#SHAREorg

SHARE
Technology · Connections · Results

DB2 for z/OS Distributed Access – Best Practices and Updates

Adrian Burke
IBM

March 13, 2012
Session 10996

Important Disclaimer

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 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 PRESENTATION OR ANY OTHER DOCUMENTATION.

NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, OR SHALL HAVE THE EFFECT OF:

- CREATING ANY WARRANTY OR REPRESENTATION FROM IBM (OR ITS AFFILIATES OR ITS OR THEIR SUPPLIERS AND/OR LICENSORS); OR
- ALTERING THE TERMS AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT GOVERNING THE USE OF IBM SOFTWARE.

Agenda

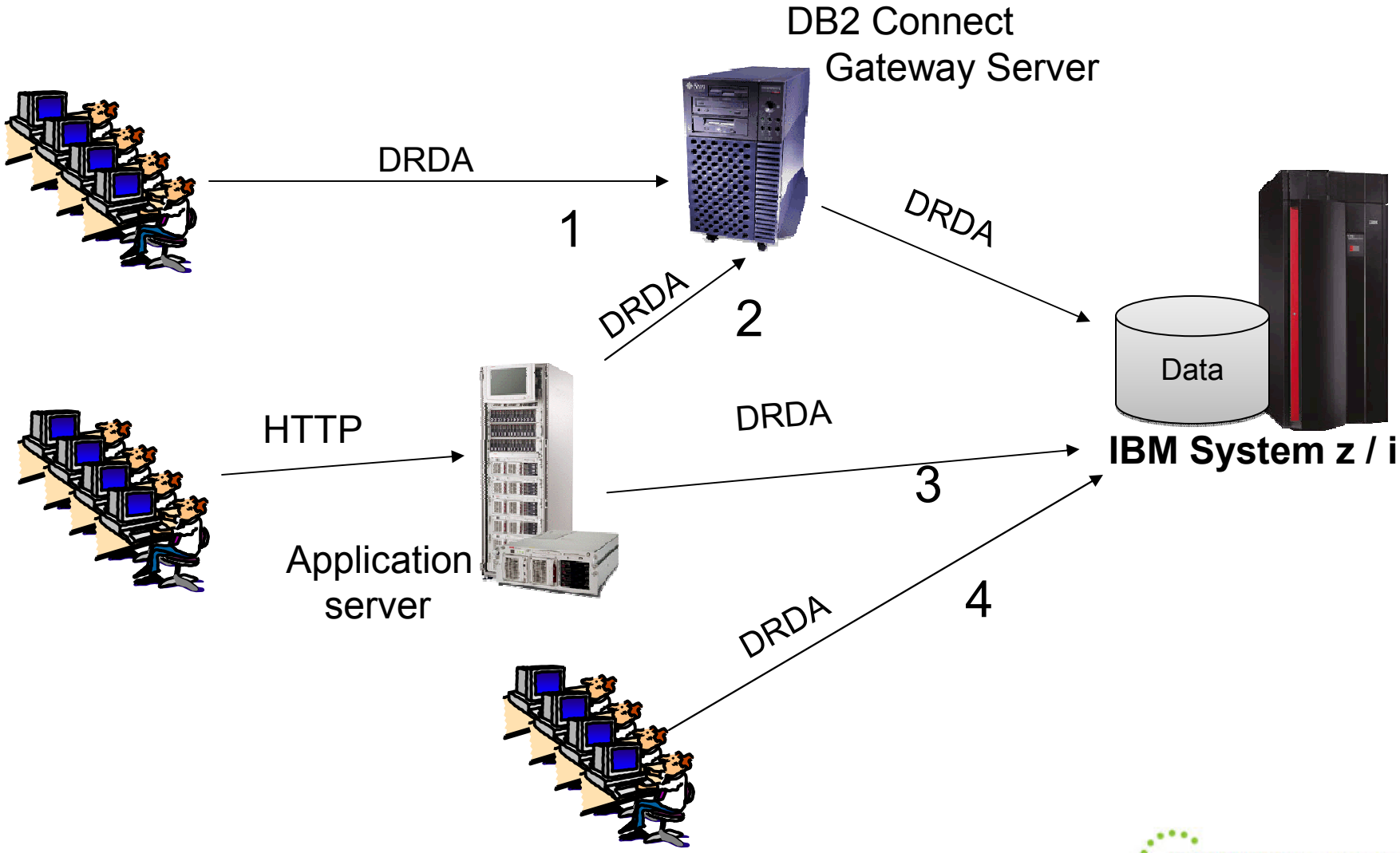
- Sysplex Workload Balancing
- Parameters
- Aliases
- Cancel Thread
- Profiling
- WLM

Adrian Burke
DB2 SWAT Team
agburke@us.ibm.com

Certified for IBM Information Management software

SHARE in Atlanta 2012

DB2 Connect Configurations



Why Would I use DB2 Connect Server

- Two Phase Commit
  - Many transaction managers still require a DB2 Connect Server for two phase commit if they use a dual transport model
    - For example, Tuxedo, Encina
  - DB2 and MS DTC are two transaction managers that do not require a DB2 Connect Server
- Federation
  - Homogeneous Federation is possible with DB2 Connect Server
    - Use of nicknames to other DB2 and Informix data servers
- Licensing
  - DB2 Connect Enterprise Edition with concurrent user licensing requires DB2 Connect Server configurations only
  - Ensure you have license for MSUs and Host (based on # of subsystems or D.S. group

Sysplex Workload Balancing

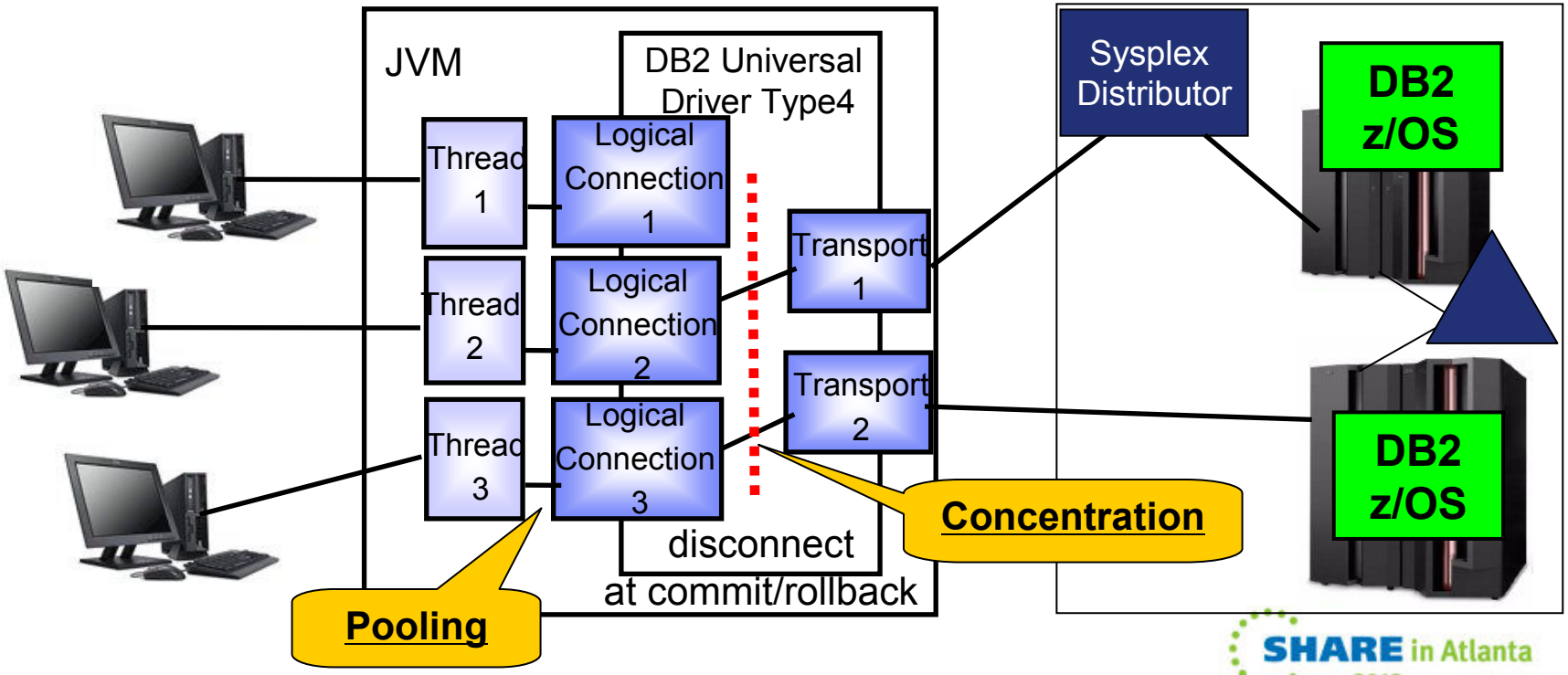
- The Sysplex Distributor
  - Allows you to transparently establish an initial connection to an available member of the data sharing group
  - However, the workload balancing is network based so should just be for establishing the initial connection to the group
- The driver/connect server uses the WLM-provided information, together with DB2 Connect's sysplex support, to do the actual workload balancing of the different connections

So what are the key points to Pooling and Concentrator

- Both can support Data Sharing
- Both can support VIPA and **Sysplex Distributor**
- Connection Pooling reuses threads at connection time
  - Have to drop the connection to be routed to another DB2
- Connection Concentrator reuses threads at commit time
  - Restrictions:
    - Only supports SSL for outbound connections
    - If you declare global temporary tables, they must be closed explicitly at transaction or branch boundary
    - Dynamic prepare requests from embedded dynamic SQL applications will be rejected.
    - See DB2 Connect user guide for details on other restrictions
    - SAP does not support Connection Concentrator
  - Observations:
    - Most customers use Connection Pooling
    - Thoroughly test usage of Connection Concentrator – often times applications are using some of the above restrictions which do not show up until in production

Sysplex Workload Balancing

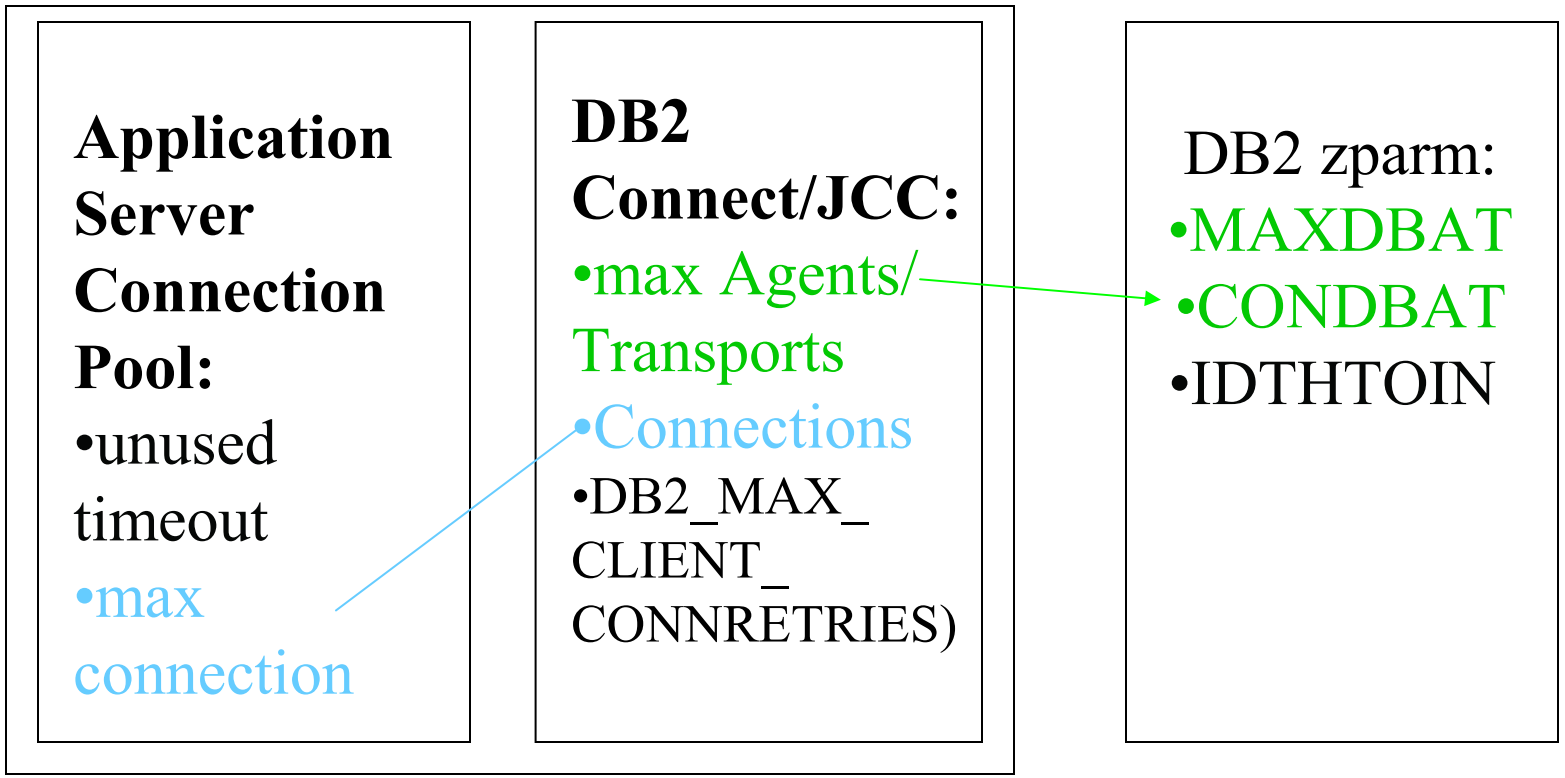
- **Sysplex Workload Balancing was supported by the JCC type 4 since V8 and by non-Java since 9.5 FP3**
  - What level driver came thru the maintenance stream…
    - https://www-304.ibm.com/support/docview.wss?q1=pk77599&rs=0&uid=swg21428742&cs=utf-8&lang=en&loc=en_US&cc=us



SysplexWLB and Failover

- Java and non-Java IBM Data Server **drivers** can exploit the sysplex as follows:
  - They can connect applications to a DB2 data sharing group as though it were a single database server, and spread the workload among the different members, based on server lists dynamically provided by WLM (**connection pooling**)
  - They can recognize when a member of a DB2 data sharing group fails and can automatically route new connections to other members. This is done after commit scope of application, otherwise connection would simply hang. (**connection concentration**)

DB2 Inactive Thread - Tuning Considerations



Thread Monitoring:

-**DIS DDF DETAIL** real time information on DBAT and CONDBAT metrics

---

```
DSNL080I  +APDB DSNLTDDF DISPLAY DDF REPORT FOLLOWS:
DSNL081I STATUS=STARTD
DSNL082I LOCATION        LUNAME          GENERICLU
DSNL083I APDB                            -NONE
DSNL084I IPADDR                          RT
DSNL085I 10.116.2.5      3231    3232
DSNL086I SQL    DOMAIN=share.org
DSNL086I RESYNC DOMAIN=share.org
DSNL090I DT=I  CONDBAT=  10000 MDBAT=  200
DSNL092I ADBAT=    0 QUEDBAT=     0 INADBAT=      0 CONQUED=      0
DSNL093I DSCDBAT=      0 INACONN=      5
DSNL099I DSNLTDDF DISPLAY DDF REPORT COMPLETE
***
```

How many threads are currently doing work

What you think you can handle.

How many threads are lounging in the pool

You hit max DBAT and some are waiting

Type 2 inactive thread/ inactive connection

Configuration Best Practices (non-JAVA)

- enableWLB – True
  - Enables WLB and seamless failover.
- Transports allocated based on app connection
  - Default max of 1000 transports per member
- ConnectionTimeout – **Network. Based on Application QoS**
  - Connect will fail when expired
- CommandTimeout - **Network (.NET 30 seconds). Based on Application QoS.**
  - Triggers interrupt when expired.
- maxAcrRetries – **1 time thru server list**
  - Number of times to retry the connection to the alternate server.
- AcrRetryInterval – **1 second**
  - Number of seconds to wait between retries.

Configuration Best Practices (non-JAVA)

- maxTransportIdleTime - 60 seconds (< than IDTHTOIN)
  - Time an unused transport hangs around before being closed.
- maxRefreshInterval – 10 seconds
  - Time between updates of WLM weights with server.
- maxTransportWaitTime – 1 second
  - Max time an app will wait for a transport to become available.
- memberConnectTimeout - 1 second
  - Timeout to be used when communicating with a member in the server list.
- tcpipKeepAlive – 15 seconds (< than TCPKPLV)
  - Timeout for all other communication requests
- interruptprocessingmode – 2
  - Controls interrupt behavior. Value=2 drops the connection when interrupt is triggered.

Configuration Best Practices…

- Cursor Hold – we do not want result sets persist across commits so…
  - JAVA – setResultHoldability =2
  - Non-JAVA – CursorHold=0
- Implicit Close – we want cursors to be automatically closed after rows are fetched so…
  - JAVA – queryCloseImplicit
  - Non-JAVA – CursorTypes
- Auto Commit – we would like the application to *intelligently* moderate the commit scope through the persistence layer or manually within the application
  - JAVA – setAutoCommit(false)
  - Non-JAVA – Autocommit configuration keyword

Key DB2 DDF Parameters

Parameter	Possible values	Default – as of V8*	Description
DDF	NO / AUTO / COMMAND	No	DDF Startup
CMTSTAT	ACTIVE / INACTIVE	INACTIVE*	Thread Pooling - Pool inactive threads
CTHREAD	1-2,000	200*	Max users - allied (local) threads RRSAF or CAF (CICS, IMS, TSO, Batch attach, SPUFI, Classic QMF, etc)
MAXDBAT	0-1,999	200*	Max remote active DDF Threads - DBM1 Address Space
CONDBAT	0-15,000	10,000*	Max remote connections - DDF Address Space
MAXTYPE1		0	Max inactive DBATs, these are used for private protocol. DRDA uses inactive connections.
POOLINAC	0-9,999	120	Approximate time, in seconds that an inactive/unused DBAT can remain idle in the pool before it is terminated. DBAT deleted after being used 200 times also.
IDTHTOIN	0-9,999	120*	The IDLE THREAD TIMEOUT (IDTHTOIN) parameter specifies the time (in seconds) that an active DBAT can remain idle before it is canceled. – Should set a "little" higher than TCPKPALV
TCPKPALV	ENABLE / DISABLE / 1-65534	120*	TCP/IP keep alive (Goes hand-in-hand with IDTHTOIN)
CONTSTOR	YES / NO	No	Periodically " contract" each thread's working storage area.
MINSTOR	YES / NO	No	Use storage management algorithms that minimize the amount of working storage consumed by individual threads.

These were mentioned on previous slides

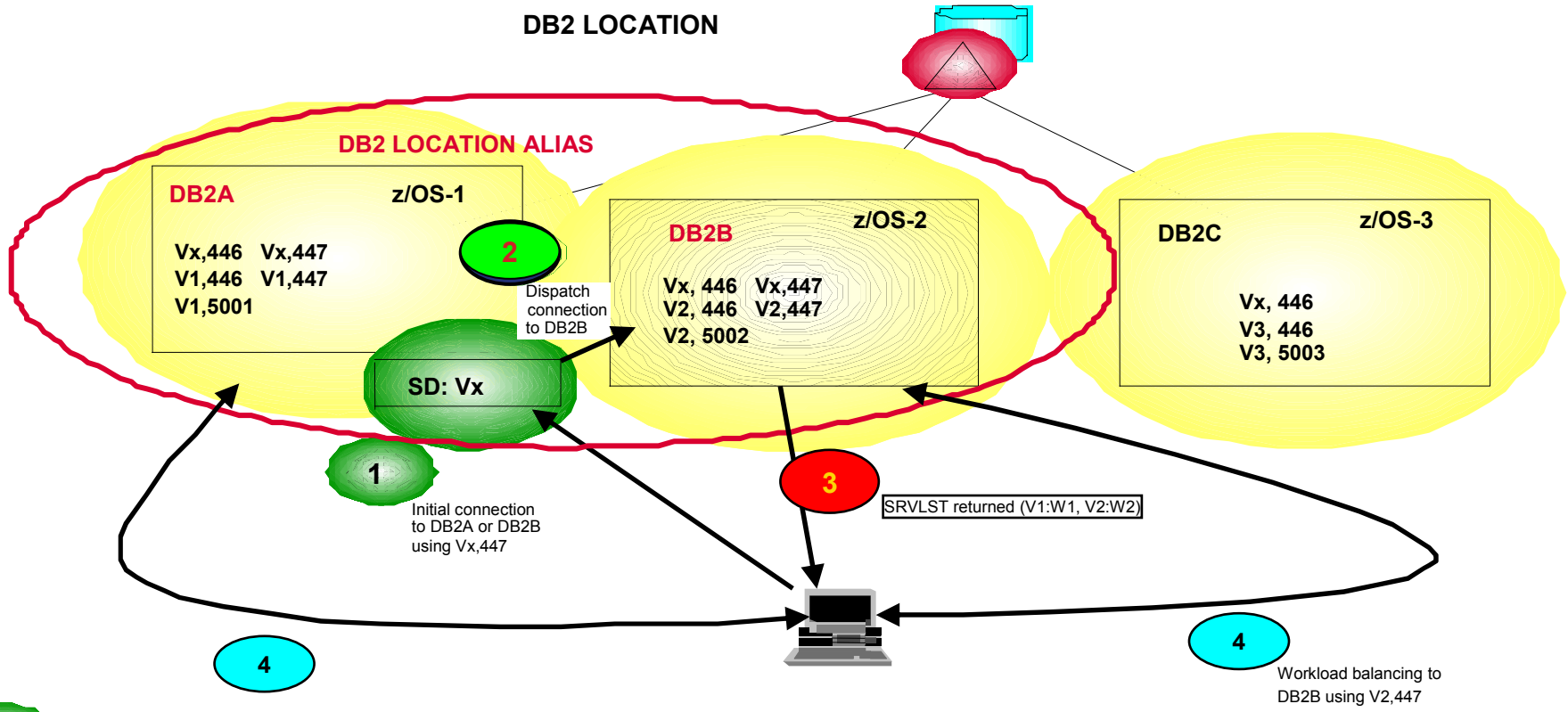
WLM Weight calculation.

There are no configuration parameters related to enabling sysplex WLB on DB2 for z/OS server. When DDF is started, it registers itself to WLM, unless MAXDBAT is set to 0 or DDF is stopped. In either of these exception cases, that member's DDF does not appear in the server list. The drivers use the updated server list on each transaction to distribute workload to DB2 data sharing members.

There are several factors WLM will take into account when creating and updating weights.

1. Displaceable capacity of systems (CPU).
2. Enclave service class achievement (Performance Index, or PI)
   - WLM goal should be attainable when system is not under stress, which would result in a PI < 1.
3. Enclave service class queuing.
4. DB2 for z/OS Health
   - DB2 will report its health factor of 0 to 100 to WLM based on the current storage consumption within the DBM1 and DIST address spaces.

Subsetting configuration



1 First connection goes through Sysplex Distributor (DVIPA)

2 First based on IEAFSSNxx order in SYS1.PARMLIB

3 A list of available DB2 subsystems and their respective weightings are returned

4 Based on weighting following connections and threads are routed

Types of location aliases

- ***Static location aliases***

  You can use DSNJU003 (change log inventory) utility to define and modify as many as 8 static location aliases. Changes to these aliases require you to stop both DDF and DB2, thus requiring an outage.

- ***Dynamic location aliases (New in v10)***

  You can use the MODIFY DDF command with the ALIAS option to define and manage as many as 40 location aliases dynamically. You can start, stop, cancel, modify, and delete dynamic location aliases without stopping either DDF or DB2. These aliases **cannot** be defined or managed by the DSNJU003 utility, and the DSNJU004 utility does not print any information about them. You have to use the DISPLAY DDF command to find information about these aliases. Before you can define dynamic location aliases, DB2 must be started, but DDF may or may not be started.

Dynamic Location Aliases

Business Challenge : *Need the ability to manage poorly performing app servers without taking a DB2 server outage.*

Solution: You can associate a location alias with an app server and use the –MODIFY DDF command to dynamically exclude poorly behaved applications from sysplexWLB (change PORT)

Business Challenge : *Stopping DDF causes all the app servers connected to a member to **failover**. DBAs are looking for a more granular failover where only certain app servers fail over while others don't.*

Solution: You can associate a location alias with an app server and use the –MODIFY DDF command to dynamically stop the alias which only causes the app servers connected to that alias to failover, without affecting others.

Dynamic Location Aliases

Business Challenge *: Block incoming traffic from certain app servers without disrupting* ***outbound*** *traffic. Setting MAXDBAT=0 blocks all inbound traffic.*

Solution: You can associate a location alias with an app server and use the –MODIFY DDF command to dynamically stop the alias on a member which blocks incoming traffic from only those app servers that are connected to that alias without affecting other inbound or any outbound traffic.

Business Challenge *: The default IP addresses provided to the client drivers may not always be usable to establish connections for sysplexWLB or failover. So, DBAs want to control the IP addresses provided to the client drivers.*

Solution : You can use the –MODIFY DDF command with the ALIAS keyword to dynamically specify the IP address that will be returned in the server list when client connects to the specified alias.

Dynamic Location Aliases - Continued

Examples :

-MODIFY DDF ALIAS(alias1) ADD

Alias1 is created and is stopped by default.

-MODIFY DDF ALIAS(alias1) PORT(9000)

Alias1 is associated with port 9000.

-MODIFY DDF ALIAS(alias1) IPv4(2.2.2.2)

Alias1 is associated with IP address 2.2.2.2

-MODIFY DDF ALIAS(alias1) START

DDF will accept requests for alias1 on port 9000.
When a client connects to alias1, IP address 2.2.2.2 is returned in the server list.

-MODIFY DDF ALIAS(alias1) STOP

Alias1 is stopped and will not accept new requests.
Existing requests will be allowed to complete.

Displaying location alias-specific information

-DISPLAY DDF ALIAS(*alias1*) DETAIL

```
DSNL080I @ DSNLTDDF DISPLAY DDF (alias1) REPORT FOLLOWS:
DSNL087I ALIAS   PORT SECPORT STATUS
DSNL088I ALIAS1 9000   5005        STARTD
DSNL089I MEMBER IPADDR=::2.2.2.2
DSNL089I MEMBER IPADDR=2002:91E:610::1
DSNL096I ADBAT= 100 CONQUED= 1000 TCONS= 1000
DSNL100I LOCATION SERVER LIST:
DSNL101I WT IPADDR   IPADDR
DSNL102I 32 ::2.2.2.2    2002:91E:610::1
DSNL102I 32 ::1.2.3.4
DSNL099I DSNLTDDF DISPLAY DDF REPORT COMPLETE
```

ADBAT *:* The number of active data base access threads (DBATs) that are currently processing requests on behalf of the specified alias.

CONQUED : The number of connection requests that are currently queued and waiting to be serviced on behalf of the specified alias.

TCONS : The total number of remote connections that are currently associated with the alias.

Online CDB (Communications DataBase)

Business Need : *TCP/IP connections established after modifying the CDB should automatically use the updated values without recycling DDF.*

Solution : When a CDB table used for TCP/IP access is updated, DDF is notified and it dynamically retrieves the updated values to establish new connections. You can use the –DISPLAY LOCATION command to tell which connections are using the updated CDB values and other connection attributes like AES, SSL, IPSec, trusted context, XA or sysplexWLB.

Cancel Thread improvements

Business Challenge : *Lack of sufficient cancel detection points in DB2 server is causing certain queries to run indefinitely and consume resources, even after client disconnects. DBAs want to be able to cancel a long running query and not be charged for wasted CPU costs that will never be materialized to the applications.*

Solution : More cancel detection points have been added in DB2, mainly in areas like sort and work file processing to catch more runaway threads.

Cancel Thread improvements - Continued

Business Challenge : *DBAs want to be able to cancel requests suspended in the network effectively without issuing external commands to drop the connection. Users may not know it is suspended in the network and end up recycling DB2 to free the held resources.*

Solution : DB2 will now programmatically drop the connection using TCP/IP network manager calls that allows DB2 to get control and cancel the thread. APAR number is PM54383.

System Profiling

Business Challenge : *With direct connections, DBAs need greater flexibility and control to manage (punish) resources for specific clients according to their needs.*

Solution : You can use DB2 profile tables to monitor connections for specific clients. You can specify thresholds and actions to be taken when threshold is exceeded. You can set filters to identify client connections to be monitored.

System Profiling - Continued

Filters that you can set.

- Client IP address/domain name
- Client product id
- Client application name
- Client userid
- Client workstation name
- Role name, authid
- Collection id/package name
- Location/location alias name

System Profiling - Continued

What can be monitored ?

1. MONITOR CONNECTIONS
   - Monitor the number of concurrent inbound connections allowed.
   - Is subject to the filtering on IP address or Domain name only.
2. MONITOR THREAD
   - Monitor the number of DBATs that are allowed to be concurrently active.
   - Is supported for all filtering combinations.
3. MONITOR IDLE THREADS
   - Monitor the amount of time, in second, that an active server thread is to be allowed to remain idle.
   - Is supported for all filtering combinations.

- **You can use history tables to know what profiles and monitoring functions are in effect.**
- **When a profile warning or exception occurs, IFCID 402 (stat class 4) record is written at an interval set in the STATIME zPARM.**

Profile enhancements

- Catalog table holds profiles
  - Can limit:
    - **Warning only produces messages**
    - **Exception will take action**
- Warning or exception
  - DIAG level 1 – 1 message if any threshold reached
  - DIAG level 2 – 1 message for each unique threshold surpassed
  - Exception for active thread means 10 can be active, 10 can be in queue, but next one in gets canceled
- DSNT771I/DSNT772I console message, reason code 00E3050x alerts you of warning or exception

PROFILEID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2	ATTRIBUTE3	ATTRIBUTETIMESTAMP	REMARKS
1	MONITOR THREADS	EXCEPTION_DIAG-LEVEL2	10		2008-12-19...	
2	MONITOR CONNECTIONS	WARNING	50		2008-12-17...	
3	MONITOR IDLE THREADS	EXCEPTION_DIAG-LEVEL1	300		2009-01-25...	

Protecting DB2

Business Challenge : *DBAs want to control the connection queue depth and the wait time for a queued connection to get a DBAT, exceeding which connection should be terminated.*

Solution : Two new zPARMs MAXCONQN (depth) and MAXCONQW (wait time) are supported. -DISPLAY DDF DETAIL will show the current configured values and connections closed because limit exceeded. This info is also recorded in stats and accounting. APAR number is PM43293 (V9 & V10).

Protecting DB2 - continued

Business Challenge : *WLM weights are not updated until PI index interval of 10 seconds which causes clients to use a stale server list to route requests. The requests may thus land on members that have a connection count approaching the CONDBAT limit.*

Solution : DB2 will lower its health when connection count starts approaching the CONDBAT limit. Since the health is factored in the weight calculation, clients will be discouraged to route here. Every 1 minute, DB2 will check if connection count started declining and will raise back the health to the system value. The health reported to WLM can be displayed using –DISPLAY DDF DETAIL. APAR number is PM43293.

Location Monitoring

- **24/7 availability and bigger counters : Previously stat counters were 4 bytes and wrapped quickly. For 24/7 availability, counters have been extended to 8 bytes.**

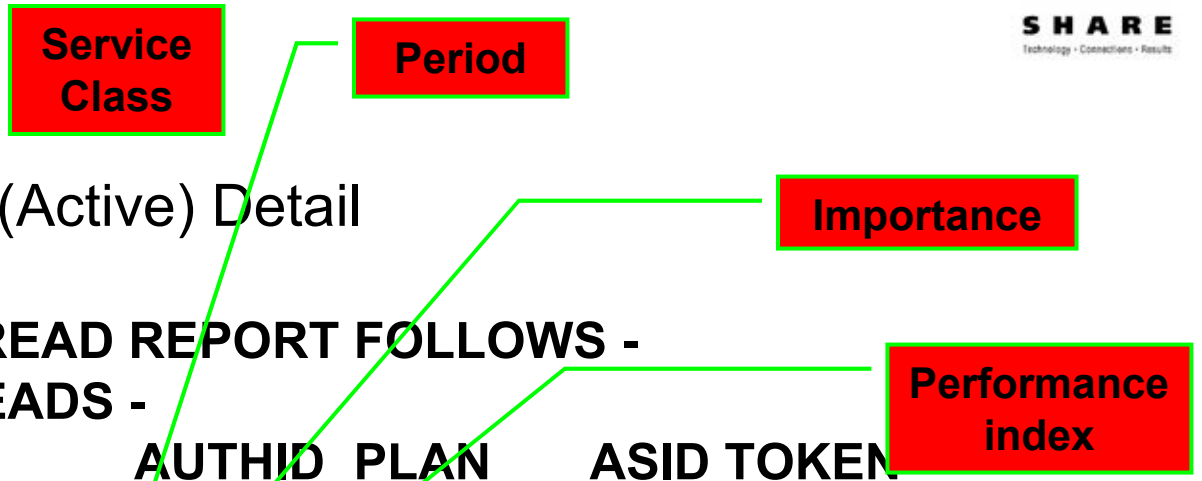
- **New IFCID 365 (class 7) : Previously, stats for all locations were grouped under DRDA REMOTE LOCS and written to SMF every minute. With IFCID 365, stats can be displayed by location and written to SMF at the interval specified by zPARM STATIME. You can provide tuning info for each app server depending on where work is coming from.**

Monitoring DBAT's WLM goals (DB2 10)

- *You can use the –**DISPLAY THREAD** command to find out the following WLM* characteristics associated with the DBAT (separated by a colon)
  - **Service Class Name –** Named group of work with similar goals and resource requirements.
  - **Importance Level of the period -** The relative importance of the service class period goal. Only used when goal is not being met. The value can be 1-5, or DISCRETIONARY(5), 1 being the highest importance.
  - **Service class Period Number -** Performance periods are available for work that has variable resource requirements and for which your goals change as the work uses more resources. You specify a goal, an importance, and a duration for a performance period. You can specify up to eight performance periods.
  - **Performance Index of the service class period -** A calculation of how well work is meeting its goal. For work with response time goals, PI is the actual divided by goal.
    - PI=1 (period is exactly meeting its goal)
    - PI<1 (period is beating its goal)
    - PI>1 (period is missing its goal)
    - DISCRETIONARY = 0.8

Example

-Display Thread(*) Type(Active) Detail




```
DSNV401I > DISPLAY THREAD REPORT FOLLOWS -
DSNV402I > ACTIVE THREADS -
NAME       ST A REQ   ID        AUTHID  PLAN     ASID TOKEN
TEST0001 RA  *   1  CORID001 SYSADM DONSQL1 0050    6
V441-ACCOUNTING=HADERLE
V482-WLM-INFO=DDFSVCLS:1:5:1.1
V442-CRTKN=9.30.113.201.5001.C68EA0176B20
V445-USIBMSY.SYEC717A.C68EA0176B20=6 ACCESSING DATA
     FOR
( 1)::FFFF:9.30.113.201
V447--INDEX SESSID A ST TIME
V448--( 1) 447:1027 W R2 1025213462763
DISPLAY ACTIVE REPORT COMPLETE
DSN9022I - DSNVDT '-DISPLAY THREAD' NORMAL COMPLETION
```

How Should I Classify Threads:

Work qualifiers are used to help identify a thread

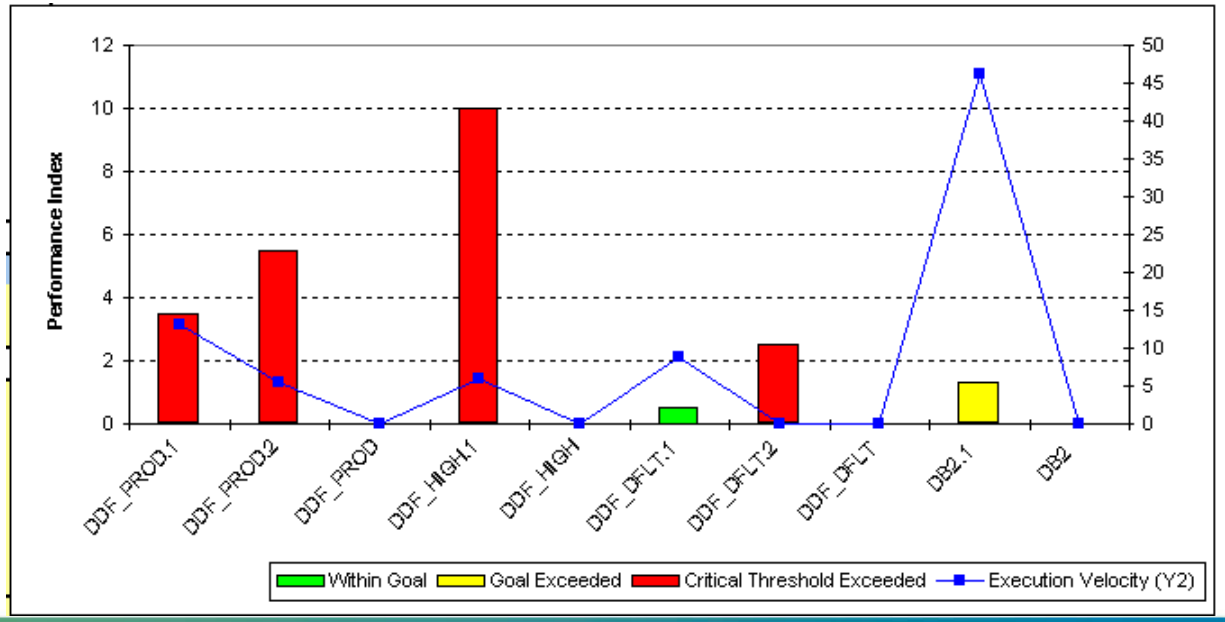
- AI - Accounting Information*
- CI - Correlation Information*
- CN - Collection Name
- CT - Connection Type
- CTG - Connection Type Group
- LU - LU Name
- LUG - LU Name Group
- NET - Net ID
- NETG - Net ID Group
- PC - Process Name
- PF - Perform
- PFG - Perform Group
- PK - Package Name *
- PKG - Package Name Group
- PN - Plan Name
- PNG - Plan Name Group
- PR - Procedure Name
- PX - Sysplex Name
- SI - Subsystem Instance
- SIG - Subsystem Instance Group
- SSC - Subsystem Collection
- UI - Userid

* Popular choices

- Best Practice
  - Ensure all remote workload is less important than DDF address space
  - Classify transactionary work based on response time goals (2 periods), not velocity goals – **AND NOT DISCRETIONARY**

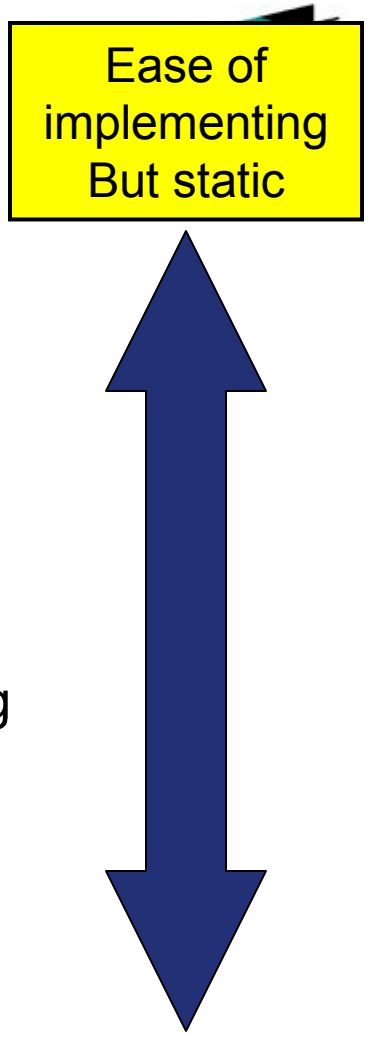
WLM classification example

- This is an example of ill-fated goals
  - DDF workload imp. 1 high velocity goal
  - DB2 DIST high velocity goal as well
    - Deadly embrace as DDF uses resources DB2 needs to accept new work
  - DDF response time could go up by factor of 10x, because duration is not part of the goal nor monitored by WLM with a velocity goal
  - If you continuously miss your goal WLM will skip you



How do I set Client-info to better classify trans.?

- In data source
  - All applications sharing data source appear the same to DB2
  - Need source per application to change information
- Calling WLM_SET_CLIENT_INFO stored procedure
  - Requires application add a call to proc and populate the information
- pureQuery product – makes JAVA into static packages
- Having application set it
  - WSConnection() method to set correlation and accounting info
- Create a wrapper from incoming getConnection() string that dynamically picks up program name and IDs
  - Can use Hibernate or Spring class to populate their intermediary config file
  - Could use a wrapper from Websphere that uses getConnection() and WSSubject class to pull the information out of the incoming request to populate client info



Example of Creating a DDF Service Class for a Specific Application Continued . . .

- If a service class of 'PC=AdrianBurke', or 'CI=agb_v9' had been created and the application code contained WSConnection.CLIENT_APPLICATION_NAME, "agb_v9", or connectionProperties.put("clientProgramName", "agb_v9"), and connectionProperties.put("clientApplicationInformation", "AdrianBurke") in connection string: then the snapshot of the enclave screen would show the following details:

```
DSNV402I  DB1S ACTIVE THREADS -
NAME     ST A   REQ ID           AUTHID   PLAN     ASID TOKEN
SERVER   RA *   209 agb_v9       DNET060  DISTSERV 008F  7606
 V437-WORKSTATION=IBM-69E1CCDBEAA, USERID=dnet060,
      APPLICATION NAME=AdrianBurke
 V441-ACCOUNTING=JCC03580IBM-69E1CCDBEAA   AdrianBurke
 V445-G9410381.G78F.C7DAE9715092=7606 ACCESSING DATA FOR
  (  1)::9.65.3.129
 V447--INDEX SESSID          A ST TIME
 V448--(  1) 5446:1935       W R2 111512142215O
         PT *     0 agb_v9       DNET060  DISTSERV 008C  7626
```

Distributed Correlation Enhancements

Previously, the correlation token used to correlate work between the client and server was only externalized in DB2 accounting data and not in DB2 messages making it impossible for users to correlate message related failures to the remote client application that is involved in the failure.

In DB2 10, the THREAD-INFO description for key messages will include this token, which will be enclosed in '<' and '>' characters, and contains three components,

separated by a period:

- **A 3 to 39 character IP address.**
- **A 1 to 8 character port address.**
- **A 12 character unique identifier.**

Example :

DSNL027I > SERVER DISTRIBUTED AGENT WITH
LUWID=G91702F8.P853.100629180434=4
THREAD-INFO=ADMF001:mask:admf001:db2bp:*:*:*:
<9.23.2.248.38995.100629180434>
RECEIVED ABEND=04E
FOR REASON=00D3001A

```
REQUESTER  : ::131.126.51.231      CORRNAME: G1PBTC00      LUW LUN: DA84
MAINPACK   : YPKSFC00              CORRNMBR: 4             LUW INS: C8A688E43E56
PRIMAUTH   : G1PDBC                CONNTYPE: DRDA          LUW SEQ:          71
ORIGAUTH   : G1PDBC                CONNECT : SERVER
```

Optional Domain name

- ***Previously, we did not allow DB2 to process TCP/IP requests without configuring a domain name, even when the IP address that domain name maps to is always fixed.***

- In v10, for users that specify fixed IP addresses in the DB2 BSDS, a domain name is no longer required to be configured to process TCP/IP requests.

  - When a domain name is unavailable, A DSNL523I message that contains the BSDS-specified IP address will be issued, in lieu of a DSNL519I message to indicate that DDF is ready to accept requests for that IP address.

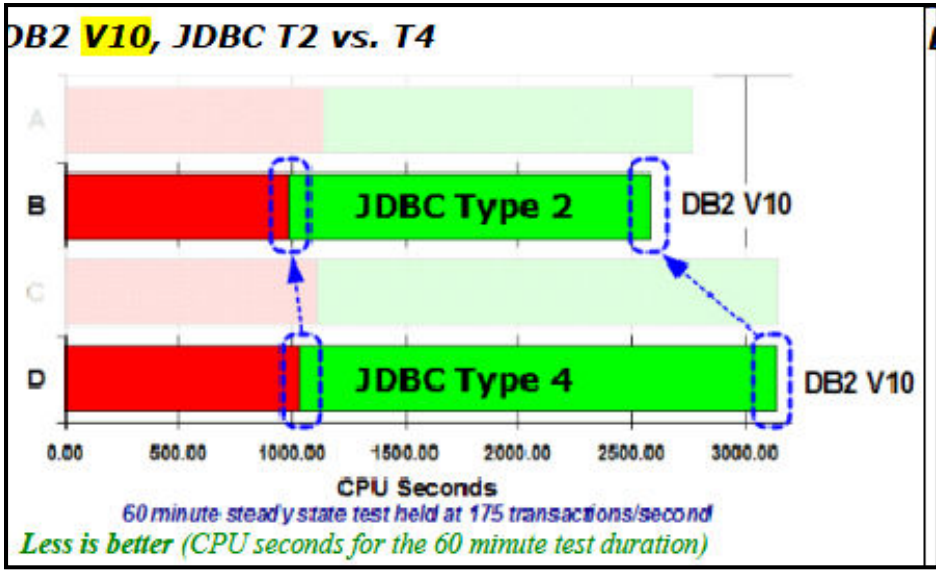
Whats new with remote threads in DB2 10…

- High Performance DBATs is a new type of distributed thread (some customershave seen XX% perf. Gain)
  - Packages must be bound with RELEASE(DEALLOCATE) to get reuse for same connection and -MODIFY DDF PKGREL(BNDOPT) must also be in effect
    - Since V6 they have been RELEASE(COMMIT)
    - Now DBAT and client connection will remain active together
    - After the Hi-Perf DBAT has been reused 200 times
      - DBAT will be purged and client connection will then go inactive
    - If Hi-Perf DBAT has not received new work for POOLINAC time
      - DBAT will be purged and the connection will go inactive
    - If # of Hi-Perf DBATs exceed 50% of MAXDBAT threshold
      - DBATs will be pooled at commit and package resources copied/allocated as RELEASE(COMMIT)
    - Hi-Perf DBATs can be purged to allow DDL, BIND, and utilities to break in
      - Via -MODIFY DDF PKGREL(COMMIT)

Cost of deployment for type 2 vs. type 4 drivers

- Type 4 saved CPU over 3 years ago
  - Even then T4 used 23% more total processing cycles (PTV8)
  - 23% less ITR (PVT8)
  - 19% increase in elapsed time (PVT8)
- Current V10 numbers show T2 saves 5% on CP, and 17% CP+zIIP

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/5cb5ed706d254a8186256c71006d2e0a/94f9d53f2e526489862575c5004b50b1/$FILE/WP101476-2%20-%20Value%20of%20Co-Location%20Update.pdf



Extras……

- No-Charge workshop (email me)
  - System z Synergy workshop focused on Websphere (LUW or z) and DB2 for z/OS, settings, best practices, lessons learned

Typical Agenda

- Introduction (15 min)
- WebSphere Application Server Performance Considerations (1.5 hours)
- DB2 Performance Considerations (1.5 hours)
- Tuning WebSphere Application Server and DB2 to work together (1.5 hours)

What can you expect?

A day filled with information to help you to gain an understanding of how you can monitor and tune DB2 for z/OS and WebSphere Application Server for z/OS for better performance, how to tune the two subsystems to work together optimally and how they can be used to benefit your enterprise.

Questions???

Adrian Burke
DB2 SWAT Team SVL Lab
agburke@us.ibm.com

- **VISIT the DB2 Best Practices**
- **VISIT the DB2 for z/OS Exchange**
- **JOIN the World of DB2 for z/OS**
- **JOIN the DB2 for z/OS group**