Performing Db2 HADR Upgrades Made Easy

Michael Roecken

IBM

Session code: D17

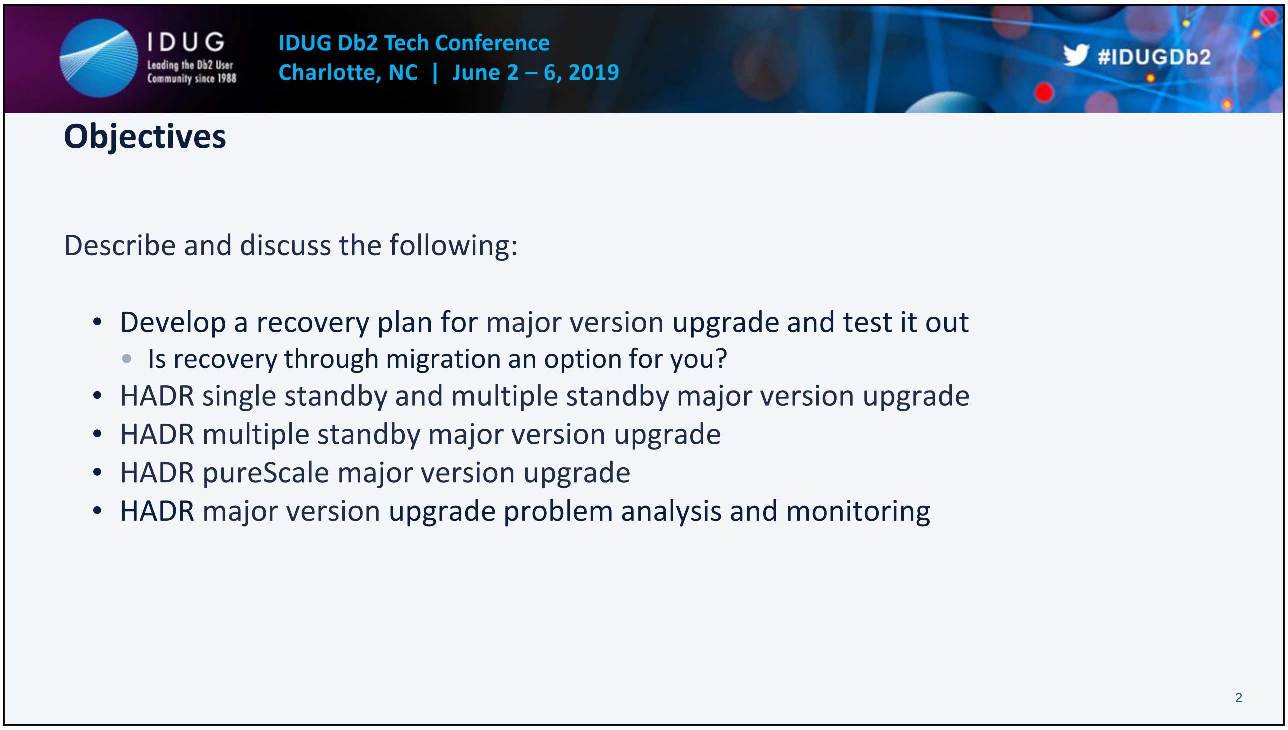
June 6, 2019, 12:00pm – 1:00pm

Platform: Db2 For Linux, UNIX, Windows

IDUG

Leading the Db2 User Community since 1988

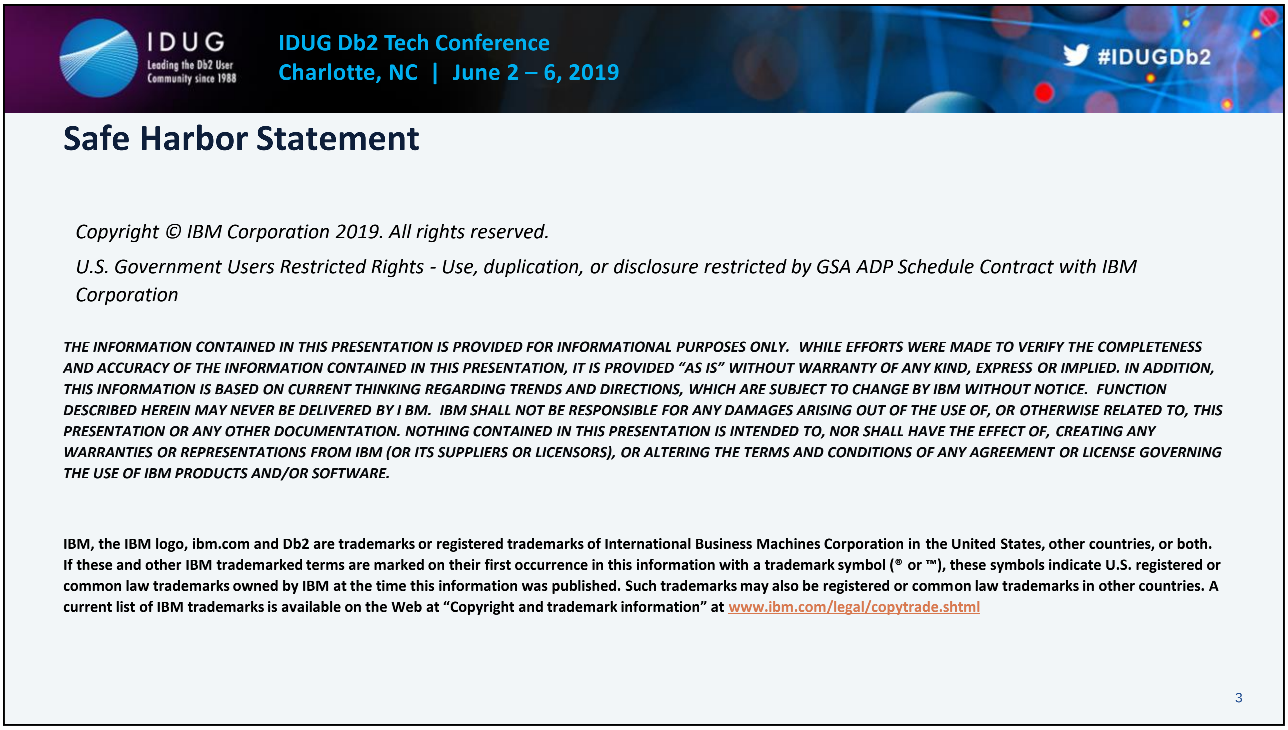
Have no fear Db2 11.5 is here. Moving to a new version of Db2 should not be a scary event for databases using the high availability disaster recovery (HADR) feature. Fear of an outage or re-initialization of your standby is no longer a concern. This presentation will introduce to you and detail the procedures to perform a major release upgrade of your HADR single standby, HADR multiple standby and HADR pureScale databases. A detailed step by step analysis, with examples, from start to end so that you can get your database to the latest version of Db2 with the least amount of concern.



Objectives

Describe and discuss the following:

- Develop a recovery plan for major version upgrade and test it out
  - Is recovery through migration an option for you?
- HADR single standby and multiple standby major version upgrade
- HADR multiple standby major version upgrade
- HADR pureScale major version upgrade
- HADR major version upgrade problem analysis and monitoring



IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

Safe Harbor Statement

U.S. Government Users Restricted Rights - Use, duplication, or disclosure restricted by GSA ADP Schedule Contract with IBM Corporation

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON CURRENT THINKING REGARDING TRENDS AND DIRECTIONS, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. FUNCTION DESCRIBED HEREIN MAY NEVER BE DELIVERED BY I BM. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS AND/OR SOFTWARE.

IBM, the IBM logo, ibm.com and Db2 are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml

The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

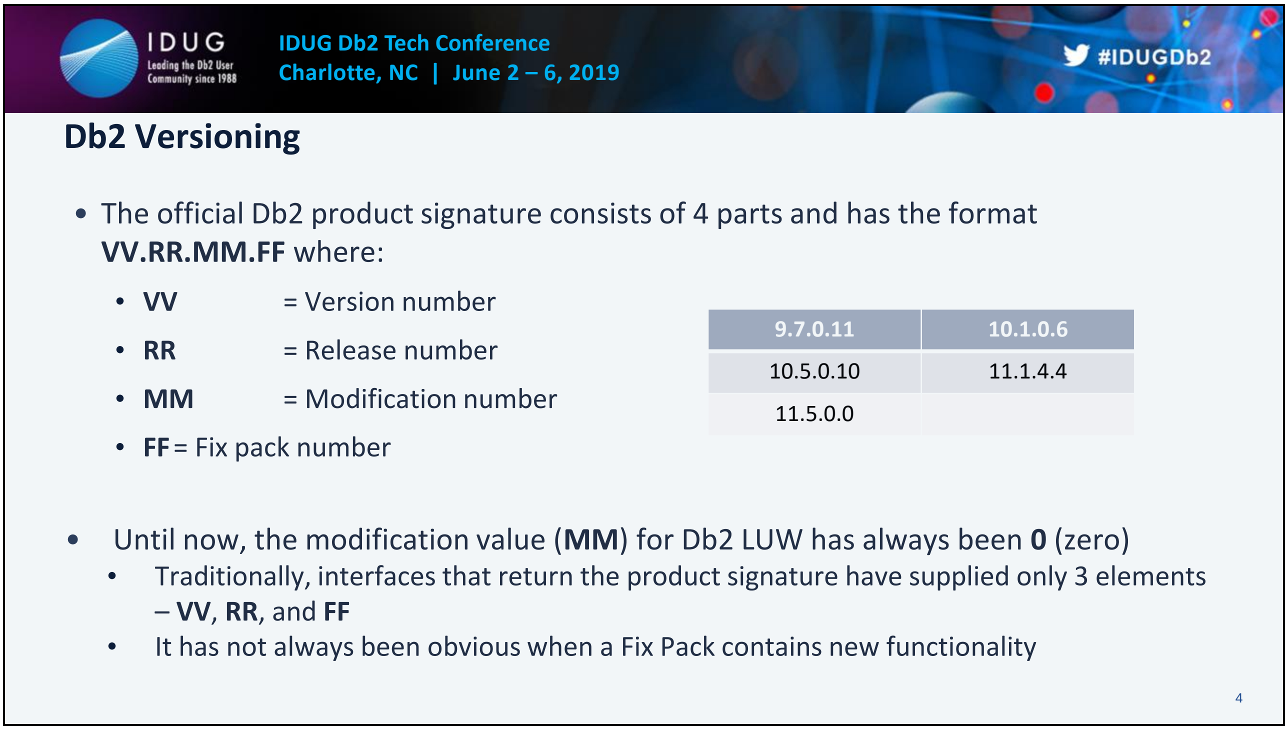
References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

IBM, and the IBM logo, are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

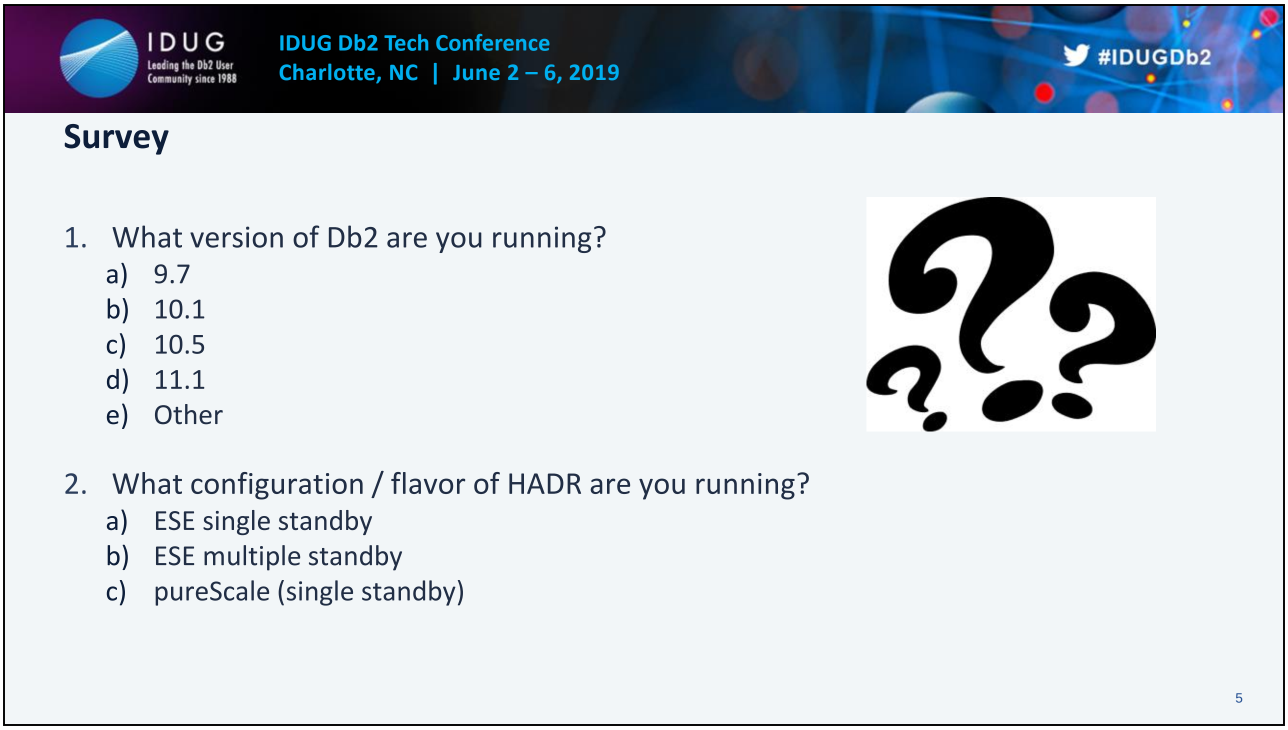


Db2 Versioning

- The official Db2 product signature consists of 4 parts and has the format **VV.RR.MM.FF** where:
  - **VV** = Version number
  - **RR** = Release number
  - **MM** = Modification number
  - **FF** = Fix pack number

9.7.0.11	10.1.0.6
10.5.0.10	11.1.4.4
11.5.0.0	

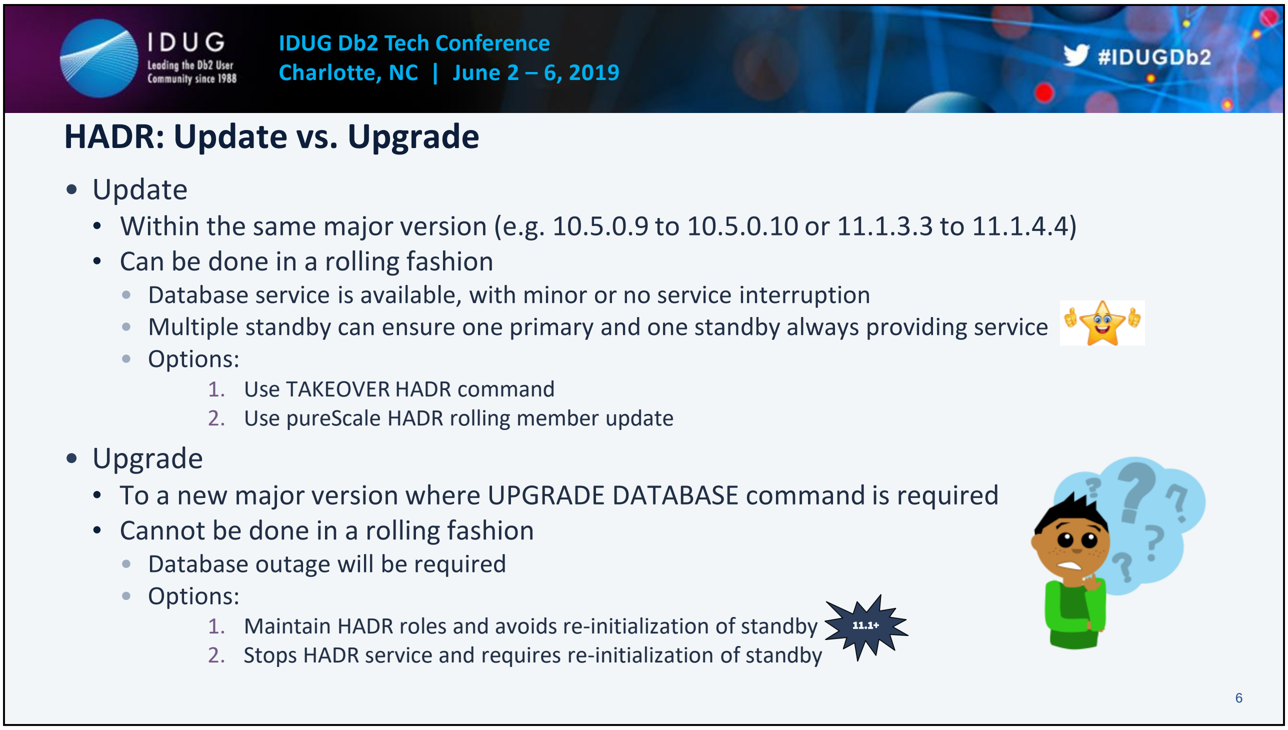
- Until now, the modification value (**MM**) for Db2 LUW has always been **0** (zero)
  - Traditionally, interfaces that return the product signature have supplied only 3 elements – **VV**, **RR**, and **FF**
  - It has not always been obvious when a Fix Pack contains new functionality



Survey

1. What version of Db2 are you running?
   a) 9.7
   b) 10.1
   c) 10.5
   d) 11.1
   e) Other

2. What configuration / flavor of HADR are you running?
   a) ESE single standby
   b) ESE multiple standby
   c) pureScale (single standby)



IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR: Update vs. Upgrade

- Update
  - Within the same major version (e.g. 10.5.0.9 to 10.5.0.10 or 11.1.3.3 to 11.1.4.4)
  - Can be done in a rolling fashion
    - Database service is available, with minor or no service interruption
    - Multiple standby can ensure one primary and one standby always providing service
    - Options:
      1. Use TAKEOVER HADR command
      2. Use pureScale HADR rolling member update
- Upgrade
  - To a new major version where UPGRADE DATABASE command is required
  - Cannot be done in a rolling fashion
    - Database outage will be required
    - Options:
      1. Maintain HADR roles and avoids re-initialization of standby 11.1+
      2. Stops HADR service and requires re-initialization of standby

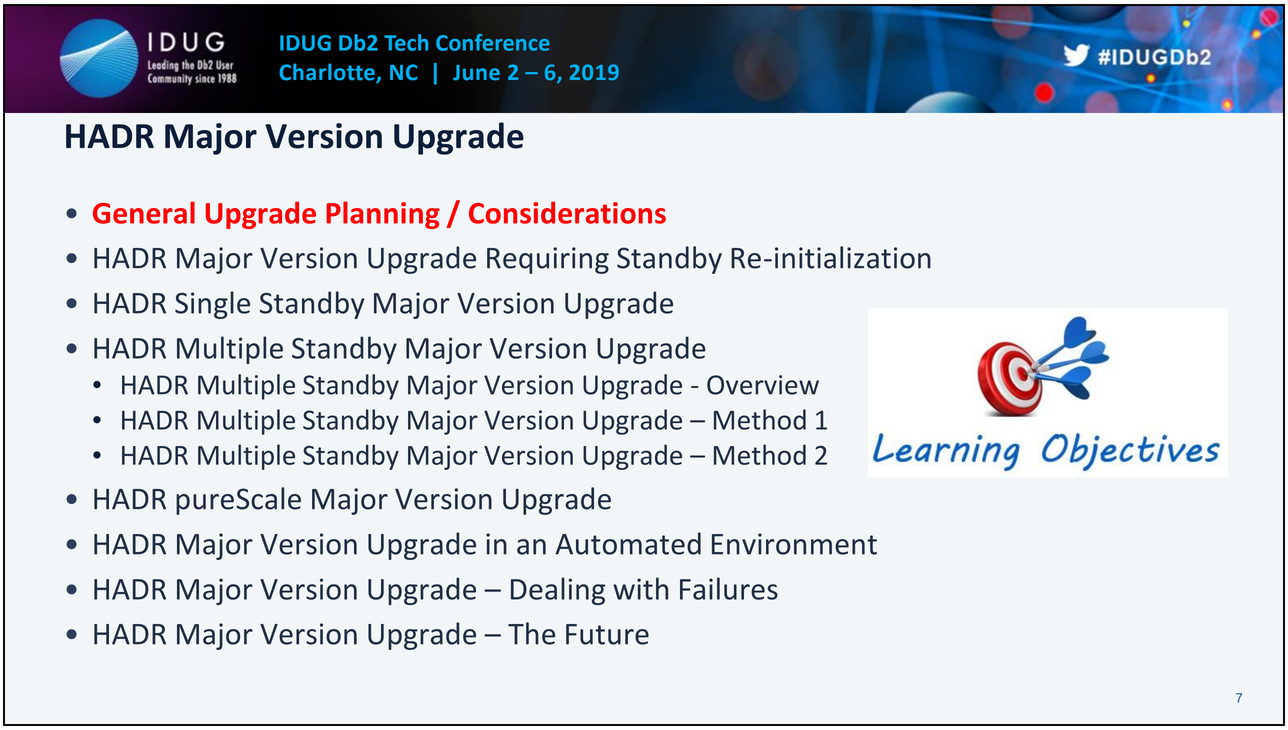
6

Update

https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.admin.ha.doc/doc/t0011766.html

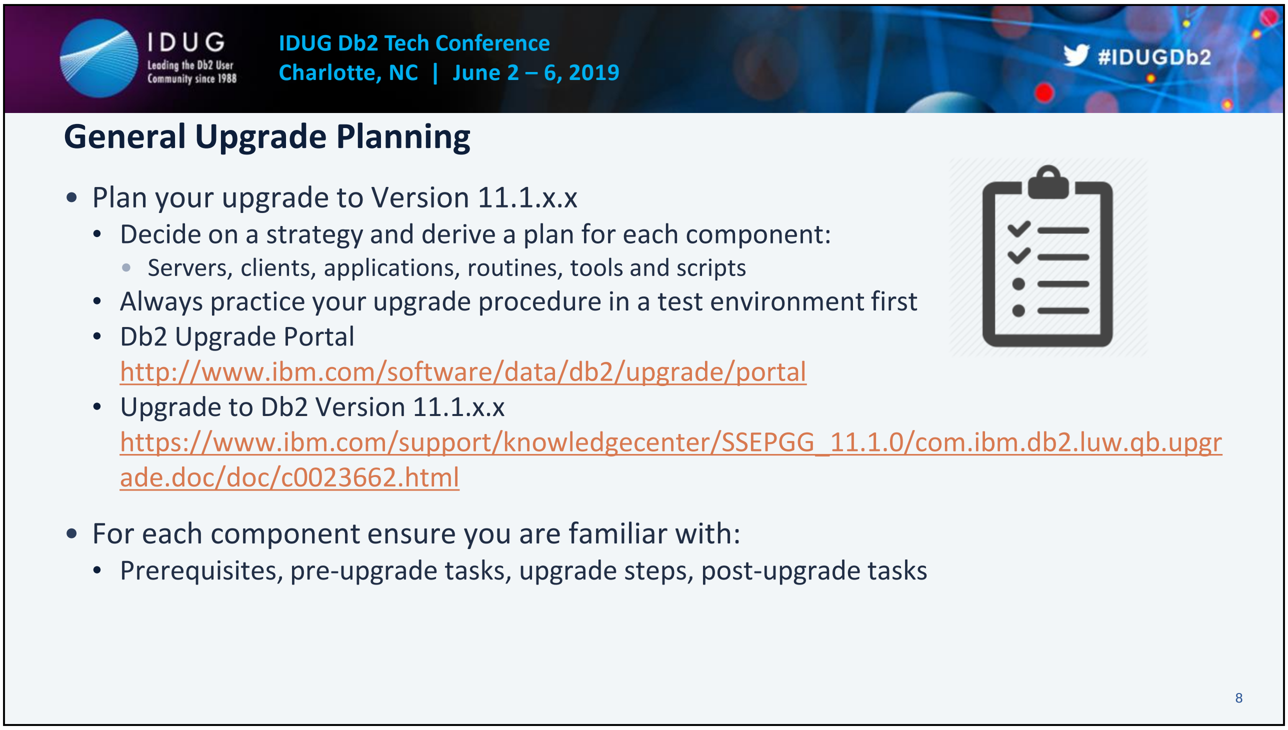
Upgrade

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/c0070028.html



HADR Major Version Upgrade

- **General Upgrade Planning / Considerations**
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future



General Upgrade Planning

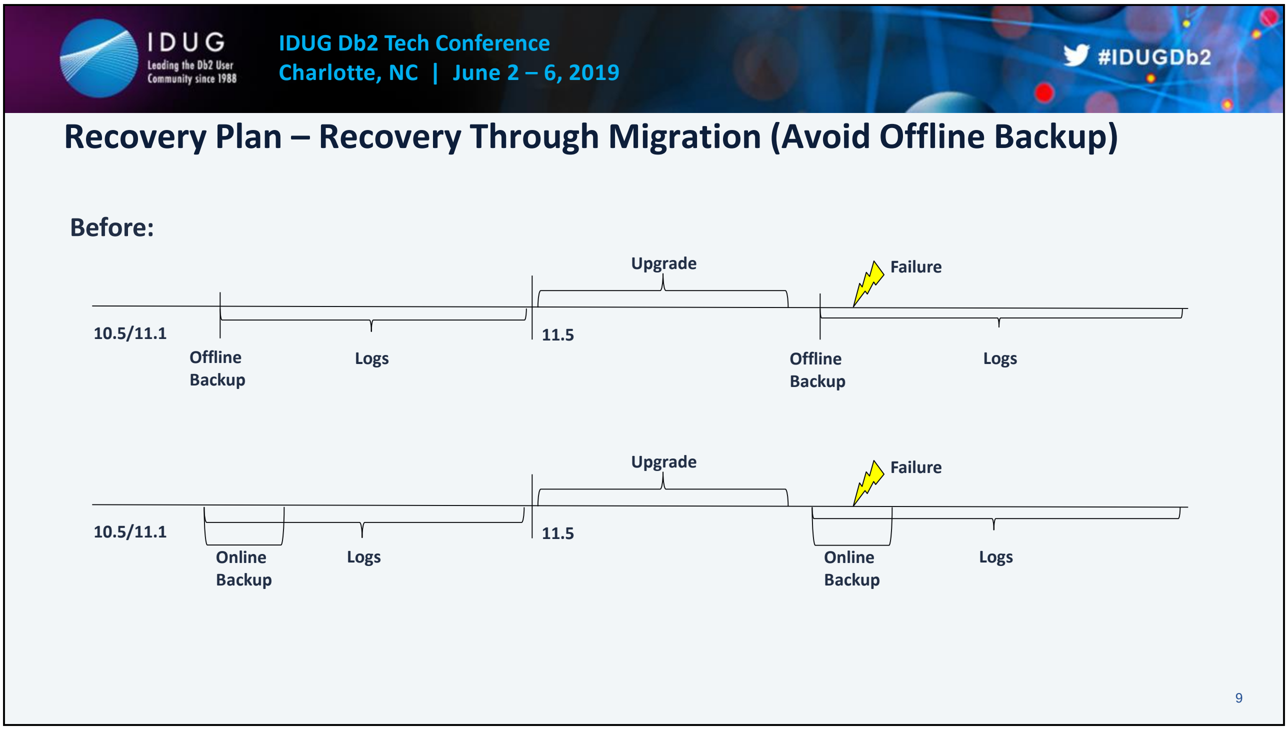
- Plan your upgrade to Version 11.1.x.x
  - Decide on a strategy and derive a plan for each component:
    - Servers, clients, applications, routines, tools and scripts
  - Always practice your upgrade procedure in a test environment first
  - Db2 Upgrade Portal
    http://www.ibm.com/software/data/db2/upgrade/portal
  - Upgrade to Db2 Version 11.1.x.x
    https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/c0023662.html
- For each component ensure you are familiar with:
  - Prerequisites, pre-upgrade tasks, upgrade steps, post-upgrade tasks

Upgrading a Db2 server (Windows)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007199.html

Upgrading a Db2 server (Linux and UNIX)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007200.html



IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

Recovery Plan – Recovery Through Migration (Avoid Offline Backup)

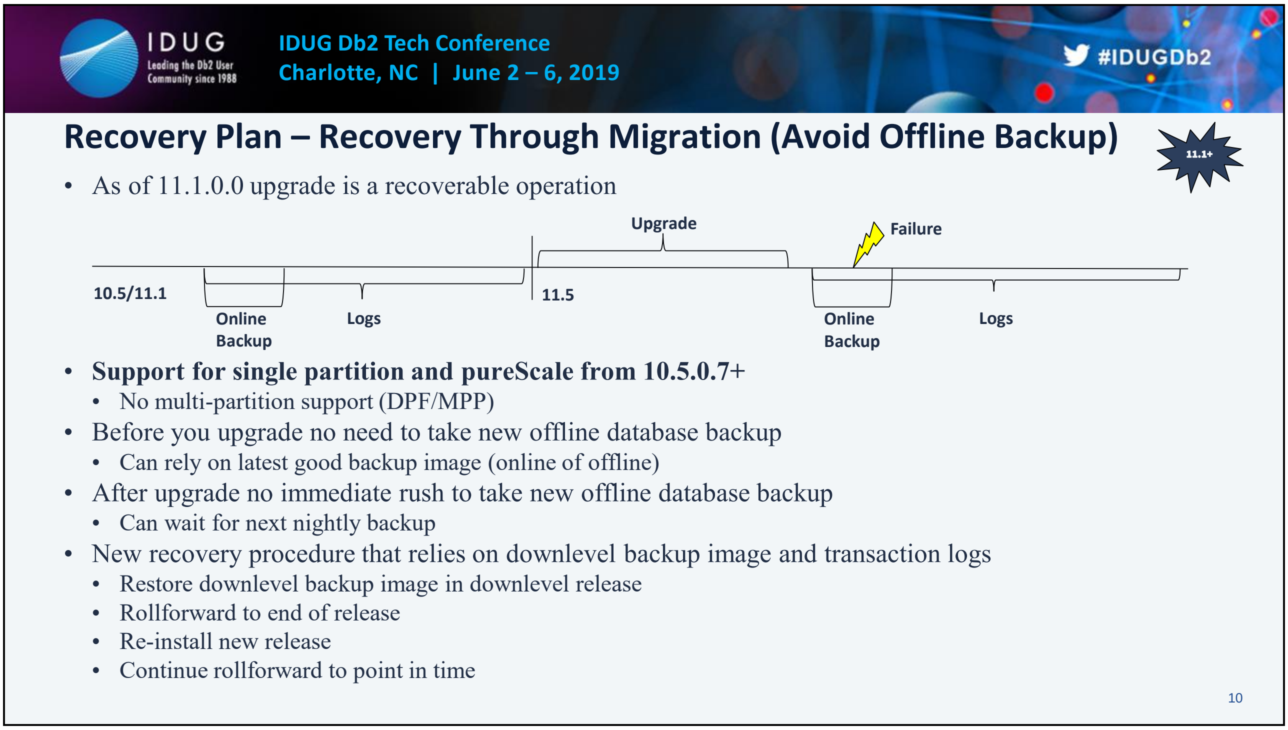
Before:

Upgrading a Db2 server (Windows)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007199.html

Upgrading a Db2 server (Linux and UNIX)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007200.html



IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

Recovery Plan – Recovery Through Migration (Avoid Offline Backup)

11.1+

- As of 11.1.0.0 upgrade is a recoverable operation

Upgrade
Failure
10.5/11.1
11.5
Online Backup
Logs
Online Backup
Logs

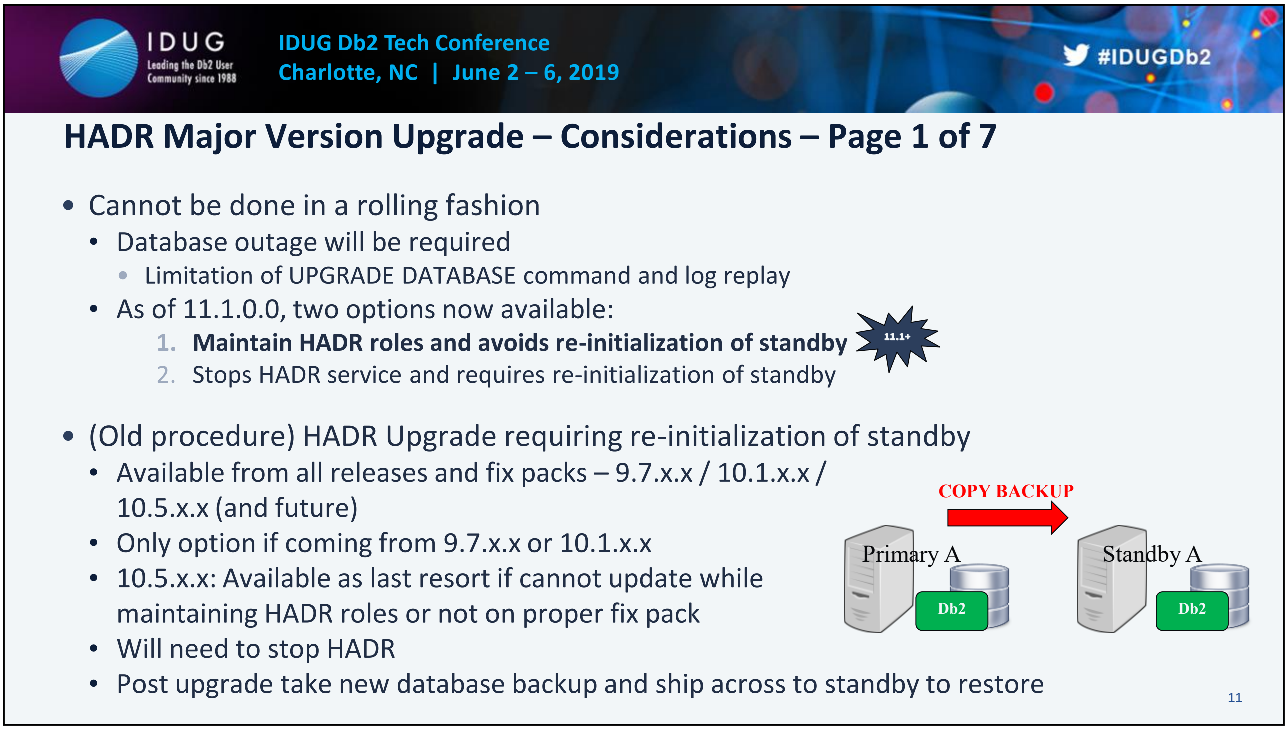
- **Support for single partition and pureScale from 10.5.0.7+**
  - No multi-partition support (DPF/MPP)
- Before you upgrade no need to take new offline database backup
  - Can rely on latest good backup image (online of offline)
- After upgrade no immediate rush to take new offline database backup
  - Can wait for next nightly backup
- New recovery procedure that relies on downlevel backup image and transaction logs
  - Restore downlevel backup image in downlevel release
  - Rollforward to end of release
  - Re-install new release
  - Continue rollforward to point in time

Upgrading a Db2 server (Windows)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007199.html

Upgrading a Db2 server (Linux and UNIX)

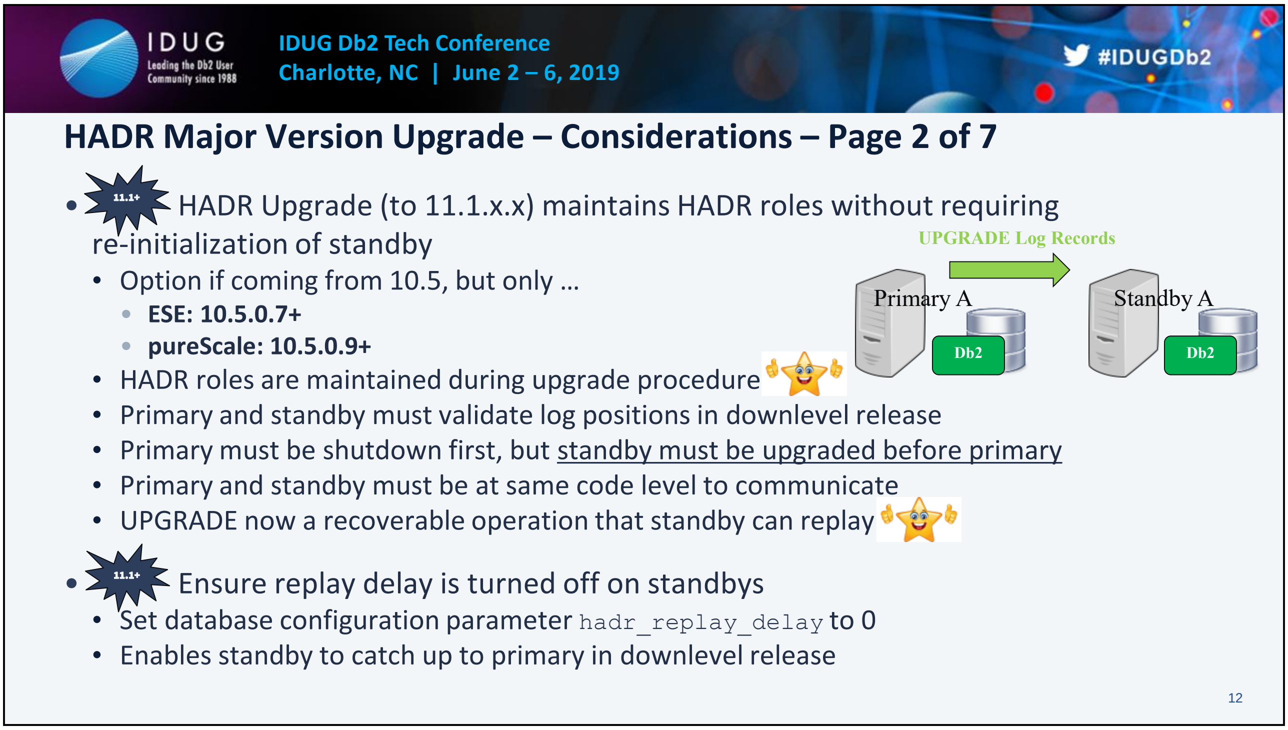
https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0007200.html



HADR Major Version Upgrade – Considerations – Page 1 of 7

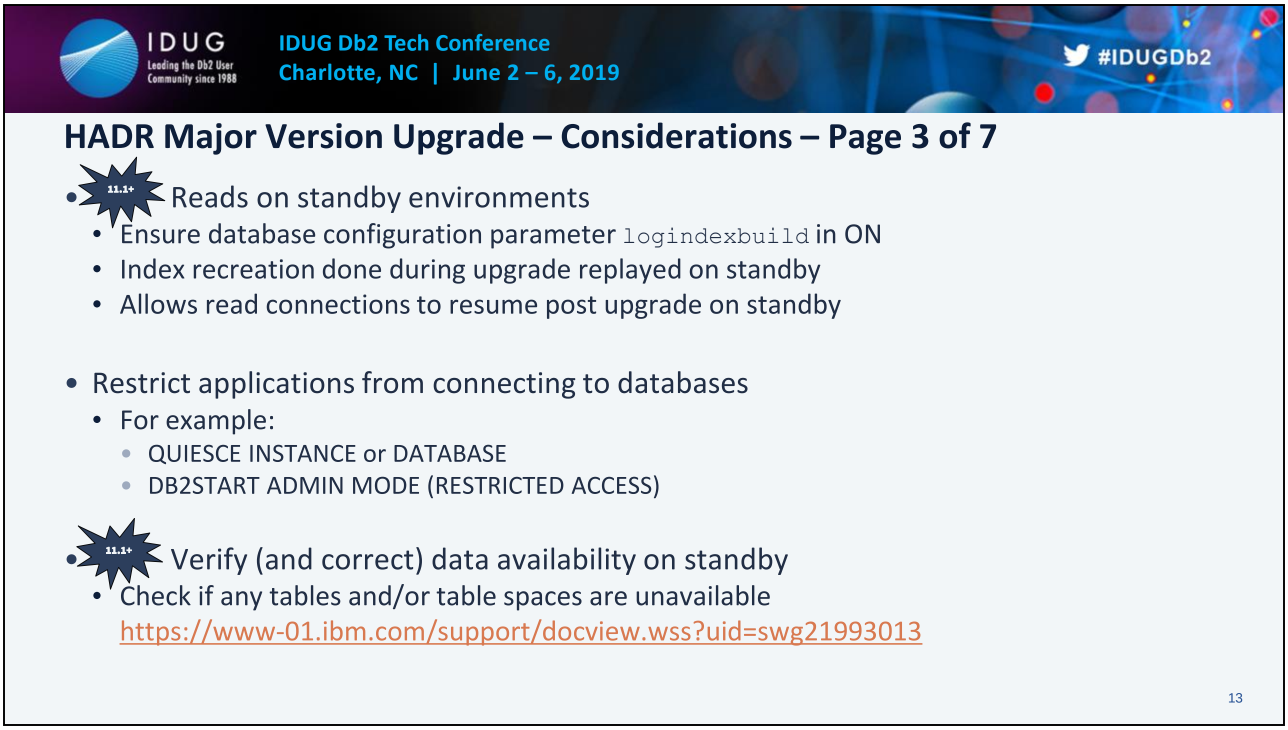
- Cannot be done in a rolling fashion
  - Database outage will be required
    - Limitation of UPGRADE DATABASE command and log replay
  - As of 11.1.0.0, two options now available:
    1. **Maintain HADR roles and avoids re-initialization of standby** 11.1+
    2. Stops HADR service and requires re-initialization of standby

- (Old procedure) HADR Upgrade requiring re-initialization of standby
  - Available from all releases and fix packs – 9.7.x.x / 10.1.x.x / 10.5.x.x (and future)
  - Only option if coming from 9.7.x.x or 10.1.x.x
  - 10.5.x.x: Available as last resort if cannot update while maintaining HADR roles or not on proper fix pack
  - Will need to stop HADR
  - Post upgrade take new database backup and ship across to standby to restore



HADR Major Version Upgrade – Considerations – Page 2 of 7

- 11.1+ HADR Upgrade (to 11.1.x.x) maintains HADR roles without requiring re-initialization of standby
  - Option if coming from 10.5, but only …
    - **ESE: 10.5.0.7+**
    - **pureScale: 10.5.0.9+**
  - HADR roles are maintained during upgrade procedure
  - Primary and standby must validate log positions in downlevel release
  - Primary must be shutdown first, but standby must be upgraded before primary
  - Primary and standby must be at same code level to communicate
  - UPGRADE now a recoverable operation that standby can replay
- 11.1+ Ensure replay delay is turned off on standbys
  - Set database configuration parameter `hadr_replay_delay` to 0
  - Enables standby to catch up to primary in downlevel release



IDUG
Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade – Considerations – Page 3 of 7

- 11.1+ Reads on standby environments
  - Ensure database configuration parameter `logindexbuild` in ON
  - Index recreation done during upgrade replayed on standby
  - Allows read connections to resume post upgrade on standby

- Restrict applications from connecting to databases
  - For example:
    - QUIESCE INSTANCE or DATABASE
    - DB2START ADMIN MODE (RESTRICTED ACCESS)

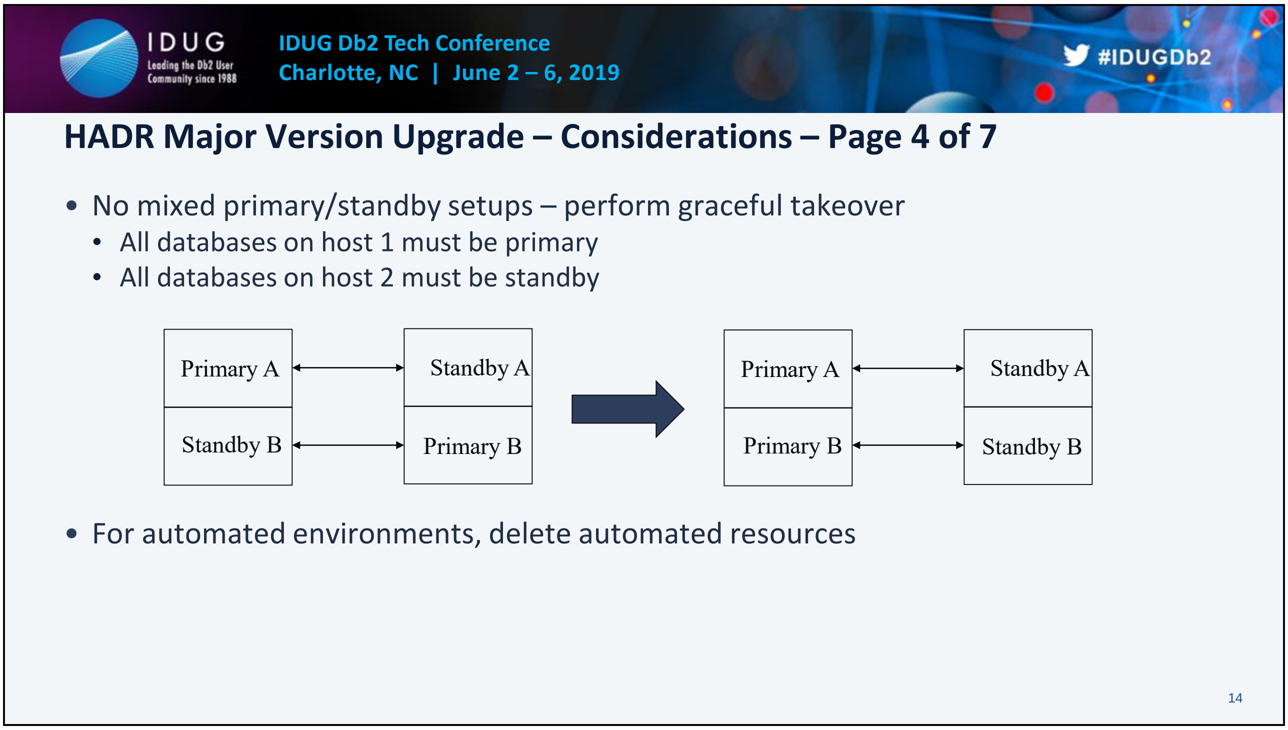
- 11.1+ Verify (and correct) data availability on standby
  - Check if any tables and/or table spaces are unavailable
    https://www-01.ibm.com/support/docview.wss?uid=swg21993013

QUIESCE command

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.admin.cmd.doc/doc/r0008635.html

db2start - Start Db2 command

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.admin.cmd.doc/doc/r0001939.html



IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade – Considerations – Page 4 of 7

- No mixed primary/standby setups – perform graceful takeover
  - All databases on host 1 must be primary
  - All databases on host 2 must be standby

Primary A ↔ Standby A
Standby B ↔ Primary B

→

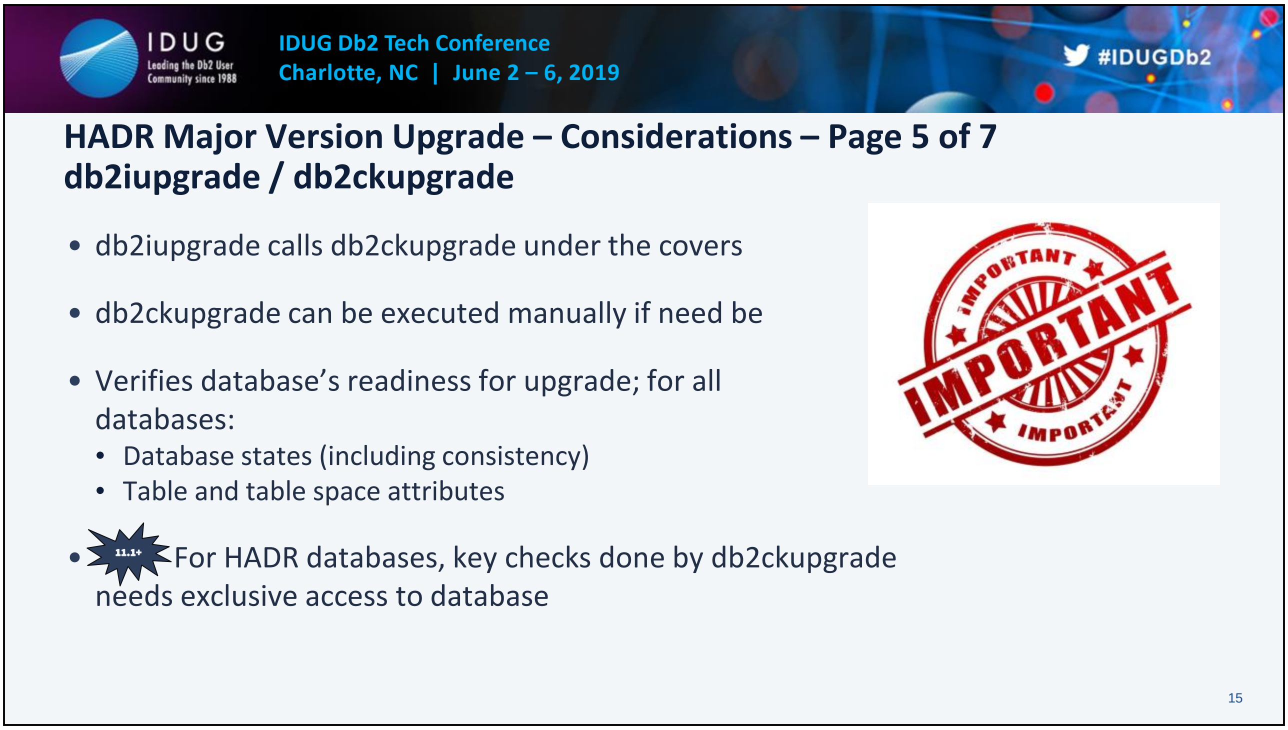
Primary A ↔ Standby A
Primary B ↔ Standby B

- For automated environments, delete automated resources

14

Db2 high availability instance configuration utility (db2haicu)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.admin.ha.doc/doc/c0051371.html


IDUG
Leading the Db2 User Community since 1988
IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019
#IDUGDb2
IMPORTANT

HADR Major Version Upgrade – Considerations – Page 5 of 7 db2iupgrade / db2ckupgrade

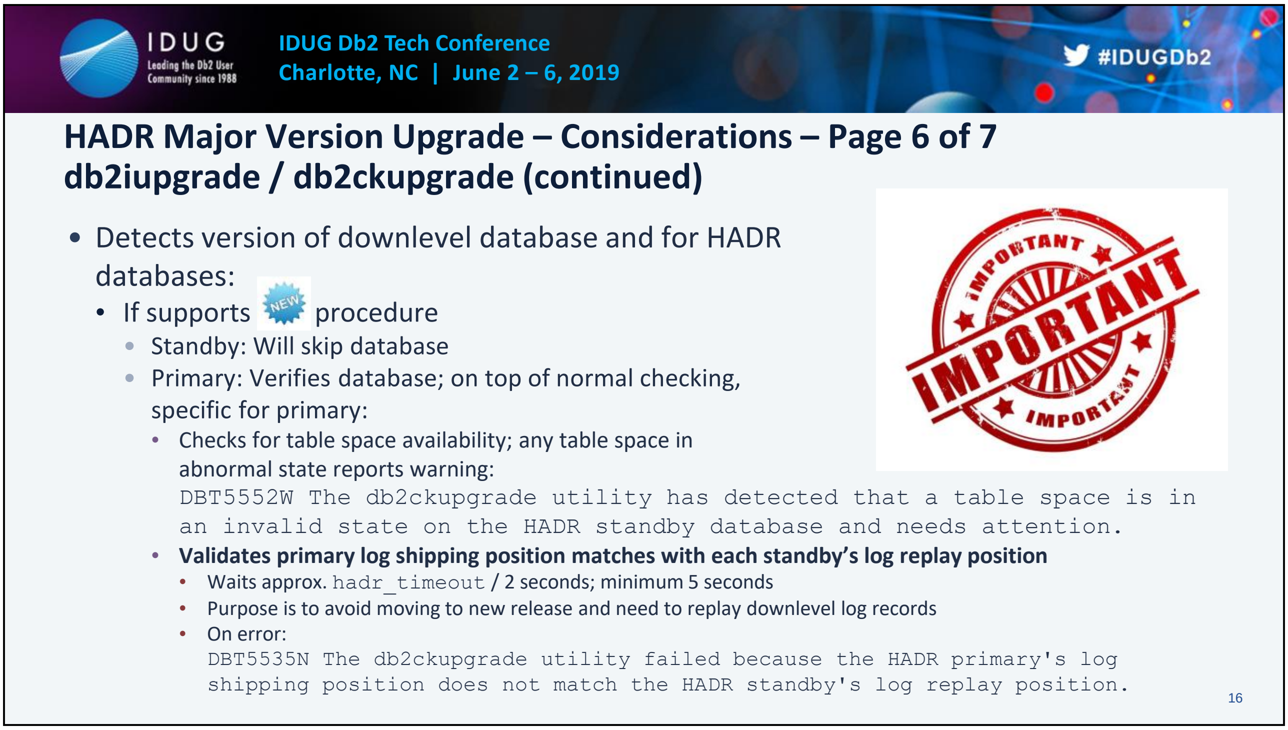
- db2iupgrade calls db2ckupgrade under the covers
- db2ckupgrade can be executed manually if need be
- Verifies database's readiness for upgrade; for all databases:
  - Database states (including consistency)
  - Table and table space attributes
- 11.1+ For HADR databases, key checks done by db2ckupgrade needs exclusive access to database

db2iupgrade - Upgrade instance command

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.admin.cmd.doc/doc/r0002055.html

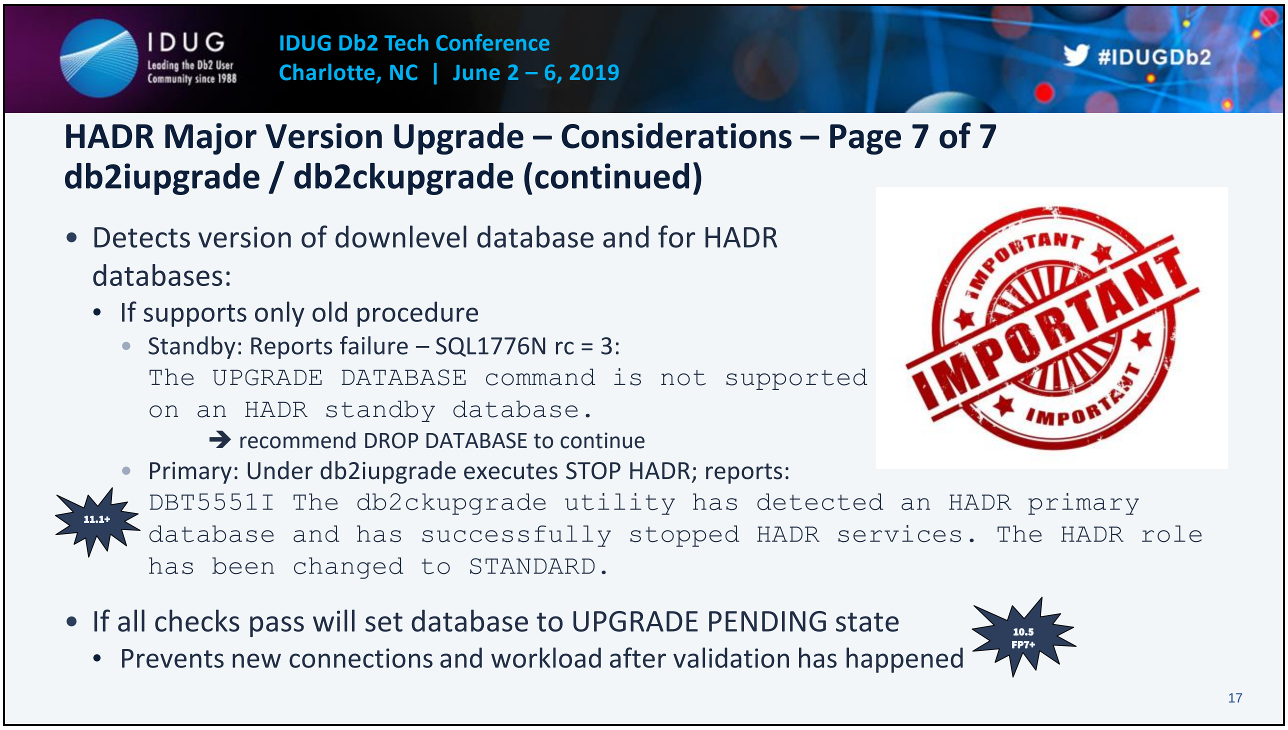
db2ckupgrade - Check database for upgrade command

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.admin.cmd.doc/doc/r0002028.html



HADR Major Version Upgrade – Considerations – Page 6 of 7
db2iupgrade / db2ckupgrade (continued)

- Detects version of downlevel database and for HADR databases:
  - If supports NEW procedure
    - Standby: Will skip database
    - Primary: Verifies database; on top of normal checking, specific for primary:
      - Checks for table space availability; any table space in abnormal state reports warning:
        `DBT5552W The db2ckupgrade utility has detected that a table space is in an invalid state on the HADR standby database and needs attention.`
      - **Validates primary log shipping position matches with each standby's log replay position**
        - Waits approx. `hadr_timeout` / 2 seconds; minimum 5 seconds
        - Purpose is to avoid moving to new release and need to replay downlevel log records
        - On error:
          `DBT5535N The db2ckupgrade utility failed because the HADR primary's log shipping position does not match the HADR standby's log replay position.`



IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade – Considerations – Page 7 of 7
db2iupgrade / db2ckupgrade (continued)

- Detects version of downlevel database and for HADR databases:
  - If supports only old procedure
    - Standby: Reports failure – SQL1776N rc = 3:
      `The UPGRADE DATABASE command is not supported on an HADR standby database.`
      ➔ recommend DROP DATABASE to continue
    - Primary: Under db2iupgrade executes STOP HADR; reports:

11.1+

```
DBT5551I The db2ckupgrade utility has detected an HADR primary
database and has successfully stopped HADR services. The HADR role
has been changed to STANDARD.
```

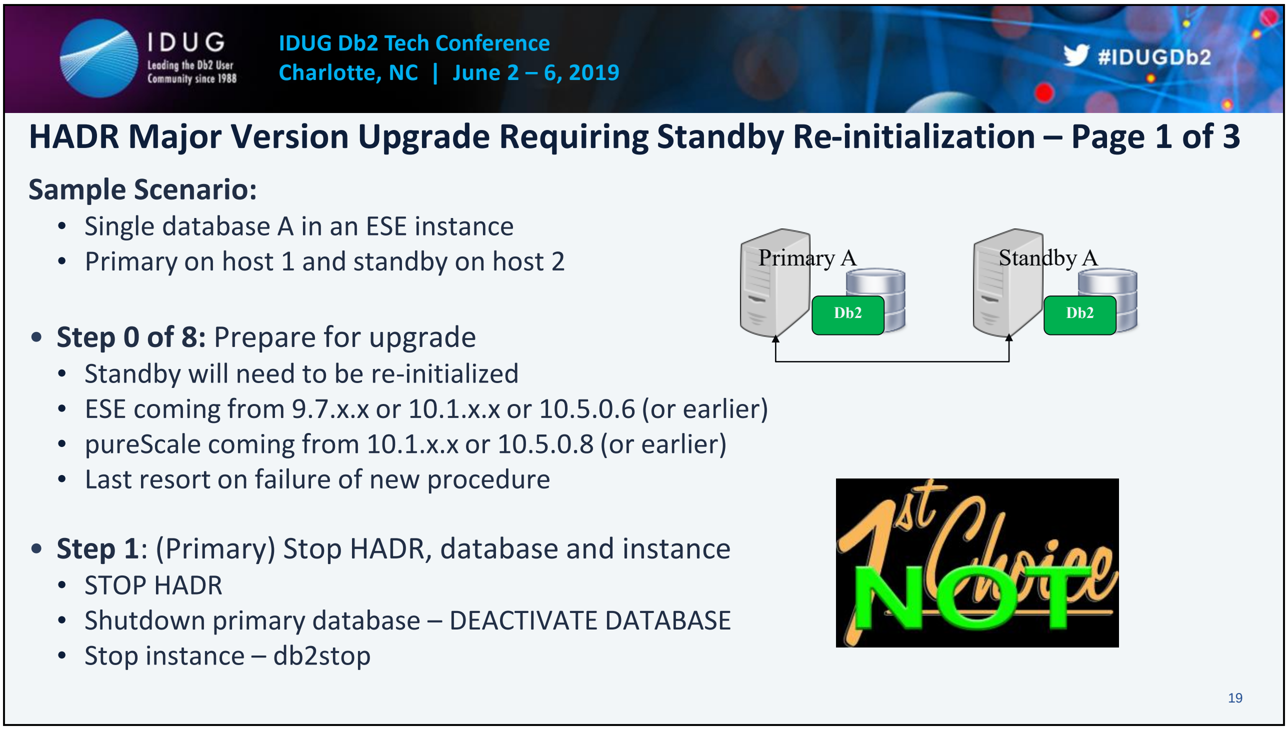
- If all checks pass will set database to UPGRADE PENDING state
  - Prevents new connections and workload after validation has happened

10.5 FP7+

17

HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- **HADR Major Version Upgrade Requiring Standby Re-initialization**
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future


Primary A
Db2
Standby A
Db2
1st Choice
NOT

IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade Requiring Standby Re-initialization – Page 1 of 3

Sample Scenario:

- Single database A in an ESE instance
- Primary on host 1 and standby on host 2

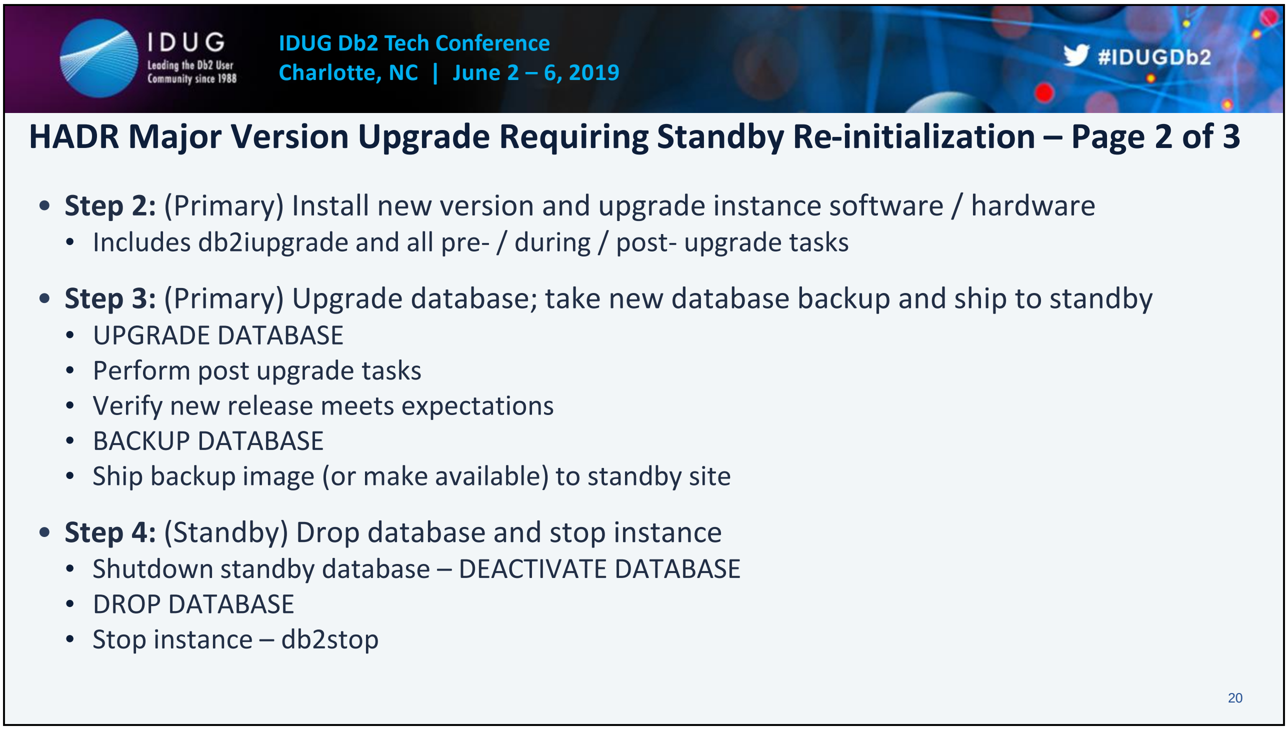
- **Step 0 of 8:** Prepare for upgrade
  - Standby will need to be re-initialized
  - ESE coming from 9.7.x.x or 10.1.x.x or 10.5.0.6 (or earlier)
  - pureScale coming from 10.1.x.x or 10.5.0.8 (or earlier)
  - Last resort on failure of new procedure

- **Step 1**: (Primary) Stop HADR, database and instance
  - STOP HADR
  - Shutdown primary database – DEACTIVATE DATABASE
  - Stop instance – db2stop

19

Upgrading Db2 servers in HADR environments

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070030.html

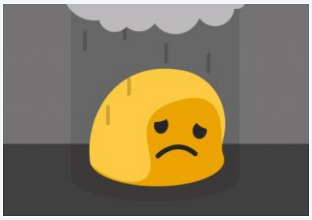


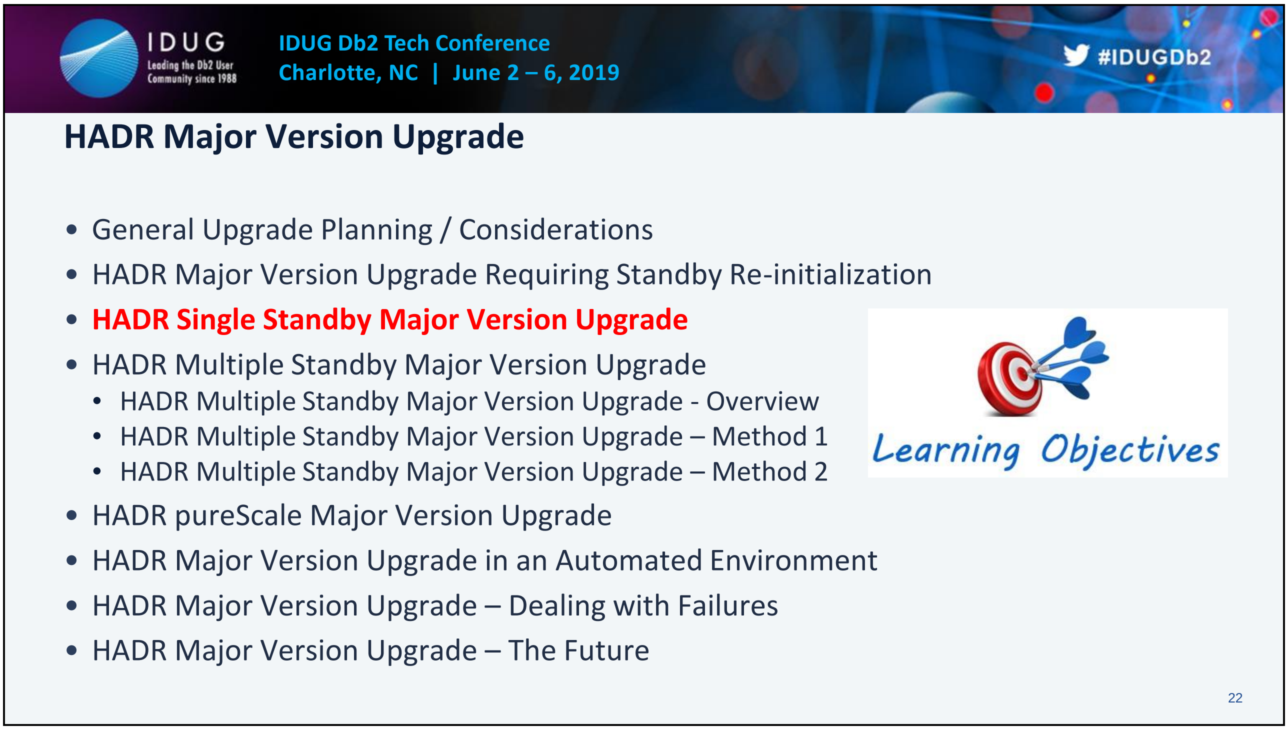
HADR Major Version Upgrade Requiring Standby Re-initialization – Page 2 of 3

- **Step 2:** (Primary) Install new version and upgrade instance software / hardware
  - Includes db2iupgrade and all pre- / during / post- upgrade tasks
- **Step 3:** (Primary) Upgrade database; take new database backup and ship to standby
  - UPGRADE DATABASE
  - Perform post upgrade tasks
  - Verify new release meets expectations
  - BACKUP DATABASE
  - Ship backup image (or make available) to standby site
- **Step 4:** (Standby) Drop database and stop instance
  - Shutdown standby database – DEACTIVATE DATABASE
  - DROP DATABASE
  - Stop instance – db2stop

HADR Major Version Upgrade Requiring Standby Re-initialization – Page 3 of 3

- **Step 5:** (Standby) Install new version and upgrade instance software / hardware
- **Step 6:** (Standby) Restore database backup image from primary
  - RESTORE DATABASE
- **Step 7:** (Standby) Configure and start HADR
  - Set configuration parameters
  - START HADR … AS STANDBY
- **Step 8:** (Primary) Configure and start HADR
  - Set configuration parameters
  - START HADR … AS PRIMARY



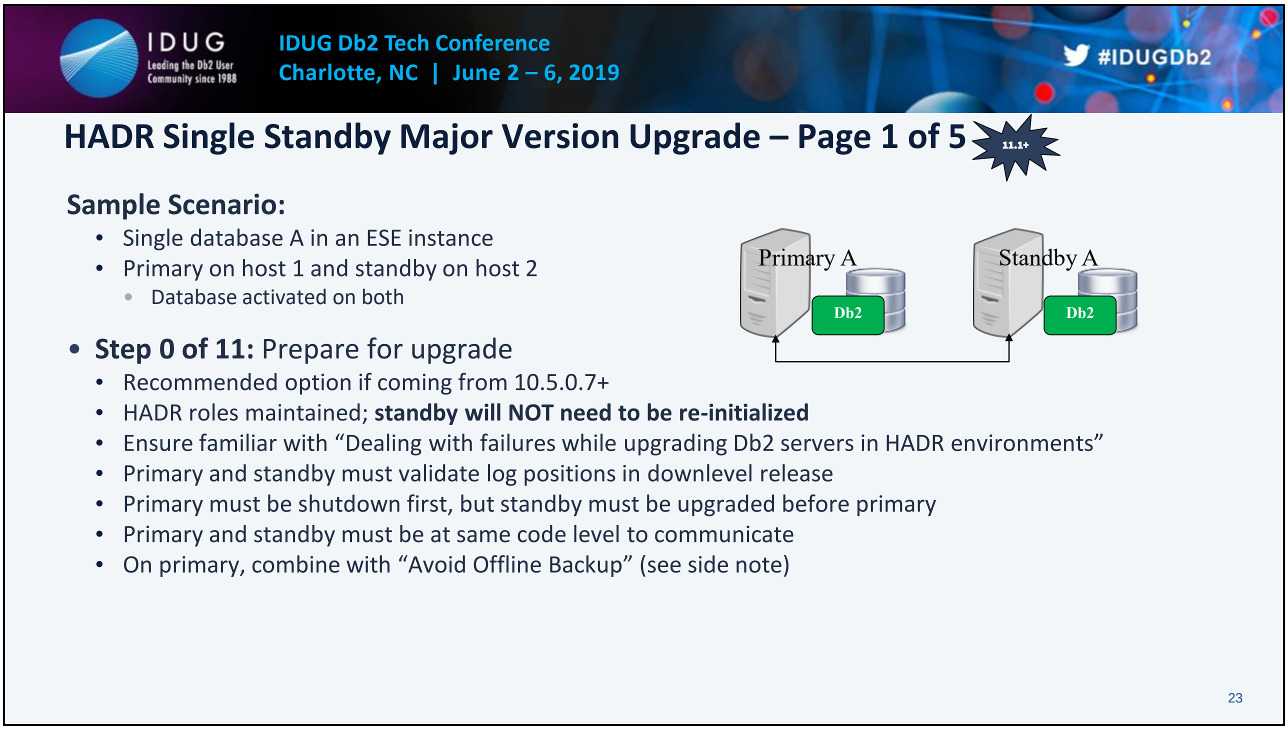


IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- **HADR Single Standby Major Version Upgrade**
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future



HADR Single Standby Major Version Upgrade – Page 1 of 5 11.1+

Sample Scenario:

- Single database A in an ESE instance
- Primary on host 1 and standby on host 2
  - Database activated on both

Primary A
Db2
Standby A
Db2

- **Step 0 of 11:** Prepare for upgrade
  - Recommended option if coming from 10.5.0.7+
  - HADR roles maintained; **standby will NOT need to be re-initialized**
  - Ensure familiar with “Dealing with failures while upgrading Db2 servers in HADR environments”
  - Primary and standby must validate log positions in downlevel release
  - Primary must be shutdown first, but standby must be upgraded before primary
  - Primary and standby must be at same code level to communicate
  - On primary, combine with “Avoid Offline Backup” (see side note)

Upgrading Db2 servers in HADR environments (without standby reinitialization)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070029.html

Recovering through a Db2 server upgrade

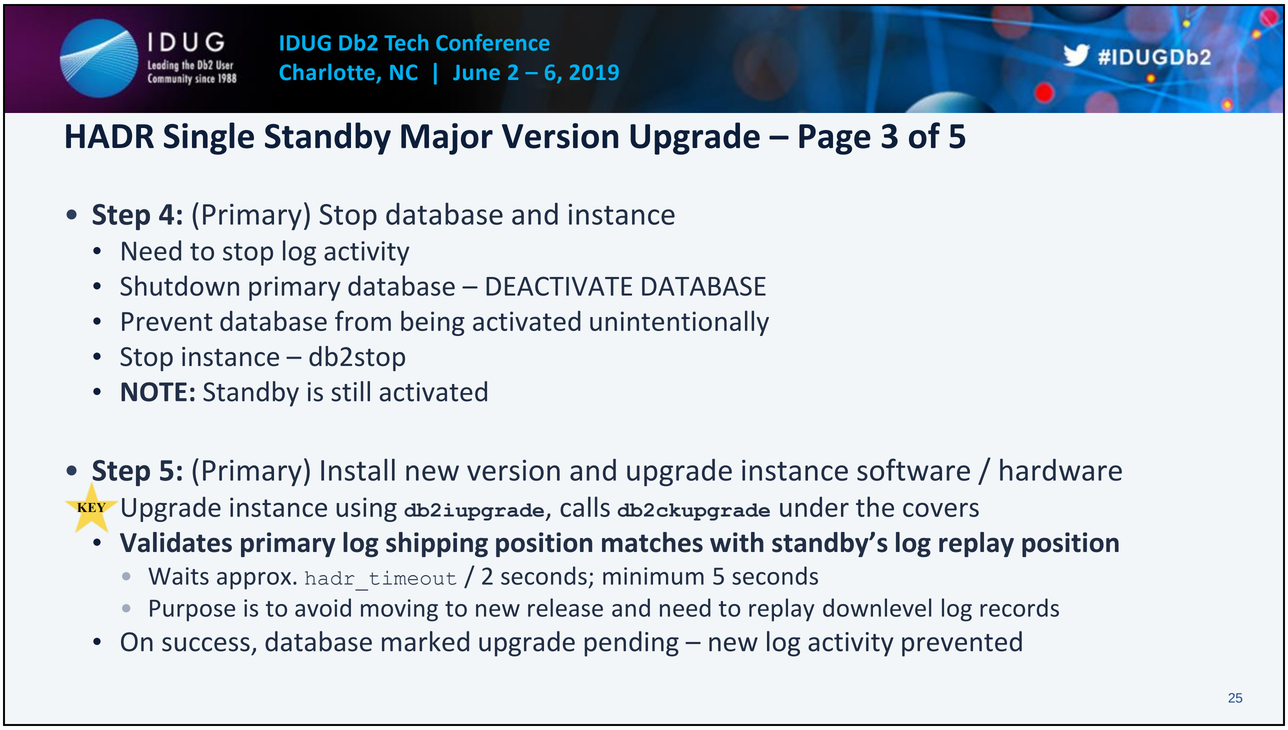
https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070050.html

HADR Single Standby Major Version Upgrade – Page 2 of 5

- **Step 1:** (Primary) [RoS] Turn on `logindexbuild`
  - Index recreation done during upgrade replayed on standby
  - Allows read connections to resume post upgrade on standby

- **Step 2:** (Standby) Turn off `hadr_replay_delay`
  - Set `hadr_replay_delay` to 0
  - Allows standby’s log replay position to catch up to the primary’s log shipping position

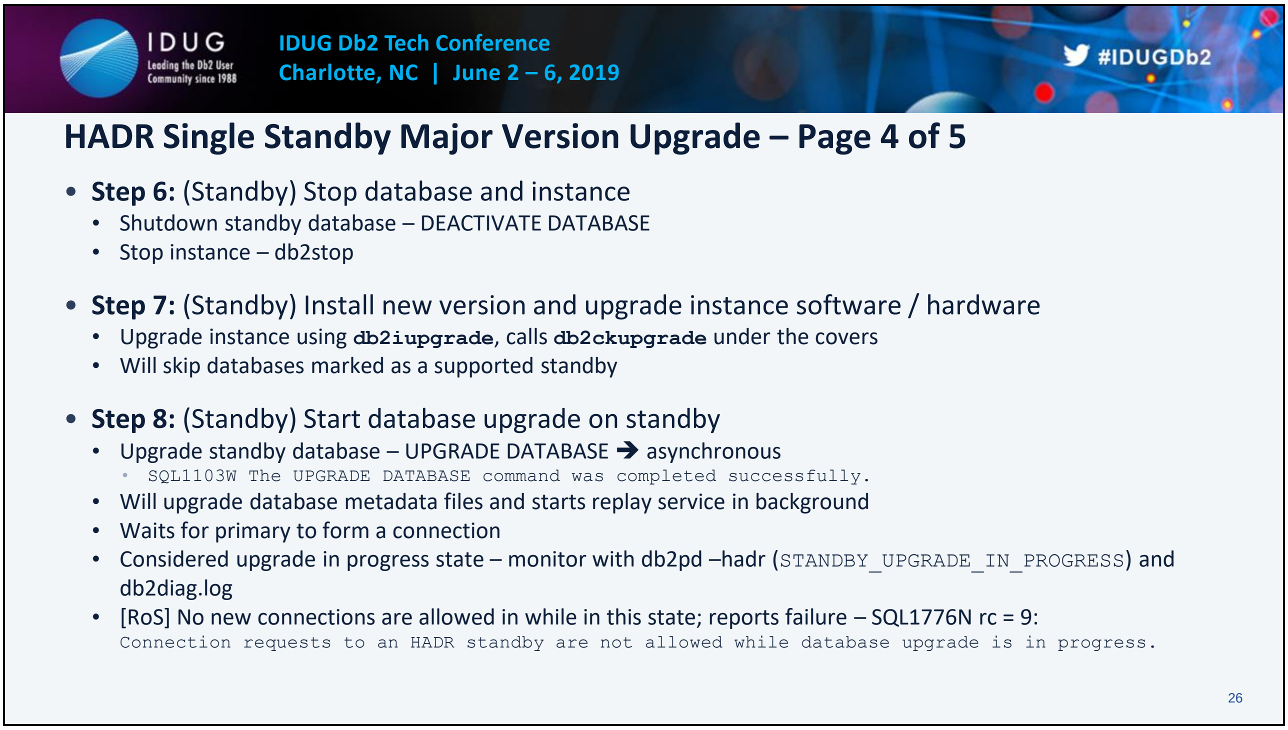
- **Step 3:** (Primary) Monitor log positions
  - Ensure primary log shipping and standby log replay positions are “healthy”
  - Helps to reduce the chance of failures later in the process
  - Use db2pd –hadr or MON_GET_HADR
  - Adjust `hadr_timeout` accordingly to prepare for log position validation



HADR Single Standby Major Version Upgrade – Page 3 of 5

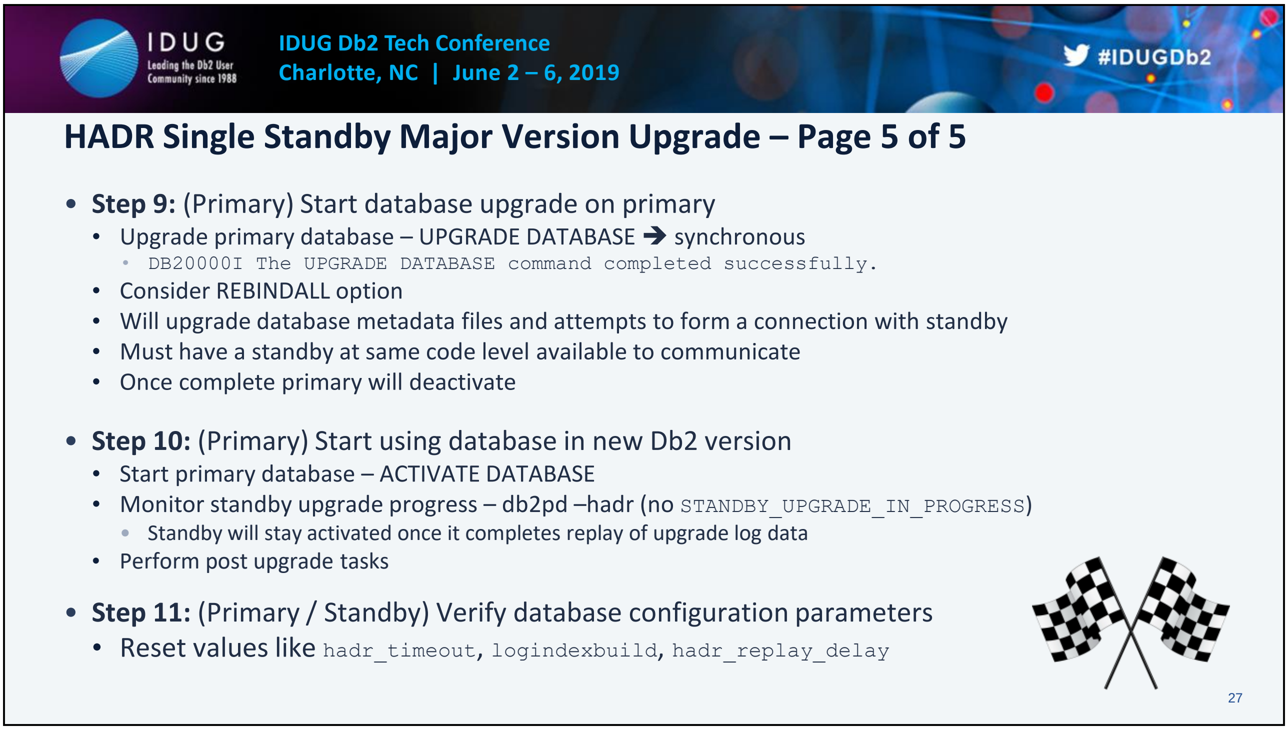
- **Step 4:** (Primary) Stop database and instance
  - Need to stop log activity
  - Shutdown primary database – DEACTIVATE DATABASE
  - Prevent database from being activated unintentionally
  - Stop instance – db2stop
  - **NOTE:** Standby is still activated

- **Step 5:** (Primary) Install new version and upgrade instance software / hardware
  - KEY Upgrade instance using **`db2iupgrade`**, calls **`db2ckupgrade`** under the covers
  - **Validates primary log shipping position matches with standby's log replay position**
    - Waits approx. `hadr_timeout` / 2 seconds; minimum 5 seconds
    - Purpose is to avoid moving to new release and need to replay downlevel log records
  - On success, database marked upgrade pending – new log activity prevented



HADR Single Standby Major Version Upgrade – Page 4 of 5

- **Step 6:** (Standby) Stop database and instance
  - Shutdown standby database – DEACTIVATE DATABASE
  - Stop instance – db2stop
- **Step 7:** (Standby) Install new version and upgrade instance software / hardware
  - Upgrade instance using **`db2iupgrade`**, calls **`db2ckupgrade`** under the covers
  - Will skip databases marked as a supported standby
- **Step 8:** (Standby) Start database upgrade on standby
  - Upgrade standby database – UPGRADE DATABASE → asynchronous
    - `SQL1103W The UPGRADE DATABASE command was completed successfully.`
  - Will upgrade database metadata files and starts replay service in background
  - Waits for primary to form a connection
  - Considered upgrade in progress state – monitor with db2pd –hadr (`STANDBY_UPGRADE_IN_PROGRESS`) and db2diag.log
  - [RoS] No new connections are allowed in while in this state; reports failure – SQL1776N rc = 9:
    `Connection requests to an HADR standby are not allowed while database upgrade is in progress.`



HADR Single Standby Major Version Upgrade – Page 5 of 5

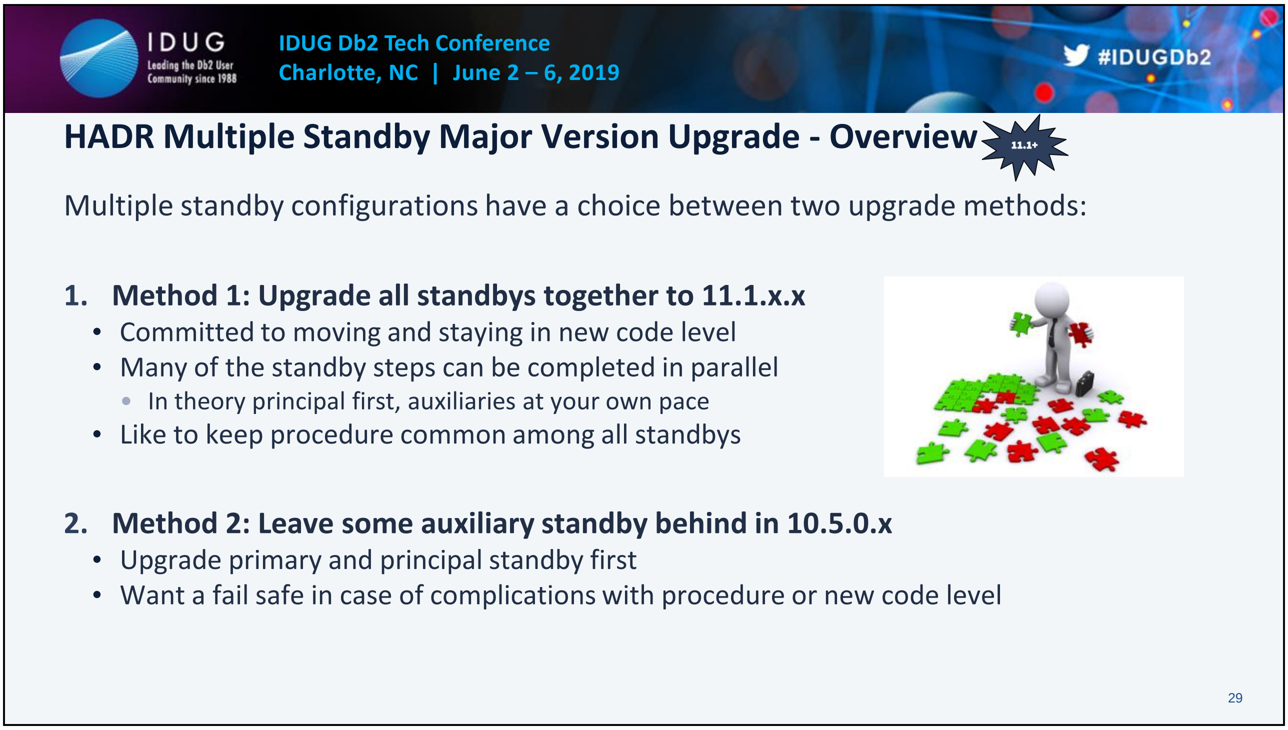
- **Step 9:** (Primary) Start database upgrade on primary
  - Upgrade primary database – UPGRADE DATABASE ➔ synchronous
    - `DB20000I The UPGRADE DATABASE command completed successfully.`
  - Consider REBINDALL option
  - Will upgrade database metadata files and attempts to form a connection with standby
  - Must have a standby at same code level available to communicate
  - Once complete primary will deactivate

- **Step 10:** (Primary) Start using database in new Db2 version
  - Start primary database – ACTIVATE DATABASE
  - Monitor standby upgrade progress – db2pd –hadr (no `STANDBY_UPGRADE_IN_PROGRESS`)
    - Standby will stay activated once it completes replay of upgrade log data
  - Perform post upgrade tasks

- **Step 11:** (Primary / Standby) Verify database configuration parameters
  - Reset values like `hadr_timeout`, `logindexbuild`, `hadr_replay_delay`

HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- **HADR Multiple Standby Major Version Upgrade**
  - **HADR Multiple Standby Major Version Upgrade - Overview**
  - **HADR Multiple Standby Major Version Upgrade – Method 1**
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future



HADR Multiple Standby Major Version Upgrade - Overview

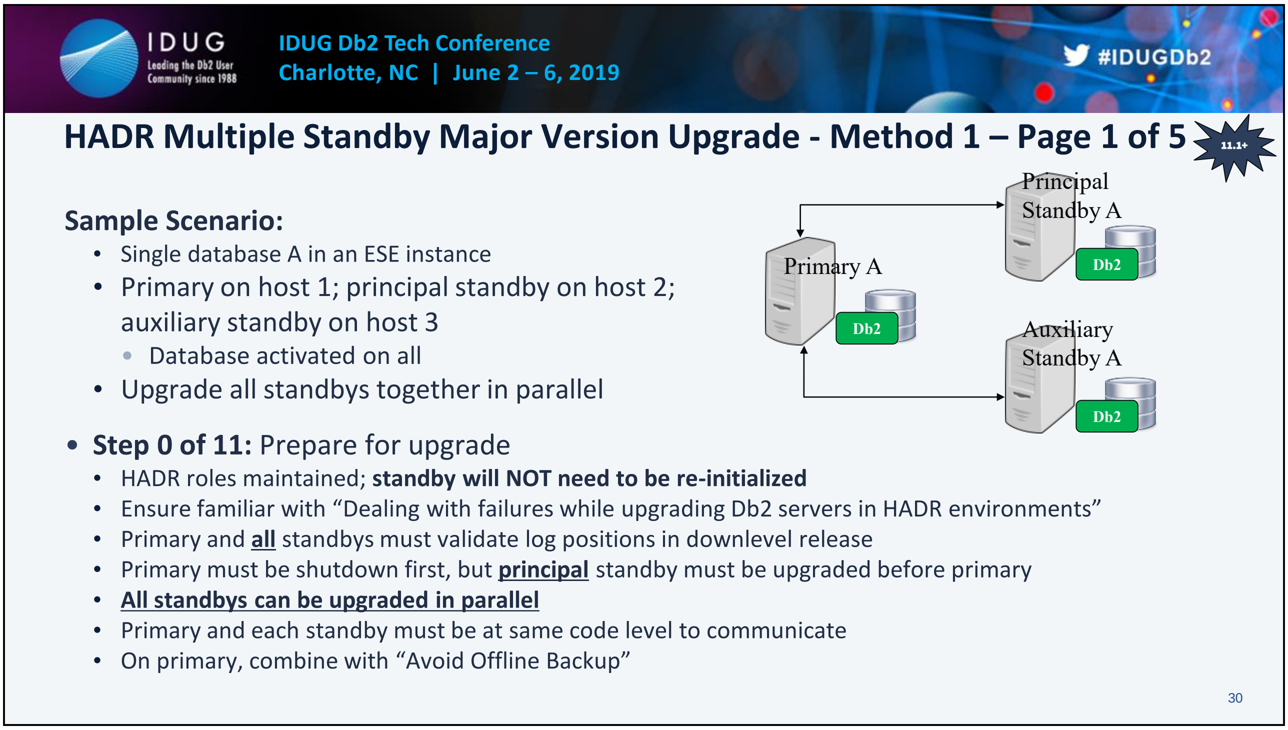
Multiple standby configurations have a choice between two upgrade methods:

1. **Method 1: Upgrade all standbys together to 11.1.x.x**
   - Committed to moving and staying in new code level
   - Many of the standby steps can be completed in parallel
     - In theory principal first, auxiliaries at your own pace
   - Like to keep procedure common among all standbys

2. **Method 2: Leave some auxiliary standby behind in 10.5.0.x**
   - Upgrade primary and principal standby first
   - Want a fail safe in case of complications with procedure or new code level

Upgrading Db2 servers in HADR multiple standby environments (without standby reinitialization)

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070095.html



IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Multiple Standby Major Version Upgrade - Method 1 – Page 1 of 5

11.1+

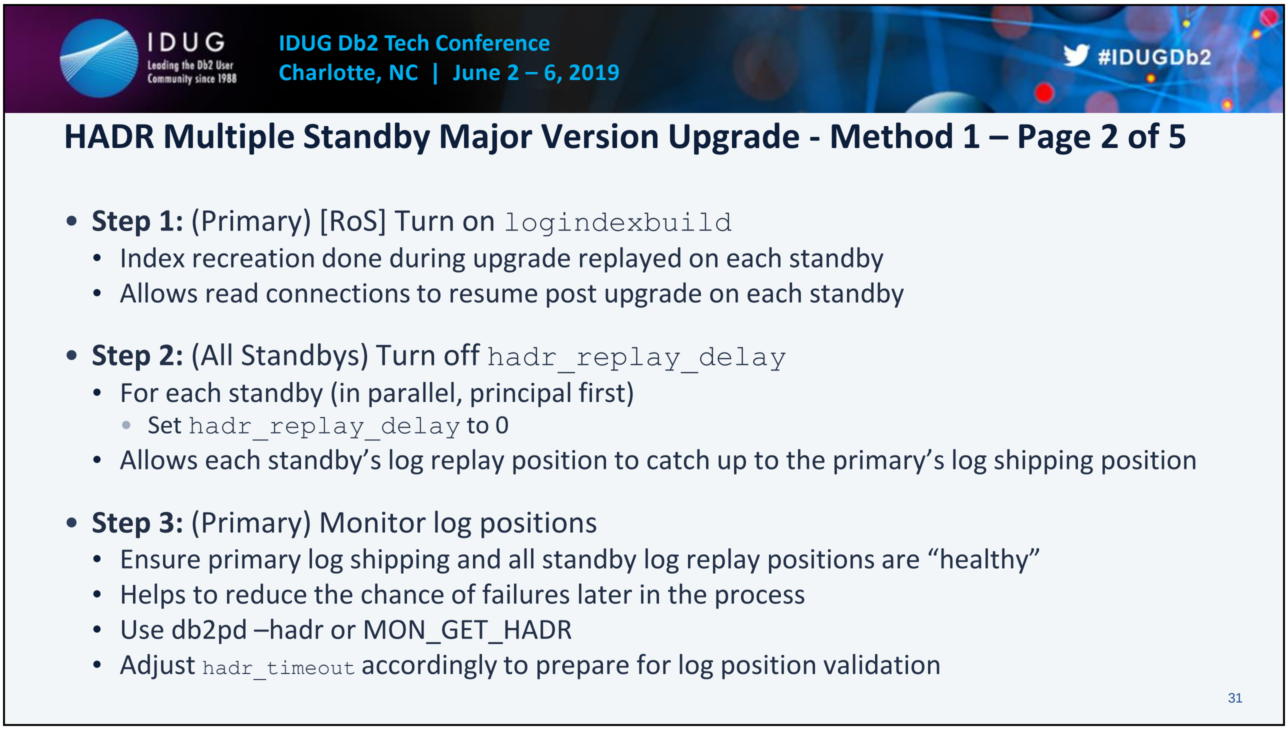
Sample Scenario:

- Single database A in an ESE instance
- Primary on host 1; principal standby on host 2; auxiliary standby on host 3
  - Database activated on all
- Upgrade all standbys together in parallel

- **Step 0 of 11:** Prepare for upgrade
  - HADR roles maintained; **standby will NOT need to be re-initialized**
  - Ensure familiar with “Dealing with failures while upgrading Db2 servers in HADR environments”
  - Primary and **all** standbys must validate log positions in downlevel release
  - Primary must be shutdown first, but **principal** standby must be upgraded before primary
  - **All standbys can be upgraded in parallel**
  - Primary and each standby must be at same code level to communicate
  - On primary, combine with “Avoid Offline Backup”

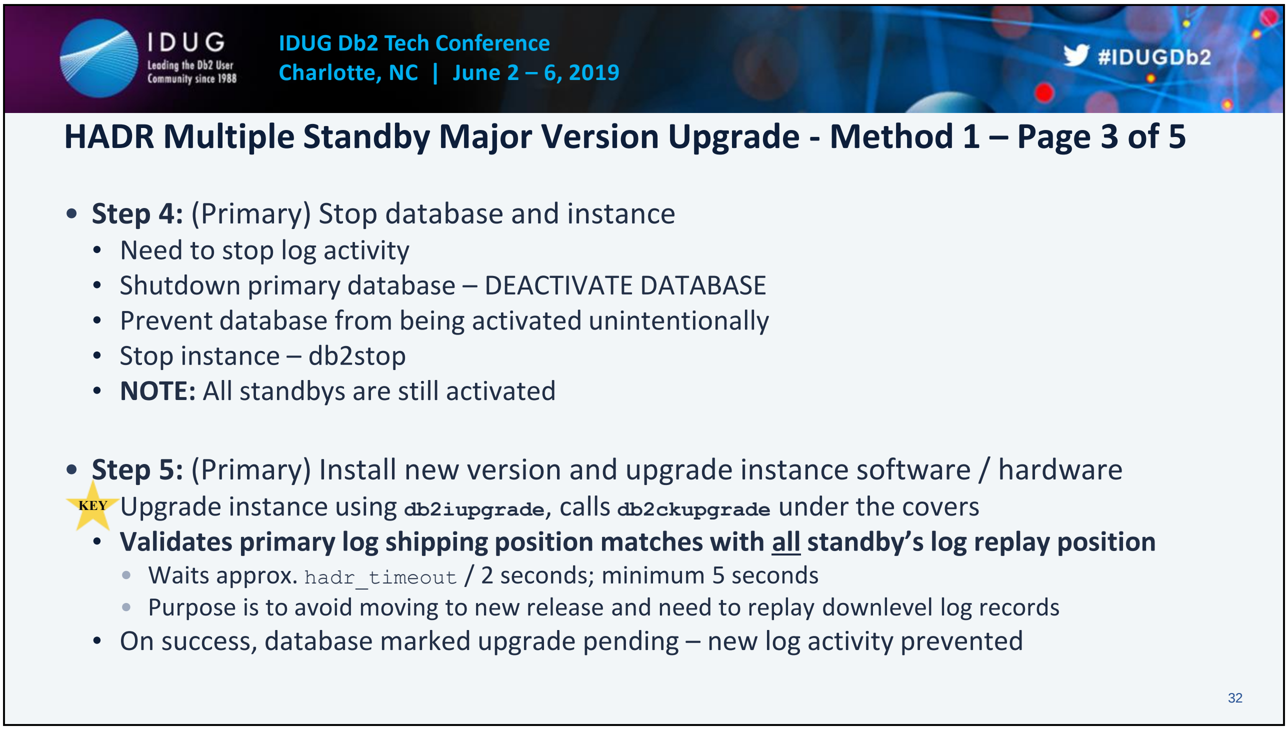
Recovering through a Db2 server upgrade

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070050.html



HADR Multiple Standby Major Version Upgrade - Method 1 – Page 2 of 5

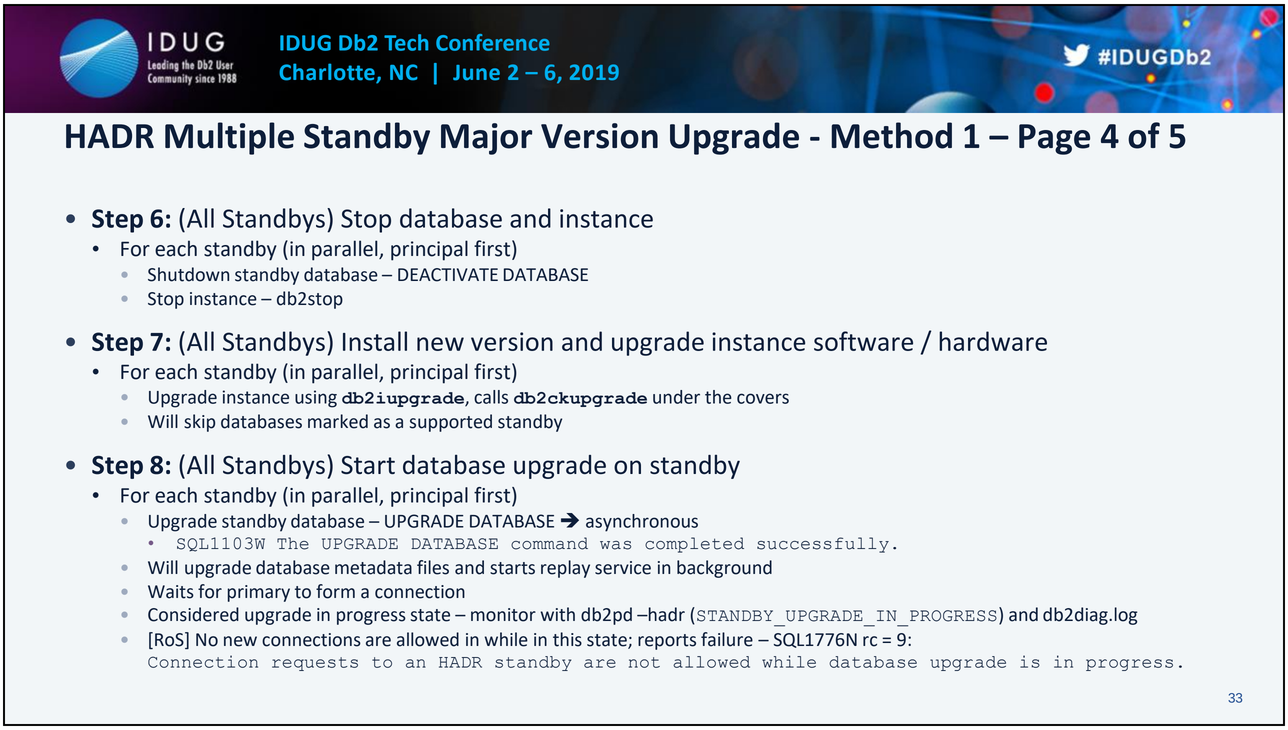
- **Step 1:** (Primary) [RoS] Turn on `logindexbuild`
  - Index recreation done during upgrade replayed on each standby
  - Allows read connections to resume post upgrade on each standby
- **Step 2:** (All Standbys) Turn off `hadr_replay_delay`
  - For each standby (in parallel, principal first)
    - Set `hadr_replay_delay` to 0
  - Allows each standby’s log replay position to catch up to the primary’s log shipping position
- **Step 3:** (Primary) Monitor log positions
  - Ensure primary log shipping and all standby log replay positions are “healthy”
  - Helps to reduce the chance of failures later in the process
  - Use db2pd –hadr or MON_GET_HADR
  - Adjust `hadr_timeout` accordingly to prepare for log position validation



HADR Multiple Standby Major Version Upgrade - Method 1 – Page 3 of 5

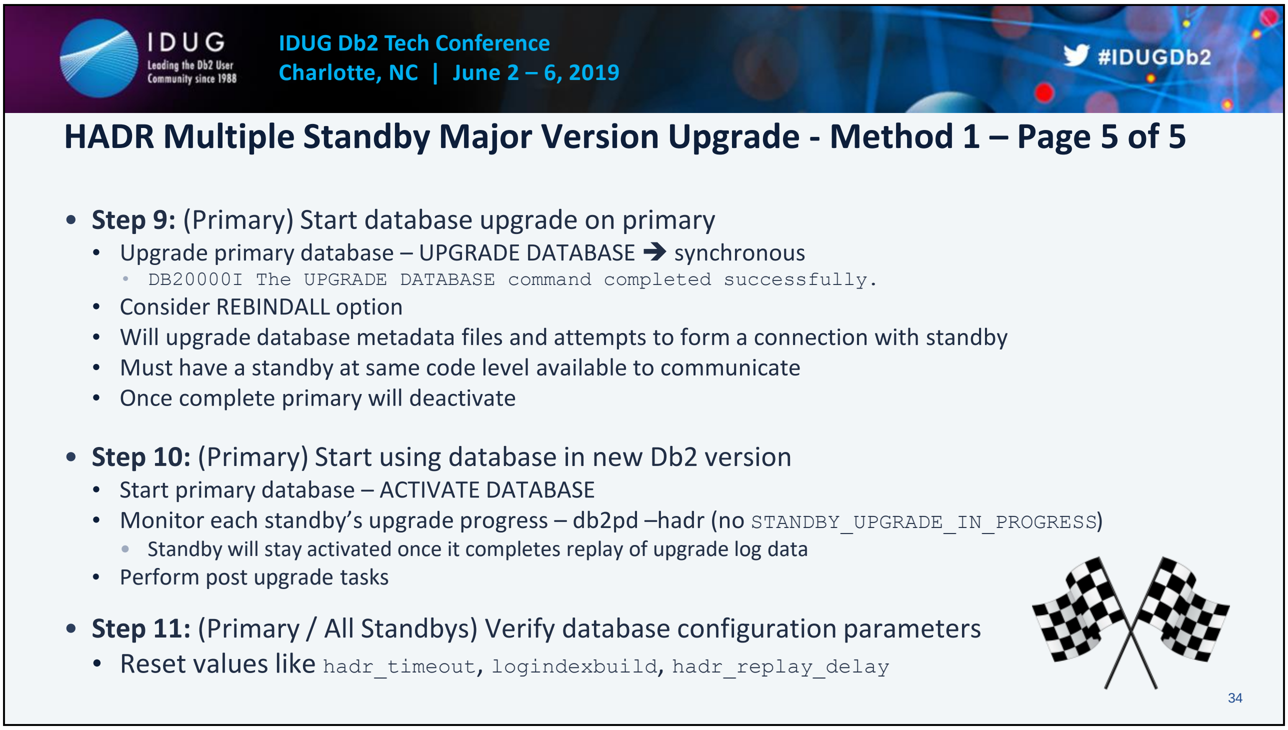
- **Step 4:** (Primary) Stop database and instance
  - Need to stop log activity
  - Shutdown primary database – DEACTIVATE DATABASE
  - Prevent database from being activated unintentionally
  - Stop instance – db2stop
  - **NOTE:** All standbys are still activated

- **Step 5:** (Primary) Install new version and upgrade instance software / hardware
  - KEY Upgrade instance using **`db2iupgrade`**, calls **`db2ckupgrade`** under the covers
  - **Validates primary log shipping position matches with all standby's log replay position**
    - Waits approx. `hadr_timeout` / 2 seconds; minimum 5 seconds
    - Purpose is to avoid moving to new release and need to replay downlevel log records
  - On success, database marked upgrade pending – new log activity prevented



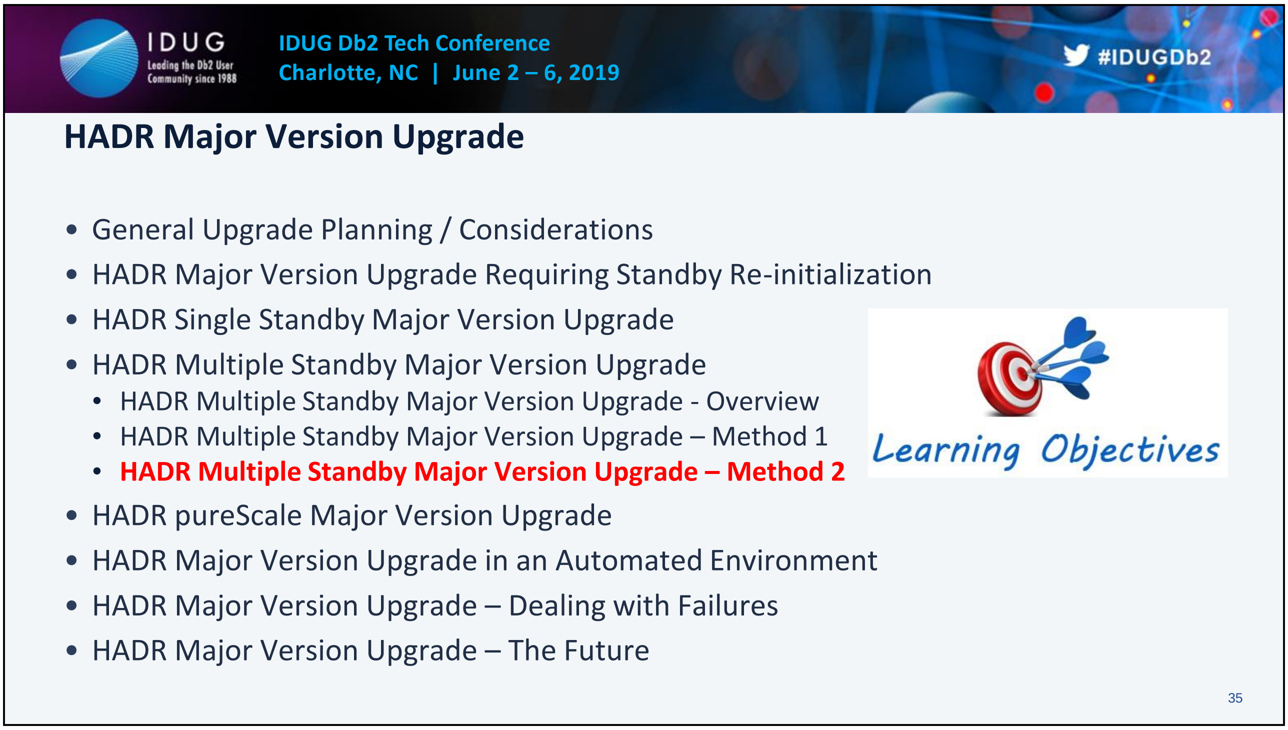
HADR Multiple Standby Major Version Upgrade - Method 1 – Page 4 of 5

- **Step 6:** (All Standbys) Stop database and instance
  - For each standby (in parallel, principal first)
    - Shutdown standby database – DEACTIVATE DATABASE
    - Stop instance – db2stop
- **Step 7:** (All Standbys) Install new version and upgrade instance software / hardware
  - For each standby (in parallel, principal first)
    - Upgrade instance using `db2iupgrade`, calls `db2ckupgrade` under the covers
    - Will skip databases marked as a supported standby
- **Step 8:** (All Standbys) Start database upgrade on standby
  - For each standby (in parallel, principal first)
    - Upgrade standby database – UPGRADE DATABASE ➔ asynchronous
      - `SQL1103W The UPGRADE DATABASE command was completed successfully.`
    - Will upgrade database metadata files and starts replay service in background
    - Waits for primary to form a connection
    - Considered upgrade in progress state – monitor with db2pd –hadr (`STANDBY_UPGRADE_IN_PROGRESS`) and db2diag.log
    - [RoS] No new connections are allowed in while in this state; reports failure – SQL1776N rc = 9: `Connection requests to an HADR standby are not allowed while database upgrade is in progress.`



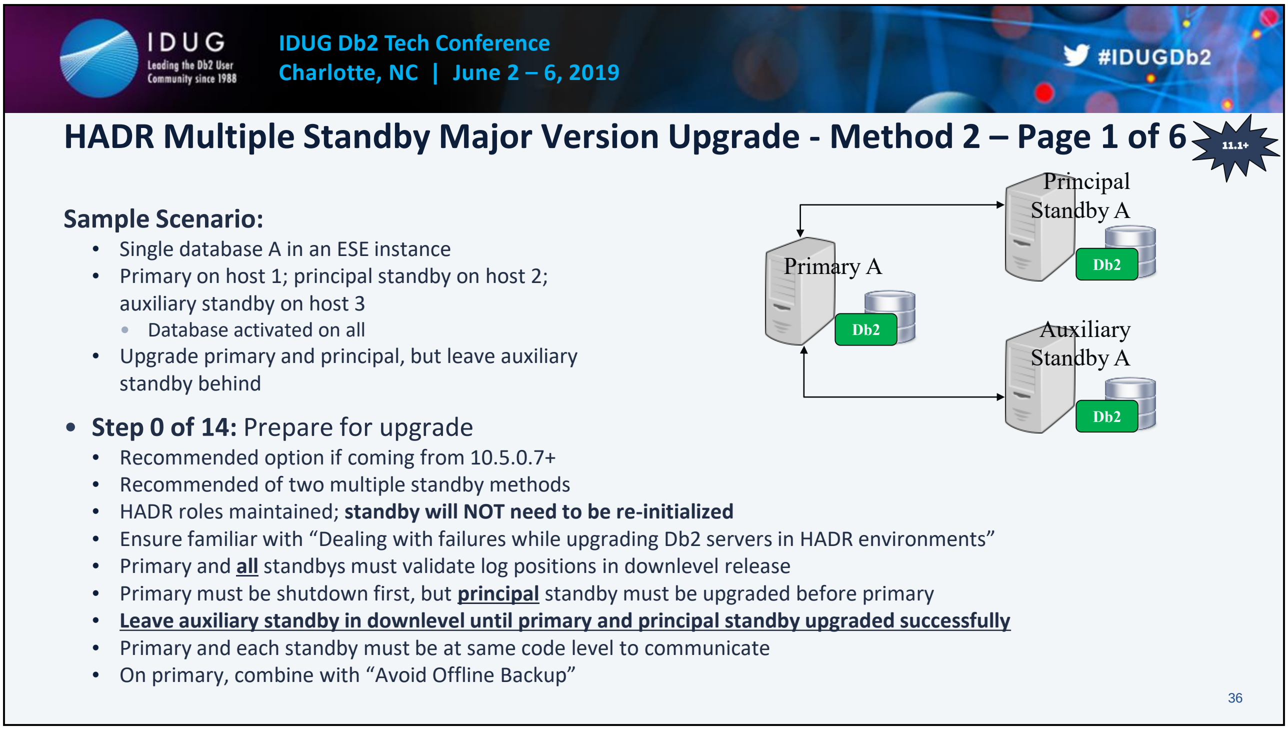
HADR Multiple Standby Major Version Upgrade - Method 1 – Page 5 of 5

- **Step 9:** (Primary) Start database upgrade on primary
  - Upgrade primary database – UPGRADE DATABASE ➔ synchronous
    - `DB20000I The UPGRADE DATABASE command completed successfully.`
  - Consider REBINDALL option
  - Will upgrade database metadata files and attempts to form a connection with standby
  - Must have a standby at same code level available to communicate
  - Once complete primary will deactivate
- **Step 10:** (Primary) Start using database in new Db2 version
  - Start primary database – ACTIVATE DATABASE
  - Monitor each standby's upgrade progress – db2pd –hadr (no `STANDBY_UPGRADE_IN_PROGRESS`)
    - Standby will stay activated once it completes replay of upgrade log data
  - Perform post upgrade tasks
- **Step 11:** (Primary / All Standbys) Verify database configuration parameters
  - Reset values like `hadr_timeout`, `logindexbuild`, `hadr_replay_delay`



HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - **HADR Multiple Standby Major Version Upgrade – Method 2**
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future



IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Multiple Standby Major Version Upgrade - Method 2 – Page 1 of 6

11.1+

Primary A
Db2

Principal Standby A
Db2

Auxiliary Standby A
Db2

Sample Scenario:

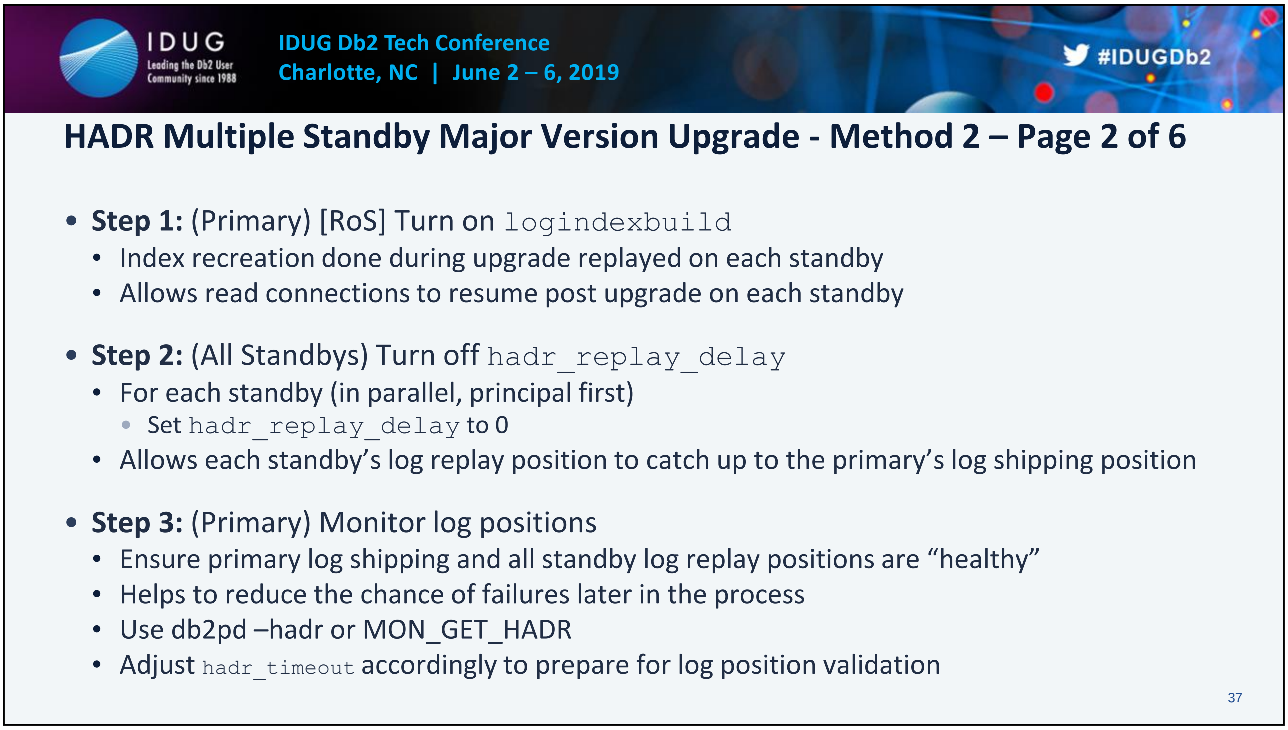
- Single database A in an ESE instance
- Primary on host 1; principal standby on host 2; auxiliary standby on host 3
  - Database activated on all
- Upgrade primary and principal, but leave auxiliary standby behind

- **Step 0 of 14:** Prepare for upgrade
  - Recommended option if coming from 10.5.0.7+
  - Recommended of two multiple standby methods
  - HADR roles maintained; **standby will NOT need to be re-initialized**
  - Ensure familiar with “Dealing with failures while upgrading Db2 servers in HADR environments”
  - Primary and **all** standbys must validate log positions in downlevel release
  - Primary must be shutdown first, but **principal** standby must be upgraded before primary
  - **Leave auxiliary standby in downlevel until primary and principal standby upgraded successfully**
  - Primary and each standby must be at same code level to communicate
  - On primary, combine with “Avoid Offline Backup”

36

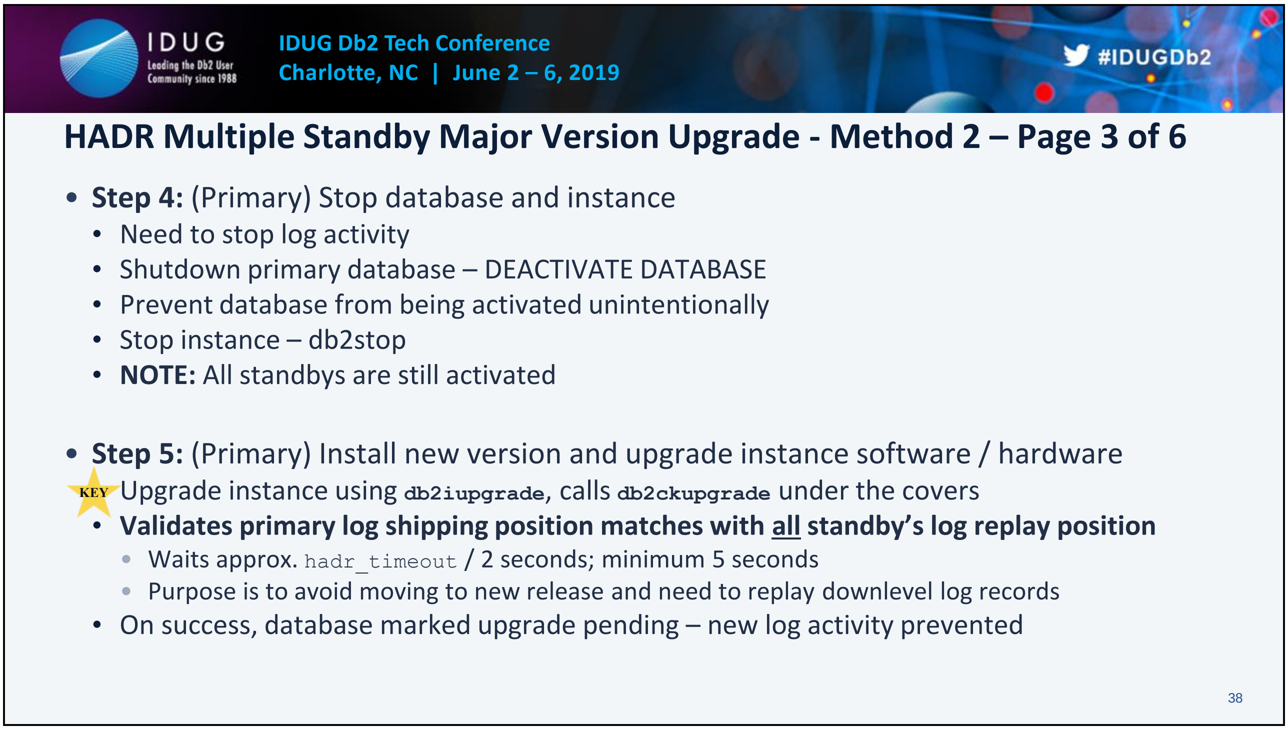
Recovering through a Db2 server upgrade

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070050.html



HADR Multiple Standby Major Version Upgrade - Method 2 – Page 2 of 6

- **Step 1:** (Primary) [RoS] Turn on `logindexbuild`
  - Index recreation done during upgrade replayed on each standby
  - Allows read connections to resume post upgrade on each standby
- **Step 2:** (All Standbys) Turn off `hadr_replay_delay`
  - For each standby (in parallel, principal first)
    - Set `hadr_replay_delay` to 0
  - Allows each standby’s log replay position to catch up to the primary’s log shipping position
- **Step 3:** (Primary) Monitor log positions
  - Ensure primary log shipping and all standby log replay positions are “healthy”
  - Helps to reduce the chance of failures later in the process
  - Use db2pd –hadr or MON_GET_HADR
  - Adjust `hadr_timeout` accordingly to prepare for log position validation



IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

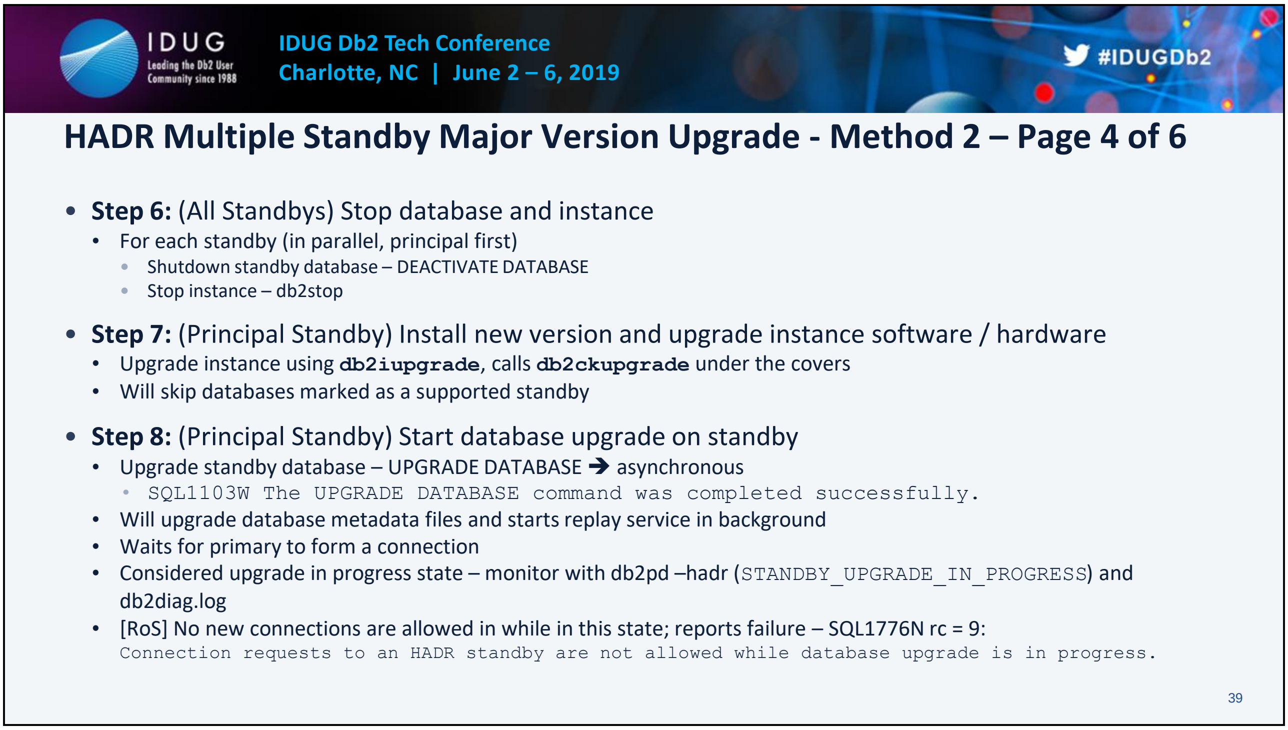
#IDUGDb2

HADR Multiple Standby Major Version Upgrade - Method 2 – Page 3 of 6

- **Step 4:** (Primary) Stop database and instance
  - Need to stop log activity
  - Shutdown primary database – DEACTIVATE DATABASE
  - Prevent database from being activated unintentionally
  - Stop instance – db2stop
  - **NOTE:** All standbys are still activated

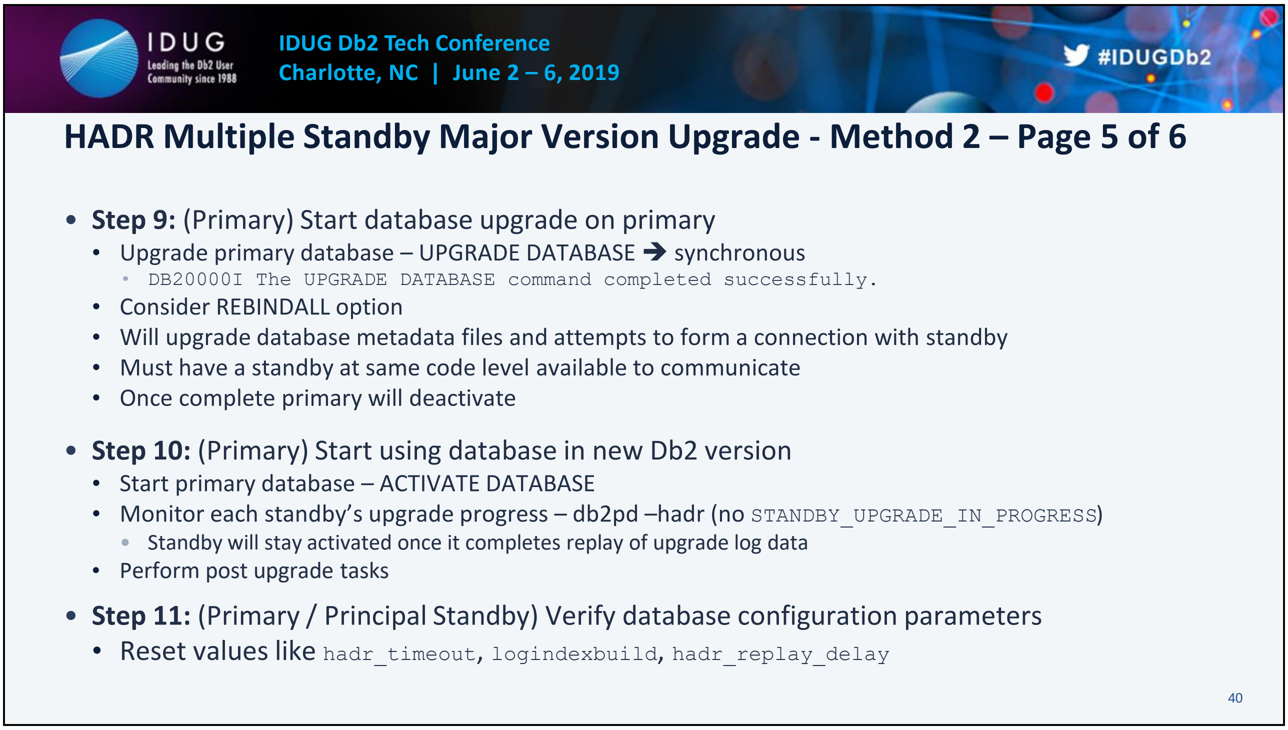
- **Step 5:** (Primary) Install new version and upgrade instance software / hardware
  - KEY Upgrade instance using `db2iupgrade`, calls `db2ckupgrade` under the covers
  - **Validates primary log shipping position matches with all standby's log replay position**
    - Waits approx. `hadr_timeout` / 2 seconds; minimum 5 seconds
    - Purpose is to avoid moving to new release and need to replay downlevel log records
  - On success, database marked upgrade pending – new log activity prevented

38



HADR Multiple Standby Major Version Upgrade - Method 2 – Page 4 of 6

- **Step 6:** (All Standbys) Stop database and instance
  - For each standby (in parallel, principal first)
    - Shutdown standby database – DEACTIVATE DATABASE
    - Stop instance – db2stop
- **Step 7:** (Principal Standby) Install new version and upgrade instance software / hardware
  - Upgrade instance using **`db2iupgrade`**, calls **`db2ckupgrade`** under the covers
  - Will skip databases marked as a supported standby
- **Step 8:** (Principal Standby) Start database upgrade on standby
  - Upgrade standby database – UPGRADE DATABASE ➔ asynchronous
    - `SQL1103W The UPGRADE DATABASE command was completed successfully.`
  - Will upgrade database metadata files and starts replay service in background
  - Waits for primary to form a connection
  - Considered upgrade in progress state – monitor with db2pd –hadr (`STANDBY_UPGRADE_IN_PROGRESS`) and db2diag.log
  - [RoS] No new connections are allowed in while in this state; reports failure – SQL1776N rc = 9:
    `Connection requests to an HADR standby are not allowed while database upgrade is in progress.`



HADR Multiple Standby Major Version Upgrade - Method 2 – Page 5 of 6

- **Step 9:** (Primary) Start database upgrade on primary
  - Upgrade primary database – UPGRADE DATABASE ➔ synchronous
    - `DB20000I The UPGRADE DATABASE command completed successfully.`
  - Consider REBINDALL option
  - Will upgrade database metadata files and attempts to form a connection with standby
  - Must have a standby at same code level available to communicate
  - Once complete primary will deactivate
- **Step 10:** (Primary) Start using database in new Db2 version
  - Start primary database – ACTIVATE DATABASE
  - Monitor each standby's upgrade progress – db2pd –hadr (no `STANDBY_UPGRADE_IN_PROGRESS`)
    - Standby will stay activated once it completes replay of upgrade log data
  - Perform post upgrade tasks
- **Step 11:** (Primary / Principal Standby) Verify database configuration parameters
  - Reset values like `hadr_timeout`, `logindexbuild`, `hadr_replay_delay`

HADR Multiple Standby Major Version Upgrade - Method 2 – Page 6 of 6

- **Step 12:** (Auxiliary Standby) Install new version and upgrade instance software / hardware
  - Once satisfied with new code level can begin with auxiliary
  - Same as principal standby

- **Step 13:** (Auxiliary Standby) Start database upgrade on standby
  - Same as principal standby

- **Step 14:** (Auxiliary Standby) Verify database configuration parameters
  - Reset values like `hadr_replay_delay`

HADR Major Version Upgrade

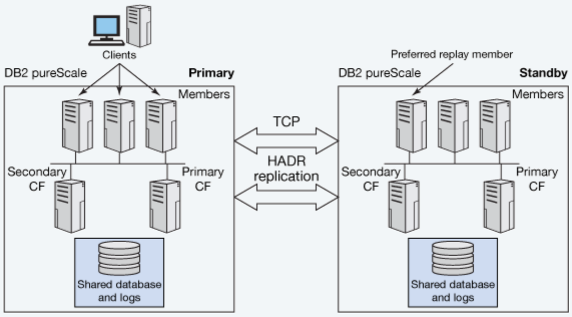
- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- **HADR pureScale Major Version Upgrade**
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future

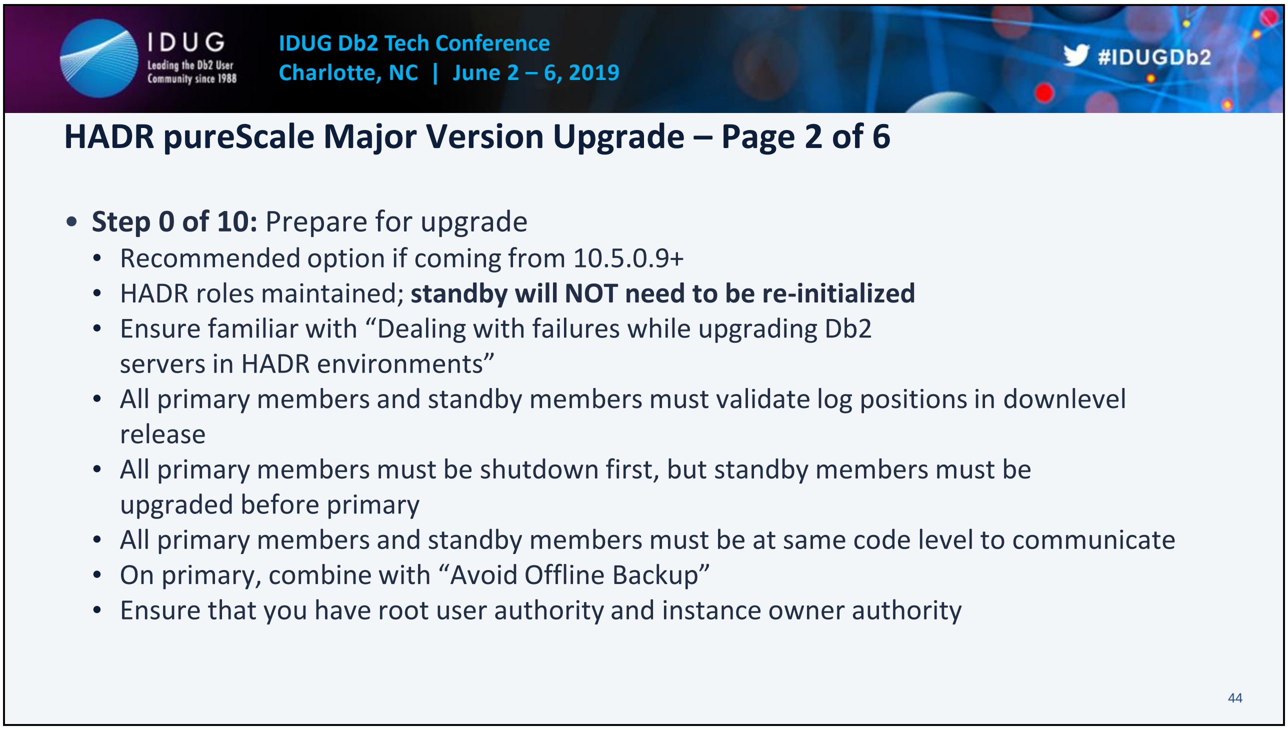
HADR pureScale Major Version Upgrade – Page 1 of 6

11.1+

Sample Scenario:

- Single database A in a pureScale instance
- Primary cluster:
  - Two CF servers
  - Members m0 / m1 / m2
- Secondary cluster:
  - Two CF servers
  - Members m0 / m1 / m2
- Database activated on both





IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR pureScale Major Version Upgrade – Page 2 of 6

- **Step 0 of 10:** Prepare for upgrade
  - Recommended option if coming from 10.5.0.9+
  - HADR roles maintained; **standby will NOT need to be re-initialized**
  - Ensure familiar with “Dealing with failures while upgrading Db2 servers in HADR environments”
  - All primary members and standby members must validate log positions in downlevel release
  - All primary members must be shutdown first, but standby members must be upgraded before primary
  - All primary members and standby members must be at same code level to communicate
  - On primary, combine with “Avoid Offline Backup”
  - Ensure that you have root user authority and instance owner authority

44

Upgrading Db2 servers in HADR pureScale environments (without standby reinitialization)

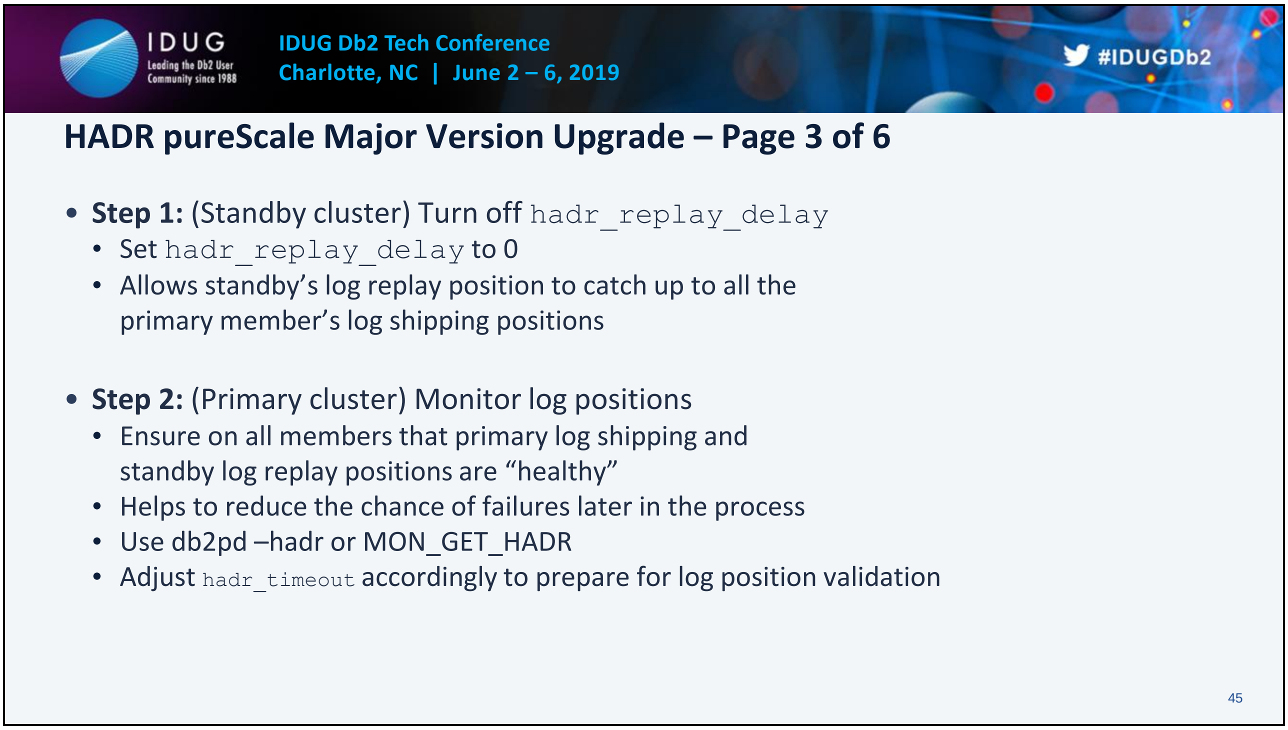
https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070096.html

Upgrading a Db2 pureScale server

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0060571.html

Recovering through a Db2 server upgrade

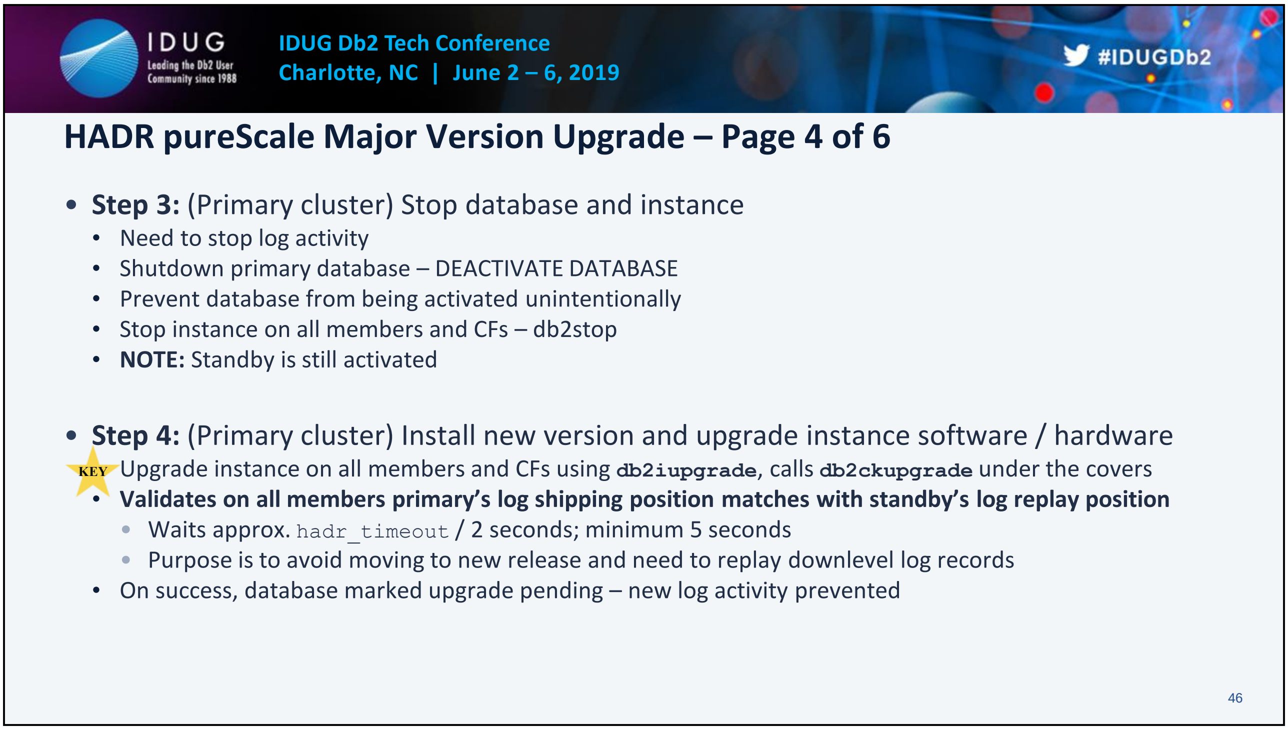
https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070050.html



HADR pureScale Major Version Upgrade – Page 3 of 6

- **Step 1:** (Standby cluster) Turn off `hadr_replay_delay`
  - Set `hadr_replay_delay` to 0
  - Allows standby's log replay position to catch up to all the primary member's log shipping positions

- **Step 2:** (Primary cluster) Monitor log positions
  - Ensure on all members that primary log shipping and standby log replay positions are "healthy"
  - Helps to reduce the chance of failures later in the process
  - Use db2pd –hadr or MON_GET_HADR
  - Adjust `hadr_timeout` accordingly to prepare for log position validation

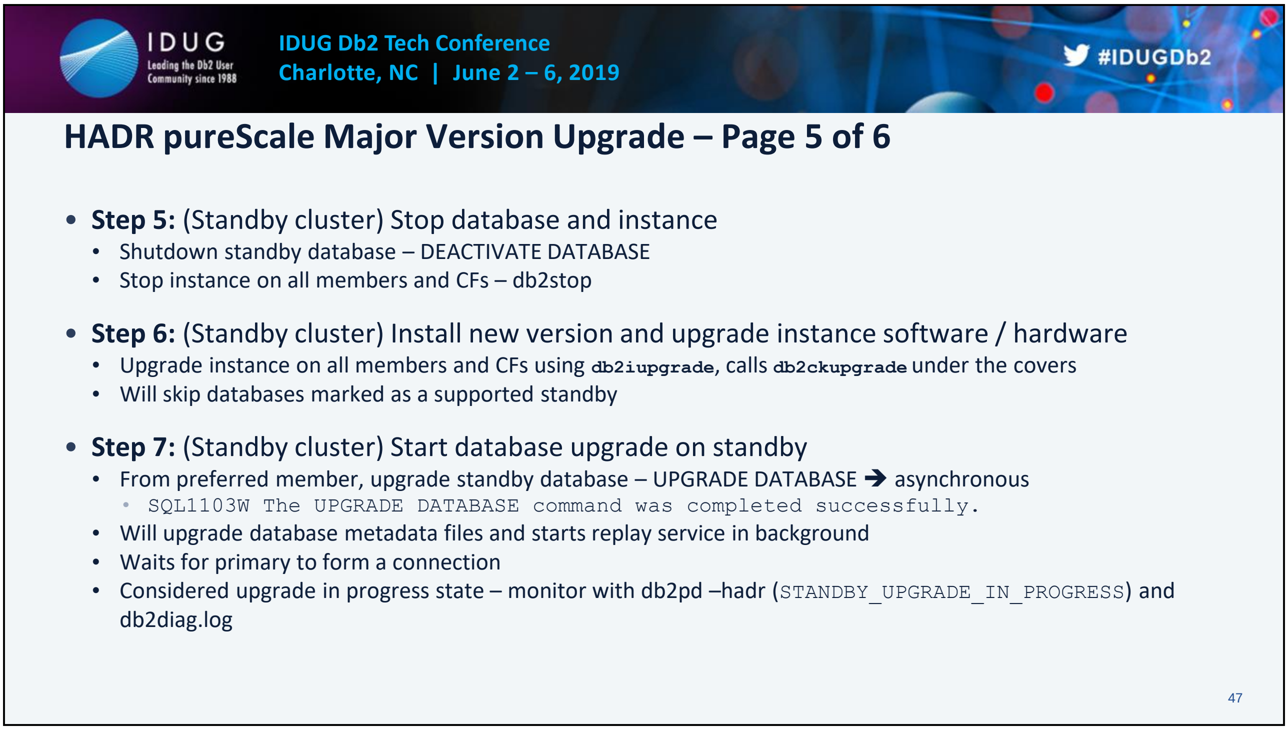


IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR pureScale Major Version Upgrade – Page 4 of 6

- **Step 3:** (Primary cluster) Stop database and instance
  - Need to stop log activity
  - Shutdown primary database – DEACTIVATE DATABASE
  - Prevent database from being activated unintentionally
  - Stop instance on all members and CFs – db2stop
  - **NOTE:** Standby is still activated
- **Step 4:** (Primary cluster) Install new version and upgrade instance software / hardware
  - KEY Upgrade instance on all members and CFs using `db2iupgrade`, calls `db2ckupgrade` under the covers
  - **Validates on all members primary's log shipping position matches with standby's log replay position**
    - Waits approx. `hadr_timeout` / 2 seconds; minimum 5 seconds
    - Purpose is to avoid moving to new release and need to replay downlevel log records
  - On success, database marked upgrade pending – new log activity prevented



HADR pureScale Major Version Upgrade – Page 5 of 6

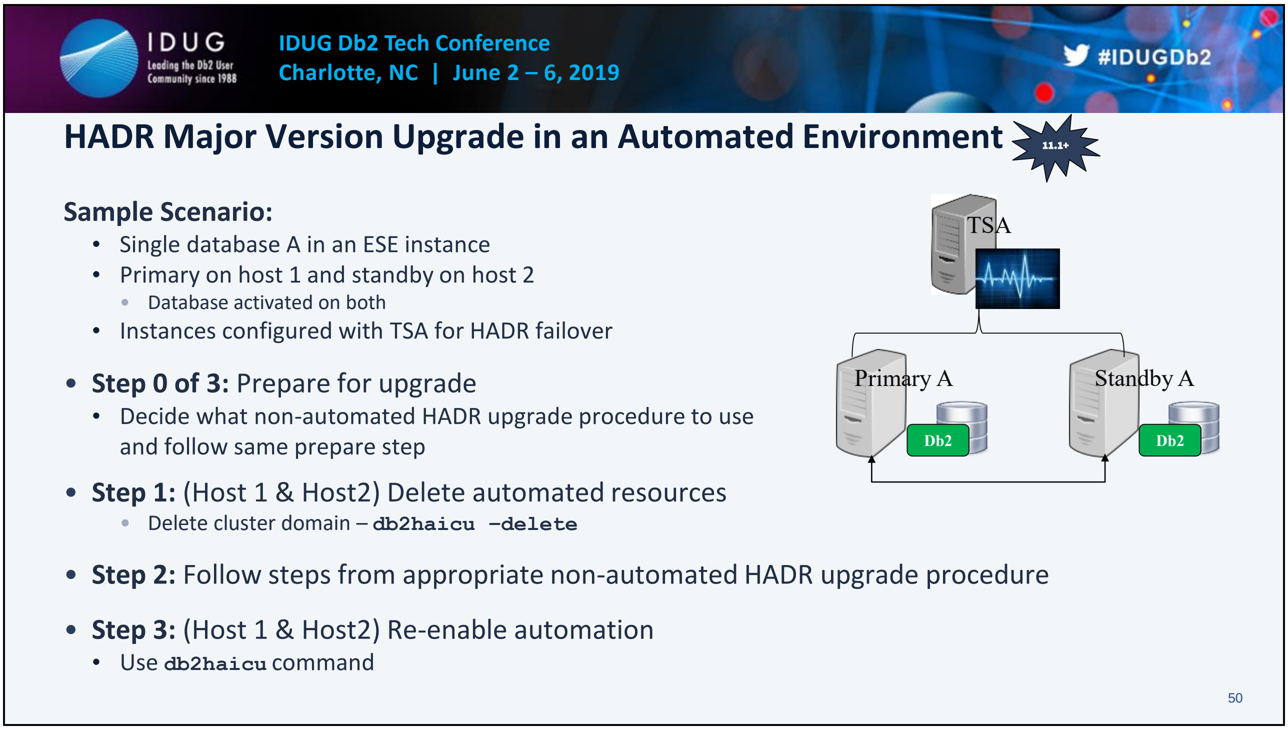
- **Step 5:** (Standby cluster) Stop database and instance
  - Shutdown standby database – DEACTIVATE DATABASE
  - Stop instance on all members and CFs – db2stop
- **Step 6:** (Standby cluster) Install new version and upgrade instance software / hardware
  - Upgrade instance on all members and CFs using **`db2iupgrade`**, calls **`db2ckupgrade`** under the covers
  - Will skip databases marked as a supported standby
- **Step 7:** (Standby cluster) Start database upgrade on standby
  - From preferred member, upgrade standby database – UPGRADE DATABASE ➔ asynchronous
    - `SQL1103W The UPGRADE DATABASE command was completed successfully.`
  - Will upgrade database metadata files and starts replay service in background
  - Waits for primary to form a connection
  - Considered upgrade in progress state – monitor with db2pd –hadr (`STANDBY_UPGRADE_IN_PROGRESS`) and db2diag.log

HADR pureScale Major Version Upgrade – Page 6 of 6

- **Step 8:** (Primary cluster) Start database upgrade on primary
  - Upgrade primary database – UPGRADE DATABASE ➔ synchronous
    - `DB20000I The UPGRADE DATABASE command completed successfully.`
  - Consider REBINDALL option
  - Will upgrade database metadata files and attempts to form a connection with standby
  - Must have all standbys at same code level available to communicate
  - Once complete primary will deactivate
- **Step 9:** (Primary cluster) Start using database in new Db2 version
  - Start primary database – ACTIVATE DATABASE
  - Monitor standby upgrade progress – db2pd –hadr (no `STANDBY_UPGRADE_IN_PROGRESS`)
    - Standby will stay activated once it completes replay of upgrade log data
  - Perform post upgrade tasks
- **Step 10:** (Primary / Standby cluster) Verify database configuration parameters
  - Reset values like `hadr_timeout`, `hadr_replay_delay`

HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- **HADR Major Version Upgrade in an Automated Environment**
- HADR Major Version Upgrade – Dealing with Failures
- HADR Major Version Upgrade – The Future



HADR Major Version Upgrade in an Automated Environment

11.1+

Sample Scenario:

- Single database A in an ESE instance
- Primary on host 1 and standby on host 2
  - Database activated on both
- Instances configured with TSA for HADR failover

TSA
Primary A
Db2
Standby A
Db2

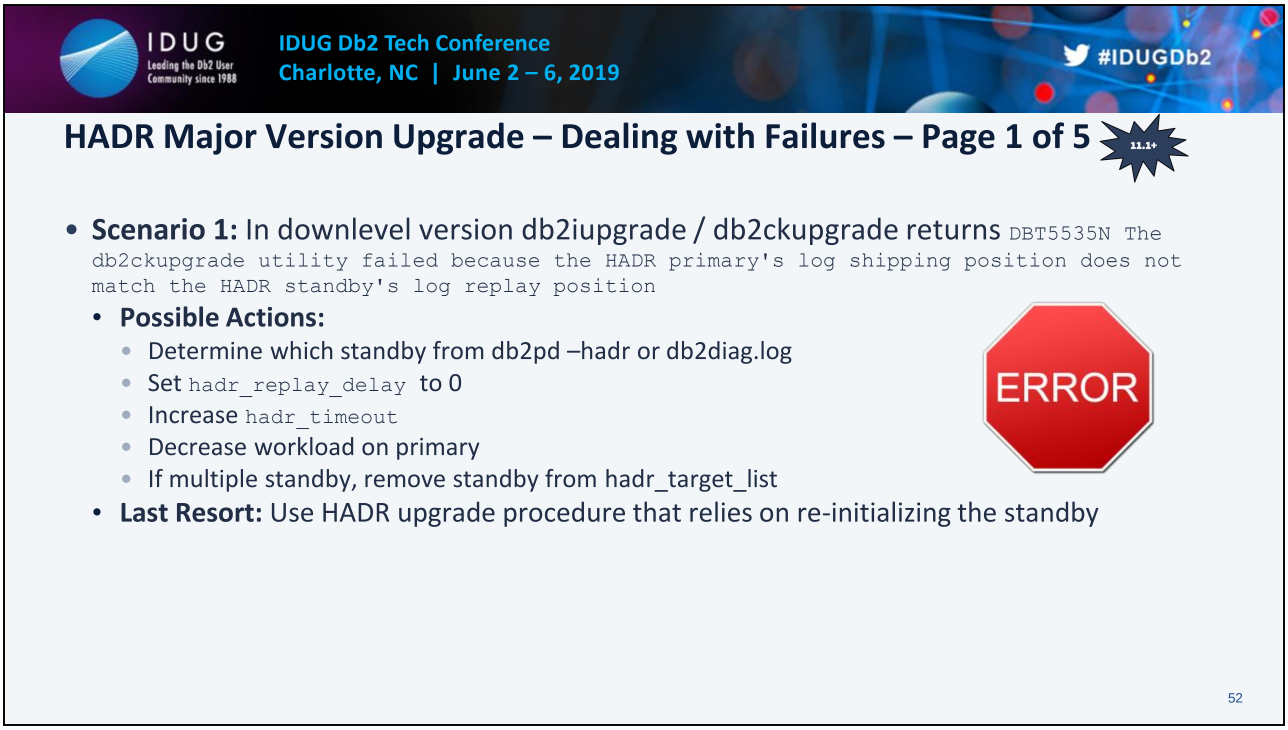
- **Step 0 of 3:** Prepare for upgrade
  - Decide what non-automated HADR upgrade procedure to use and follow same prepare step
- **Step 1:** (Host 1 & Host2) Delete automated resources
  - Delete cluster domain – `db2haicu -delete`
- **Step 2:** Follow steps from appropriate non-automated HADR upgrade procedure
- **Step 3:** (Host 1 & Host2) Re-enable automation
  - Use `db2haicu` command

Upgrading Db2 servers in an automated HADR environment

https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/t0070016.html

HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- **HADR Major Version Upgrade – Dealing with Failures**
- HADR Major Version Upgrade – The Future

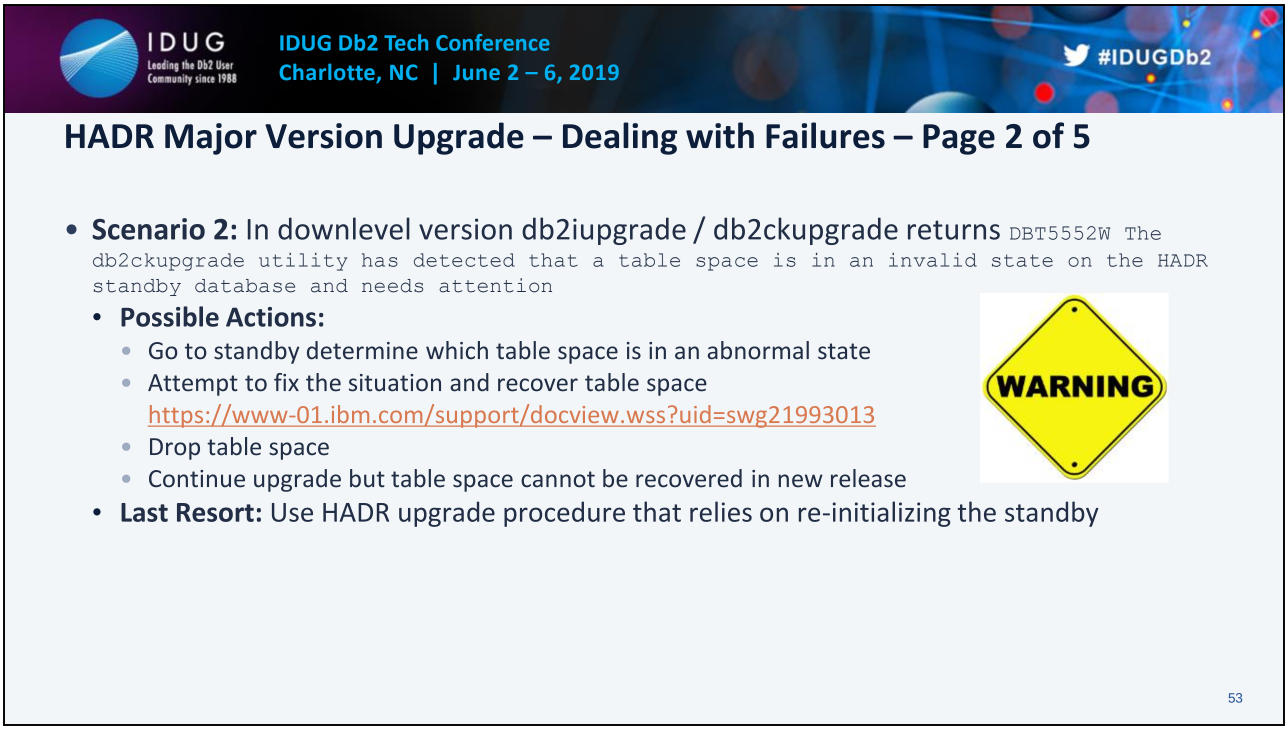


HADR Major Version Upgrade – Dealing with Failures – Page 1 of 5

- **Scenario 1:** In downlevel version db2iupgrade / db2ckupgrade returns `DBT5535N The db2ckupgrade utility failed because the HADR primary's log shipping position does not match the HADR standby's log replay position`
  - **Possible Actions:**
    - Determine which standby from db2pd –hadr or db2diag.log
    - Set `hadr_replay_delay` to 0
    - Increase `hadr_timeout`
    - Decrease workload on primary
    - If multiple standby, remove standby from hadr_target_list
  - **Last Resort:** Use HADR upgrade procedure that relies on re-initializing the standby

Scenario 1: In Db2 Version 10.5 Fix Pack 7 or later, if the primary's log shipping functionality and the standby's log replay functionality are not healthy causing db2iupgrade/db2ckupgrade to fail.

If the issue cannot be fixed within the upgrade window, then follow the previous HADR procedure that requires the stopping of HADR and reinitialization discussed in Upgrading Db2 servers in HADR environments.

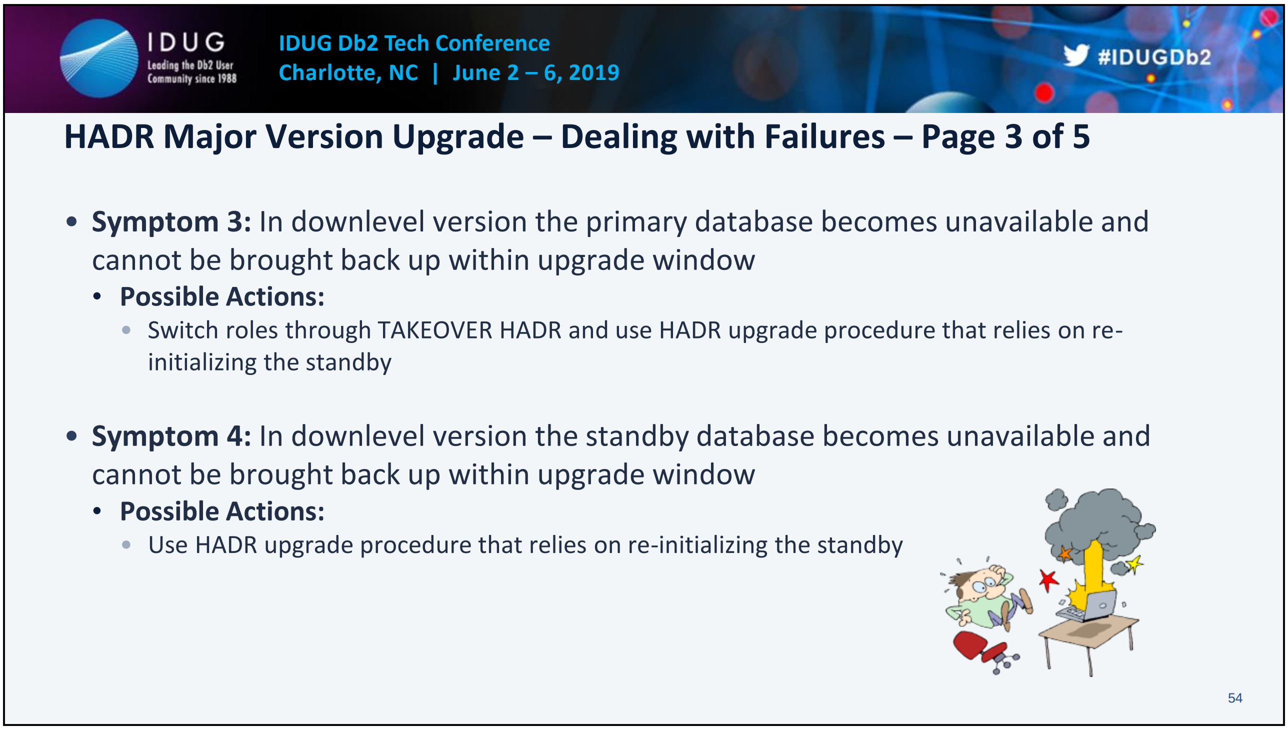


HADR Major Version Upgrade – Dealing with Failures – Page 2 of 5

- **Scenario 2:** In downlevel version db2iupgrade / db2ckupgrade returns `DBT5552W The db2ckupgrade utility has detected that a table space is in an invalid state on the HADR standby database and needs attention`
  - **Possible Actions:**
    - Go to standby determine which table space is in an abnormal state
    - Attempt to fix the situation and recover table space https://www-01.ibm.com/support/docview.wss?uid=swg21993013
    - Drop table space
    - Continue upgrade but table space cannot be recovered in new release
  - **Last Resort:** Use HADR upgrade procedure that relies on re-initializing the standby

Scenario 2: In Db2 Version 10.5 Fix Pack 7 or later, if the primary's log shipping functionality and the standby's log replay functionality are healthy but the standby's replay position is still behind the primary's log shipping position causing db2iupgrade/db2ckupgrade to fail.

Ensure that replay delay is turned off by setting hadr_replay_delay to 0. Try to allow more time for the standby to catch up by increasing the hadr_timeout value. If neither of these options allow for the log positions to match within the upgrade window, then follow the previous HADR procedure that requires the stopping of HADR and reinitialization discussed in Upgrading Db2 servers in HADR environments.



IDUG Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade – Dealing with Failures – Page 3 of 5

- **Symptom 3:** In downlevel version the primary database becomes unavailable and cannot be brought back up within upgrade window
  - **Possible Actions:**
    - Switch roles through TAKEOVER HADR and use HADR upgrade procedure that relies on re-initializing the standby
- **Symptom 4:** In downlevel version the standby database becomes unavailable and cannot be brought back up within upgrade window
  - **Possible Actions:**
    - Use HADR upgrade procedure that relies on re-initializing the standby

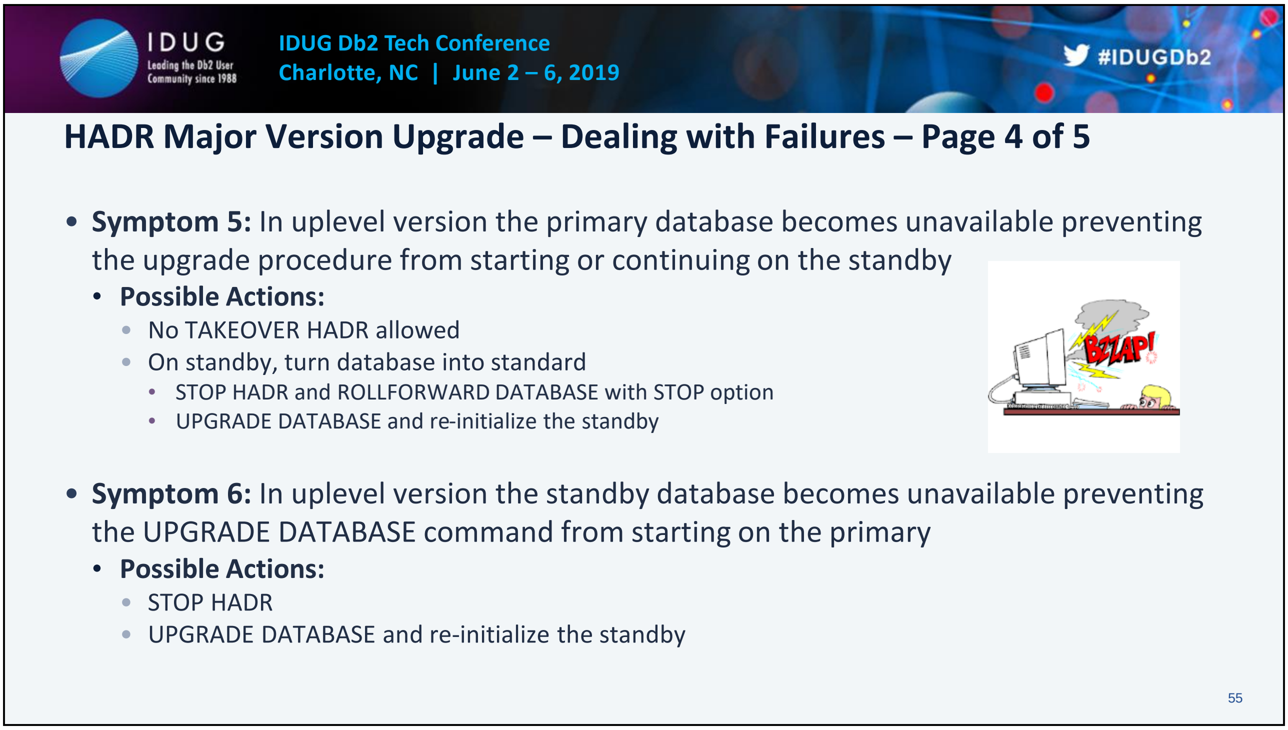
54

Scenario 3: In Db2 Version 10.5 Fix Pack 7 or later, if the primary database becomes unavailable preventing db2iupgrade/db2ckupgrade from being run.

If the primary database cannot be brought back up within the upgrade window, switch roles on the standby and then follow the previous HADR procedure that requires the stopping of HADR and reinitialization discussed in Upgrading Db2 servers in HADR environments.

Scenario 4: In Db2 Version 10.5 Fix Pack 7 or later, if the standby database becomes unavailable preventing db2iupgrade/db2ckupgrade from being run.

If the standby database cannot be brought back up within the upgrade window, then follow the previous HADR procedure that requires the stopping of HADR and reinitialization discussed in Upgrading Db2 servers in HADR environments.



HADR Major Version Upgrade – Dealing with Failures – Page 4 of 5

- **Symptom 5:** In uplevel version the primary database becomes unavailable preventing the upgrade procedure from starting or continuing on the standby
  - **Possible Actions:**
    - No TAKEOVER HADR allowed
    - On standby, turn database into standard
      - STOP HADR and ROLLFORWARD DATABASE with STOP option
      - UPGRADE DATABASE and re-initialize the standby

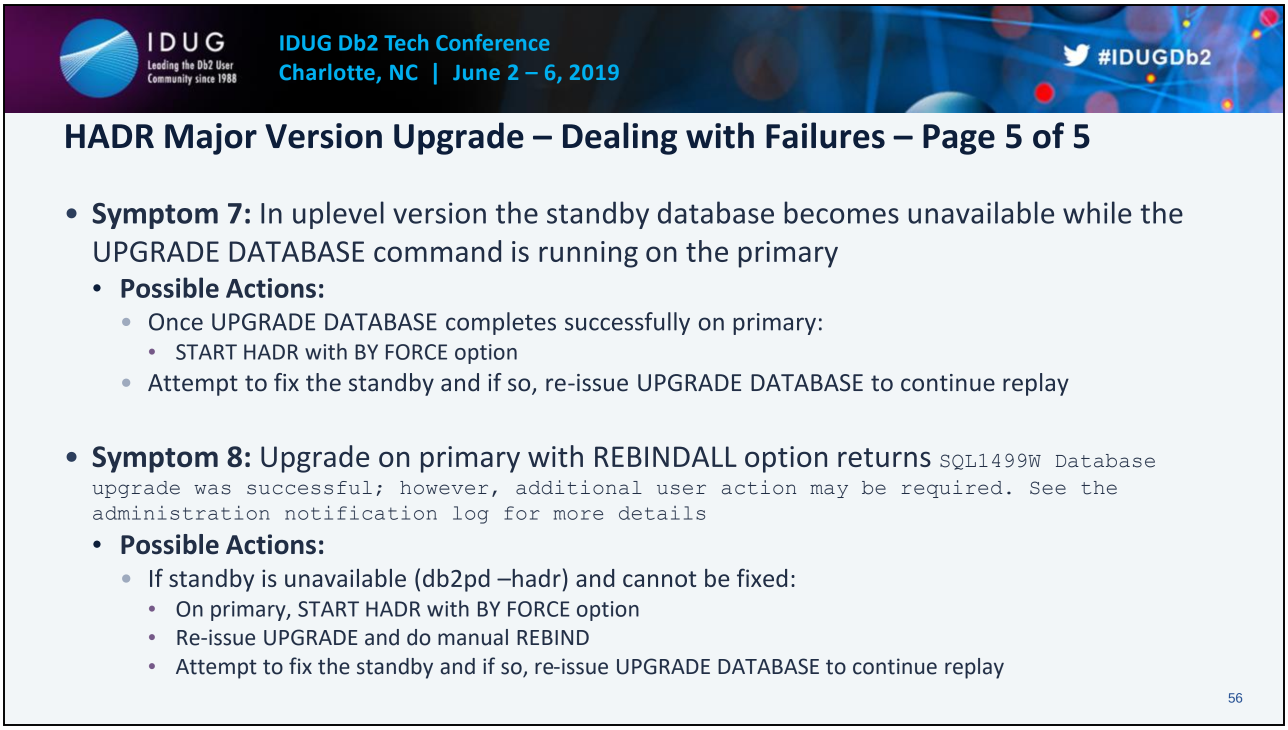
- **Symptom 6:** In uplevel version the standby database becomes unavailable preventing the UPGRADE DATABASE command from starting on the primary
  - **Possible Actions:**
    - STOP HADR
    - UPGRADE DATABASE and re-initialize the standby

Scenario 5: In Db2 Version 11.1, if the primary database becomes unavailable preventing the upgrade procedure from continuing on the standby.

If the primary database cannot be brought back up within the upgrade window, on the standby issue STOP HADR followed by ROLLFORWARD DATABASE with the STOP option. This will turn the database into a non-HADR database. The database will now be upgrade pending and so issue the UPGRADE DATABASE command to continue the upgrade. Once complete refer to Post-upgrade tasks for Db2 servers and Verifying upgrade of Db2 servers. HADR must be reinitialized.

Scenario 6: In Db2 Version 11.1, if the standby database becomes unavailable preventing the UPGRADE DATABASE command from starting up on the primary.

If the standby database cannot be brought back up within the upgrade window, on the primary issue STOP HADR. This turns the database into a non-HADR database. The database will still be upgrade pending so reissue the UPGRADE DATABASE command to continue the upgrade. Once complete refer to Post-upgrade tasks for Db2 servers and Verifying upgrade of Db2 servers. HADR will have to be reinitialized.



IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

HADR Major Version Upgrade – Dealing with Failures – Page 5 of 5

- **Symptom 7:** In uplevel version the standby database becomes unavailable while the UPGRADE DATABASE command is running on the primary
  - **Possible Actions:**
    - Once UPGRADE DATABASE completes successfully on primary:
      - START HADR with BY FORCE option
    - Attempt to fix the standby and if so, re-issue UPGRADE DATABASE to continue replay

- **Symptom 8:** Upgrade on primary with REBINDALL option returns `SQL1499W Database upgrade was successful; however, additional user action may be required. See the administration notification log for more details`
  - **Possible Actions:**
    - If standby is unavailable (db2pd –hadr) and cannot be fixed:
      - On primary, START HADR with BY FORCE option
      - Re-issue UPGRADE and do manual REBIND
      - Attempt to fix the standby and if so, re-issue UPGRADE DATABASE to continue replay

56

Scenario 7: In Db2 Version 11.1, if the standby database becomes unavailable while in upgrade in progress state.

Once the UPGRADE DATABASE command is issued on the primary and the primary forms a connection with a standby database, the upgrade will proceed without issue on the primary and will eventually complete successfully. The concern is that there is no standby database replaying log data, which leaves an exposure to a loss of the primary. Post upgrade the primary database can still be brought up through the START HADR command specifying the BY FORCE option. At this point, all attempts should be made to resolve the issues with the standby. Once resolved, since the standby was in upgrade in progress state, the UPGRADE DATABASE command should be issued. The standby continues to replay the upgrade log data shipped by the primary until it completes and is no longer in the upgrade in progress state.

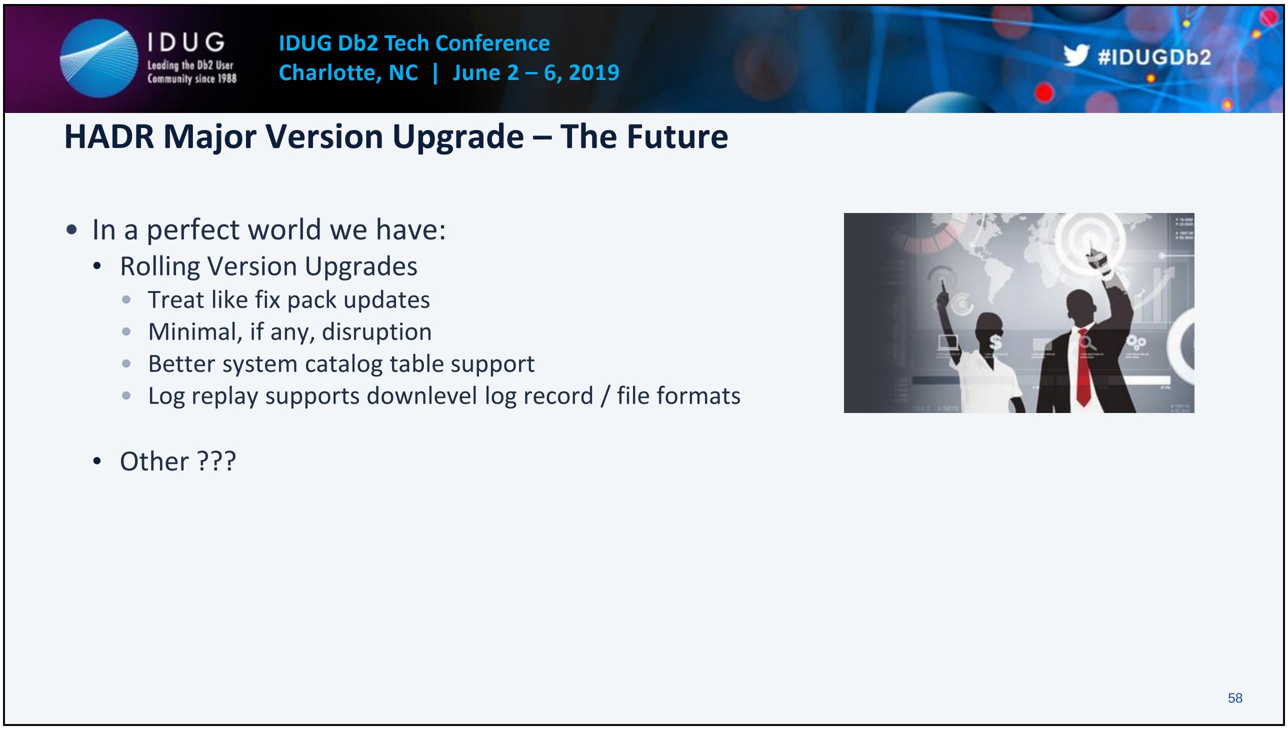
Scenario 8: In Db2 Version 11.1, if the UPGRADE DATABASE command with the REBINDALL option was specified on the primary and the standby database becomes unavailable while in upgrade in progress state.

The difference from Scenario 7 is that on the primary the UPGRADE DATABASE command was specified with the REBINDALL option. In this case, the UPGRADE DATABASE command requires and attempts a new connection to the database. If the standby database is not available during this second connection attempt, the UPGRADE DATABASE command returns SQL1499W. SQL1499W can be returned for many other reasons so the Db2 diagnostics log may be required to tell what failed and whether this scenario applies. If so, the primary database can still be brought up through the START HADR command specifying the BY FORCE option. Rebinding can still take place manually at this point. But, all attempts should be made to resolve the issues with the standby. Once resolved, since the standby was in upgrade in progress state, the UPGRADE DATABASE command should be issued. The standby continues to replay the upgrade log data shipped by the primary until it completes and is no longer in the upgrade in progress state.

At any time, if there are issues with the upgrade to Db2 Version 11.1, you can reverse the upgrade or fall back from Db2 Version 11.1 to a pre-Db2 Version 11.1 release. See Reversing Db2 server upgrade to learn all the required steps to reverse a database upgrade.

HADR Major Version Upgrade

- General Upgrade Planning / Considerations
- HADR Major Version Upgrade Requiring Standby Re-initialization
- HADR Single Standby Major Version Upgrade
- HADR Multiple Standby Major Version Upgrade
  - HADR Multiple Standby Major Version Upgrade - Overview
  - HADR Multiple Standby Major Version Upgrade – Method 1
  - HADR Multiple Standby Major Version Upgrade – Method 2
- HADR pureScale Major Version Upgrade
- HADR Major Version Upgrade in an Automated Environment
- HADR Major Version Upgrade – Dealing with Failures
- **HADR Major Version Upgrade – The Future**



IDUG

Leading the Db2 User Community since 1988

IDUG Db2 Tech Conference
Charlotte, NC | June 2 – 6, 2019

#IDUGDb2

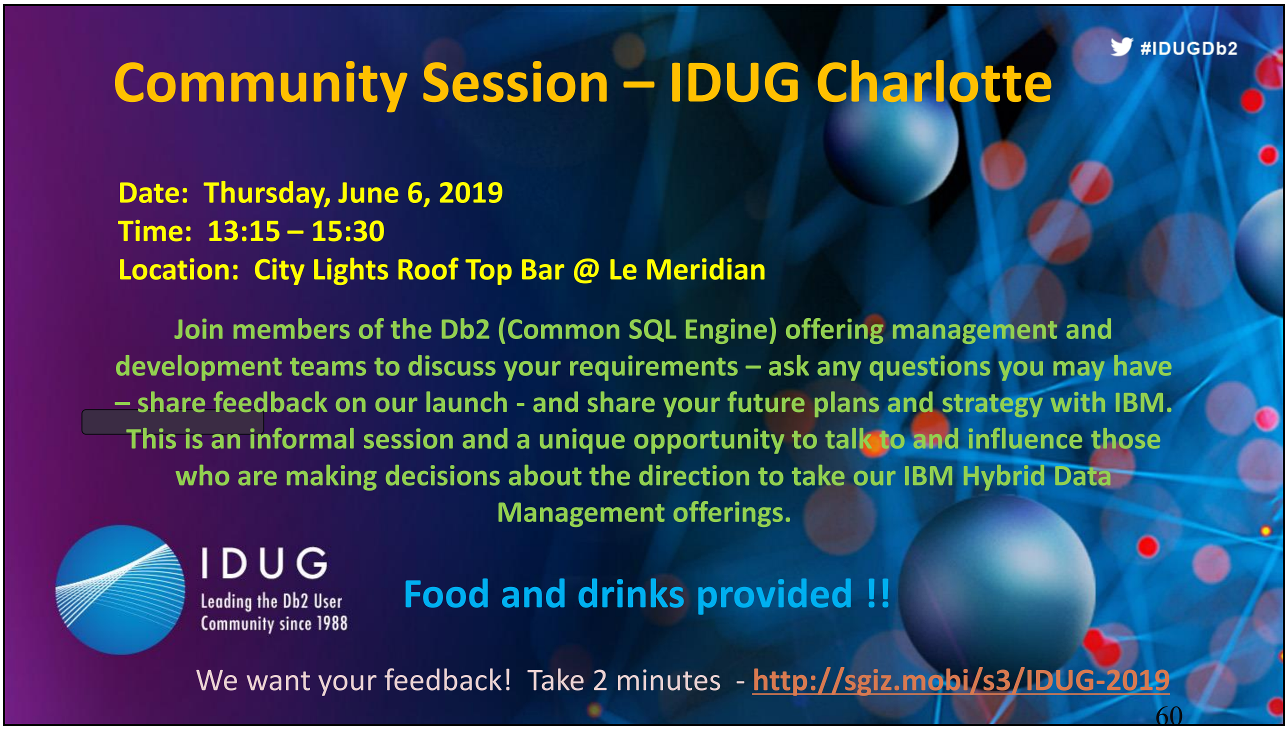
HADR Major Version Upgrade – The Future

- In a perfect world we have:
  - Rolling Version Upgrades
    - Treat like fix pack updates
    - Minimal, if any, disruption
    - Better system catalog table support
    - Log replay supports downlevel log record / file formats
  - Other ???

The future is no difference between rolling updates and rolling upgrades. Be able to update/upgrade versions with no or very minimal database outage.

Resources

- Db2 11.1 Knowledge Center – Performing Rolling Updates in HADR
  - https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.admin.ha.doc/doc/t0011766.html
- Db2 11.1 Knowledge Center – Upgrading HADR Environments
  - https://www.ibm.com/support/knowledgecenter/en/SSEPGG_11.1.0/com.ibm.db2.luw.qb.upgrade.doc/doc/c0070028.html
- Db2 Upgrade Portal
  - http://www.ibm.com/software/data/db2/upgrade/portal
- My Blog - Db2 Availability & Recovery Insider
  - https://www.idug.org/p/bl/et/blogid=703



#IDUGDb2

Community Session – IDUG Charlotte

Date: Thursday, June 6, 2019
Time: 13:15 – 15:30
Location: City Lights Roof Top Bar @ Le Meridian

Join members of the Db2 (Common SQL Engine) offering management and development teams to discuss your requirements – ask any questions you may have – share feedback on our launch - and share your future plans and strategy with IBM. This is an informal session and a unique opportunity to talk to and influence those who are making decisions about the direction to take our IBM Hybrid Data Management offerings.

IDUG
Leading the Db2 User
Community since 1988

Food and drinks provided !!

We want your feedback! Take 2 minutes - http://sgiz.mobi/s3/IDUG-2019

60

Michael Roecken
IBM
roecken@ca.ibm.com
@roecken

Session code: D17

IDUG
Leading the Db2 User Community since 1988

Please fill out your session evaluation before leaving!

Please complete your evaluations before leaving.

Michael Roecken is a senior software developer with Db2 for Linux, UNIX, and Windows platforms at the IBM Toronto Lab. Michael has worked since 2000 designing, implementing and supporting various features and capabilities in the areas of: backup/restore, crash/rollforward recovery, high availability/disaster recovery, and logging/transaction management.

Please connect with Michael on Twitter at @roecken and LinkedIn.