



Fear no Threads: Secure and Monitor Db2 Connections with Profile Tables

Toine Michielse

Sr. Manager Technical Consulting

toine.michielse@broadcom.com

Agenda

Technology landscape driving requirements for profile tables

Evolution of profile tables

Setting up and activating profile tables to:

- Control DDF workload
- Secure connection to Db2
- Break in to apply DDL changes for local application

Sample customer use case:
Large U.S. financial services firm

- Coerce unwanted java drivers from system



Profile tables have improved with each new Db2 release

New features and enhancements are added in each successive version of Db2



Db2 9

System profile monitoring introduced

- Primary use was for influencing characteristics of complex SQL

Db2 10

Manage increasing DDF workloads

- Monitor DBAT threads and connections
- Set subsystem parameters for SQL statements
- Set subsystem parameters for test
- Set thresholds for query acceleration
- Evaluate dynamic SQL for acceleration

Db2 11

Set values for special registers

- Keyword: SPECIAL_REGISTER
- Attribute is a SET statement

Db2 12

Set values for built-in Global Variables

- Keyword: GLOBAL_VARIABLE
- Attribute is a valid SET statement

Db2 13

Added support for local threads

- Keyword: RELEASE_PACKAGE



Profile tables explained

Objects created in DSNTIJS

Use to monitor and/or control connections and threads

- More granular ZPARM settings or special register settings
- Select Global Variable settings
- Set thresholds to manage and optimize for system performance

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

| Activating profile tables after defining

Profiles with PROFILE_ENABLED = 'Y' are activated through

- START PROFILE command

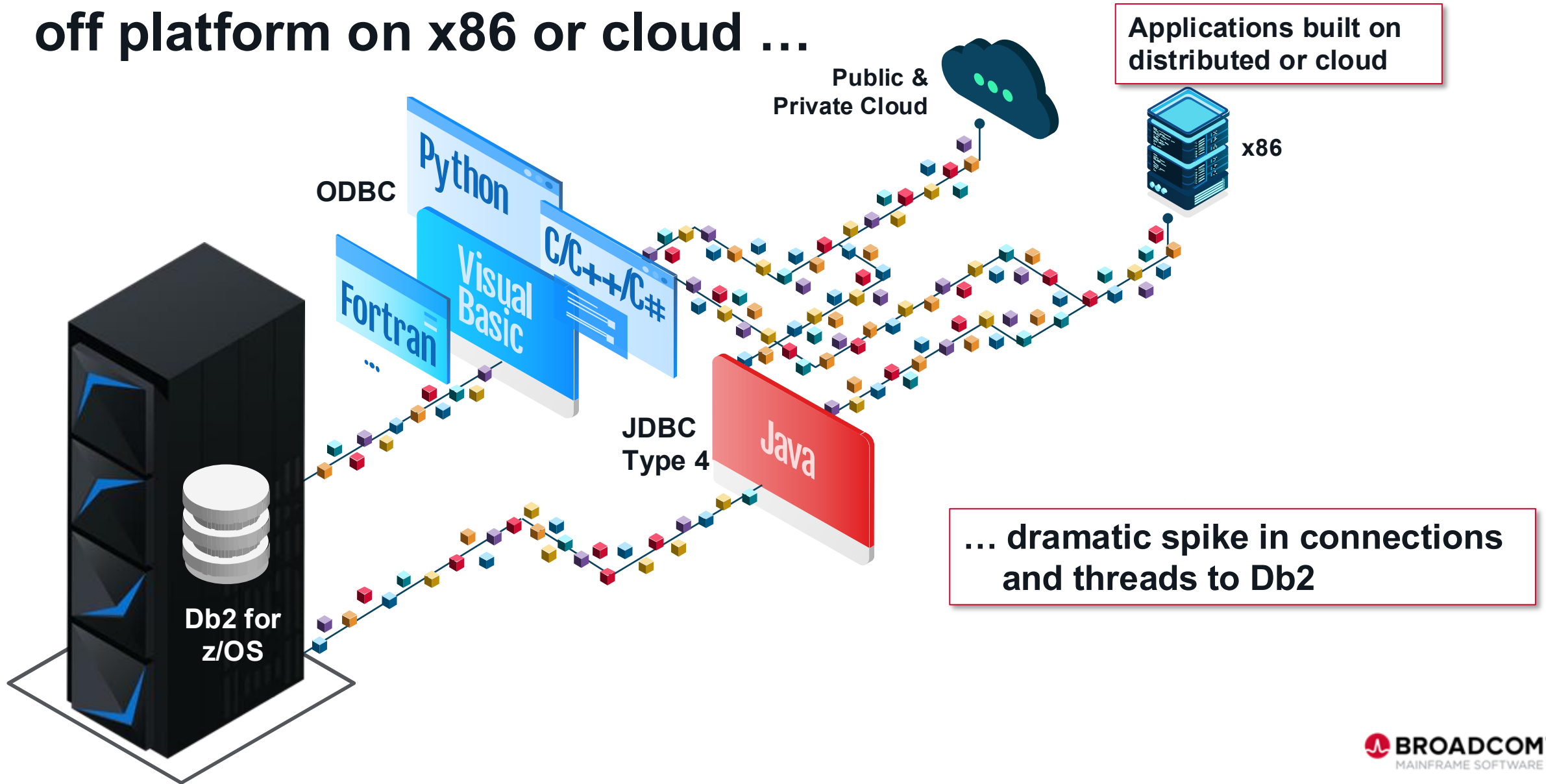
 - Local to each member

After start profile, must verify:

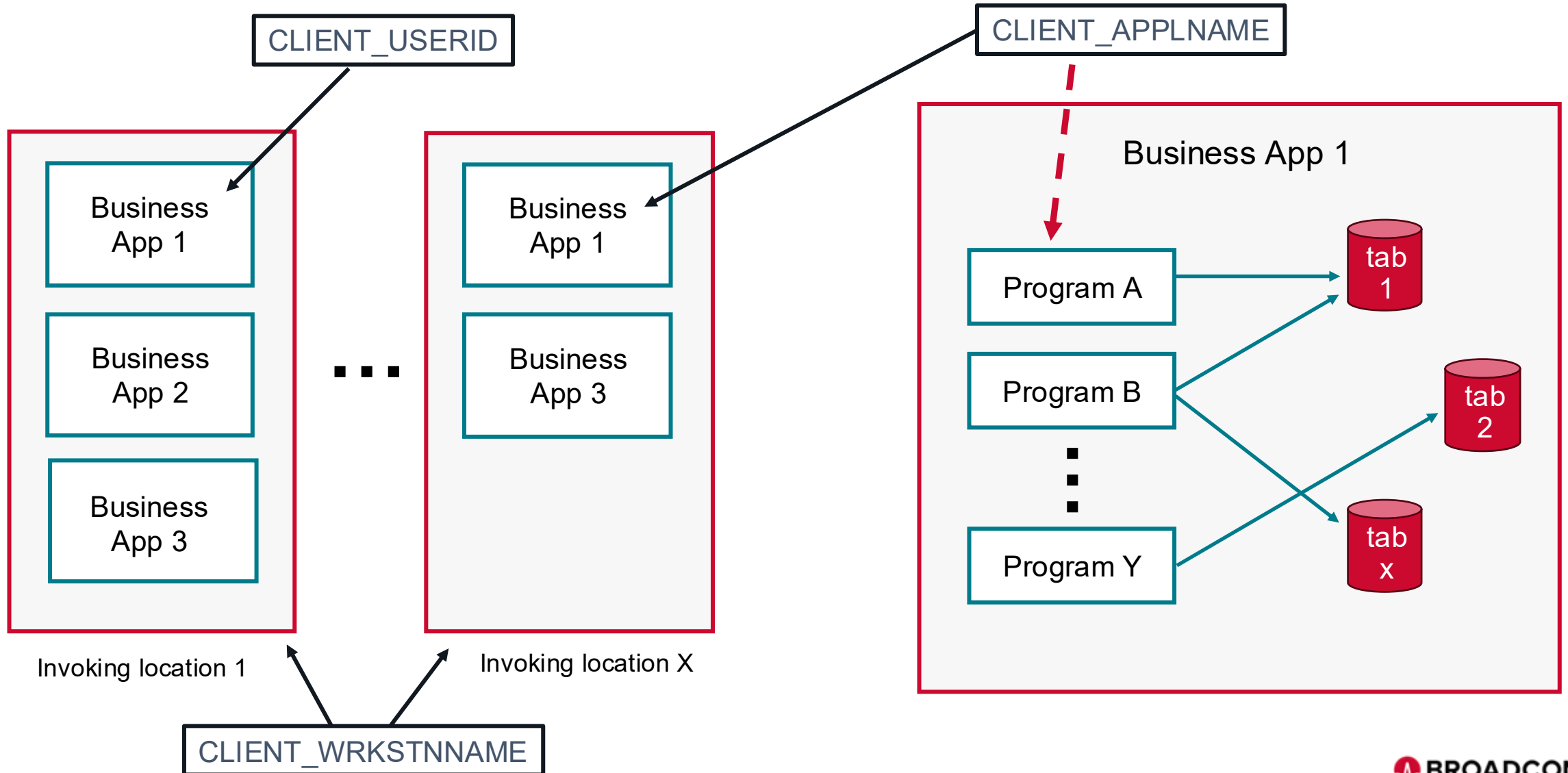
- STATUS column in SYSIBM.DSN_PROFILE_HISTORY /
ATTRIBUTES_HISTORY

 - 'ACCEPTED' or 'REJECTED'

Exponential growth in applications built off platform on x86 or cloud ...



Positioning applications in a distributed world



Start with proper program identification

Once distributed applications have been identified, there are numerous benefits, including:

- Enable profiling in Db2 for z/OS
- Enable workload protection using WLM classification
- Enable different level of monitoring, dash boarding and capacity insights
- Enable proper charge back

Steps to defining and activating profiles

1. Define applications	2. Enter value for Keyword	3. Enter value for Attribute1	4. Enter value for Attribute2
Define the Scope in DSN_PROFILE_TABLE • IP Address • AuthID • Driver levels • Roles • ...	Define function to perform in DSN_PROFILE_ATTRIBUTES • Monitor threads • Monitor connections • SPECIAL_REGISTER • GLOBAL_VARIABLES • RELEASE_PACKAGE	i.e. Tell Db2 to trigger a Warning or an Exception	i.e. Define the threshold for triggering the Warning or Exception

Once profile tables have been populated, issue the – START PROFILE command to load or reload profile tables into memory (*Broadcom's Plan Analyzer has profile services that simplifies this process*)

Sample profile table for monitoring connections and threads

DSN_PROFILE

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

DSN_PROFILE_ATTRIBUTES

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

Another example to monitor all connections for a set of locations

DSN_PROFILE

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

DSN_PROFILE_ATTRIBUTES

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

Db2 13 added new keyword to apply profiles to local connections

- To save CPU and improve elapsed time, packages are bound with RELEASE(DEALLOCATE); however, that prevents the DBA from being able to break in with critical DDL updates
- The new RELEASE_PACKAGE keyword in Db2 13 allows the DBA to break in with changes

DSN_PROFILE

PROFILEID	LOCATION	PROFILE_ENABLED	COLLID	...	...
99		Y	DEALLOC		

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2	...
99	RELEASE_PACKAGE	COMMIT	2	

- *DBA can start profile to force package to release at commit: ATTRIBUTE1 = COMMIT overrides the RELEASE setting for the collection*
- *ATTRIBUTE2 = 2 indicates scope applies to both local and distributed*

Profiles to support performance management

Model production environment in dev/test

- From IBM_SERVICE_DATA (plan_table in production)
 - Gather SIMULATED_CPU_SPEED, SIMULATED_CPU_COUNT, SORTPOOL_SIZE, MAX_RID_BLOCKS
- From Prod ZPARM:
 - NPGTHRSH, PARAMDEG, STARJOIN
- Create ZPARM with:
 - SIMULATED_CPU_SPEED, SIMULATED_CPU_COUNT, NPGTHRSH, PARAMDEG, STARJOIN



Profiles to support performance management

DSN_PROFILE

PROFILEID	LOCATION	PROFILE_ENABLED	CLIENT_APPLNAME	...	...
101		Y			

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2
101	SORT_POOL_SIZE	NULL	<SORTPOOL_SIZE>
101	MAX_RIDBLOCKS	NULL	<MAX_RID_BLOCKS>
101	BP0	NULL	500
101	BP1	NULL	10000

Sample
customer use
case:

U.S. financial
services firm



Manage driver level connections

Enforce applications to be compliant



DBAs were spending a lot of their time manually identifying these connections that were out of compliance.

Challenge: Threads were connecting to Db2 for z/OS using drivers that were out of support or at less than desired standards from:

- Developers/users did not upgrade drivers
- Developers/users did not go to central repositories to pick up the latest configurations

Impact: Significant stress on the system staff from back-level drivers, eating into DBA time to manage

DBAs can take back control with profile tables

- Activate distributed location statistics stats class(7) – IFCID(365) – to identify driver levels
- Set up profile table parameters using profile services in Broadcom's Plan Analyzer
 - Define scope using PRDID of non-conforming driver level
 - Set KEYWORDS to SPECIAL_REGISTER and ATTRIBUTE1 to SET CURRENT PACKAGE = 'NULLID_V11R1' to prevent the non-compliant drivers from accessing latest NULLID packages bound with Db2 v12 APPICompat behavior
- Application connections with non-conforming drivers were not allowed to access the latest features in Db2

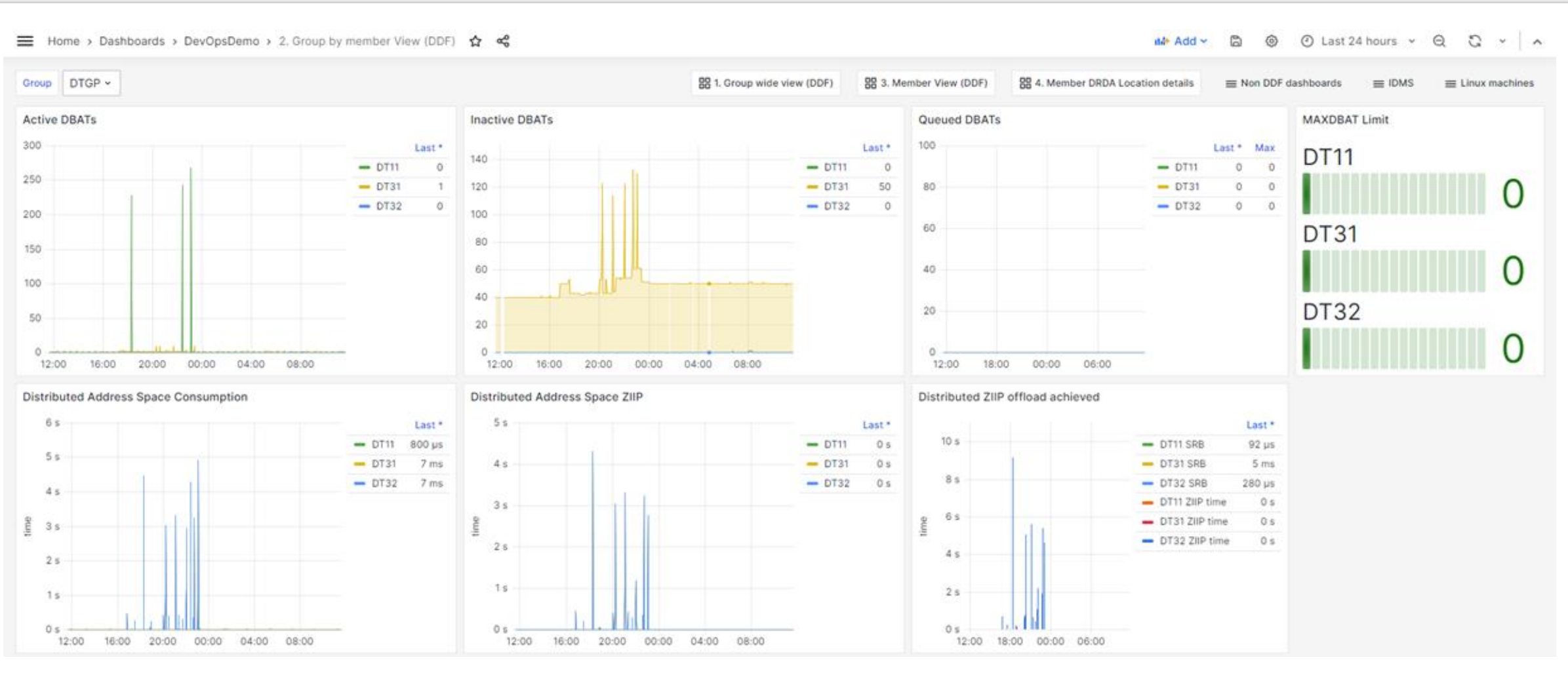
DSN_PROFILE

PROFILEID	LOCATION	PRDID	COLLID	AUTHID	...
60		JCC03720			

DSN_PROFILE_ATTRIBUTES

PROFILEID	KEYWORDS	ATTRIBUTE1	ATTRIBUTE2	...
60	SPECIAL_REGISTER	SET CURRENT PACKAGE = 'NULLID_V11R1'		

Consider using dashboards like this ...



Questions ...

