IDUG

2023 NA **Db2** Tech Conference

IDUG

Leading the **Db2** User Community for **35** Years

Who's afraid of DDF

Toine Michielse, Broadcom

Session Code: A16

Philadelphia

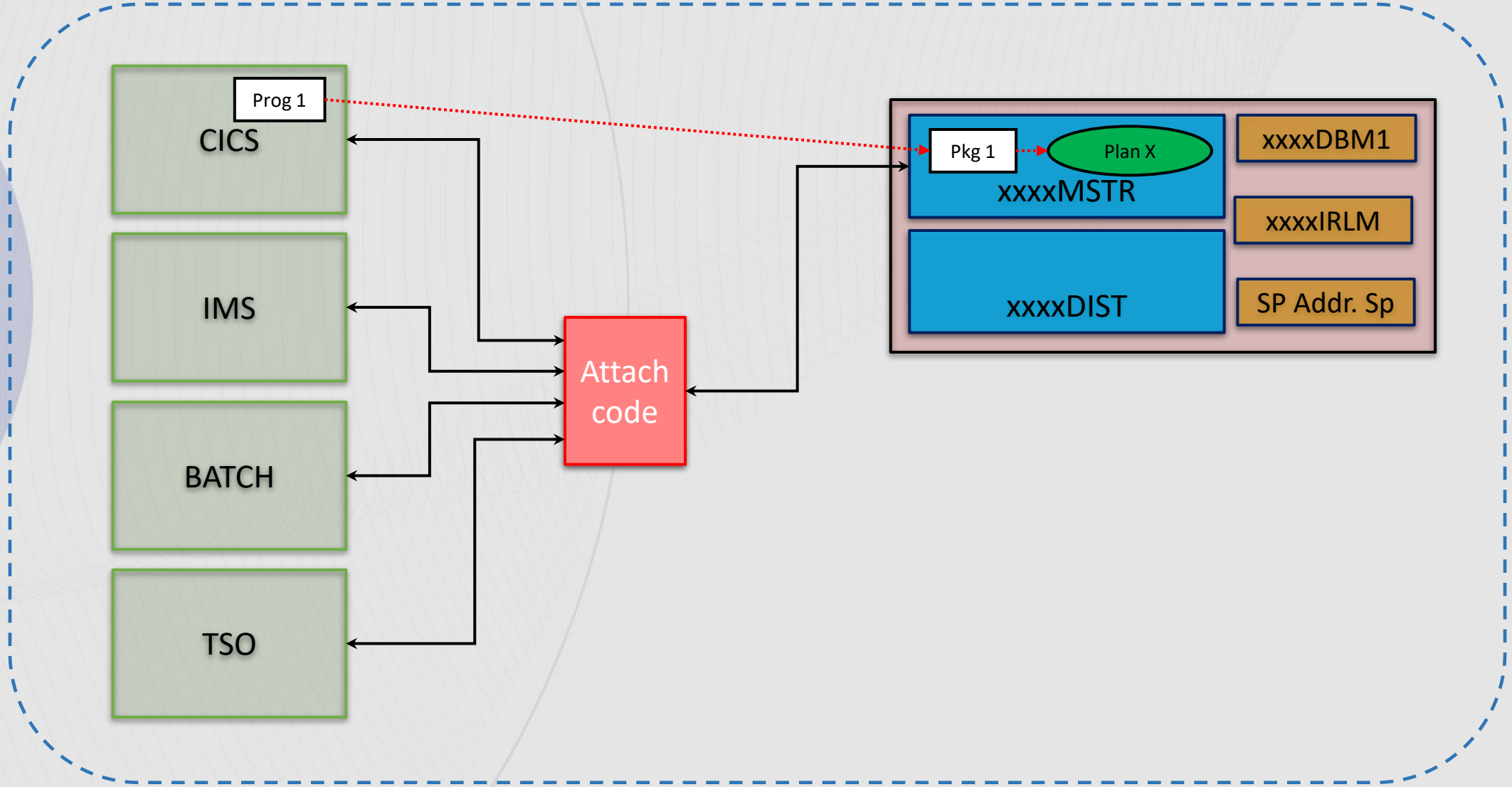
Agenda

- Introduction
- Identification
- Application topics
- Taking control
- Questions

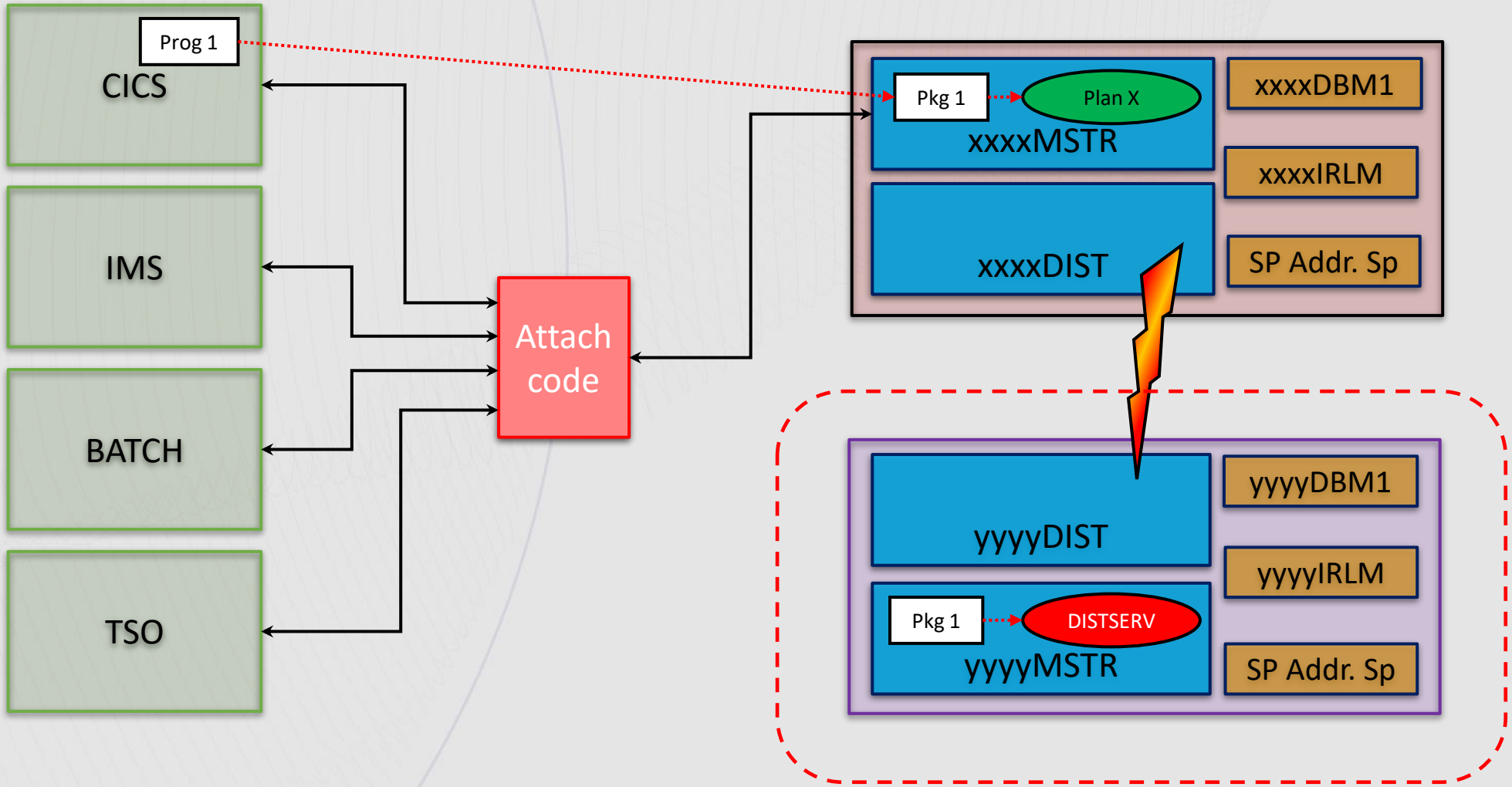
Let me quickly introduce myself

- Toine Michielse, born in The Netherlands
  - DB2 programmer, DBA, System Engineer, Architect
- Worked for many years for IBM as Db2 for z/OS Lab Advocate
- Came to Madrid from Switzerland
  - Busy learning Spanish (and padel)
- My passions:
  - Db2, data, mainframe modernization
  - Paragliding
  - Playing drums with “Ciencia Urbana”

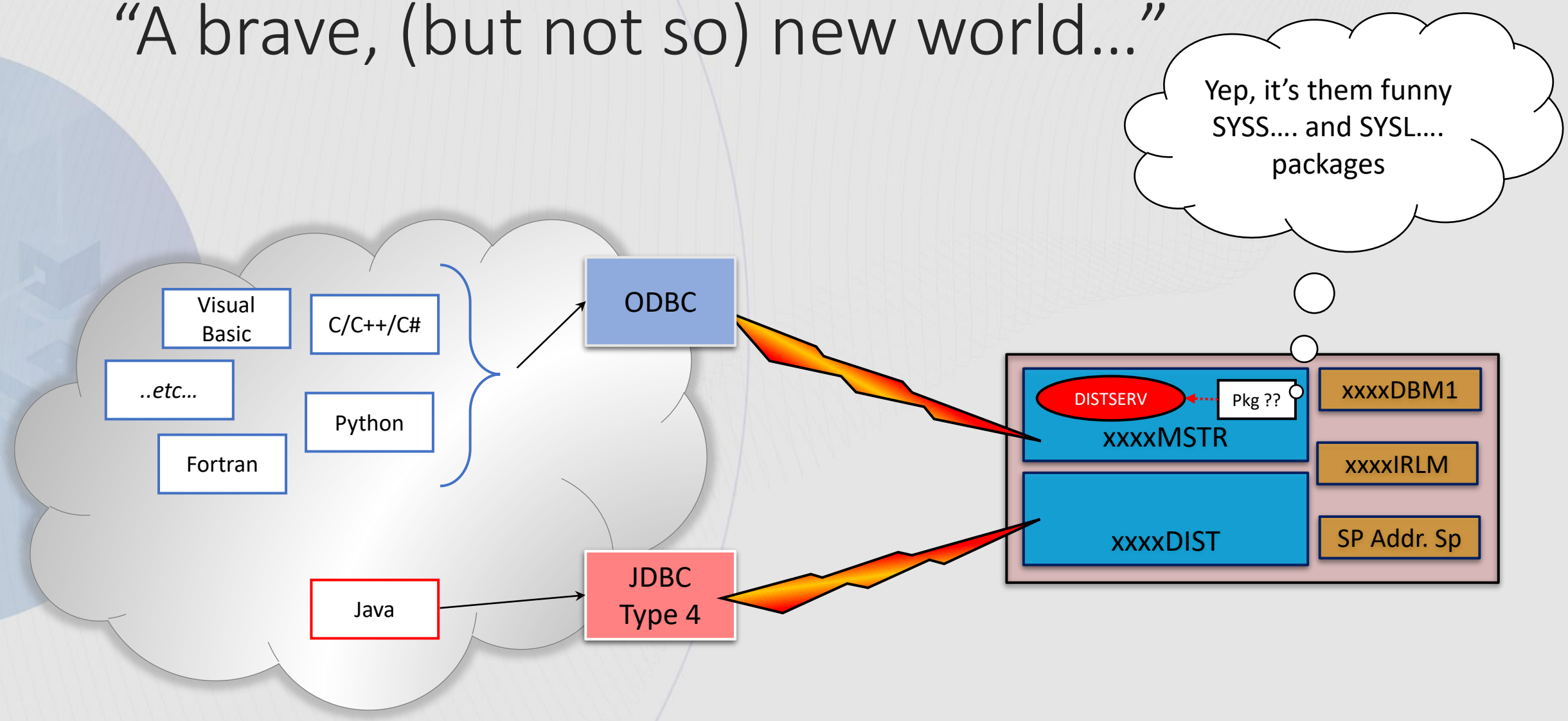
“Traditional” application connect architecture



“Traditional” application connect architecture



“A brave, (but not so) new world…”



Tell us who you are!!!

An error occurred…

Something went wrong…

Tell us who you are!!!

- Without proper identification, solving performance or capacity problems becomes a nightmare

- The reason DISTSERV and generic packages

Naming Convention for CLI packages:

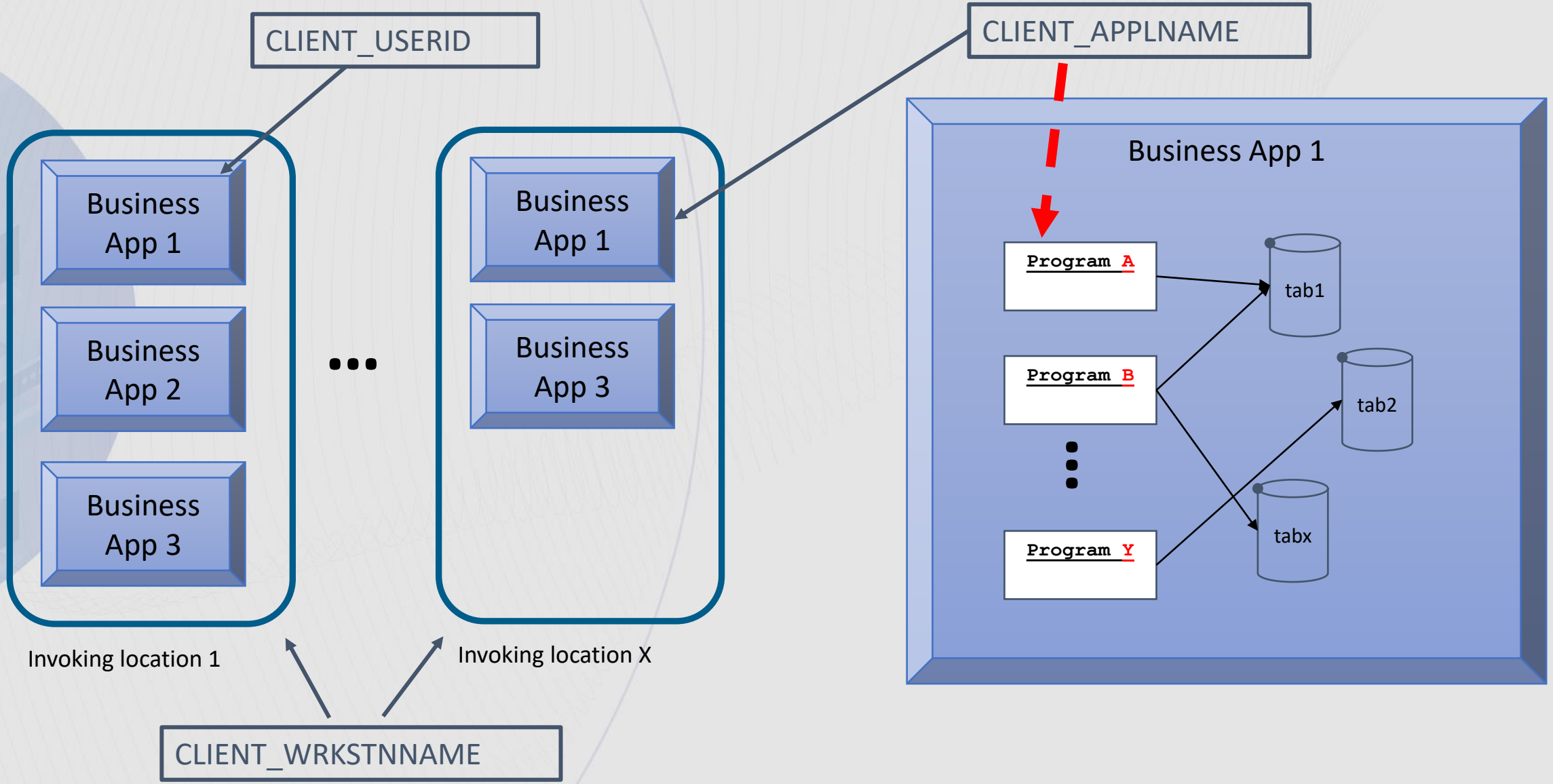
SYSSxyy and SYSLxyy

'S' represents a small package, and 'L' represents a large package
'H' represents WITH HOLD, and 'N' represents NOT WITH HOLD
'x' is the isolation level: 0=NC, 1=UR, 2=CS, 3=RS, 4=RR
'yy' is the package iteration 00 through FF

Benefits of proper identification

- Enable profiling in DB2 for z/OS
  - Set limits for specific applications, avoid single application monopolizing resources
  - Capacity management and protection
- Enable workload protection using WLM classification
  - Protect your bread-and-butter applications from being impacted by adhoc SQL
  - Avoid having to add capacity with associated monetary impact
- Enable different level of monitoring, dash boarding and capacity insights
  - IMMEDIATE identification of **program** causing an issue in Detector/Apptune/Query Monitor for instance
- Enable proper charge back
  - Properties end up in SMF records / MICS

Positioning applications in a distributed world



Identifying SQL initiated off-Mainframe

Java	.NET / .CFG	WLM	Profile	Special register	CLI/ODBC	DB2 SP setable
ClientAccountingInformation	ClientAccountingString	CAI	N	CLIENT_ACCTNG	QWDASUFX	Y
ClientProgramName	<none>	CI	N		QWHCCV	N
ClientCorrelationToken	ClientCorrelationToken	<none>	N	CLIENT_CORR_TOKEN		N
ClientUser	ClientUserID	CUI	Y	CLIENT_USERID	QWHCEUID_Var	Y
ClientHostName	ClientWorkStationName	CWN	Y	CLIENT_WRKSTNNAME	QWHCEUWN_var	Y
ApplicationName	ClientApplicationName	CTN	Y	CLIENT_APPLNAME	QWHCEUTX_Var	Y
ApplicationInformation	ClientApplicationName	PC	Y	CLIENT_APPLNAME	QWHCEUTX	Y

- Call the java.sql.Connection.setClientInfo / java.sql.Connection.getClientInfo method
  - i.e. conn.setClientInfo(“ApplicationName", “MyProg”)
- Use the .NET get or set methods from Db2Connection Class: public function set ClientApplicationInformation(String)
- CLI/ODBC applications use SQLSetConnectAttr() function
- As a ***last resort***: use SQL: SQL CALL WLM_SET_CLIENT_INFO stored procedure (part of Db2 for z/OS)

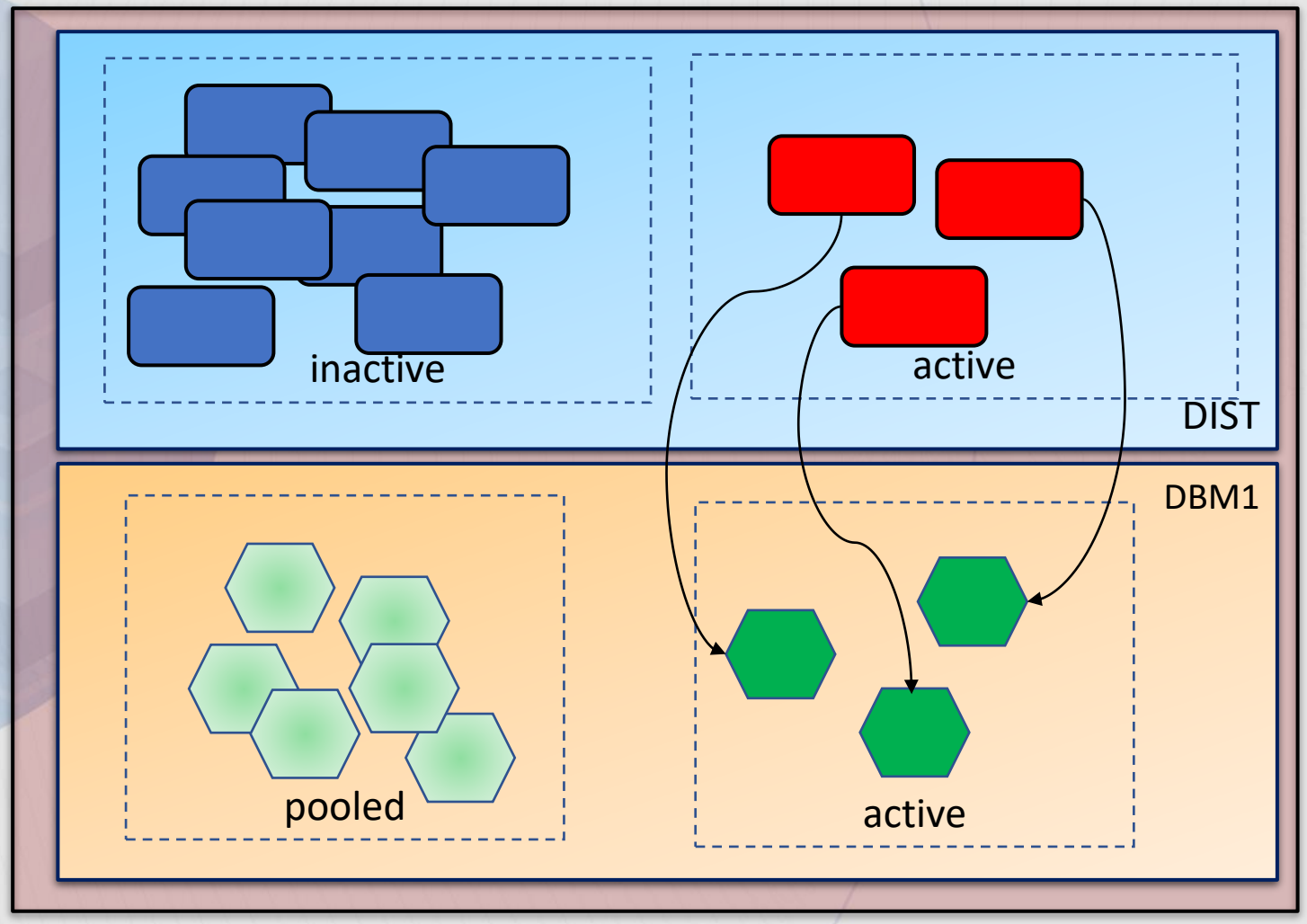
What if it is too late…..

- Websphere DataSourceDefinition
  - ClientApplicationInformation
    - In Data Source extended property or Resource Reference of the given application
- DB2JccConfiguration.properties
  - ClientApplicationName
- Db2cli.ini
  - ClientApplName
- Db2dsdriver.cfg
  - ClientApplName

Performance considerations

- Managing connections and DBATs
  - Avoid delays due to resource shortage
  - Avoid overwhelming the system by over configuration and roque applications
- SQL / Data now needs to be transmitted over a network protocol
  - Minimize message exchange
- More opportunities for SQL binds
  - Access path changes impose a risk to performance stability

How about those connections and threads?



- CMSTAT INACTIVE
  - Allow DBAT disassociated from connection
  - At commit (*)
    - Active thread to pool
      - POOLINAC drives cleanup
    - Active connection, inactive
  - RELEASE(DEALLOCATE)
    - DBAT remains associated with connection (HiperDBAT)
    - **Beware of “unhealthy” application behavior with RELEASE(DEALLOCATE)**
    - Allow processes to break in:
      - MODIFY DDF PKGREL

Thread control….

- Active DBAT vs connection
  - Avoid queuing for a DBAT
  - Promote thread pooling
  - CONDBAT, MAXDBAT, POOLINAC

```
DSNL080I  !XXXX DSNLTDDF DISPLAY DDF REPORT FOLLOWS:
DSNL081I STATUS=STARTD
DSNL082I LOCATION           LUNAME            GENERICLU
DSNL083I XXXXXXXX           -NONE             -NONE
DSNL084I TCPPORT=XXXXX SECPORT=0     RESPORT=XXXXX IPNAME=XXXXXXXX
DSNL085I IPADDR=::XX.XXX.XX.XX
DSNL086I SQL    DOMAIN=your.domain.name
DSNL090I DT=I  CONDBAT=    600 MDBAT=  600
DSNL092I ADBAT=    1 QUEDBAT=      0 INADBAT=      0 CONQUED=      0
DSNL093I DSCDBAT=      1 INACONN=      5 IUDBAT=      0 PQDBAT=      0
DSNL105I CURRENT DDF OPTIONS ARE:
DSNL106I PKGREL = COMMIT
DSNL106I SESSIDLE = 001440
DSNL099I DSNLTDDF DISPLAY DDF REPORT COMPLETE
```

- Consider using profiling for more granular control
  - MONITOR (ALL) CONNECTIONS
  - MONITOR (ALL) THREADS

Profile tables

- Objects created in DSNTIJSG
- Use to monitor and/or control connections and threads
  - More granular ZPARM settings or special register settings
- DSN_PROFILE_TABLE defines the scope of a given profile
  - AUTHID
  - IP Address (es), (also location or location alias)
  - Various other attributes
- DSN_PROFILE_ATTRIBUTES defines what needs to be done for a match on a given profile
  - Relevant categories: REMOTE CONNECTIONS, REMOTE THREADS

Profile tables: activation

- Profiles with PROFILE_ENABLED = ‘Y’ are activated through
  - -START PROFILE command
    - Local to each member
- After start profile, must verify:
  - STATUS column in SYSIBM.DSN_PROFILE_HISTORY / ATTRIBUTES_HISTORY
    - ‘ACCEPTED’ or ‘REJECTED’

Profile tables: sample 1

DSN_PROFILE

PROFILEID	LOCATION	PROFILE_ENABLED	CLIENT_APPLNAME	…	…
1		Y	APP1		
2	appsrv2.bc.com	Y			
3		Y	APP3		

DSN_PROFILE_ATTRIBUTES

PROFILE ID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2
1	MONITOR THREADS	WARNING	50
2	MONITOR CONNECTIONS	EXCEPTION	10
3	MONITOR IDLE THREADS	EXCEPTION	20
3	MONITOR THREADS	WARNING	50

Profile tables: sample 2

DSN_PROFILE

PROFILEID	LOCATION	PROFILE_ENABLED	CLIENT_APPLNAME	…	…
1	appsrv1.bc.com	Y			
2	::0	Y			

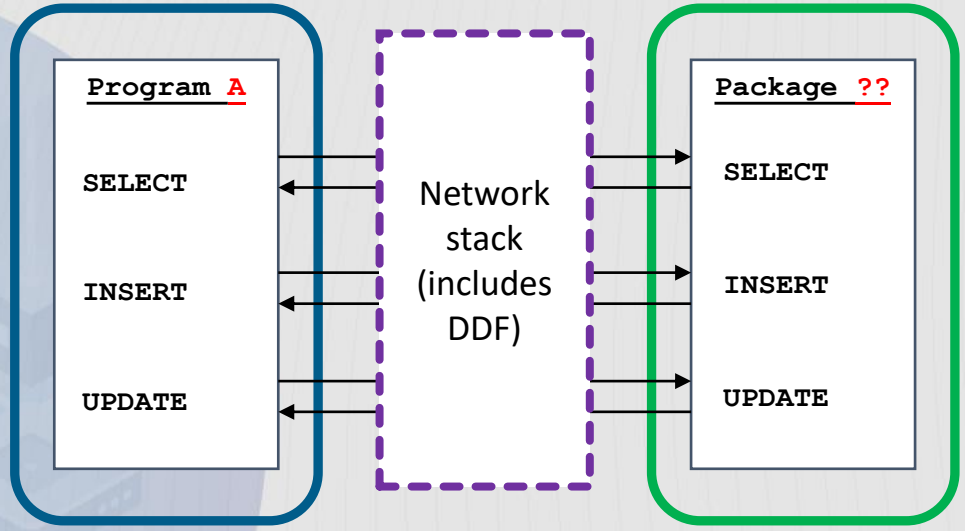
DSN_PROFILE_ATTRIBUTES

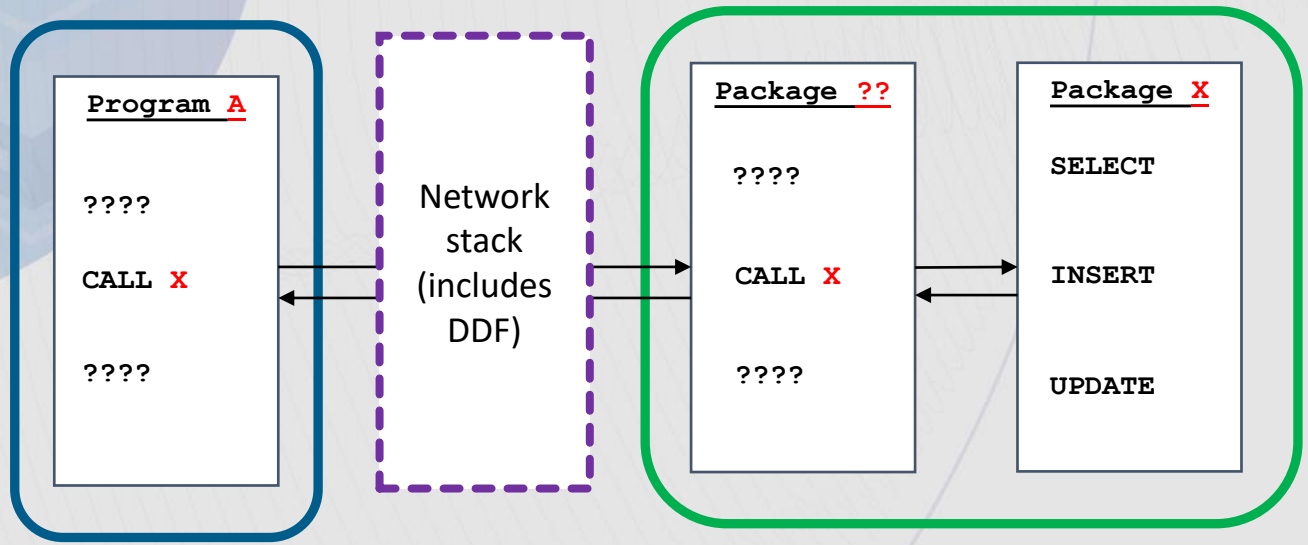
PROFILE ID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2
1	MONITOR CONNECTIONS	EXCEPTION	50
2	MONITOR CONNECTIONS	EXCEPTION	5
2	MONITOR ALL CONNECTIONS	EXCEPTION	500

SQL best practices distributed access

- Use parameter markers instead of values in the SQL statement text
  - Improve dynamic cache residency / hitratio dramatically
  - Exceptions maybe parameter markers for columns with extreme skew
- Always consider block fetch for read SQL
  - Save network message exchanges
  - Need to look at isolation level
- Minimize impact on other applications
  - Healthy SQL, healthy locking behavior
  - Isolation level. Warning: understand difference in Java isolation level and DB2 for z/OS isolation level

Where to code the SQL





- Major benefit of SP is avoiding network traffic
  - At cost of more complex architecture
  - Still uses DDF to invoke the SP code
- Package **??** can be a generic package like SYSxxyyy
  - Of interest is finding **A**, the source that ships the SQL
- Side benefit, you now have an identification for **PART** of the application
  - Only identify SQL wrapped in package X
  - The usage of SP X in Program A may be part of the problem though
  - Identification can be achieved by other means
- This doesn't necessarily help in identifying 'business application' or even caller

So now it’s all dynamic…. Or is it???

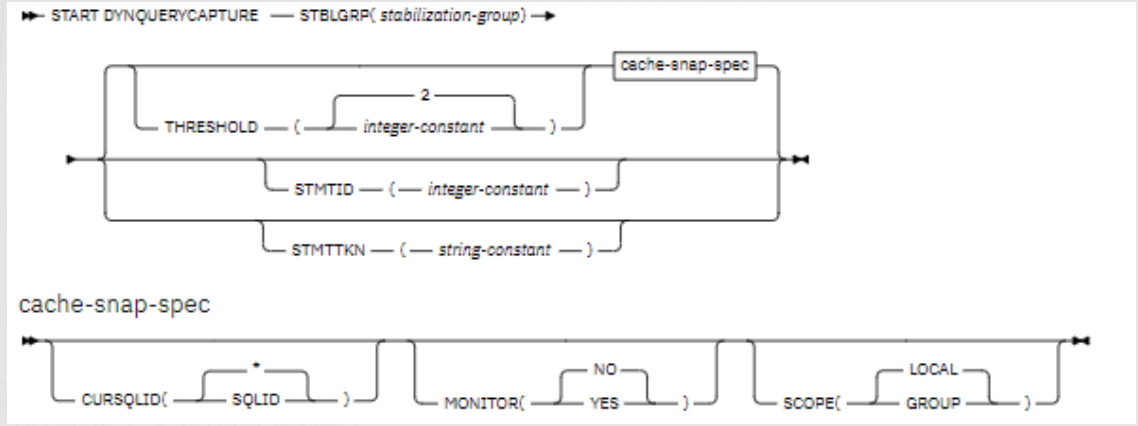
- Dynamic Statement Cache
  - Static SQL behavior under conditions
  - Keep an eye on relevant statistics

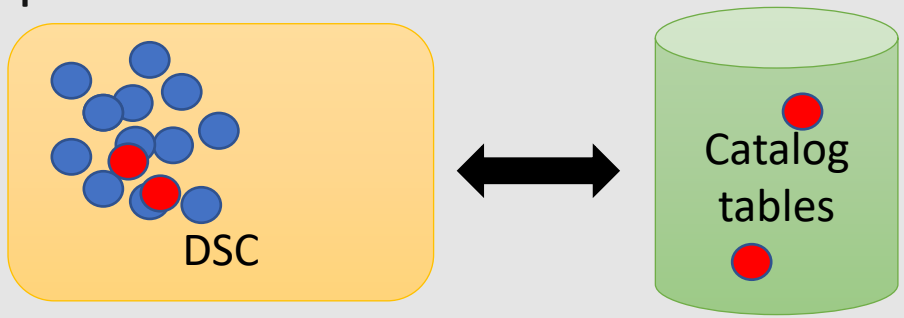
- Pay attention to local and global statement cache.
  - Monitor regularly for “cache thrashers”

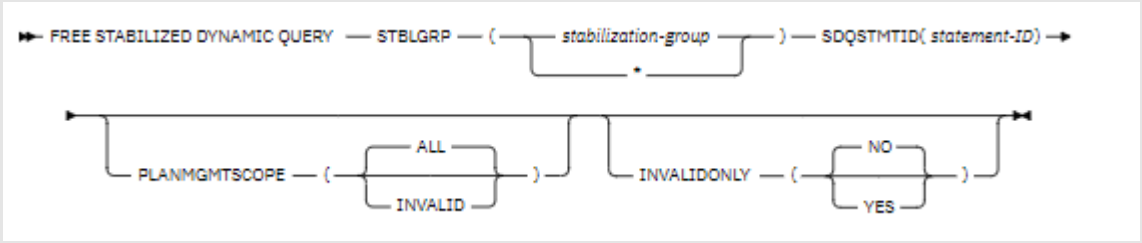
- Consider usage of “concentrate literals”
  - But beware of the downside (skewed data)

So now it’s all dynamic…. Or is it???

- Dynamic access path stabilisation
- Can be snapshot or continuous monitoring
- Statements can be invalidated
  - i.e. by DROPs
- When matching SQL text is not found in DSC,
  - Structure are loaded from catalog, avoiding full prepare
- Statements can be removed
  - FREE







SQL best practices for distributed access

- Optimize message exchanges
  - Application level (multi-row processing)
- Enable block fetch
  - Read only cursors
    - FOR FETCH ONLY
    - ISOLATION and CURRENT DATA option
    - OPTIMIZE FOR x ROWS
- Block fetch types:
  - Limited
    - Single stream, synchronous processing
  - Continuous
    - Multiple query blocks on each message exchange

SQL best practices for distributed access

- Ensure to disable auto commit in JDBC (perhaps also other protocols)
  - In Java the default is auto commit yes.

- Use the JDBC and ODBC batching interface

```
SQLCHAR *BatchStmt =
   "INSERT INTO Orders (OrderID, CustID, OpenDate, SalesPerson, Status)"
      "VALUES (2002, 1001, {fn CURDATE()}, 'Garcia', 'OPEN');"
   "INSERT INTO Lines (OrderID, Line, PartID, Quantity) VALUES (2002, 1, 1234, 10);"
   "INSERT INTO Lines (OrderID, Line, PartID, Quantity) VALUES (2002, 2, 987, 8);"
   "INSERT INTO Lines (OrderID, Line, PartID, Quantity) VALUES (2002, 3, 566, 17);"
   "INSERT INTO Lines (OrderID, Line, PartID, Quantity) VALUES (2002, 4, 412, 500)";

SQLExecDirect(hstmt, BatchStmt, SQL_NTS);
```

zIIP offload

- When DDF is involved, you have zIIP eligible workload. Use to your advantage
- Blocked workload support is not enabled for zIIP specialty engines
- The zIIP "needs help" function (IIPHONORPRIORITY = YES) can get delayed in a zIIP-constrained environment.
- Work classified as Discretionary to WLM does not benefit from the zIIP “needs help” function.
- JDBC Type 2 versus Type 4 driver
  - zIIP offload may come at a high cost……

Consider using dashboards like this…

Questions….

Broadcom Mainframe Technical Exchanges

In-person events are back!

- **North American in-person event in Plano, TX: June 13-15**
- **Global virtual event: October 3-5**

Make plans to attend

- Network with peers and Mainframe technical experts
- Technical education, product update, how-to and roundtable sessions
- No registration fee! Open to all Broadcom customers
- Learn more: https://bit.ly/MainframeTechEx

BROADCOM MAINFRAME SOFTWARE

Thank You

Speaker: Toine Michielse

Company: Broadcom

Email Address: toine.michielse@broadcom.com

Session Code: A16

Please fill out your session evaluation before leaving!