IDUG

2024 NA **Db2** Tech Conference

What can I do with Db2 profile tables

Tammie Dang
IBM Corp

Session Code: ZLN1 | Platform: Db2 for z/OS

Agenda

- Profile table overview
- Functionalities
  - ➤ Setting special registers and global variables
  - ➤ Monitoring remote connections and threads
  - ➤ Modeling and influencing query's access path and acceleration
  - ➤ Optimizing chance of successful online schema change

Profile table functionalities

User can create and populate the Db2 profile tables to

1. Set special registers and global variables
2. Monitor distributed application threads and connections
3. Discover and control secure TCP/IP connectivity
4. Allow stored procedures to share locks with any transactions that they invoke in an RRS context
5. Override the RELEASE(DEALLOCATE) package bind option
6. Optimize certain subsystem parameters used for SQL statement's access path selection
7. Model a production environment on a test subsystem
8. Evaluate queries for IDAA server acceleration

1-5: system profile
6-8: query's access path related

Session ZLN6
Tuesday 2PM

https://www.ibm.com/docs/en/db2-for-zos/13?topic=tables-dsn-profile-attributes

Profile specification

- A set of criteria that identifies a specific context (threads, connections, user ID, application package, etc) and actions to be taken

- Users insert rows as criteria (categories) in table DSN_PROFILE_TABLE. Each row defines a profile with PROFILEID column.

- Users insert rows as actions in table DSN_PROFILE_ATTRIBUTES. RI relationship with DSN_PROFILE_TABLE on the PROFILEID column.

- Tables and indexes are created by Db2 installation job DSNTIJSG

Profile table context and actions

Identify an application process context and actions to be taken

- Filtering categories in the table DSN_PROFILE_TABLE columns:

    - LOCATION (IP addr, location name/alias, domain name)
    - PRDID (product ID)
    - ROLE, AUTHID (primary authorization ID), or both
    - PLANNAME
    - COLLID (collection ID), PKGNAME (package name), or both
    - CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WRKSTNNAME

    Order of predence

- Actions in the table DSN_PROFILE_ATTRIBUTES columns KEYWORDS, ATTRIBUTE1 – ATTRIBUTE3:

    - Monitor and control remote server threads/connections
    - Set special registers/global variables for remote server threads
    - Modeling production environment on a test subsystem
    - Evaluate queries for IDAA offload
    - Override RELEASE(DEALLOCATE) bind option
    - …

- RI relationship on the PROFILEID column

Starting and stopping profiles

- The Profile tables are loaded in memory when profile is started via the –START PROFILE command

- DDF address space needs to be started

```
-DB2ASTA PROFILE
DSNT761I  -DB2A DSNT1RSP START PROFILE DETECTED
                DDF IS NOT LOADED. PROFILE ROWS FOR SYSTEM
                MONITORING WILL NOT BE ACTIVATED.
DSNT741I  -DB2A DSNT1SDV START PROFILE IS COMPLETED.
DSN9022I  -DB2A DSNT1STR 'START PROFILE' NORMAL COMPLETION
```

- Accepted and rejected rows are in DSN_PROFILE_HISTORY and DSN_PROFILE_ATTRIBUTES_HISTORY tables, STATUS column

- Changes can be restarted without stopping. Use the DSN_PROFILE_TABLE.PROFILE_ENABLE column.

6

Starting and stopping profiles

- For profiles that affect access path selection, invalidate cached dynamic queries

- Max 4096 rows active in the DSN_PROFILE_TABLE

- Profile can be started automatically when Db2 is started. Subsystem parameter PROFILE_AUTOSTART=YES

Profile criteria

- Certain categories are not applicable to certain actions

- Wild card, partial wildcard are allowed for certain categories

- Rules:

1. Accept only the newest duplicate profile in PROFILE_TIMESTAMP column

2. Apply all profiles from different filtering categories.

DSN_PROFILE_TABLE

PROFILEID	LOCATION	AUTHID	PRDID
10			DSN13011
11		TAMMIE	
12	*		

Profile criteria

3. Exact values take priority over wildcards: USRT001, USRT*, *

4. Apply only 1 profile for same filtering category

DSN_PROFILE_TABLE

PROFILEID	AUTHID
20	*
21	TAMMIE

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORDS
20	GLOBAL VARIABLE
21	MONITOR CONNECTIONS

5. Order of precedence on page 5

SET SPECIAL REGISTERS AND GLOBAL VARIABLES

Using profile table to set special registers and global variables

- No need to incur cost of changing applications

- Filtering criteria in the table DSN_PROFILE_TABLE, columns:
  - ➢ LOCATION (remote app)
  - ➢ PRDID (remote app)
  - ➢ ROLE, AUTHID, or both
  - ➢ COLLID, PKGNAME, or both
  - ➢ One of CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WORKSTNNAME

- Db2 13 GA supports setting some special register and global variable for local threads

Using profile table to set special registers and global variables

- Actions in the table DSN_PROFILE_ATTRIBUTES, columns:

  - ➢ KEYWORDS = 'SPECIAL_REGISTER' or 'GLOBAL_VARIABLE'
  - ➢ ATTRIBUTE1 is the SET SQL statement (max length 1024 bytes, no expression, etc)
  - ➢ ATTRIBUTE2
    - ➢ **Null**: applicable to remote app. SET statements are processed when the first package is loaded (1st SQL in first package is loaded)
    - ➢ **1**: applicable to local app (CURRENT LOCK TIMEOUT and DEADLOCK_RESOLUTION_PRIORITY only). SET statements are processed when each package is loaded (1st SQL in each package is executed)
    - ➢ **2**: applicable to both remote and local app's (CURRENT LOCK TIMEOUT and DEADLOCK_RESOLUTION_PRIORITY only)

Example

DSN_PROFILE_TABLE

PROFILEID	LOCATION	ROLE	COLLID	CLIENT_APPLNAME
1	COLUMBUS			
2		DBA_ROLE		
3			COLLECTION1	
4				NEWAPPL

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE1	ATTRIBUTE2
1	SPECIAL_REGISTER	SET CURRENT REFRESH AGE = 'ANY'	Null
4	GLOBAL_VARIABLE	SET SYSIBMADM.MOVE_TO_ARCHIVE ='E'	Null
2	SPECIAL_REGISTER	SET CURRENT LOCK TIMEOUT = 15	2
2	GLOBAL_VARIABLE	SET SYSIBMADM.DEADLOCK_RESOLUTION_PRIORITY = 5	2

Recommendation

Recommendation: Don't mix SQL statement SET in application and Profile table

Note: special register values are *not* saved/restored at package switching

Example: SET CURRENT LOCK TIMEOUT = 10 for PackageA in Profile table

```
PackageA:
    SELECT … FROM T1
    SELECT … FROM T2
    (Language) CALL PackageB;

    PackageB:
        SET CURRENT LOCK TIMEOUT = 20
        INSERT INTO ..
        return;
    UPDATE T3 …
```

What's the timeout interval for each SQL statement?

Recommendation (cont)

Recommendation: Don't mix SQL statement SET in application and Profile table

Note: special register values are *not* saved/restored at package switching

Example: SET CURRENT LOCK TIMEOUT = 10 for PackageA in Profile table

PackageA:	
SELECT … FROM T1	locks timeout after 10 seconds
SELECT … FROM T2	locks timeout after 10 seconds
(Language) CALL PackageB;	
PackageB:	
SET CURRENT LOCK TIMEOUT = 20	
INSERT INTO ..	locks timeout after 20 seconds
return;	
UPDATE T3 …	locks timeout after 20 seconds

MONITOR REMOTE CONNECTIONS AND THREADS

Using profile table to monitor remote connections and threads

- Manage remote application workloads -> Db2 resource consumption

- Get notifications via console messages

- Monitor all or a specific remote connection(s), and all or a specific remote thread(s)

- Filtering criteria in the table DSN_PROFILE_TABLE, column:

  ➢ LOCATION – only category for connection monitoring
  ➢ PRDID (remote app)
  ➢ AUTHID, ROLE, or both
  ➢ COLLID, PKGNAME, or both
  ➢ One of CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WORKSTNNAME

Monitoring remote connections

- Actions to be taken when number of connections reaches the specified threshold

- Columns in the table DSN_PROFILE_ATTRIBUTES:

  - ➢ KEYWORDS = 'MONITOR ALL CONNECTIONS' – total cumulative # of all remote (active/inactive) connections
    - ▪ DSN_PROFILE_TABLE.LOCATION must be: '*', '::0', or '0.0.0.0'
  - ➢ KEYWORDS = 'MONITOR CONNECTIONS' – total # of remote (active/inactive) connections from each application server
    - ▪ DSN_PROFILE_TABLE.LOCATION: IP address or domain name
  - ➢ ATTRIBUTE1: fail or allow next request. Issue console message DSNT771I, DSNT772I, DSNT774I or DSNT773I
    - ▪ EXCEPTION, EXCEPTION_DIAGLEVEL1, EXCEPTION_DIAGLEVEL2, EXCEPTION_DIAGLEVEL3. SQLCODE -30041.
    - ▪ WARNING, WARNING_DIAGLEVEL1, WARNING_DIAGLEVEL2, WARNING_DIAGLEVEL3
  - ➢ ATTRIBUTE2: threshold (less than or equal to CONDBAT subsystem parameter)

Monitoring remote threads

- Actions to be taken when number of concurrent active threads reaches the specified threshold

- Columns in the table DSN_PROFILE_ATTRIBUTES:

  - ➢ KEYWORDS = 'MONITOR ALL THREADS' – total # of concurrent active threads from all application servers
    - ▪ DSN_PROFILE_TABLE.LOCATION must be: '*', '::0', or '0.0.0.0'
  - ➢ KEYWORDS = 'MONITOR THREADS' – total # of remote connections from each application server
    - ▪ Filtering criteria on IP address, domain, location, product ID, role, authID, collection ID, package, and client user, client application, client workstation names
  - ➢ ATTRIBUTE1: queue/suspend next request or just issue console message DSNT771I , T772I, T774I or T773I
    - ▪ EXCEPTION, EXCEPTION_DIAGLEVEL1, EXCEPTION_DIAGLEVEL2, EXCEPTION_DIAGLEVEL3. Fail request with SQLCODE -30041 if too many suspended threads already (Attribute3 threshold).
    - ▪ WARNING, WARNING_DIAGLEVEL1, WARNING_DIAGLEVEL2, WARNING_DIAGLEVEL3
  - ➢ ATTRIBUTE2: threshold (less than or equal to MAXDBAT subsystem parameter)
  - ➢ ATTRIBUTE3: suspended thread threshold

Monitoring remote idle threads

- Actions to be taken when a thread has been idle for longer than the specified time threshold

- Columns in the table DSN_PROFILE_ATTRIBUTES:

  ➢ KEYWORDS = 'MONITOR IDLE THREADS'
    - DSN_PROFILE_TABLE.LOCATION must be: '*', '::0', or '0.0.0.0'
    - Filtering criteria on IP address, domain, location, product ID, role, authID, collection ID, package, and client user, client application, client workstation names

  ➢ ATTRIBUTE1: abort the thread, pool the DBAT, terminate connection or allow thread to remain idle Issue console message DSNT771I , DSNT772I, or DSNT773I
    - EXCEPTION, EXCEPTION_DIAGLEVEL1, EXCEPTION_DIAGLEVEL2, EXCEPTION_ROLLBACK, EXCEPTION_ROLLBACK_DIAGLEVEL1, EXCEPTION_ROLLBACK_DIAGLEVEL2
    - WARNING, WARNING_DIAGLEVEL1, WARNING_DIAGLEVEL2, WARNING_MESSAGE_FOR_IDLE_TIMEOUT

  ➢ ATTRIBUTE2: time threshold in seconds
    - 0 – 9999
    - Can be shorter or longer than IDTHTOIN subsystem parameter
    - Threads are checked every 2 minutes

Example

DSN_PROFILE_TABLE

PROFILEID	LOCATION	ROLE	AUTHID	PRDID
1	TEST.INSTITUTE.COM			
2		DEVELOPER		
3			DATA_SCIENTIST*	
4				JCC*
5		DBA_ROLE	SMIKE	

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE1	ATTRIBUTE2	ATTRIBUTE3
1	MONITOR CONNECTIONS	WARNING	50 (connections)	NULL
2	MONITOR THREADS	EXCEPTION	100 (active threads)	100 (suspended threads)
3	MONITOR IDLE THREADS	EXCEPTION	150 (seconds)	NULL
4	MONITOR IDLE THREADS	EXCEPTION_ROLLBACK	150 (seconds)	NULL
5	MONITOR THREADS	EXCEPTION	200 (active threads)	NULL

Example

DSN_PROFILE_TABLE

PROFILEID	AUTHID
30	USER1
31	USER*
32	*

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE1	ATTRIBUTE2
30	MONITOR THREADS	EXCEPTION	100
31	MONITOR THREADS	EXCEPTION	50
32	MONITOR THREADS	EXCEPTION	20

USER1 can have max 100 threads. USER99 can have max 50 threads. TAMMIE can have max 20 threads. If USER1 reaches 20 threads, no other auth ID can execute.

Messages

DSNT771I A MONITOR PROFILE *WARNING/EXCEPTION* CONDITION OCCURRED *10* TIME(S)

DSNT772I A MONITOR PROFILE *WARNING/EXCEPTION* CONDITION OCCURRED *10* TIME(S) IN PROFILE ID=*X* WITH FILTER SCOPE=*Y*

- issued at a minimum interval of 5 minutes. Check statistics class 4 IFCID 402 trace record for any profile warning or exception condition that occurs in a statistics interval.

DSNT773I SERVER DISTRIBUTED AGENT WITH LUWID=*luwid* THREAD-INFO=*thread-information* FOR LOCATION=*location* RECEIVED *event-type* WARNING DUE TO PROFILE ID=*profile-id* OCCURRED *10* TIME(S)

- issued every time a thread exceeds threshold with WARNING_LEVEL3

DSNT774I SERVER DISTRIBUTED AGENT WITH LUWID=*luwid* THREAD-INFO=*thread-information* FOR LOCATION=*location* RECEIVED *event-type* EXCEPTION DUE TO PROFILE ID=*profile-id* OCCURRED *10* TIME(S)

- issued every time a thread exceeds threshold with EXCEPTION_LEVEL3

SHARE LOCKS STORED PROC <-> RRS TRANSACTIONS

Timeout problem

1. A remote application executes an SQL statement which obtains locks

2. The remote application invokes a stored procedure

3. The stored procedure invokes a CICS or IMS transaction in RRS context

4. CICS or IMS transaction executes an SQL statement and gets timeout error on resources acquired by step 1.

Solution: remote application thread (step 1) and CICS/IMS transaction (step 4) in a global transaction and share locks to avoid lock contention

a) User specifies stored procedure names and keyword 'SHARE_LOCKS' in the Db2 profile tables
b) Stored procedure sets the GXID (global transaction ID) on the private context
c) CICS/IMS extracts the GXID and signons to Db2 with that GXID
d) Db2 sets the same lock compatibility token for all agents in the same global transaction

Share locks

- Filtering criteria in the table DSN_PROFILE_TABLE, column:

  ➢ LOCATION only
  ➢ PRDID only
  ➢ AUTHID, ROLE, or both
  ➢ COLLID, PKGNAME, or both
  ➢ One of CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WRKSTNNAME

- Columns in the table DSN_PROFILE_ATTRIBUTES:

  ➢ KEYWORDS = 'SHARE_LOCKS'
     ▪ ATTRIBUTES1: 'PROCEDURE LIST = schema1.name1, schema2.name2, …'

DSN_PROFILE_TABLE

PROFILEID	AUTHID
50	GAYA

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE1
50	SHARE_LOCKS	PROCEDURE LIST = 'DEPT0F1.RESOURCEPLAN, DEPT0F2.EMAILSCRUB'

OPTIMIZE SUBSYSTEM PARMS FOR SQL

Optimizing subsystem parameters for SQL statements

- Overriding the subsystem parameters for specific SQL statement's access path selections:
  - NPGTHRSH – compared to SYSTABLES.NPAGESF, SYSTABSTATS.NPAGES to choose index access
  - STARJOIN – to allow star join access path
  - SJTABLES – minum number of tables for star join processing
- Filtering criteria in the table DSN_PROFILE_TABLE, columns:
  - PLANNAME = '*', COLLID, and PKGNAME (specify all 3)

Optimizing subsystem parameters for SQL statements

- Columns in the table DSN_PROFILE_ATTRIBUTES:

  - KEYWORDS = 'NPAGES THRESHOLD'
    - ATTRIBUTES2: threshold value
  - KEYWORDS = 'STAR JOIN'
    - ATTRIBUTES1 = 'DISABLE' or 'ENABLE'
  - KEYWORDS = 'MIN STAR JOIN TABLES'
    - ATTRIBUTES2: threshold value between 3-225

DSN_PROFILE_TABLE

PROFILEID	PLANNAME	COLLID	PKGNAME
60	*	MYCOLLID	MYPACKAGE

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE2
60	NPAGES THRESHOLD	50000
60	STAR JOIN	DISABLE

MYCOLLID.MYPACKAGE

```
SELECT * FROM T1 WHERE C1 > 'abc';
SELECT * FROM SALES S, TIME T,   PRODUCT P,
LOCATION L
    WHERE S.TIME=T.ID AND
            S.PRODUCT=P.PRODUCT AND
            S.LOCATION=L.ID AND
            T.YEAR=2005 AND
            P.CLASS='AUDIO' AND
            L.LOCATION='SAN JOSE';
```

MODEL A PRODUCTION SYSTEM

Modeling a production environment

Improve accuracy of an SQL statement's access path testing in a test subsystem

1. Gather subsystem parameters on production system and set them in test system (NPGTHRSH, PARAMDEG, STARJOIN)

   ➢ DSNTEJ6Z invoking ADMIN_INFO_SYSPARM stored procedure

2. EXPLAIN the SQL on production system , then query the PLAN_TABLE.IBM_SERVICE_DATA to gather:

   ➢ Processor speed, number of processors –> set the subsystem parameters SIMULATED_CPU_SPEED and SIMULATED_CPU_COUNT in test system

   ➢ Maximum number of RID blocks, sort pool size –> use profile table to specify them in test system

   ➢ Query is on *https://www.ibm.com/docs/en/db2-for-zos/13?topic=performance-modeling-production-environment-test-subsystem*

3. Use –DIS BUFFERPOOL command on production system and use profile table to specify bufferpool sizes in test system

Modeling a production system

- Filtering criteria in the table DSN_PROFILE_TABLE, column:

  ➢ Only PROFILEID

- Columns in the table DSN_PROFILE_ATTRIBUTES:

  ➢ KEYWORDS = 'MAX_RIDBLOCKS'
    - ATTRIBUTES2: value from PLANTABLE.IBM_SERVICE_DATA on production system
  ➢ KEYWORDS = 'SORT_POOL_SIZE'
    - ATTRIBUTES2 = value from PLANTABLE.IBM_SERVICE_DATA on production system
  ➢ KEYWORDS = 'BPx'   *where x is the bufferpool ID*
    - ATTRIBUTES2: value from –DISPLAY BUFFPERPOOL command on production system

EVALUATE QUERY FOR ACCELERATION

Acceleration on the IDAA

Influence IDAA offload criteria

- Filtering criteria in the table DSN_PROFILE_TABLE, column :
  - None (global scope)
  - AUTHID and LOCATION
  - PLANNAME, COLLID, and PKGNAME
- DSN_PROFILE_ATTRIBUTES.KEYWORDS:
  - ➢ ACCEL_NAME_EXPLAIN - the Db2 Optimizer evaluates whether a dynamic SQL query can be offloaded to a specific IBM Db2 Analytic Accelerator
  - ➢ ACCEL_TABLE_THRESHOLD - query having a table cardinality less than threshold is not accelerated
  - ➢ ACCEL_RESULTSIZE_THRESHOLD - query returning over number of threshold rows is not accelerated
  - ➢ ACCEL_TOTALCOST_THRESHOLD – query having a total cost less than threshold is not accelerated
- CURRENT QUERY ACCELERATION or QUERYACCELERATION bind option = 'ENABLE', 'ENABLE WITH FAILBACK' or 'ELIGIBLE', or 'ALL'

Acceleration on the IDAA

- Columns in the table DSN_PROFILE_ATTRIBUTES:

    ➢ KEYWORDS = 'ACCEL_NAME_EXPLAIN'
    - ATTRIBUTES1: name of the real or virtual accelerator
    ➢ KEYWORDS = 'ACCEL_TABLE_THRESHOLD'
    - ATTRIBUTES2:
        - number of rows
        - -1: this check is not used
        - default is 1,000,000 rows
    ➢ KEYWORDS = 'ACCEL_RESULTSET_THRESHOLD'
    - ATTRIBUTES2:
        o number of rows in thousands
        o -1: this check is not used (default)
    ➢ KEYWORDS = 'ACCEL_TOTALCOST_THRESHOLD'
    - ATTRIBUTES3:
        - floating point value
        - -1: this check is not used
        - Default: 5,000

DSN_PROFILE_TABLE

PROFILEID	AUTHID	LOCATION
17	SMITH	
18		CHARLOTTE

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE2
17	ACCEL_TABLE_THRESHOLD	800000
18	ACCEL_RESULTSET_THRESHOLD	20

DDL BREAK-IN ENHANCEMENT

Problem statement

DBA cannot easily route threads using RELEASE(DEALLOCATE) packages to RELEASE(COMMIT) behavior when issuing DDL statements, resulting in DDL timeout.

- *Packages are bound with RELEASE(DEALLOCATE) for better performance with the drawback that they may prevent successful DDL.*

- *DDL and **static SQL** in packages are serialized using the package lock*

    Package A has SELECT * FROM ORDERS   -> share lock on package A

    ALTER TABLE ORDERS ADD COLUMN SHIPDATE DATE   -> exclusive lock on package A

- *Subsystem parameter PKGREL_COMMIT: requires synchronized COMMIT among threads*

- *-MODIFY DDF,PKGREL(COMMIT): applicable to all remote applications*

Db2 12 scenario

- The DBA binds package DEALLOCATE.A with RELEASE(DEALLOCATE) option
- Threads execute DEALLOCATE.A
- The DBA issues a DDL statement
- All threads running DEALLOCATE.A are quiesced
- DDL statement may complete or timeout/deadlock

- Pain point:
- failed DDL prevents promotion of new applications

Goal

The DBA can easily route applications to RELEASE(COMMIT) behavior to optimize the likelihood of DDL success

Db2 13 scenario

1. The DBA binds package DEALLOCATE.A with RELEASE(DEALLOCATE)

2. Thread 1 executes package DEALLOCATE.A

3. The DBA inserts a row in the Profile table with the RELEASE_PACKAGE and COMMIT attribute for the application process

4. The DBA issues the –STA PROFILE command

5. The DBA issues the SET CURRENT LOCK TIMEOUT, SET SYSIBMADM.DEADLOCK_RESOLUTION_PRIORITY before the DDL statement

6. New threads load package DEALLOCATE.A and set RELEASE(COMMIT) option

7. Existing thread 1 releases package DEALLOCATE.A on next COMMIT

The solution

- New keyword RELEASE_PACKAGE in DSN_PROFILE_ATTRIBUTES to demote RELEASE(DEALLOCATE) to RELEASE(COMMIT) behavior
  - ATTRIBUTE1 = COMMIT
  - ATTRIBUTE2
    - ➢ 1 - local threads only
    - ➢ null – remote threads only
    - ➢ 2 – both local and remote threads
- Filtering criteria in DSN_PROFILE_TABLE for local threads:
  - AUTHID, ROLE, or both
  - COLLID, PKGNAME, or both
  - One of CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WORKSTNNAME

The solution (cont)

- Filtering criteria in DSN_PROFILE_TABLE for remote server threads:
  - ➤LOCATION only
  - ➤PRDID only
  - ➤AUTHID, ROLE, or both
  - ➤COLLID, PKGNAME, or both
  - ➤One of CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WORKSTNNAME

- Profile is applied at
  - ➤Each package is loaded (1st SQL statement execution)
  - ➤COMMIT, ROLLBACK

Finding package dependency

SELECT BNAME, BQUALIFIER, BTYPE, DCOLLID, DNAME FROM SYIBM.SYSPACKDEP
WHERE BNAME = '*xxxx*' AND BQUALIFIER = '*yyyy*' AND BTYPE = '*z*'

SYSPACKDEP

BNAME	BQUALIFIER	BTYPE	DCOLLID	DNAME
EMPLOYEE	REGION1	T	COLLID1	PKGA
EMPLOYEE	REGION1	T	COLLID2	PKGB
EMP_IDX1	REGION1	I	COLLID1	PKGA
EMP_IDX2	REGION2	I	COLLID3	PKGC
EMP_IDX2	REGION2	I	COLLID2	PKGB

Finding package dependency

SELECT BNAME, BQUALIFIER, BTYPE, DCOLLID, DNAME FROM SYIBM.SYSPACKDEP
WHERE BNAME = '*EMPLOYEE*' AND BQUALIFIER = '*REGION1*' AND BTYPE = '*T*'

SYSPACKDEP

BNAME	BQUALIFIER	BTYPE	DCOLLID	DNAME
EMPLOYEE	REGION1	T	**COLLID1**	**PKGA**
EMPLOYEE	REGION1	T	**COLLID2**	**PKGB**
EMP_IDX1	REGION1	I	COLLID1	PKGA
EMP_IDX2	REGION2	I	COLLID3	PKGC
EMP_IDX2	REGION2	I	COLLID2	PKGB

Profile tables

DSN_PROFILE_TABLE

PROFILEID	COLLID	PKGNAME	PROFILE_TIMESTAMP
99	COLL1	PKGA	2021-02-28-14.55.56.780277
99	COLL2	PKGB	2021-02-28-14.55.57.780277

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORD	ATTRIBUTE1	ATTRIBUTE2
99	RELEASE_PACKAGE	COMMIT	2

Notes

- RELEASE(COMMIT) may keep package lock at end of unit of work
  - ➢WITH HOLD cursors
  - ➢KEEPDYNAMIC(YES) bind option
- -STOP PROFILE command or disable/delete rows in DSN_PROFILE_ATTRIBUTES table and restart Profile
- Selectively disable High-Performance DBATs:
  - ➢connection will be made inactive
  - ➢thread will be pooled
- IFCID 177 written at package loading