Data and AI on IBM Z

Adrian Burke
Db2 for z/OS SWAT Team
agburke@us.ibm.com

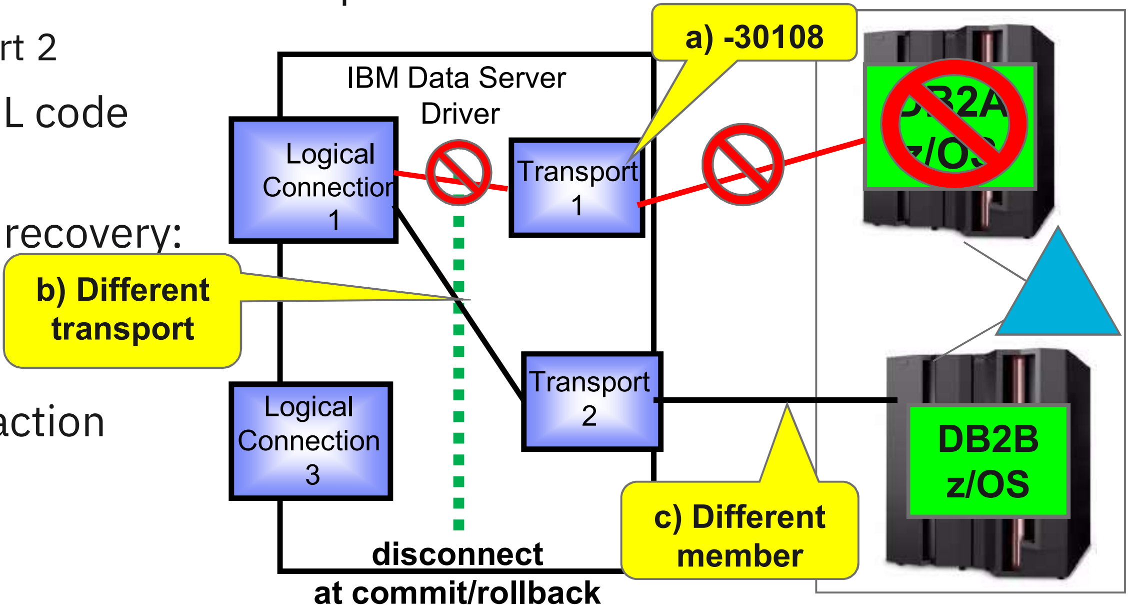
Distributed Best Practices for Db2 Today … and Tomorrow

Agenda

- **High Availability**
  - Enabling Sysplex workload balancing
- **Why is my workload imbalanced?**
- **Controlling connection usage**
- **Controlling thread usage**
- **Monitoring the ‘Black Box’ of distributed access**
- **Caution for zIIPs**

High Availability – enable Sysplex Workload Balancing

- **With WLB, application connectivity is *NOT* lost if DB2A is brought down or crashes**
  - The intent is to route around connection failures and mask them from the applications
    - This also avoids stale connection exceptions in app server connection pools
  - If connectivity to DB2A is lost i.e. taken down for maintenance or crashes AND if the first SQL statement of a transaction encounters this situation:
    a) Driver receives network failure (-30108)
       ✓ That transport would then be recycled to avoid a stale connection exception
    b) Driver seamlessly routes transaction to another transport on another member
       ✓ Logical connection 1 moves to transport 2
    c) Application receives no negative SQL code and continues processing
  - If the connection is lost during a unit of recovery:
    a) Thread is rolled back and connection re-established to Db2
    b) But application must re-drive transaction



High Availability – Sysplex WLB …

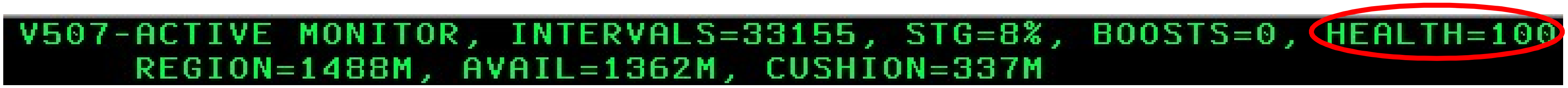
- **Recommendation : enableSysplexWLB=TRUE**
- **Provides intelligent workload distribution at transaction level**
  - At transaction boundary next SQL can be routed to any member of the group based on 3 main factors:
    - Displaceable CPU (GCP+zIIP) on the LPAR
    - Enclave WLM Service class goal attainment and queuing
    - Db2 health
      - ✓ Based on 31-bit virtual storage constraints
      - ✓ Also lowered as the number of connections in use approaches CONDBAT
      - ✓ Can be monitored with -DIS DDF DETAIL and -DIS THREAD(*) TYPE(SYSTEM)

Workload imbalance …

- **Enabling Sysplex WLB means the driver should distribute the work to the appropriate Db2 based on (IWMSRSRS WLM interface) 3 main factors:**
  - – Displaceable capacity
    - Sysplex WLB favors LPARs with more CPs/zIIPs over smaller LPARs
      - ✓ Also favors single member per LPAR because system weight is divided by # of servers on it
    - After the initial connection using sysplex distributor the driver takes over
      - ✓ Changing the Sysplex Distributor DISTMETHOD will not have any effect on DDF workload distribution
    - **Recommendation**:
      - ✓ Try to define and maintain a symmetric configuration
  - – Performance index of enclaves as well as queue time
    - If threads delayed on LPAR 1 miss their WLM goal, they may ‘slosh’ to a Db2 on another LPAR
    - If the WLM goal is too loose the driver will not re-route work in a timely manner and DBAT requests will queue, possibly encroaching on CONDBAT
    - **Recommendation**:
      - ✓ WLM goals must be realistic and ACHIEVABLE i.e. may mean moving from Velocity to Response time goal
      - ✓ Monitor -DIS THREAD(*) DETAIL DSNV482I during peak times as well as RMF 72-3 Workload activity reports to determine effectiveness of the WLM policy and adjust if necessary

Workload imbalance …

- **Enabling Sysplex WLB means the driver should distribute the work to the appropriate Db2 based on (IWMSRSRS WLM interface) 3 main factors …**
  - Db2 system health reported to WLM
    - Can be seen in DSNL094I WLMHEALTH=*x* And DSNV507I message below

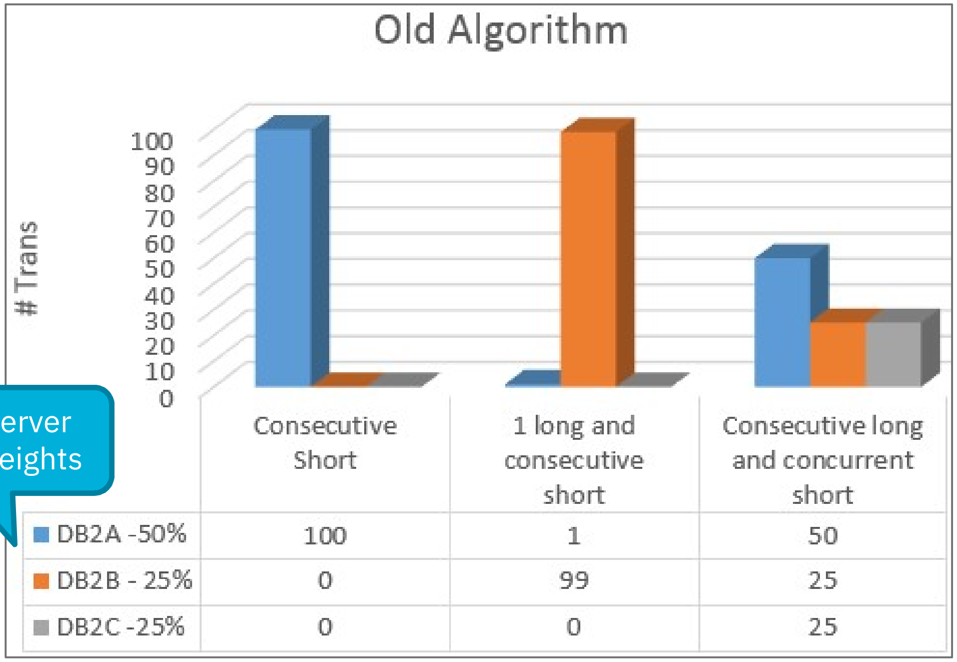


```
V507-ACTIVE MONITOR, INTERVALS=33155, STG=8%, BOOSTS=0, HEALTH=100
    REGION=1488M, AVAIL=1362M, CUSHION=337M
```

    - **Recommendations**:
      - ✓ Ensure active+inactive connections is < 80% of CONDBAT reached
        - » At 80% (DSNL074I issued) health reduced to 50%, at 90% health reduced to 25%
      - ✓ Monitor 31-bit storage via IFCID 225 and maintain a cushion based on threads/data sets
        - » DSNV508I issued at 88% of 31-bit private virtual storage consumed and Db2 health is lowered
      - ✓ Health also lowered if BOOSTS > 0
        - » Monitor DSNV522I (PI29671) displays boosted agent

NEW Sysplex WLB algorithm

- **Historically the distribution of transactions (accounting records) was not aligned with the relative WLM weighting**
  - For consecutive trans the highest ranked system (member) would always receive the next transaction until it was loaded to its respective weight with concurrent transports, only then work would spread around the group
  - Customer experience:
    - SYS1 – DB2A Weight=10
      - ✓ 918 trans ~ 86%
    - SYS2 – DB2B Weight=6 & DB2C Weight=6
      - ✓ 182 trans ~ 14% combined
      - ✓ ** If > 1 member on LPAR weight is divided by # of subsystems
  - With a combination of long and short running transactions:
    - If long a tran ran first it would occupy the highest weighted member and the other trans would run on the next member
    - Concurrency was needed to load balance the rest of the data sharing members

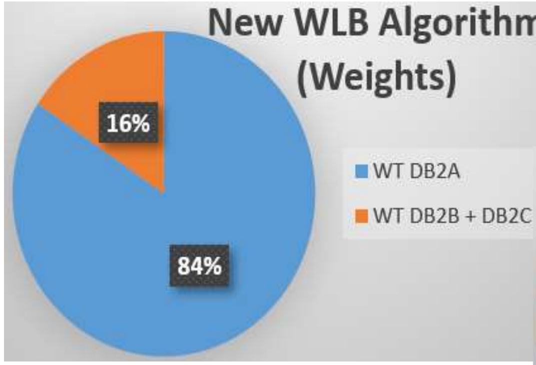


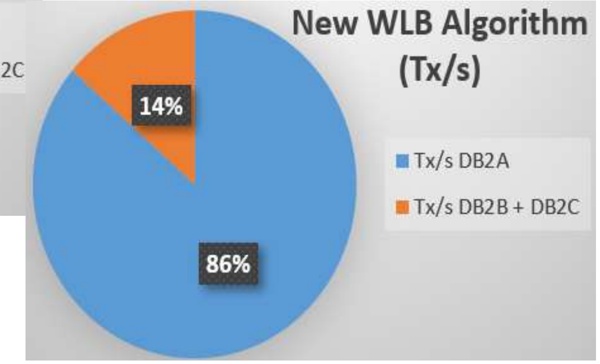
	Consecutive Short	1 long and consecutive short	Consecutive long and concurrent short
🟦 DB2A -50%	100	1	50
🟧 DB2B - 25%	0	99	25
◼ DB2C -25%	0	0	25

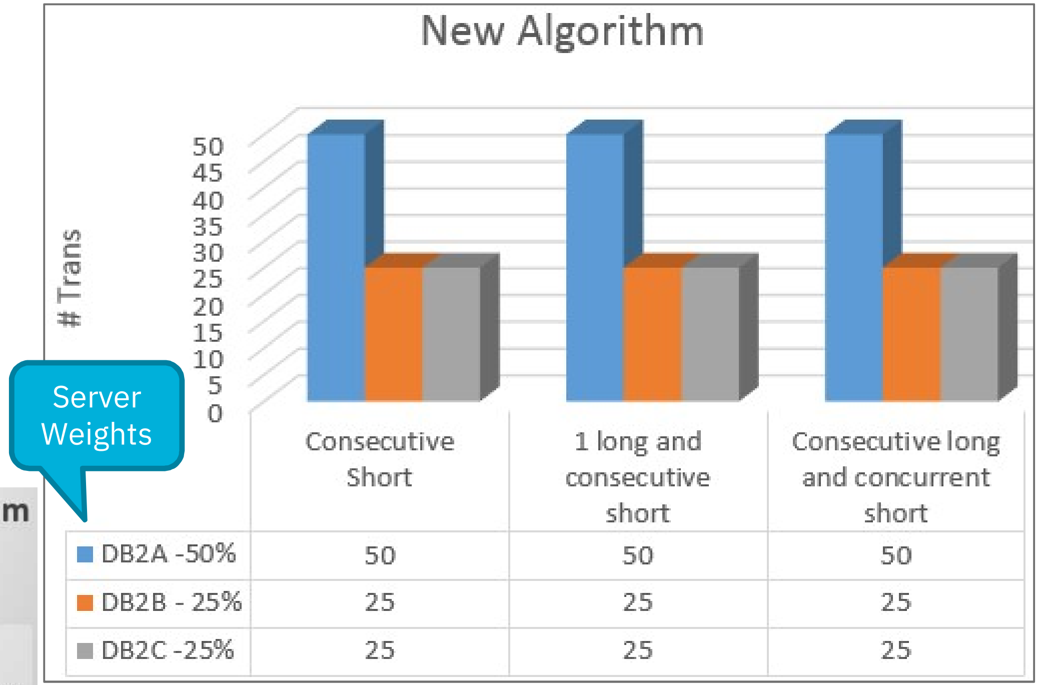
Server Weights

NEW Sysplex WLB algorithm

- **With Db2 Connect v11.1 M4 FP4 the algorithm will be adjusted - PH05977 JCC 4.25.14**
  - Upon the next getTransport() instead of starting over with the highest weighted member, the algorithm will resume where it left off in the list, thus ensuring workload is routed around the group based on the server listing preference
  - Some customers in the past have resorted to DISTMETHOD ROUNDROBIN in the Sysplex distributor, and turning off Sysplex WLB to more evenly distribute the work
    - ***BUT*** this only distributes work at a connection boundary, and will NEVER re-distribute those connections
    - Moving work to other servers requires net-new connections
  - Customer experience: *within 2% of ideal*







Server Weights

	Consecutive Short	1 long and consecutive short	Consecutive long and concurrent short
DB2A -50%	50	50	50
DB2B - 25%	25	25	25
DB2C -25%	25	25	25

Profile monitoring

- **System profiling monitoring**
  - – Goal is to allow the specification of ZPARM/Special Register values at more granular level
  - – Monitor profiles can be used to warn about, or limit the consumption of Db2 resources
    - Monitor profiles remote threads and connections (TCP/IP)
      - ✓ MONITOR CONNECTIONS – more granular CONDBAT
      - ✓ MONITOR THREADS – more granular MAXDBAT
      - ✓ MONITOR IDLE THREADS – more granular IDTHTOIN (idle thread timeout)
    - Special Registers – CURRENT DEGREE, CURRENT PACKAGE PATH, CURRENT APPLICATION COMPATABILITY, CURRENT QUERY ACCELERATION…etc
  - – Monitor profiles implemented via Db2 Catalog starting in Db2 10 NFM (DSNTIJSG job)
    - SYSIBM.DSN_PROFILE_TABLE – what you filter on
      - ✓ DSN_PROFILE_HISTORY contains STATUS column containing ‘ACCEPTED’ or ‘REJECTED’
    - SYSIBM.DSN_PROFILE_ATTRIBUTES – what you monitor
      - ✓ DSN_PROFILE_ATTRIBUTES_HISTORY contains why was it REJECTED
  - – Once defined in the catalog they must be -STARTED each time DBM1 comes up to keep track of those resources and can either warn of thresholds being reached or take action
    - PROFILE_AUTOSTART Zparm avoids this via APAR PI89912
      - ✓ They <u>must</u> be started again if their definitions are altered in the Db2 catalog

Profile monitoring …

- **DSN_PROFILE_TABLE_ATTRIBUTES**
  - MONITOR CONNECTIONS – limit must be lower than CONDBAT
    - Total number of remote connections from a TCP/IP requestor, includes active and inactive connections
      - ✓ ** If you set this to 100 then the 101st connection trips the exception
    - Filtering on LOCATION column only: IP Address or Domain Name
  - MONITOR THREADS – limit must be lower than MAXDBAT
    - Total number of concurrent active remote threads that use TCP/IP on the Db2 subsystem
      - ✓ ** If set to 100, then **100** are **active** and another **100** can be **suspended** before the exception is tripped EXCEPT when filtering on IP ADDR or Domain where the request is queued from #101 to CONDBAT
    - Filtering on: LOCATION column (IP Address, domain name, location, database, location alias), or PRDID, or ROLE and/or AUTHID, or COLLID and/or PKGNAME, or ONE of CLIENT_APPLNAME, CLIENT_USERID, or CLIENT_WORKSTNNAME
  - MONITOR IDLE THREADS – timeout can be greater or less than IDTHTOIN
    - Removes IDTHTOIN for qualifying threads, so an EXCEPTION is needed for each WARNING
      - ✓ Same filtering as MONITOR THREADS
- **Statistics class 4 IFCID 402 contains information on profile activities**
  - CUR TYPE 1 INACTIVE DBATS **(**QDSTQCIT) and HWM Type 1 record suspended threads in Db2

Using Profiles for Upgrades

- **Setting 'SPECIAL_REGISTER' in profiles to manipulate the CURRENT PACKAGE PATH for:**
  - Toggling between 'old' and 'new' NULLID collection ID to allow for REBINDs
    - Set profile for all inbound AUTHIDs to use new collection, until all the users of the old collection have drained off
    - Can cycle Db2 members in Data Sharing group if needed to close off last holders
  - Segregate well-behaved apps from others when using performance enhancing bind parameters
    - i.e. KEEPDYNAMIC, RELEASE(DEALLOCATE), DEGREE(ANY)
  - **For V12 Function Levels and continuous delivery**
    - Segregate driver packages into Collections based on APPLCOMPAT that need/want new function from those with no desire to change
      - ✓ Min of 2 collections: NULLID which never changes and NULLID501, NULID502, NULLID503, ect. Or just 1 which is at the same APPLCOMPAT as the Function Level Db2 z/OS is at
    - **Always keep one collection bound with APPLCOMPAT V12R1M500

Client application compatibility …

- **DDF only allows the new client behavior based on APPLCOMPAT setting**
  - **If clientApplcompat is set and NULLID collection APPLCOMPAT <501, or NULLID APPLCOMPAT > 500 and clientApplcompat is not set then SQLCODE -30025, connection rejected
    - PH08482 makes clientApplcompat optional for Db2 Connect V11.1 FP1 and higher
      - ✓ http://www-01.ibm.com/support/docview.wss?crawler=1&uid=swg1PH08482
    - **However** customers not setting clientApplcompat must test to ensure compatibility, and be aware if the DRDA flow changes it could break applications
    - **Mitigate** risk by always keeping 1 set of the NULLID packages bound at FL500 or below
      - ✓ **Must still get driver to V11.1 FP1 to use functionality beyond FL500**
- **Dynamic JCC packages (NULLID) will need to be rebound with APPLCOMPAT level**
  - Run job DSNTIJLC, modify the property that controls the current package set in your applications to match the collection ID of the new package copies i.e. NULID501 (M501)
    - For CLI or ODBC drivers, change the CLI/ODBC CurrentPackageSet configuration keyword value
    - For the IBM Data Server Driver for JDBC and SQLJ, change the DB2BaseDataSource.currentPackageSet Connection or DataSource property value
    - **OR** from the Db2 for z/OS side create a System Monitor Profile to force certain apps to:
      - ✓ Pick-up the currentPackagePath special register point to the new collection ID

EXAMPLE system profile for COLLECTION ID

```
INSERT INTO "SYSIBM".DSN_PROFILE_TABLE (
  AUTHID,   PLANNAME,   "COLLID",   PKGNAME,   LOCATION,   PROFILEID,   PROFILE_TIMESTAMP,   PROFILE_ENABLED,
  GROUP_MEMBER,   REMARKS,   "ROLE",   PRDID,   CLIENT_APPLNAME,   CLIENT_USERID,   CLIENT_WRKSTNNAME)  VALUES ()
VALUES (
‘*’                  , --VARCHAR(128)    AUTHID
NULL                 , --VARCHAR(24)     PLANNAME
NULL                 , --VARCHAR(128)    "COLLID"
NULL                 , --VARCHAR(128)    PKGNAME
NULL                 , --VARCHAR(254)    LOCATION
2                    , --INTEGER         PROFILEID
CURRENT TIMESTAMP    , --TIMESTMP        PROFILE_TIMESTAMP
'Y'                  , --CHAR(1)         PROFILE_ENABLED
NULL                 , --VARCHAR(24)     GROUP_MEMBER
‘SET CURRENT PATH TEST' , --VARCHAR(762)    REMARKS
NULL                 , --VARCHAR(128)    "ROLE"
NULL                 , --CHAR(8)         PRDID
NULL                 , --VARCHAR(255)    CLIENT_APPLNAME
NULL                 , --VARCHAR(255)    CLIENT_USERID
NULL                   --VARCHAR(255)    CLIENT_WRKSTNNAME );
INSERT INTO "SYSIBM".DSN_PROFILE_ATTRIBUTES (
    PROFILEID, KEYWORDS, ATTRIBUTE1, ATTRIBUTE2, ATTRIBUTE3, ATTRIBUTE_TIMESTAMP,REMARKS)
VALUES (
1                                , --INTEGER                                                      PROFILEID
'SPECIAL_REGISTER’               , --VARCHAR(128)                                                 KEYWORDS
'SET CURRENT PACKAGE PATH=NULLID501, NULLID’,                                                               -
-  ATTRIBUTE1
NULL                             , --INTEGER                                                      ATTRIBUTE2
NULL                             , --FLOAT                                                        ATTRIBUTE3
CURRENT TIMESTAMP                , --TIMESTMP
           ATTRIBUTE_TIMESTAMP
'TEST SETTING PACKAGE PATH’      , --VARCHAR(762)                                      REMARKS
```

- Apply **PH11200** ** HIPER ** to address DIST CPU increase after PI99529

Profile monitoring...

- **Here is an example of some entries for profiles (not all columns are shown)**
  - Correlate the rows based on PROFILEID

SYSIBM.DSN_PROFILE_TABLE

PROFILED	AUTHID/ ROLE	LOCATION – IP address, domain name, Alias	PRDID	COLLID	PKGNAME	CLIENT_APPLNAME / CLIENT_USERID / CLIENT_WRKSTNNAME	GROUP_ MEMBER	PROFILE_ ENABLED	Remarks
1	WASPRD*						DPG1	Y	DPG1 only needs PI70250/PI63531
2			JCC03* OR HTS*					Y	
3		SVL.IBM.COM						Y	Need PI63531

SYSIBM.DSN_PROFILE_ATTRIBUTES

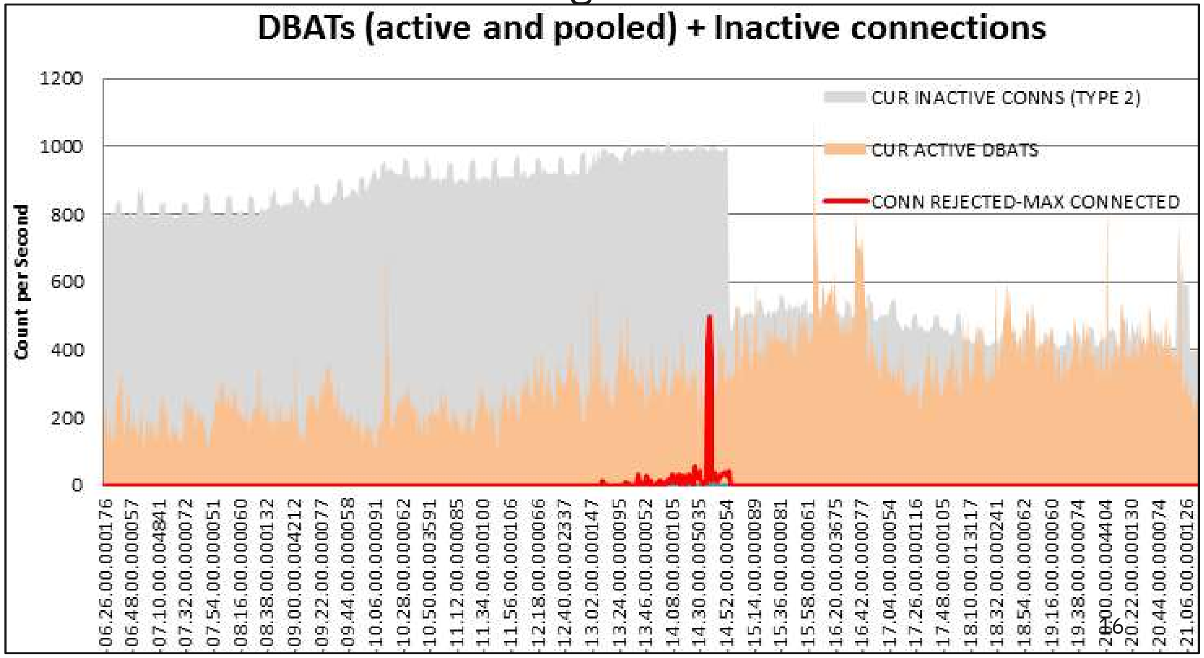
PROFILEID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2	ATTRIBUTE3	ATTRIBUTE TIMESTAMP
1	MONITOR THREADS	EXCEPTION_DIAGLEVEL3	100	NULL	2016-10-17...
2	MONITOR IDLE THREADS	WARNING_DIAGLEVEL2	180	NULL	2016-10-17…
3	MONITOR CONNECTIONS	EXCEPTION_DIAGLEVEL3	100	NULL	2016-10-17…

Profile monitoring …

- **Monitor profile example**
  - PROFILEID=1 (needs PI99529 / PH11200 (RSU1907))
    - Requests coming into member DPG1 using primary AUTHID of ‘WASPRD*’ (WASPRD001, WASPRD002, etc.) are only allowed to have 100 active threads
      - ✓ 100 more requests will queue and then the 201st connection attempt will be canceled
    - The message will have the DSNT774I for every occurrence and the detail would be in the **IFCID 406** records (PH08313 June 2019)
      - ✓ Return code 00E30507 and SQLCODE -30041
  - PROFILEID=2 **Need to add a corresponding exception threshold at IDTHTOIN limit
    - Requests coming in from PRODUCTID ‘JCC03*’ (3.0 spec of driver) **OR** all REST services using SSL that hold a DBAT are only allowed to remain idle for 180 seconds before tripping the profile
      - ✓ ProductID would also be good for blocking back-level drivers from connecting
    - If triggered this will have the DSNT772I message and 00E30501 in it, at most every **5 minutes**
  - PROFILEID=3 (needs PI63531 for DIAGLEVEL3)
    - Requests coming in from SVL.IBM.COM would only be allowed to hold 100 active and/or inactive connections
      - ✓ 101st connection will cause DSNT772I and 00E30504 reason code and SQLCODE -30081
      - ✓ DSNT774I for every occurrence of another connection attempt and the detail would be in the **IFCID 406**

Controlling connection usage

- **Common problem: Applications gain and hold connections**
  - *Applications do not release their logical connections when their transaction is complete
  - *Validation queries from app server keep connection alive by continuously pinging it
  - Lack of connection pooling leads to growth of unnecessary connections to Db2 (especially in .NET)
  - If there is pooling they are based on bigger-is-better so maxConnections parameter is overinflated
  - Timeout values for connections in the pool are disabled or do not exist
  - Drivers not enabled with Sysplex WLB (no maxtransportObjectIdleTime)
  - DSNL074I at 80% of CONDBAT, then denial of service after reaching CONDBAT
  - Graph of customer experience →
    - Inactive connections grew over the course of a few hours
    - DBAT usage was normal, only a few hundred in use, never hit MAXDBAT
    - Around 13:00 CONDBAT was hit blocking all other applications access to this member
      - ✓ Why? An OLEDB application had over 800 inactive connections associated with it



Controlling connection usage …

- **Recommendations**
  - Enable Sysplex WLB for data sharing systems (Data Server Driver level 9.7 FP 6 and up)
    - Avoid stale or abandoned connection with default maxTransportObjectIdleTime of 10 seconds
    - Even with a stand-alone member this behavior can alleviate the need to modify application server parameters
      - ✓ Can also help avoid stale connection exceptions to same member if minConnections >0
    - Set CONDBAT on each member = total maxConnections from all app server connection pools coming into the data sharing group x 1.25 (allow cushion for 1-off connections)
      - ✓ From WebSphere Admin Console Data sources > *data source* > Connection Pools
        - » General Properties Maximum connections = 10 (*default)*
      - ✓ Option to use MAXCONQN to force the redirection of connections to other members
        - » ACR is utilized under the covers so the redirect is transparent to the application
  - Disable validation queries at the application server
    - ✓ Resources > JDBC > Data Sources > *data_source* > Connection Validation Properties (uncheck the box)
  - Consider using dynamic location aliases to subset the data sharing group for ‘unruly’ applications
    - These can be created, started and stopped dynamically to quiesce workload out of the group

Controlling connection usage ...

- **Recommendations ...**
  - Set up a wildcard ‘0.0.0.0’ profile with warning based on peak number of connections from a location

```
INSERT INTO SYSIBM.DSN_PROFILE_TABLE (PROFILEID,LOCATION) VALUES (1,‘0.0.0.0');

INSERT INTO SYSIBM.DSN_PROFILE_ATTRIBUTES (PROFILEID, KEYWORDS, ATTRIBUTE1, ATTRIBUTE2)
VALUES (1,'MONITOR CONNECTIONS',‘WARNING',100);
```

    - -DIS LOCATION DETAIL output can be put into a spreadsheet to manually track peaks
    - -DIS THREAD(*) TYPE(INACTIVE) output can then be used to search on the IP address from the location to determine the AUTHID or more detailed information to go after them

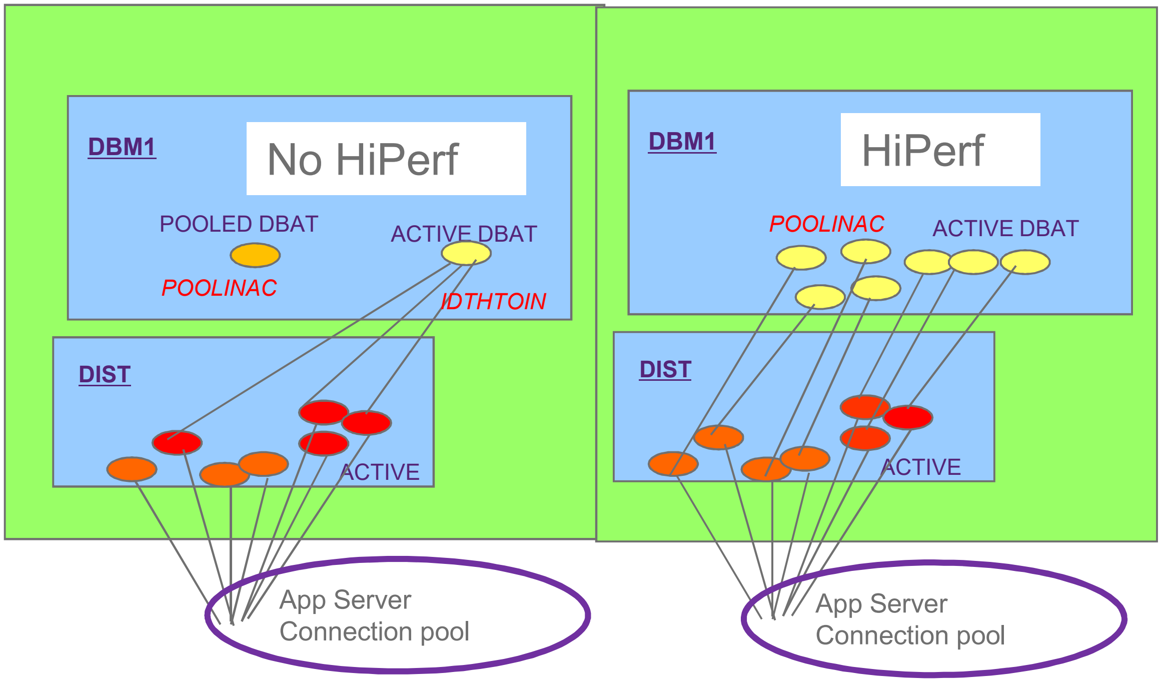
  - Once comfortable use ‘EXCEPTION_DIAGLEVEL3’ (PI31957) for more granular information
    - Warning: this will create 1 message per occurrence (several a second could be a nuisance)
    - Automatic client reroute will direct the blocked connections to another member
    - Can also tune down to DIAGLEVEL2 and monitor with IFCID 406 for details

Controlling thread usage

- **Common problem: Applications gobble threads**
  - Lack of a commit scope for transaction
    - Intentionally left open to avoid re-signon
    - Even with commit() KEEPDYNAMIC, DGTTs, and held cursors are problems
  - Rush of connection requests cause a slow down, CPU pegged, DBATs pile up
    - DBAT is required for authentication of inbound connection – NOT zIIP eligible
  - Overuse of RELEASE(DEALLOCATE) and -MODIFY DDF PKGREL(BNDOPT|BNDPOOL)
    - Rebinding NULLID collection with RELEASE(DEALLOCATE) so everyone becomes a High Performance DBAT
    - Allowing any application to take advantage of the DEALLOCATE SYSLN###, SYSLH###, SYSSTATS packages drives up the number of concurrent DBATs because they are not pooled as quickly for reuse by another requestor
      - ✓ If any part of the applications touches a DEALLOCATE package (even nested stored proc) it becomes a HPDBAT
  - Timeout values like idle thread timeout based on worst behaving application

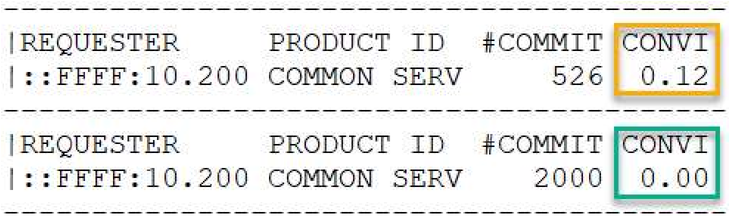
Remote connection monitoring for High Perf. DBATs

- **Using HiPerf DBATs means number of active DBATS in Db2 will now be a direct reflection of the number of max connections in the app server connection pools**
- **The number of threads seen increases as a function of the frequency with which each connection in the pool is used**
  - So 100 trans with CL2 Elapsed of 10ms could run each second on **1** thread with pooling, but those trans could take **100** threads with HiPerf DBATs
  - Remember you can disable HiPerf DBATs at any time with:
    - –MODIFY DDF PKGREL(COMMIT)
    - At COMMIT threads will revert to RELEASE(COMMIT)
- **TIMEOUTS – min/default of 120 seconds**
  - *IDTHTOIN* works on active threads that have not issued SQL in x seconds
  - *POOLINAC* works on inactive DBATs including HiPerf DBATs that have not issued SQL in x seconds



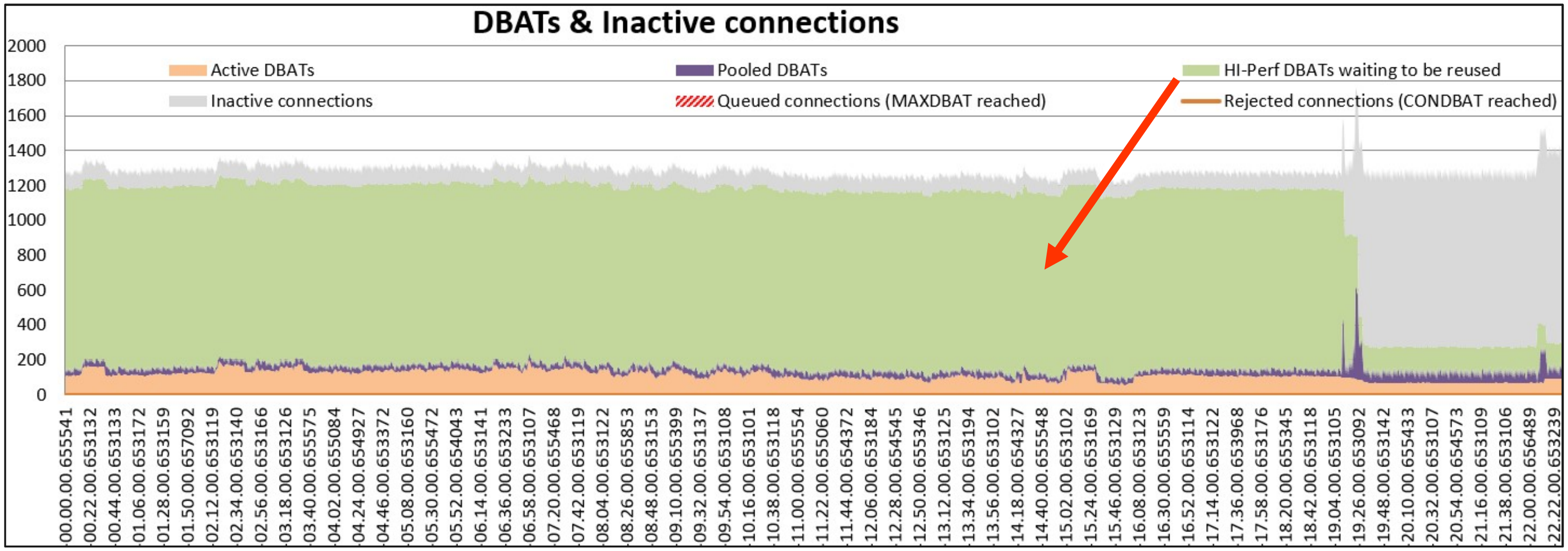
High Performance DBATs

- **Budget for HiPerf DBATs because DBAT goes away when the connection deallocates**
  - Ensure application #COMMITS * CONVI > 200 in accounting records, based on location to avoid wasted DBAT destruction and CPU overhead in DIST address space



```
---------------------------------------------------
|REQUESTER      PRODUCT ID   #COMMIT  CONVI
|::FFFF:10.200  COMMON SERV      526   0.12
---------------------------------------------------
|REQUESTER      PRODUCT ID   #COMMIT  CONVI
|::FFFF:10.200  COMMON SERV     2000   0.00
---------------------------------------------------
```

- **Statistics field for CUR ACTIVE DBATS-BND DEALLC indicates thread is only active because of bind option, not actually doing work (otherwise it would be active)**
  - When HiPerf DBATs turned off (-MODIFY DDF PKGREL) enormous drop in DBATS



Controlling thread usage …

- **Recommendations**
  - Set up a Profile
    - Monitor Threads with an initial setting based on HWM ACTIVE AND DISCON DBATS (QDSTHWAT)
      - ✓ Start with warning → automate –DIS commands to manually track offenders (monitor PI63531)
      - ✓ Base profile on AUTHID, client application name… or wildcard '*' (do not use wildcard in LOCATION)
    - Monitor Idle Threads with an initial setting of ½ IDTHTOIN
      - ✓ Include EXCEPTION at current IDTHTOIN threshold
      - ✓ Start with warning to understand what a realistic timeout is, and how frequently it is hit
  - Before utilizing high performance DBATs:
    - Segregate RELEASE(DEALLOCATE) packages to a separate collection, and use the currentPackageSet special register in application request or data source
      - ✓ **Monitor Profiles** can also be used to SET CURRENT PACKAGE PATH from the Db2 for z/OS side

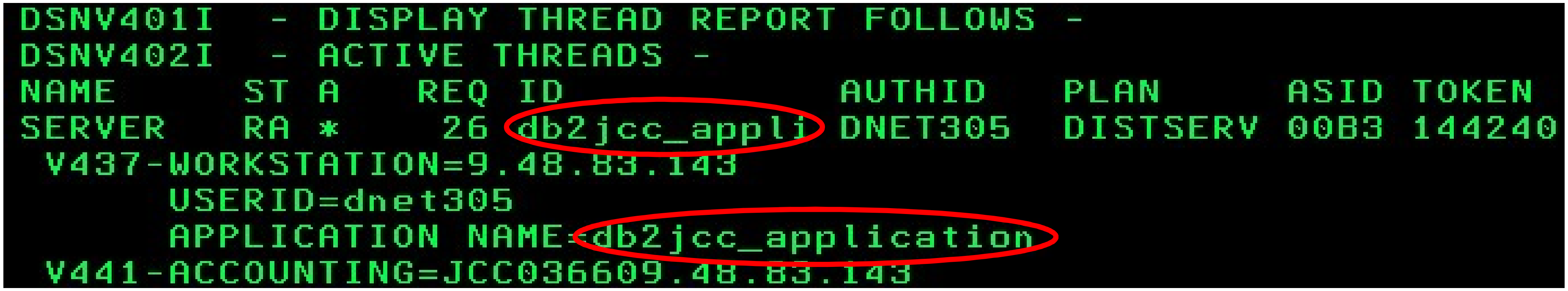
```
INSERT INTO SYSIBM.DSN_PROFILE_TABLE ( PROFILEID, COLLID, PROFILE_ENABLED)
   VALUES   (xxxx, 'NULLID', 'Y' ) ;
INSERT INTO SYSIBM.DSN_PROFILE_ATTRIBUTES (PROFILEID, KEYWORDS,  ATTRIBUTE1 )
  VALUES (xxxx, 'SPECIAL_REGISTER', 'SET CURRENT PACKAGE PATH = HIPERF');
```

Monitoring the ‘black box’

- **Common problem: How does one differentiate remote applications coming in?**
  - Historically locally attached applications were simpler to track based on the Plan name, correlation ID/AUTHID, and corresponding joblog/syslog entries of the local address space
  - The move to distributed applications means control is decentralized
    - Plan name DISTSERV
    - Generic AUTHIDs at the application server level
      - ✓ These IDs/privileges are inherited by the calling application
    - Larger middle-tier servers mean a wider variety of applications coming in
    - Generic HTTP/web servers spraying connections into the data sharing group
  - Connections and DBATs coming in are no longer differentiable
    - No specific WLM service classes
    - No internal chargeback methodology
    - Dynamic statement cache can only be used to tune RUNSTATs not SQL
    - Problem resolutions cannot be applied effectively
    - Standards for the driver and application coding practices cannot be upheld

Monitoring the ‘black box’ …

- **Common problem: How does one differentiate remote applications coming in? …**
  - Example of a –DISPLAY thread command
    - i.e. Correlation ID = Application Name = Process Name = ‘db2jcc_application’
    - The ACCOUNTING information contains the driver level and the originating IP Address of the requestor, but there could be many applications using that driver
      - ✓ By default Db2 z/OS populates this information on behalf of the application
    - This information is propagated to the –DISPLAY commands, accounting SMF 101 records, as well as the WLM SMF 72 records which describe the Enclave



```
DSNV401I  - DISPLAY THREAD REPORT FOLLOWS -
DSNV402I  - ACTIVE THREADS -
NAME     ST A   REQ ID           AUTHID   PLAN     ASID TOKEN
SERVER   RA *    26 db2jcc_appli DNET305  DISTSERV 00B3 144240
 V437-WORKSTATION=9.48.83.143
      USERID=dnet305
      APPLICATION NAME=db2jcc_application
 V441-ACCOUNTING=JCC036609.48.83.143
```

Monitoring the ‘black box’ …

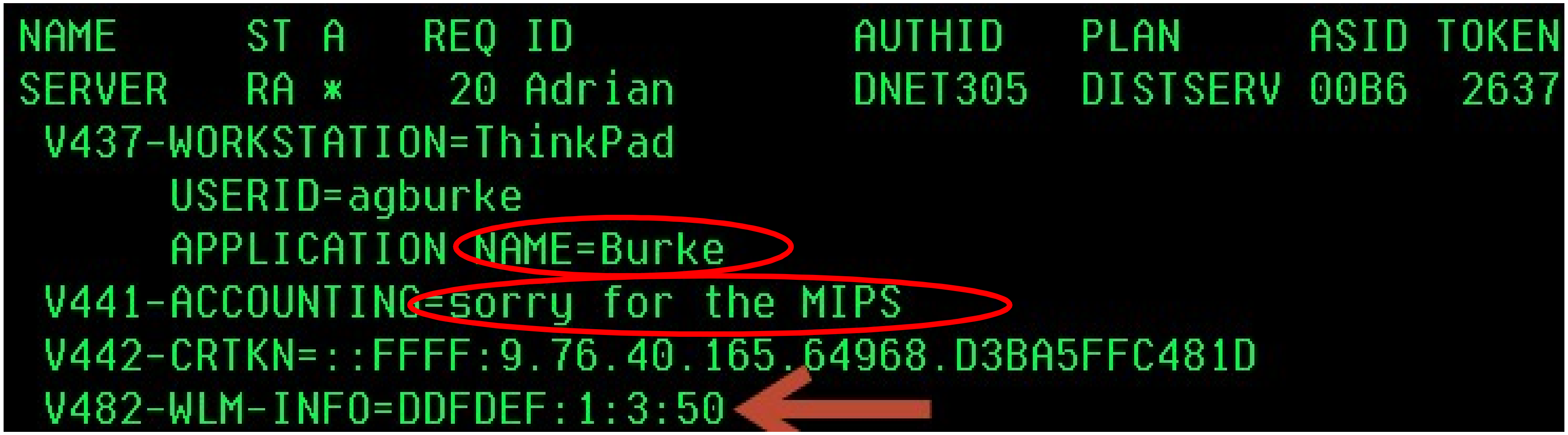
- **Common problem: How does one differentiate remote applications coming in? …**
  - The application can set these client information fields using JAVA APIs, set methods for .NET and CLI, data source properties, driver connection properties, and Db2 special registers
  - More granular workload management classifications for relative importance levels as well as software pricing discounts (i.e. Mobile workload pricing) as well as internal chargeback
    - Simpler problem determination and exposure of tuning possibilities
  - Here is a break-down of what the most common fields are and where their values are found

JAVA	.NET or .CFG keyword	English	WLM Classification	Monitor Profile	Special Register	Size-bytes	Macro
ClientAccountingInformation	ClientAccountingString	Client Accounting Information	CAI		CLIENT_ACCTNG	512	QWDASUFX
ClientProgramName		Correlation ID	CI			8	QWHCCV
ClientCorrelationToken	ClientCorrelationToken	Client CorrelationToken			CLIENT_CORR_TOKEN	255	
ClientUser	ClientUserID	Client User ID (END_USER)	CUI	Y	CLIENT_USERID	128	QWHCEUID_Var
ClientHostName	ClientWorkStationName	Client Workstation Name	CWN	Y	CLIENT_WRKSTNNAME	255	QWHCEUWN_var
ApplicationName	**ClientApplicationName**	**Client Transaction or application name**	**CTN**	**Y**	**CLIENT_APPLNAME**	**255**	**QWHCEUTX_Var**
ApplicationInformation	ClientApplicationName	Process Name (client tran or appl name)	PC	Y	CLIENT_APPLNAME	39	QWHCEUTX

Monitoring the 'black box' …

- **Recommendations**
  - Set enableDefaultClientInfo = *true* property for .NET/CLI applications in the .cfg file so the applications can retrieve the values that Db2 z/OS sets for the following parameters

1. Having the application set the client information fields in their code is the most granular
   - Call the java.sql.Connection.setClientInfo method
     - ✓ *conn.setClientInfo("ApplicationName", string);*
   - Use the .NET get or set methods from Db2Connection Class
     - ✓ *public function set ClientApplicationInformation(String);*
   - CLI applications use *SQLSetConnectAttr()* function to set SQL_ATTR_INFO_APPLNAME



```
NAME     ST A   REQ ID           AUTHID   PLAN     ASID TOKEN
SERVER   RA *    20 Adrian       DNET305  DISTSERV 00B6  2637
 V437-WORKSTATION=ThinkPad
      USERID=agburke
      APPLICATION NAME=Burke
 V441-ACCOUNTING=sorry for the MIPS
 V442-CRTKN=::FFFF:9.76.40.165.64968.D3BA5FFC481D
 V482-WLM-INFO=DDFDEF:1:3:50
```

Monitoring the ‘black box’ …

- **Recommendations …**

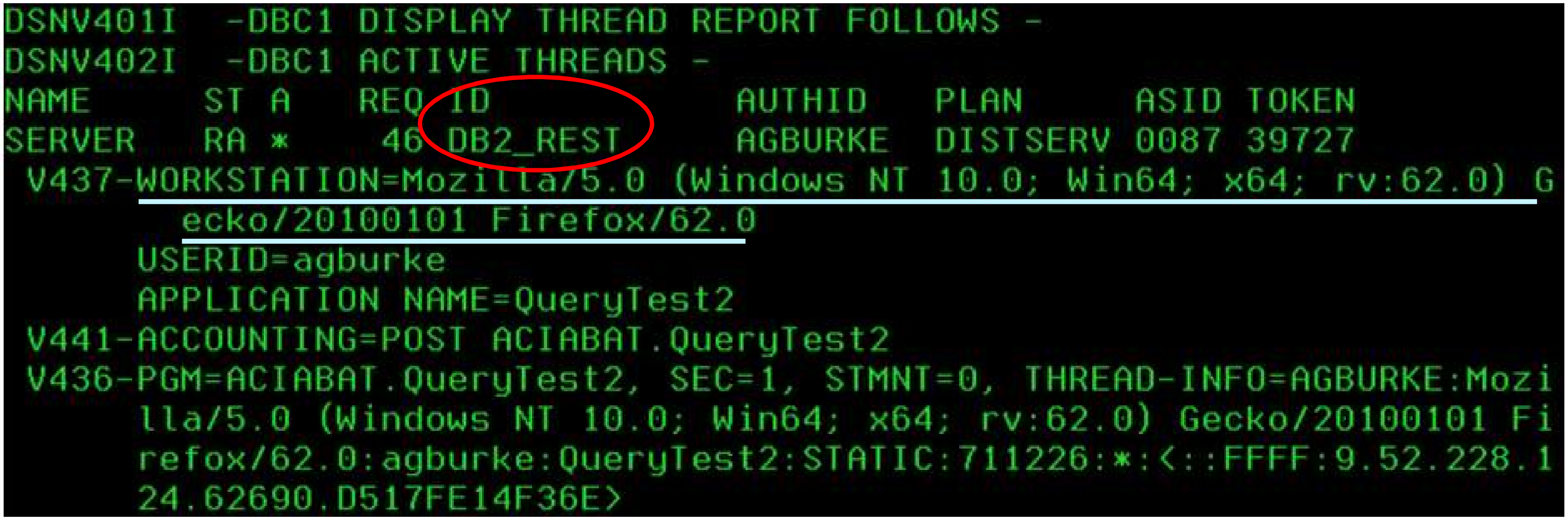
  2. If the applications cannot be modified, the next granular level is in the Data Source
     - In WebSphere Resource Reference if multiple applications share the same data source
       - ✓ In the Admin console navigate to Applications >*application name* > Resource reference >Extended properties and create *clientApplicationInformation* property and give it a value
     - In the Data Source extended property if each application has its own unique data source
       - ✓ In the Admin console navigate to Data Sources > *data souce* > Connection pools > Connection pool custom properties and create *clientApplicationInformation* property and give it a value
  3. If the applications are homogeneous on an application server, the properties can be set in the driver and persist across connection as long as the connection exists
     - DB2JccConfiguration.properties file – JDBC
       - ✓ ClientApplicationName
     - For Call level interface use db2cli.ini (text) or for .NET use db2dsdriver.cfg (XML file)
       - ✓ ClientApplName

  – In WLM, this would populate the WLM fields of CTN (client transaction name) and PC (process name) and the Db2 special register CLIENT_APPLNAME
     - Allow granular classification of work

What about Db2 Native REST services?

- **–DISPLAY THREAD(*) command shows pre-populated client accounting fields**
  - Already better than dynamic JCC calls… you are welcome..
  - – Correlation ID (CI) for a thread that processes a service is set to "DB2_REST"
  - – Workstation (CWN) is set to REST client or browser you are using
  - – Application name (CTN) is the name of the REST service "QueryTest2"

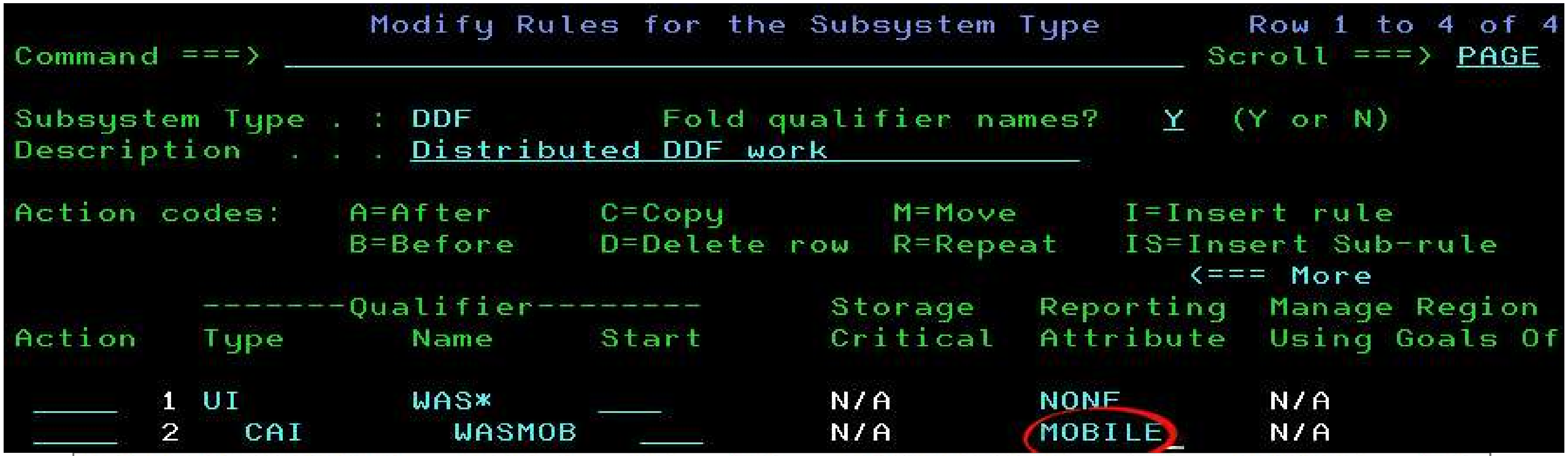
- **What can the caller/invoke control?**
  - – The "User-Agent" field in the REST header can be customized, here it is WORKSTATION='agb'



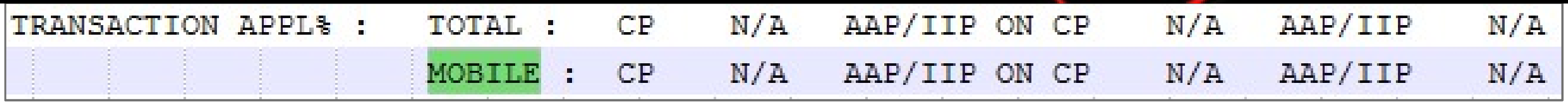
```
DSNV401I  -DBC1 DISPLAY THREAD REPORT FOLLOWS -
DSNV402I  -DBC1 ACTIVE THREADS -
NAME      ST A   REQ ID            AUTHID   PLAN     ASID TOKEN
SERVER    RA *    46 DB2_REST      AGBURKE  DISTSERV 0087 39727
 V437-WORKSTATION=Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:62.0) G
        ecko/20100101 Firefox/62.0
      USERID=agburke
      APPLICATION NAME=QueryTest2
 V441-ACCOUNTING=POST ACIABAT.QueryTest2
 V436-PGM=ACIABAT.QueryTest2, SEC=1, STMNT=0, THREAD-INFO=AGBURKE:Mozi
      lla/5.0 (Windows NT 10.0; Win64; x64; rv:62.0) Gecko/20100101 Fi
      refox/62.0:agburke:QueryTest2:STATIC:711226:*:<::FFFF:9.52.228.1
      24.62690.D517FE14F36E>
```

WLM and MOBILE Pricing

- **Define a WLM service classification rule to subset transactions with client accounting information out from those using a specific Authorization ID on the server**
  - You could also use client IP address to distinguish them; this IP address would be the same one you would see in the accounting string if you did a –DIS THREAD
- **Using a sub-rule for the classification where the accounting string is ‘WASMOB’ I can push it down into another service class if needed or just report on it**
- **The CPU consumption will then be reported in the WLM activity report as well**



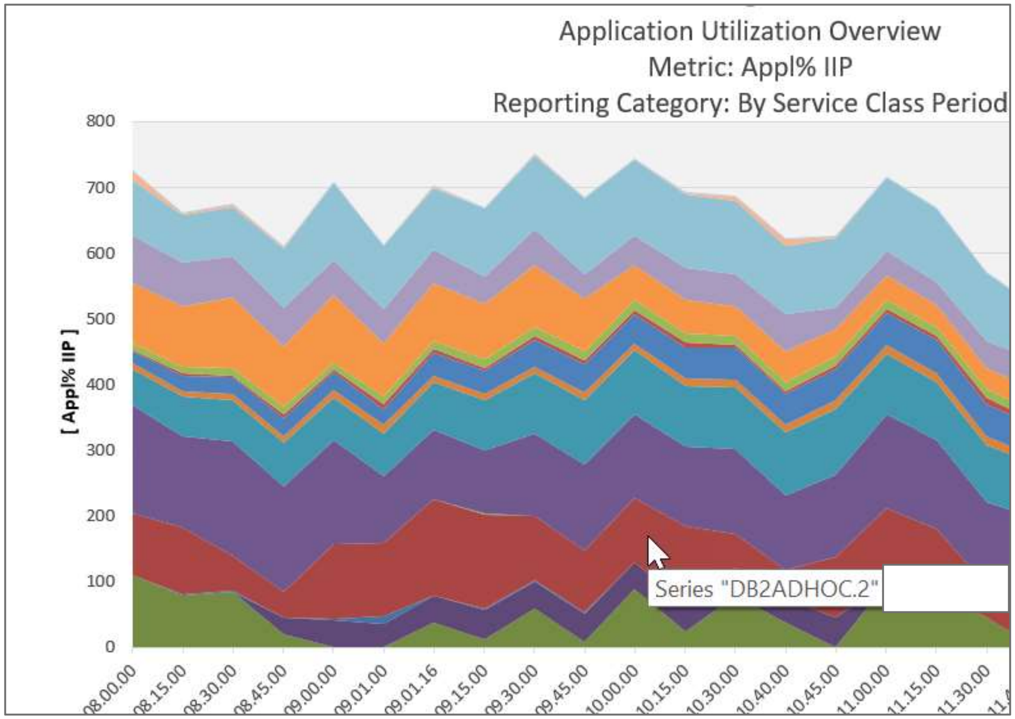
```
                     Modify Rules for the Subsystem Type          Row 1 to 4 of 4
Command ===> ____________________________________________________ Scroll ===> PAGE

Subsystem Type . : DDF           Fold qualifier names?   Y  (Y or N)
Description  . . . Distributed DDF work________________

Action codes:    A=After      C=Copy          M=Move       I=Insert rule
                 B=Before     D=Delete row    R=Repeat     IS=Insert Sub-rule
                                                                  <=== More
          --------Qualifier--------            Storage    Reporting   Manage Region
Action    Type        Name        Start        Critical   Attribute   Using Goals Of

 ____   1 UI          WAS*        ___          N/A        NONE        N/A
 ____   2  CAI          WASMOB      ___        N/A        MOBILE_     N/A
```



```
TRANSACTION APPL% :   TOTAL :   CP    N/A   AAP/IIP ON CP   N/A   AAP/IIP   N/A
                      MOBILE :  CP    N/A   AAP/IIP ON CP   N/A   AAP/IIP   N/A
```

zIIP work and WLM

- **Any workload which lands in a Discretionary service class *(i.e. SYSOTHER)* AND is zIIP eligible will not be redirected to a GCP**
  - Even with IIPHONORPRIORITY=YES Discretionary DDF work will queue for a zIIP and not fall back to a GCP
  - Blocked Workload Support is also not available for zIIP eligible work
    - Prefetch, deferred write, GBP castout engine in data sharing, log write and prefetch, parallel index I/O
- **WLM now allows HONOR PRIORITY at service class level, do not say 'NO' for work that changes Db2 data**
- **Normally we look for IIPCP CPU zIIP redirect to general purpose engine to determine zIIP shortages BUT will not show up in this case, must monitor WLM delays for zIIP**
- **Recommend: avoid any Db2 dependent workload, especially those that are zIIP eligible from being classified as discretionary (monitor in RMF)**



References

- **Db2 for z/OS News from the Lab**
  - Monitoring and controlling resources for distributed workloads with profile tables
    - https://www.ibm.com/developerworks/community/blogs/897a7c98-57af-4523-9cfa-07ebc3f996b4?lang=en
- **Clientinfo setter methods for DB2Connection class – enableDefaultClientInfo property allows app to retrieve the default values**
  - http://www.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.swg.im.dbclient.adonet.ref.doc/doc/DB2ConnectionClassMembers.html
- **Registry values set by JCC driver with V11**
  - http://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.swg.im.dbclient.config.doc/doc/r0061010.html
- **Special register client_applename in Db2 z**
  - https://www.ibm.com/support/knowledgecenter/SSEPEK_11.0.0/sqlref/src/tpc/db2z_currentclientapplname.html
- **Sample driver properties file**
  - https://www.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.swg.im.dbclient.config.doc/doc/r0061060.html
- **Export current .NET parameters**
  - https://www.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.swg.im.dbclient.config.doc/doc/c0054555.html

Db2 SWAT team engagements

- **Db2 Master Class– held twice a year, one in the US and one in the UK**
  - https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/Wc05a3bbc003d_44bf_8673_d5dd7683d239/page/Db2%20for%20zOS%20Master%20Class%202019%20-%20Workshop%20Announcement
  - Hursley Lab – usually a week in June
  - Silicon Valley Lab – San Jose usually a week in September
    - Spend a week with John Campbell and the Db2 SWAT team covering performance and availability topics including how to analyze statistics and accounting data

It's all about robustness.

- **Db2 360 Degree Continuous Availability Assessment Study**
  - Comprehensive study performed by the Db2 SWAT team aimed at discovering exposures in continuous availability, performance, and speed of recovery
- ***Please contact me or Chunyang Xia (cxia@us.ibm.com)***

References

- **PI95306 - with Product ID wildcarding in a monitor profile, Db2 does not correctly attribute that connection to a profile**
- **PI94667 - remove implicit queuing for DDF threads when System Monitor Profile is operative**
- **APARs needed to eliminate BSDS requirement for SECPORT definition, so they can be defined in the TCPIP.PROFILE dataset**
  - PI85058 - Db2 DDF ALIAS will not automatically started after Db2 or DDF recycle if the ALIAS is defined with a SECPORT only
  - PI92982 - Db2 DDF STATIC or DYNAMIC ALIAS (no secure port) GROUP IPADDRESSIS not added to the listener