with MON_GET_BUFFERPOOL table function - Get buffer pool metrics.
- Deprecated SNAPCONTAINER administrative view and SNAP_GET_CONTAINER table function - Retrieve tablespace_container logical data group snapshot information replaced with MON_GET_CONTAINER table function - Get table space container metrics.
- Deprecated SNAPDB administrative view and SNAP_GET_DB table function - Retrieve snapshot information from the dbase logical group replaced with MON_GET_DATABASE table function - Get database level information.
- Deprecated SNAPDBM administrative view and SNAP_GET_DBM table function - Retrieve the dbm logical grouping snapshot information replaced with MON_GET_INSTANCE table function - Get instance level information.

Deprecated SQL administrative routines and views (2)

Discontinued in V10.5	Replacement View/Routine
SNAPFCM, SNAP_GET_FCM	MON_GET_FCM
SNAPDETAILLOG, SNAP_GET_DETAILLOG	MON_GET_TRANSACTION_LOG
SNAPFCM_PART, SNAP_GET_FCM_PART	MON_GET_FCM
SNAPSTMT, SNAP_GET_TBSP	MON_GET_TABLESPACE
SNAPTbsp, SNAP_GET_TBSP	MON_GET_TABLESPACE
SNAPTbsp_PART, SNAP_GET_TBSP_PART	MON_GET_TABLESPACE
SNAPUTIL, SNAP_GET_UTIL	MON_GET_UTILITY
TBSP_UTILIZATION	MON_TBSP_UTILIZATION

96

- Deprecated SNAPFCM administrative view and SNAP_GET_FCM table function - Retrieve the fcm logical data group snapshot information replaced with MON_GET_FCM - Get FCM metrics.
- Deprecated SNAPDETAILLOG administrative view and SNAP_GET_DETAILLOG table function - Retrieve snapshot information from the detail_log logical data group replaced with MON_GET_TRANSACTION_LOG table function - Get log information.
- Deprecated SNAPFCM_PART administrative view and SNAP_GET_FCM_PART table function - Retrieve the fcm_node logical data group snapshot information replaced with MON_GET_FCM - Get FCM metrics.
- Deprecated SNAPSTMT administrative view and SNAP_GET_STMT table function - Retrieve statement snapshot information replaced with MON_CURRENT_SQL - Retrieve key metrics for all activities on all members and MON_GET_CONNECTION table function - Get connection metrics.
- Deprecated SNAPTbsp administrative view and SNAP_GET_TBSP table function - Retrieve table space logical data group snapshot information replaced with MON_GET_TABLESPACE table function - Get table space metrics.
- Deprecated SNAPTbsp_PART administrative view and SNAP_GET_TBSP_PART table function - Retrieve tablespace_nodeinfo logical data group snapshot information replaced with MON_GET_TABLESPACE table function - Get table space metrics.
- Deprecated SNAPUTIL administrative view and SNAP_GET_UTIL table function - Retrieve utility_info logical data group snapshot information replaced with MON_GET_UTILITY table function - Get utilities running on the database.
- Deprecated TBSP_UTILIZATION administrative view - Retrieve table space configuration and utilization information replaced with MON_TBSP_UTILIZATION - Retrieve monitoring metrics for all table spaces and all database partitions.

Administrative Views and Routines Changes in DB2 10.5.

- **New columns, changed descriptions, changed column data types, and increased column lengths**

- ADMIN_TABINFO and ADMIN_GET_TAB_INFO
- ENV_SYS_INFO
- MON_BP_UTILIZATION
- MON_CONNECTION_SUMMARY
- MON_CURRENT_SQL
- MON_DB_SUMMARY
- MON_FORMAT_XML_COMPONENT_TIMES_BY_ROW
- MON_FORMAT_XML_METRICS_BY_ROW
- MON_FORMAT_XML_TIMES_BY_ROW
- MON_GET_APPL_LOCKWAIT
- MON_GET_BUFFERPOOL
- MON_GET_CONNECTION
- MON_GET_CONNECTION_DETAILS
- MON_GET_PKG_CACHE_STMT
- MON_GET_SERVICE_SUBCLASS
- MON_GET_SERVERLIST
- MON_GET_TABLE
- MON_GET_TABLESPACE
- MON_GET_TABLE_USAGE_LIST
- MON_TBSP_UTILIZATION
- MON_GET_UNIT_OF_WORK
- MON_GET_UNIT_OF_WORK_DETAILS
- MON_GET_WORKLOAD
- MON_GET_WORKLOAD_DETAILS
- MON_WORKLOAD_SUMMARY

For complete list of changes and deprecated table functions and views in DB2 9.7, 10.1 and 10.5 see URLs in notes.

97

The above administrative views and routines have changed in DB2 10.5. Most modifications consist of new columns, new values, changed column data types, and increased column lengths.

Review the list of the deprecated administrative routines and their replacement routines or views to determine additional changes that might impact your applications and scripts.

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.sql.rtn.doc/doc/0023171.html>

New Routines and Views in DB2 10.5

- **New monitor routines and views:**
 - MON_CAPTURE_ACTIVITY_IN_PROGRESS procedure
 - MON_COLLECT_STATS procedure
 - MON_GET_ACTIVITY table function
 - MON_GET_AGENT table function
 - MON_GET_DATABASE table function
 - MON_GET_DATABASE_DETAILS table function
 - MON_GET_INSTANCE table function
 - MON_GET_QUEUE_STATS table function
 - MON_GET_SERVICE_SUBCLASS_STATS table function
 - MON_GET_SERVICE_SUPERCLASS_STATS table function
 - MON_GET_UTILITY table function
 - MON_GET_WORKLOAD_STATS table function
 - MON_GET_WORK_ACTION_SET_STATS table function
 - MON_TRANSACTION_LOG_UTILIZATION administrative view
- **New environment routines:**
 - ENV_GET_DB2_EDU_SYSTEM_RESOURCES table function
 - ENV_GET_INSTANCE_CODE_LEVELS table function
- **New workload management routines:**
 - WLM_ALTER_MEMBER_SUBSET procedure
 - WLM_CREATE_MEMBER_SUBSET procedure
 - WLM_DROP_MEMBER_SUBSET procedure

98

Adjust your applications to the changes in existing system catalog objects such as new columns, or modified data type in columns. Also, start using new system catalog objects that provide more comprehensive views or routines, or access to information about new functionality.

To minimize the impact of changes to built-in routines and views, review Best practices for calling built-in routines and views in applications.

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.sql.rtn.doc/doc/c0056435.html>

For a list of the data dictionary-compatible views, see Data dictionary-compatible views at:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.wn.doc/doc/i0058717.html>

System Catalog Views – New and Changed

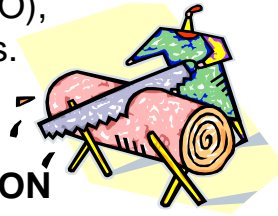
- **New System Catalog Views**
 - SYSCAT.MEMBERSUBSETATTRS
 - SYSCAT.MEMBERSUBSETMEMBERS
 - SYSCAT.MEMBERSUBSETS
- **New columns, changed descriptions, changed column data types, and increased column lengths**
 - SYSCAT.ATTRIBUTES catalog view
 - SYSCAT.CHECKS catalog view
 - SYSCAT.COLUMNS catalog view
 - SYSCAT.CONTROLS catalog view
 - SYSCAT.DATATYPES catalog view
 - SYSCAT.INDEXCOLUSE catalog view
 - SYSCAT.INDEXES catalog view
 - SYSCAT.PACKAGES catalog view
 - SYSCAT.ROUTINEPARMS catalog view
 - SYSCAT.ROUTINES
 - SYSCAT.ROWFIELDS
 - SYSCAT.SERVICECLASSES
 - SYSCAT.STOGROUPS
 - SYSCAT.TABDEP
 - SYSCAT.TABLES-new column called TABLEORG indicates table organization.
 - SYSCAT.TABLESPACES
 - SYSCAT.TRIGGERS
 - SYSCAT.VARIABLES
 - SYSCAT.VIEWS
 - SYSSTAT.COLUMNS
 - SYSSTAT.INDEXES
 - SYSSTAT.TABLES

To support new features in DB2 Version 10.5, database catalog objects such as system catalog views, built-in functions and global variables, built-in administrative routines and views have been added and modified.

Most modifications to catalog views consist of new columns, changed descriptions, changed column data types, and increased column lengths.

REF: DB2 10.5 Deprecated Functionality – Raw Logs

- Use of raw devices for database logging has been deprecated since DB2 9.1 and will be removed in a future release.
- Using file system with non-buffered I/O capabilities enabled, such as Concurrent I/O (CIO) or Direct I/O (DIO), yields performance comparable to raw devices.



- Change **newlogpath** to file system directory:

**db2 UPDATE DATABASE CONFIGURATION
USING newlogpath /disk2/newlogdir**

- Change does not become effective until database in consistent state and all users are disconnected from database.
- Database manager moves logs to new location after first user connects to database.

100

You should be aware of functionality that is deprecated or discontinued in DB2 10.5 that can affect the upgrade of your DB2 server. Also, you should be aware of the DB2 products that are no longer supported because upgrade from these products to DB2 10.5 is unsupported. To deal with these functionality changes, you must perform additional tasks before or after upgrade. The above list describes changes that are not included in the pre-upgrade and post-upgrade tasks for DB2 servers.

If you are upgrading from DB2 V10.5 versions, you must also review the changes made in DB2 Version 10.1 or Version 9.7 that might impact your environment after upgrading to DB2 Version 10.5. Review the following topic to learn about additional possible impacts on the upgrade of your DB2 server:

Deprecated or discontinued functionality in DB2 Version 10.1 for upgrade from DB2 Version 10.1.

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.qb.upgrade.doc/doc/c0022309.html>

Deprecated or discontinued functionality in DB2 Version 9.7 for upgrade from DB2 Version 9.7.

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.qb.upgrade.doc/doc/c0022309.html>

REF: Snapshot Backup Scripts

- Allows for integrated snapshot backup capabilities for storage devices not supported by DB2 Advanced Copy Services (ACS)
 - Works with pureScale
- Custom script implements the DB2 ACS API
 - Users or storage vendors can write their own scripts
 - Write operations to database are automatically suspended and resumed by DB2 during backup process
- Benefits include
 - Wider storage support
 - Avoids need for manual snapshot backup process in pureScale
 - Manually running `SET WRITE SUSPEND`, `SET WRITE RESUME`, `db2inidb`, and storage vendor commands can be error prone
 - History file record is generated



```
BACKUP DATABASE PRODDATA USE SNAPSHOT SCRIPT '/scripts/snapshot.sh'  
RESTORE DATABASE PRODDATA USE SNAPSHOT SCRIPT '/scripts/snapshot.sh'  
TAKEN AT 20130614120000
```


08

If you were performing a snapshot operation in DB2 10.1, you either had to use storage hardware that provided a vendor library that supported the DB2 ACS API (for non-pureScale only), or you had to write your own script -- which included having to suspend and resume writes to the database and call the underlying storage commands to take the snapshot. There are some drawbacks to writing a script like this. For instance, they can be difficult and error-prone to write, especially in regards to suspending and resuming database operations. Also, they do not generate a history file entry, so you cannot monitor the progress and success of the snapshot operation.

In DB2 10.5, these trade-offs have been eliminated. The DB2 ACS API is now wrapped in the library for DB2 ACS. The library invokes a custom script to perform the snapshot operation. DB2 takes over the error-prone actions like issuing the `SET WRITE SUSPEND`, `SET WRITE RESUME`, and `db2inidb` commands at the correct time. At the same time, because the DB2 ACS API is being used as part of a true DB2 backup operation, an entry is made in the recovery history file for every snapshot operation, allowing you to monitor successful and unsuccessful backups.

We document the API requirements within the Information Center and it is possible for DB2 users to write their own scripts. However, it is suggested that people reach out to their storage vendor to have them write and provide a script instead.

REF: Upgrading to a new DB2 10.5 Server (1)

- Perform a full **offline** database backup of your existing databases
- If database backups performed recently, can perform incremental *offline* database backup instead.
- Log on to new DB2 server as root on Linux and UNIX or user with Local Administrator authority on Windows.
- Install DB2 10.5 on new DB2 server.
- Create DB2 10.5 instances on new DB2 server by running **db2icrt**.
- If similar resources, restore DBM CFG values for each instance.
- Transfer pre-DB2 10.5 backup files for all databases want to upgrade to new server.
- Upgrade database using RESTORE DATABASE:
db2 RESTORE DATABASE *sample* FROM /db2/backups
- If performed incremental *offline* database backup before upgrade, must have access to most recent full *offline* database backup and incremental *offline* database backup and use automatic incremental restore to upgrade database.
 - manual incremental restore will fail because each RESTORE DATABASE  tries to upgrade database before database is completely recovered.

db2 RESTORE DATABASE *sample* INCREMENTAL AUTOMATIC TAKEN AT *timestamp* WITHOUT PROMPTING

- In partitioned database, must execute RESTORE DATABASE on all database partitions starting with catalog partition first.

102

If you want to upgrade to a new DB2 10.5 server, re-create your instances and then upgrade your databases by restoring a pre-DB2 10.5 database backup. After restoring the database backup, the RESTORE DATABASE command automatically runs the UPGRADE DATABASE command. A database backup from one platform family can only be restored on any system within the same platform family. For Windows operating systems, you can restore a database that was created on DB2 10.1 on a DB2 10.5 database system. For Linux and UNIX operating systems, as long as the endianness (big endian or little endian) of the backup and restore platforms is the same, you can restore backups that were produced on down level versions. The target system must have the same (or later) version of the DB2 database product as the source system. You cannot restore a backup that was created on one version of the database product to a system that is running an earlier version of the database product. For example, you can restore a DB2 10.1 on a DB2 10.5 database system, but you cannot restore a DB2 10.5 backup on a DB2 10.1 database system. Database configuration parameters are set to their defaults if the values in the backup are outside of the allowable range for the environment in which the database is being restored.

You can restore a database from a backup image that was taken on a 32-bit level into a 64-bit level, but not vice versa. The DB2 backup and restore utilities should be used to back up and restore your databases. Moving a file set from one machine to another is not recommended as this can compromise the integrity of the database. In situations where certain backup and restore combinations are not allowed, you can move tables between DB2 databases using other methods:

- The db2move command
- The export command followed by the import or the load command

REF: Upgrading to a new DB2 10.5 Server (2)

- If log file size is still not large enough, the following error is returned:
 - SQL1704N Database upgrade failed. Reason code "3".
You must increase the log file size and attempt to upgrade the database again.
 - After database upgrade completed, reset size of log files to pre-upgrade values.
- Optional: Configure new server to use new resources available to calculate buffer pool sizes, and DBM and DB CFG values. Run command to display only recommended values:
**db2 CONNECT TO sampledb2 AUTOCONFIGURE
USING MEM_PERCENT 80 WORKLOAD_TYPE complex
NUM_STMTS 1 TPM 73 ADMIN_PRIORITY performance IS_POPULATED YES
NUM_REMOTE_APPS 15 ISOLATION CS APPLY NONE;**
- If choose not to run and apply, manually configure DB2 server to use new resources.
- Restore any external routines that backed up in pre-upgrade tasks.
- Verify database upgrade successful. Connect to upgraded databases and issue query:
cd samplefile-dir-clp
db2 connect to sample

db2 -tvf testdata.db2
where samplefile-dir-clp is DB2DIR/samples/clp on Linux and UNIX and DB2DIR\samples\clp on Windows, DB2DIR represents the location specified during DB210.1 installation, and sample is database name.
- After upgrading DB2 server, perform recommended post-upgrade tasks such as resetting diagnostic error level, adjusting log space size, and rebinding packages. In addition, verify that upgrade of DB2 server was successful.

103

If you want to upgrade to a new DB2 10.5 server, re-create your instances and then upgrade your databases by restoring a pre-DB2 10.5 database backup. After restoring the database backup, the RESTORE DATABASE command automatically runs the UPGRADE DATABASE command.

REF: Upgrading server using online backups from previous release

- Perform this task only under the following conditions:
 - If cannot upgrade existing instances and databases.
 - If did not perform full *offline* database backups recently or incremental *offline* database backups as indicated in Pre-upgrade tasks for DB2 servers.
- Transfer pre-DB2 10.5 online database backup files for all databases that want to upgrade to DB2 server.
- If have DB2 copy with same version as online database backups, install DB2 copy of same version. If performed online database backups from DB2 10.1 copy, must have DB2 10.1 copy installed on DB2 server.
- If do not have instance running on DB2 copy with same version as online backups, create instance under DB2 copy.
- As SYSADM, rebuild databases by:
**RESTORE DB *db-name* REBUILD WITH ALL TABLESPACES IN DATABASE
TAKEN AT *timestamp-backup*;
ROLLFORWARD DB *db-name* TO END OF LOGS AND STOP;**
- Can choose to rebuild database with just subset of table spaces, but must drop all table spaces in restore pending state after issue ROLLFORWARD DATABASE.
- Verify databases that rebuild are in consistent state by issuing GET DB CFG:
GET DB CFG FOR sample | FIND "consistent"

104

You can rebuild your database on a previous release by using online database backups from the same release and then upgrade to DB2 10.5.

Before upgrading your DB2 server, ensure that you have root access on Linux and UNIX operating systems or Local Administrator authority on Windows. All necessary full or incremental online pre-DB2 10.5 database backups of your databases so that you can rebuild your databases by using these online backups.

Perform this task only under the following conditions:

- If you cannot upgrade the existing instances and databases.
- If you did not perform full *offline* database backups recently or incremental *offline* database backups as indicated in Pre-upgrade tasks for DB2 servers.

REF: Post-upgrade tasks for DB2 servers (1)

- If reset diaglevel to 3 or higher as recommended in pre-upgrade tasks, reset to value before upgrade.
- Existing pre-DB2 10.5 tables with row compression enabled have classic row compression enabled. If want to use adaptive compression, alter tables.
- If changed log space setting as recommended in pre-upgrade tasks, reset logfilsiz, logprimary, and logsecond to pre-upgrade values.
- Ensure existing libraries for external routines remain on original location before upgrade.
- Activate database to start database and all necessary database services.
- Automatic storage table spaces inherit media attribute values, including overhead, device read rate and data tag attributes, from the storage group it is using by default. After upgrading to DB2 10.5, existing table spaces retain their settings and OVERHEAD and DEVICE READ RATE attributes for the storage group are set to undefined. Can set media attributes with the ALTER STOGROUP statement. .
- Consider setting new registry variables and configuration parameters that can impact behavior of DB2.
- If automatic collection of statistics failed on certain system catalog tables during database upgrade, update statistics on these tables.
- If did not use REBINDALL on UPGRADE DATABASE then rebind packages to validate packages and use updated statistics or new index information.

105

After upgrading DB2 servers, you should perform several post-upgrade tasks to ensure that your DB2 servers perform as expected and at their optimum level.

Perform the above post-upgrade tasks that apply to your DB2 server.

REF: Post-upgrade tasks for DB2 servers (2)

- Refresh data in existing MQTs by using REFRESH TABLE. MQTs on unicode databases using language aware collation and involves LIKE predicate or substring function involved in a basic predicate, need to be refreshed.
- Migrate Explain Tables to retain explain information previously gathered.
- If obtained customized code page conversion tables from DB2 support service, copy all files for those tables from the *DB2101or97/conv* to *DB2105/conv*. Do not need to copy standard code page conversion tables.
- If upgraded existing DB2 10.1 or 9.7 on Windows, can restore customized code page conversion tables that backed up as part of pre-upgrade tasks.
- Upgrade existing target tables for event monitors that write to tables and to unformatted event (UE) tables by using new EVMON_UPGRADE_TABLES procedure.
- Test applications and tools to ensure DB2 server is working as expected.
- Back up your databases after DB2 server upgrade is complete.
- If recoverable databases, UPGRADE DATABASE renamed all log files in active log path using .MIG extension. After verifying database upgrade successful and backing up databases, can delete S*.MIG files.
- Must migrate SQL Replication in order to support new LSN formats.

106

After upgrading DB2 servers, you should perform several post-upgrade tasks to ensure that your DB2 servers perform as expected and at their optimum level.

Perform the above post-upgrade tasks that apply to your DB2 server.

If you obtained customized code page conversion tables from the DB2 support service, you need to backup all of the files in the DB2OLD\conv directory where DB2OLD is the location of your existing pre-10.5 DB2 copy. You do not need to backup standard code page conversion tables. Upgrading your pre-10.5 DB2 copy removes these tables because standard code page tables are contained in a DB2 10.5 library.

REF: Post upgrade - What to do next?

- If upgraded existing DB2 9.7 copy, database log directories are changed. Review db2diag.log file which has entries detailing new log directories. If user defined log directory is used, after upgrade the location of log files will be /usr/logpath/NODE0000/LOGSTREAM0000. Old log directory will only contain renamed log files. If default database directory is used, /home/db2user/db2inst/NODE0000/SQL00001/SQLOGDIR, after upgrade location of log files will be /home/db2user/db2inst/NODE0000/SQL00001/LOGSTREAM0000. Old log directory will only contain renamed log files.
- If upgrade DB2 server running HADR, initialize HADR replication. During upgrade to DB2 10.5, database role is changed from primary to standard. Upgrade of standby databases is not supported.
- When DB2 server performance is stable, take advantage of optimizer improvements and collect statistics for new functionality by updating statistics for your upgraded databases. During database upgrade to DB2 10.5, statistics collected from existing database tables retain their values. Statistics for new characteristics on tables and indexes have a value of -1 to indicate there is no information gathered. Only need these statistics if using new functionality.
- After updating statistics for upgraded databases, determine if index or table reorganization is necessary by running REORGCHK. Table and index reorganization can help improve performance.
- At this point, resume all of maintenance activities such as backing up databases and updating statistics. Remove any DB2 10.1, 9.7 or 9.8 copies that no longer need.

107

Perform the above post-upgrade tasks that apply to your DB2 database products or add-on features.

HADR Changes

- As DB2 10.5, **including environments with one standby database**, always set **HADR_TARGET_LIST** as part of initializing HADR databases. Initialization of HADR without setting HADR_TARGET_LIST is deprecated.
- HADR log spooling now enabled by default. **HADR_SPOOL_LIMIT** has default of AUTOMATIC for new databases. For upgraded databases, value of `hadr_spool_limit` is not changed.
- HADR now supported in DB2 pureScale.
 - In DB2 pureScale environments, must set HADR_TARGET_LIST.

For details see:

http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.admin.ha.doc/doc/t0011725.html?lang=en

108

DB2 pureScale environments now support DB2 high availability disaster recovery (HADR). This enhancement combines the continuous availability of the DB2 pureScale Feature with the robust disaster recovery capabilities of HADR.

HADR log spooling is now enabled by default. The `hadr_spool_limit` database configuration parameter, which specifies log spooling behavior, now has a default setting of AUTOMATIC. For DB2 pureScale databases, the `hadr_spool_limit` parameter is set to AUTOMATIC. Because HADR was previously not supported with the DB2 pureScale Feature, this change has no impact.

For all other DB2 databases, the value of the `hadr_spool_limit` parameter is not changed. Introduced in 10.1, HADR log spooling allows transactions on the primary to make progress without having to wait for the log replay on the standby. Testing has shown that spooling can significantly reduce any impact that HADR might have on the primary's workload but has a negligible impact on a standby's replay performance.

As of Version 10.5, the steps for initializing your HADR databases have changed. You should now use the `hadr_target_list` configuration parameter even if you are only configuring one standby. The `hadr_target_list` database configuration parameter was introduced in Version 10.1 to configure multiple HADR standby databases. You could use this parameter to specify up to three standby databases. This parameter was not required for HADR setups that had only one standby database.

As of Version 10.5, in DB2 environments that do not use the DB2 pureScale Feature, you should always set the `hadr_target_list` database configuration parameter as part of initializing your HADR databases, including environments with one standby database. Initialization of HADR without setting the `hadr_target_list` database configuration parameter is deprecated and might be discontinued in a future release. In DB2 pureScale environments, you must set the `hadr_target_list` database configuration parameter to initialize HADR.

If you are creating a new HADR setup, perform all the steps in the initializing HADR task, including setting the `hadr_target_list` database configuration parameter. If your HADR is already set up using the old method (that is, the `hadr_target_list` parameter is empty), see:

<http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/topic/com.ibm.db2.luw.admin.ha.doc/doc/t0060257.html>

REF: HADR (High Availability Disaster Recovery)

- Consider **logbufsz, logfilsiz, logprimary logsecond, locklist & maxlock** changes
- Backup upgraded Primary Database
- Restore on Standby
- Reinitialize HADR

Primary db2 stop hadr on db dbx

**Standby db2 deactivate database dbx
db2 stop hadr on db dbx**

**Primary UPGRADE INSTANCE & DATABASE
db2 backup db dbx online to ...**

Standby db2iupgrade -u <fencedid> <standbyinstance>

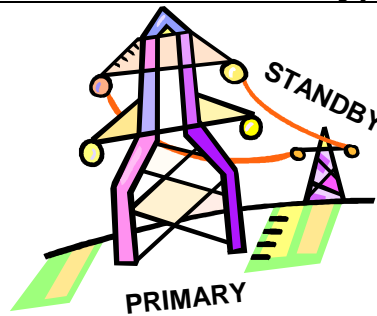
move log files from previous DB2 version out of log path of new DB2 version

db2 restore db dbx from ...

db2 start hadr on db dbx as standby

Primary db2 start hadr on db dbx as primary

- See notes: Upgrade Procedure for TSAMP Automated HADR



109

Upgrading a primary database to DB2 10.5 changes the database role from primary to standard. Upgrading standby databases to DB2 10.5 is not supported because these databases are in roll forward pending state. Because of these restrictions, upgrading an HADR environment to DB2 10.5 requires that you stop HADR, upgrade your DB2 server where the primary database resides, and then re-initialize HADR. The primary database is taken out of HADR (its status is changed from “primary” to “standard”).

Before you activate your standby database, you must move the log files from the previous version of DB2 out of the log path of the new version of DB2. If you do not move the log files, DB2 might try to use the old files and fail to initialize.

Rolling Upgrade Procedure for TSAMP Automated HADR environment resources:

- White Paper:
<http://www-01.ibm.com/support/docview.wss?uid=swg21586363>
- Information Center:
<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.admin.ha.doc/doc/t0011766.html>

REF: Adjusting log space size if upgrade from DB2 9.7

- Set appropriate size for log files since is one of important factors in tuning server.
- If increased log files sizes as pre-upgrade task, can restore additional free space to server.
- On partitioned database, must adjust log space size on catalog database partition.
- Restore log file size settings to values before upgrade:
db2 UPDATE DB CFG FOR sample using LOGSECOND *previous-value*
- If you enabled infinite active logging, disable it by running the following commands:
db2 UPDATE DB CFG FOR sample using LOGARCHMETH1 *previous-value*
db2 UPDATE DB CFG FOR sample using LOGSECOND *previous-value*
- To support larger log record headers, increase log space setting, by approximately 10% - 15% over what used for DB2 9.7.
- To support larger log record headers, increase logbufsz and softmax by 10% - 15% over what used for DB2 9.7.
- logfilsiz changes take effect only when database is reactivated.



110

Adjust the log space size in upgraded databases as needed.

REF: Checklist: Best practices for upgrading DB2 servers

- Review changes in existing DB2 database product functionality
- Perform hardware and operating system upgrades before DB2 database product upgrade
- Benchmark DB2 server performance
- Devise a plan to reverse an upgrade
- Perform pre-upgrade tasks
- Determine whether to upgrade DB2 servers or clients first
- Upgrade database applications and routines
- Upgrading DB2 High Availability Disaster Recovery (HADR) environments
 - only primary upgrade supported
- Migrating SQL replication environments – See URL in notes.
- Upgrading DB2 Spatial Extender – See URL in notes.
- Upgrading Microsoft Cluster Server environments – See URL in notes.
- Upgrading from Query Patroller to Workload Manager - See URL in notes.
 - Query Patroller is discontinued

111

When planning your DB2 server upgrade, there are a number of best practices to consider. Review these best practices before you start your upgrade. After upgrading your database servers, you can optionally migrate your SQL replication environment to DB2 Version 10.5. See:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/index.jsp?topic=/com.ibm.swg.im.iis.db.repl.sqlmig.doc/topics/iyrsmigluw10ovu.html>

Upgrading DB2 Spatial Extender requires that you upgrade your DB2 server first and then upgrade specific database objects and data in spatially enabled databases. Upgrading from 32-bit DB2 Spatial Extender Version 10.1 or Version 9.7 to 64-bit DB2 Spatial Extender V10.5 requires that you upgrade your DB2 server to 64-bit DB2 V10.5 and then upgrade specific database objects and data in spatially enabled databases. See:

<http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/topic/com.ibm.db2.luw.spatial.topics.doc/doc/db2sb26.html>

In a Microsoft Cluster Server (MSCS) environment, install DB2 10.5 as a new copy and then run the db2iupgrade command to upgrade the MSCS instance. See:

Query Patroller is discontinued. See Migrating from Query Patroller to DB2 workload manager for details on how to migrate.

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.qb.upgrade.doc/doc/t0053461.html>

IBM Database Conversion Workbench (DCW)



- **IBM Database Conversion Workbench (DCW)** is no-charge plug-in that adds database migration capabilities to IBM Data Studio.
- DCW can help you
 - move your Oracle databases to DB2 for LUW and DB2 for z/OS in no time
 - move existing DB2 databases from different versions to other DB2 environments, including IBM PureData System for Transactions or DB2 pureScale.
- [Database Conversion Workbench 2.0](#) (IBM Data Studio 4.1 or 3.2)
 - Download URL in student notes

112

As of July 12th, 2013, DCW 2.0 is now available as a no-charge plug-in that adds database migration capabilities to IBM Data Studio.

For more details see:

<https://www.ibm.com/developerworks/community/groups/service/html/communit>

New Course - DB2 10.5 for LUW New Features and Database Upgrade Considerations

- Course Title: **DB2 10.5 for LUW New Features and Database Upgrade Considerations**

Course Code: **CL314** (classroom)
3L314 (ILO- Instructor Led Online)



- This course can be customized to meet your needs.
- Check Out IBM Education Offerings at:
<http://www.ibm.com/software/data/education/>

113

Course Code CL314 provides training on DB2 10.5 for LUW New Features and Database Upgrade Considerations.
See the above URL for more details.

DB2 10.5 Additional Information



Technical Talks:

<http://www-01.ibm.com/software/data/db2/tech-talk.html>

DB2 10.5 with BLU Acceleration - New Dynamic In-Memory Analytics for the Era of Big Data

<http://bit.ly/DB2BLUFlashbook>

Download additional information about DB2 10.5 for free from the above urls.

DB2 10.5 for LUW Certification Exam

- **IBM Certified Database Administrator – DB2 10.5 DBA for LUW Upgrade from DB2 10.1**

Test 000-311: IBM DB2 10.5 DBA for LUW Upgrade from DB2 10.1



For more details:

<http://www-03.ibm.com/certify/tests/ovrC2090-311.shtml>

- *DB2 10.1/10.5 for Linux, UNIX, and Windows Database Administration: Certification Study Guide*

115

DB2 10.5 for Linux, Unix and Windows Upgrade certification examination, preparation suggestions and sample assessment tests are available.

If you are knowledgeable with DB2 10.5 and are capable of performing the intermediate to advanced skills required in the day-to-day administration of DB2 instances and databases, you may benefit from this certification role.

A resource for Exam 311 (DB2 10.5 DBA for LUW Upgrade from 10.1) is the InfoCenter. Consider cross-referencing the sections/objectives listed at <http://pic.dhe.ibm.com/infocenter/db2luw/v10r5/index.jsp?topic=%2Fcom.ibm.db2.luw.welcome.doc%2Fdoc%2Fwelcome.html>

There is one physical book for preparation of both exams: DB2 10.1/10.5 for Linux, UNIX, and Windows Database Administration: Certification Study Guide

You can order from Amazon at: http://www.amazon.com/10-1-Linux-Windows-Database-Administration/dp/1583473750/ref=sr_1_7?ie=UTF8&qid=1390947785&sr=8-7&keywords=db2+10+certification

The web source for Exam 611 (DB2 10.1 DBA for LUW) is an excellent preparation source: http://public.dhe.ibm.com/ps/products/db2/info/vr101/pdf/en_US/db2cga.pdf

There are also free tutorials for Exam 611:

http://www.ibm.com/developerworks/views/data/libraryview.jsp?sort_order=1&sort_by=Title&series_title_by=db2+10.1+dba+certification+exam

Summary

- Describe changes to DB2 10.5 packaging and software and operating system requirements.
- Understand DB2 10.5 changes to DDL, utilities, tools and monitoring.
- Configure a database and instance to implement DB2 10.5 new features.
- Analyze DB2 10.5 changes that affect applications and database runtime behavior.
- Develop a DB2 10.5 upgrade strategy



Abstract:

Are you responsible for upgrading your database to DB2 10.5? Take advantage of columnar data store integrated into the core DB2 engine which has huge benefits for Business Intelligence workloads. DB2 10.5 is not just BLU. There is additional new great functionality. Learn the entire story! If you are looking for a detailed step by step approach to implement DB2 10.5, attend this presentation to learn about often overlooked but very important tips and tricks to understand before upgrading. Planning your upgrade is much easier if you know these best practices.



DB2 Information Management
Software

Best Practices Upgrade to DB2 LUW 10.5 with BLU Acceleration

Melanie Stopfer
IBM SWG

Email: mstopfer@us.ibm.com
Twitter: [mstopfer1](#)
LinkedIn: [Melanie Stopfer](#)



5.4

Melanie Stopfer is a Consulting Learning Specialist and Developer for IBM Software Group. She is recognized worldwide as an advanced database specialist. As a Certified DB2 LUW 10.5 Technical Expert and Learning Facilitation Specialist, she has provided in-depth technical support to IM customers specializing recovery, performance and database upgrade and migration best practices since 1988. In 2009, Melanie was the first DB2 LUW speaker to be inducted into the IDUG Speaker Hall of Fame and was again selected Best Overall Speaker at IDUG EMEA 2011 and 2012 and IDUG NA 2012 and 2014. In 2013, IBM IOD rated her presentations in top two for customer satisfaction.