

---

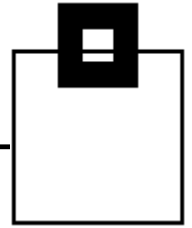
# Where's the BiF??

Roy Boxwell,  
Senior Software Architect,  
Software Engineering GmbH



# Agenda

---

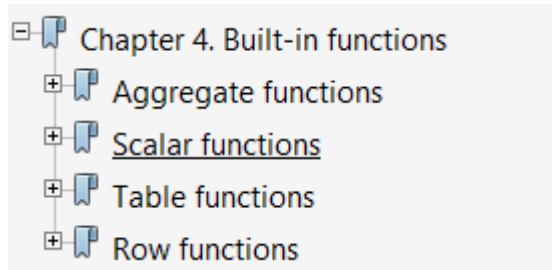


- What's a BiF?
- Where's the BiF?
- Customer Story
- Where's the BiF? – Wrap up
- Q&A

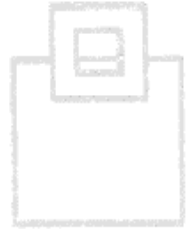
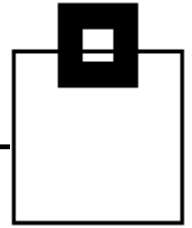


# What's a BiF?

A BiF is a Built-In Function such as CHAR, DECIMAL, etc.  
They exist in four different flavors:

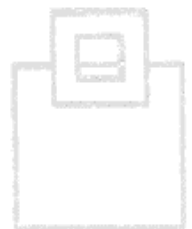
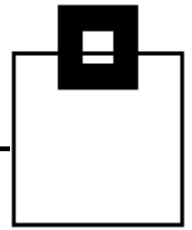
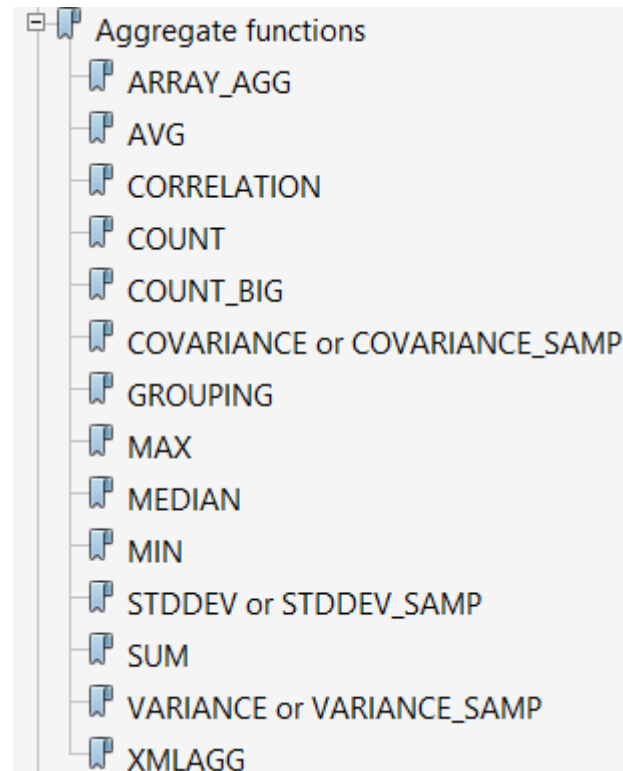


There are hundreds of them these days and every new release of DB2 brings more. In the last few DB2 releases, IBM has changed a few to make DB2 more compatible with SQL standards.



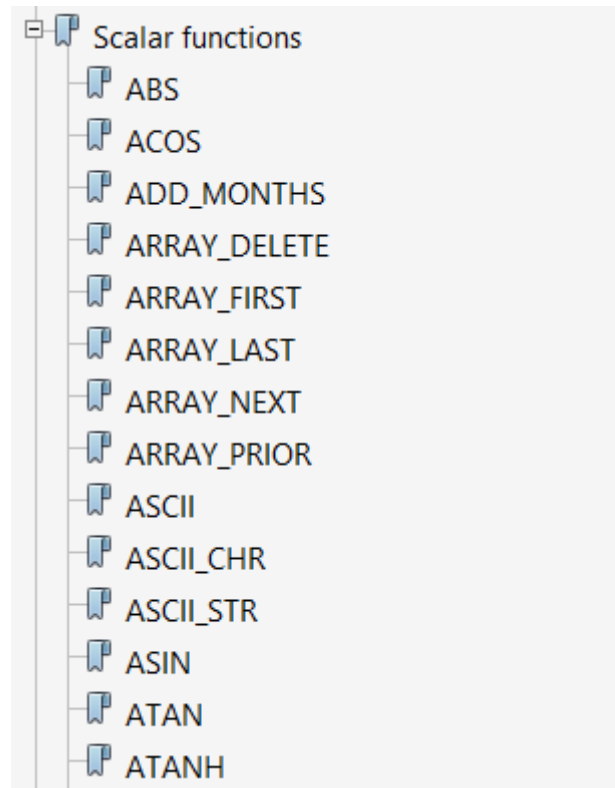
# What's a BiF?

Aggregate functions we all know - but mostly we use just AVG, COUNT, MAX, MIN, and SUM:



# What's a BiF?

Scalar functions are the ones we really use all the time. Here are the first few in DB2 11:



There are 184 of these...

# What's a BiF?

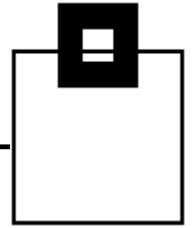
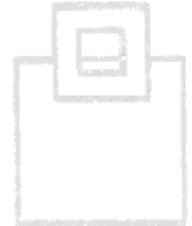
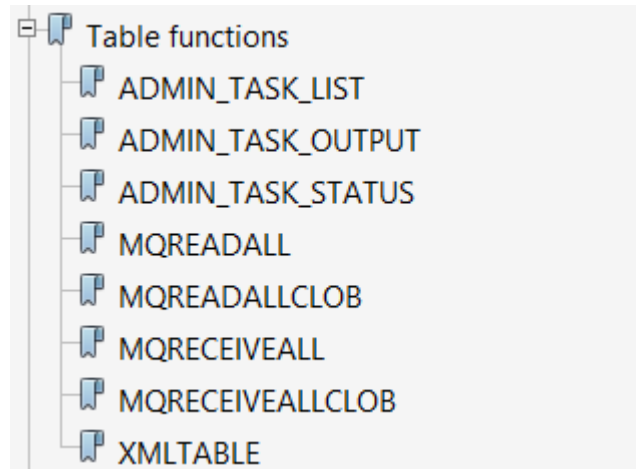


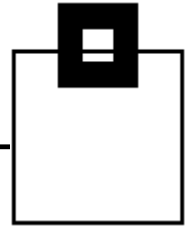
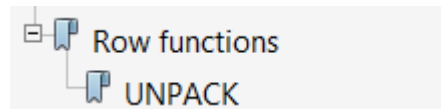
Table functions are a bit exotic and not used that often:



# What's a BiF?

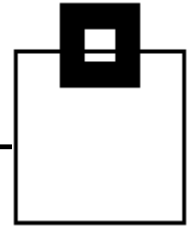
---

Row functions are brand-new, in fact there is only one at this time! Quite why it gets the plural is a puzzle!



# What's a BiF?

---



There are lots of examples of these in the DB2 Documentation and also in my newsletters, at least all new functions from DB2 V8 and up, so you can have a play in SPUFI to see what all these things can do.



So, now we know what a BiF is, Where's the BiF?

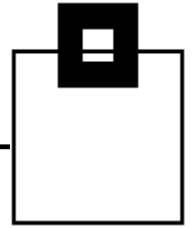
All these functions are very, very handy and useful tools that we use all the time, the problem is that the output of these BiFs sometimes changes with a DB2 release...





# Where's the BiF?

---



The one that started the ball rolling was the change of decimal data within a CHAR function.

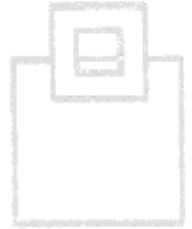
Your SQL looked like:

```
SELECT CHAR( DECCOL1 ) FROM SYSIBM.SYSDUMMY1;
```



You used to get ` 000.1` and suddenly you got `.1`

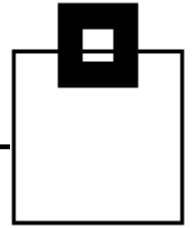
This caused a few German people to jump up and down so quick-as-a-flash IBM came up with the infamous BIF\_COMPATIBILITY flag in ZPARMs...



# Where's the BiF? BIF\_COMPATIBILITY

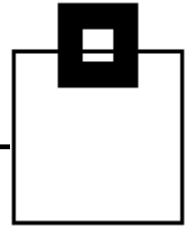
---

IBM introduced BIF\_COMPATIBILITY with values V9 for old format CHAR and CURRENT for new format.



## Where's the BiF? BIF\_COMPATIBILITY

---



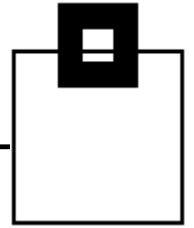
IBM introduced BIF\_COMPATIBILITY with values V9 for old format CHAR and CURRENT for new format.

Then they discovered VARCHAR...so BIF\_COMPATIBILITY got a new value: V9\_DECIMAL\_VARCHAR which included CHAR.



# Where's the BiF? BIF\_COMPATIBILITY

---



IBM introduced BIF\_COMPATIBILITY with values V9 for old format CHAR and CURRENT for new format.

Then they discovered VARCHAR...so BIF\_COMPATIBILITY got a new value: V9\_DECIMAL\_VARCHAR which included CHAR.

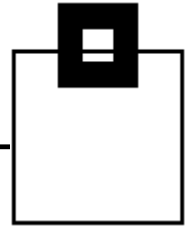


Then they discovered TIMESTAMPS...so now if BIF\_COMPATIBILITY was set to V9 or V9\_DECIMAL\_VARCHAR illegal TIMESTAMP formats were "accepted".



## Where's the BiF? BIF\_COMPATIBILITY

---



IBM introduced BIF\_COMPATIBILITY with values V9 for old format CHAR and CURRENT for new format.

Then they discovered VARCHAR...so BIF\_COMPATIBILITY got a new value: V9\_DECIMAL\_VARCHAR which included CHAR.



Then they discovered TIMESTAMPS...so now if BIF\_COMPATIBILITY was set to V9 or V9\_DECIMAL\_VARCHAR illegal TIMESTAMP formats were "accepted".



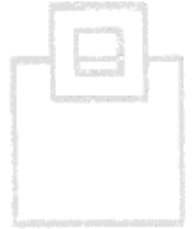
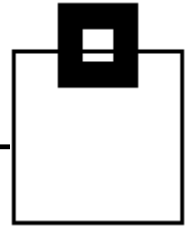
Then they discovered TRIM...so BIF\_COMPATIBILITY got a new value: V9\_TRIM for LTRIM, RTRIM, and STRIP which also included all of the above...



# Where's the BiF? Further Fixes

---

Two new SCHEMAS were then created to help sort out the mess: SYSCOMPAT\_V9 and SYSCURRENT (Just for CHAR, VARCHAR, and all the TRIM stuff. They must be added before SYSIBM in the path of course!)



# Where's the BiF? Further Fixes

---

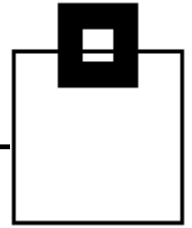
Two new SCHEMAS were then created to help sort out the mess: SYSCOMPAT\_V9 and SYSCURRENT (Just for CHAR, VARCHAR, and all the TRIM stuff. They must be added before SYSIBM in the path of course!)

DB2 11: APPLCOMPAT – New ZPARM introduced.

V10R1 after migration

V11R1 after install

If set to V10R1 then DB2 10 NFM behavior is guaranteed for the next two releases. So in DB2 13 it will all go horribly wrong...That's about seven years from now!



# Where's the BiF? Further Fixes

---

Two new SCHEMAS were then created to help sort out the mess: SYSCOMPAT\_V9 and SYSCURRENT (Just for CHAR, VARCHAR, and all the TRIM stuff. They must be added before SYSIBM in the path of course!)

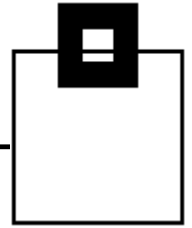
DB2 11: APPLCOMPAT – New ZPARM introduced.

V10R1 after migration

V11R1 after install

If set to V10R1 then DB2 10 NFM behavior is guaranteed for the next two releases. So in DB2 13 it will all go horribly wrong...That's about seven years from now!

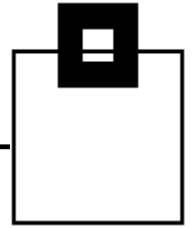
DB2 11: Two new BiF's - CHAR9 and VARCHAR9.





# Where's the BiF? Monitoring IFCIDs

---

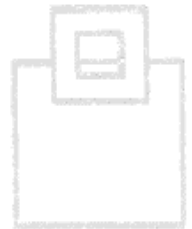


This was, to quote Monty Python, “Getting Silly”

So then the IFCID 366 was born. This IFCID is output whenever DB2 detects a *possible* change of behavior from the current release to the next release.



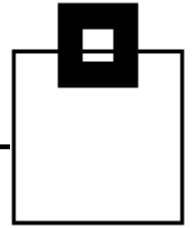
Within this IFCID are a bunch of fields (for the full details please look in your <db2hlq>.SDSNMACS(DSNDQW05) member) including a QW0366FN field that is a number (ICI – Incompatible Change Indicator, you just gotta love another TLA from IBM) that started at 1 and went to 3 with the initial version. A “1” meant a CHAR decimal problem, a “2” meant a VARCHAR decimal problem, and a “3” meant a TIMESTAMP problem.



## Where's the BiF? Monitoring IFCIDs

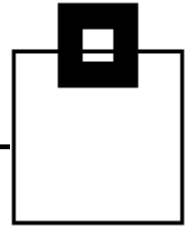
---

Now it has 10 numbers for DB2 10 and 17(!) for DB2 11. People also started getting lots of 366's and so the 376 was born, but only for DB2 11. This is an aggregated version of the 366.



# Where's the BiF? Monitoring IFCIDs

---



Now it has 10 numbers for DB2 10 and 17(!) for DB2 11. People also started getting lots of 366's and so the 376 was born, but only for DB2 11. This is an aggregated version of the 366.

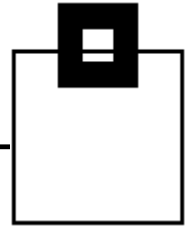


This was TMI (Too Much Information) as most shops were attempting to use SMF and filling up packs of disks with millions of records that contained Statement Ids of dynamic SQL that had long disappeared from the DSC.

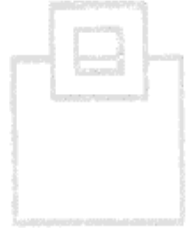


# Where's the BiF? Monitoring IFCIDs

---



Now it has 10 numbers for DB2 10 and 17(!) for DB2 11. People also started getting lots of 366's and so the 376 was born, but only for DB2 11. This is an aggregated version of the 366.



This was TMI (Too Much Information) as most shops were attempting to use SMF and filling up packs of disks with millions of records that contained Statement Ids of dynamic SQL that had long disappeared from the DSC.

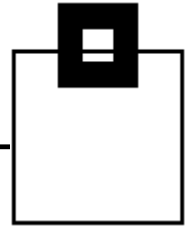


This is "sub-optimal"... (or, in English, a problem!)



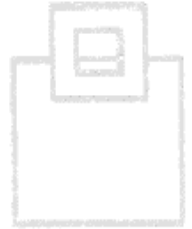
# Customer Example

---



Temporary solution for the problem until the affected applications are identified:

```
ZPARM BIF_COMPATIBILITY=V9_DECIMAL_VARCHAR
```



But this just “hides” the problem of course

Let’s have a quick look at a real customer and his problems...



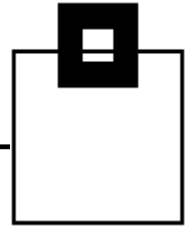
# Methodology at customer's site

---

- To get all relevant applications (daily, weekly etc.) since mid of April 2014 a performance trace with IFCID 366 was run in production

```
-STA TRACE (P) CLASS (32) IFCID (366) DEST (SMF) SCOPE (GROUP)
```

- For packages with static SQL additionally IFCID 63 was activated
- Identified were 18 programs:
  - 1 Cobol Program (Static SQL)
  - 1 C++ Program (Dynamic SQL) 3rd party
  - 4 Java Programs (Dynamic SQL)
  - 12 High Performance Unloads (Dynamic SQL)



# Evaluation of trace records

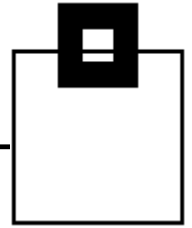
---

- Evaluation of SMF records with the help of **OMEGAMON XE FOR DB2 PERFORMANCE EXPERT**
- BMC Mainview also offers reporting capabilities described in the „Performance Reporter User Guide“

The BMC Datacollector has to be active.

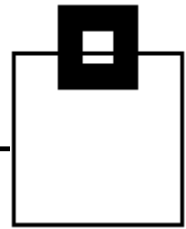
## Attention:

Depending on the frequency of program calls space requirements for the SMF records will increase drastically!

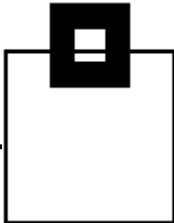


# JCL of the evaluation job (OMEGAMON)

```
//PEVS30 EXEC PGM=DB2PM
//STEPLIB DD DISP=SHR,DSN=#003,OMPE000,T0,AKANM00
// DD DISP=SHR,DSN=#003,OMPE000,SYS3,AKANM00U
//INPUTDD DD DISP=SHR,DSN=PS,SMPLOG,002,02019073
//JOBSUMDD DD SYSOUT=*
//RTTRCDD1 DD SYSOUT=*
//SYSIN DD *
GLOBAL
        TIMEZONE (-2)
        EXCLUDE (
                PRIMAUTH (
                        IS*)
        )
        INCLUDE (
                IFCID (366)
                GROUP (DBP0035)
        )
RECTRACE
        TRACE
        LEVEL (SHORT)
        DDNAME (RTTRCDD1)
EXEC
```







Top of Data

LOCATION: ~~XXXXXXXXXX~~ OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R3M0) PAGE: 1-1  
GROUP: ~~XXXXXXXXXX~~ RECORD TRACE - SHORT REQUESTED FROM: NOT SPECIFIED  
MEMBER: ~~XXXXXXXXXX~~ TO: NOT SPECIFIED  
SUBSYSTEM: ~~XXXXXXXXXX~~ ACTUAL FROM: 03/31/15 01:05:43.52  
DB2 VERSION: V10 PAGE DATE: 03/31/15

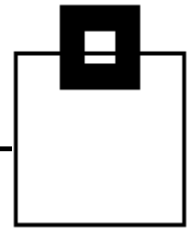
PRMAUTH	CONNECT	INSTANCE	END_USER	WS_NAME	TRANSACT	
ORIGAUTH	CORRNAME	CONNTYPE	RECORD TIME	DESTNO ACE IFC	DESCRIPTION	DATA
PLANNAME	CORRNMBR		TCB CPU TIME	ID		

CONTROL	DB2CALL	CEB99CE38ADB	CONTROL	DB2CALL		
CONTROL	<b>P12AC05</b>	DB2CALL	01:05:43.52254258	561390	1	366 INCOMPATIBLE FUNCTIONS
HPWL000	BLANK		N/P			

INCOMPATIBLE FUNCTIONS EXECUTED

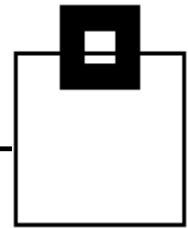
!COLLECTION ID : ~~XXXXXXXXXX~~  
!PROGRAM NAME : ~~XXXXXXXXXX~~  
!TYPE : 1 REASON : V9 SYSIBM.CHAR(DECIMAL-EXPR) FUNCTION  
!STMT NBR QUERY : 169 SECTION : 1 PLAN NAME QUERY: ~~XXXXXXXXXX~~  
!STMT ID : 4382822 STMT TYPE : DYNAMIC CONTOKEN (TS) : X'199387CC179A5487'  
!VERSION LENGTH : 7 VERSION : ~~XXXXXXXXXX~~

# SET CURRENT PATH = SYSCOMPAT\_V9 BIF\_COMPATIBILITY=V9\_DECIMAL\_VARCHAR



```
-- AUSZUG JOB PT2A205
SET CURRENT PATH = SYSCOMPAT_V9;
RESULT OF SQL STATEMENT:
DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
SET SUCCESSFUL
--
***INPUT STATEMENT:
SELECT SUBSTR (CHAR (R.RVTD_NR) , 2, 2) AS RVTD_NR
      , SUBSTR (CHAR (R.RENTE_VAID_SL), 2, 2) AS RENTE_VAID_SL
FROM DB2P.TCVR020_RENTE R
FETCH FIRST 2 ROWS ONLY WITH UR;
+-----+
! RVTD_NR ! RENTE_VAID_SL !
+-----+
1_! 70 ! 01 !
2_! 70 ! 01 !
+-----+
SUCCESSFUL RETRIEVAL OF 2 ROW(S)
```

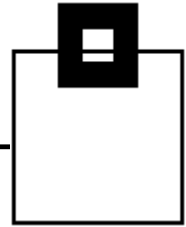
# SET CURRENT PATH = SYSCURRENT, SYSIBM BIF\_COMPATIBILITY=V9\_DECIMAL\_VARCHAR



```
-- AUSZUG JOB PFIW105
SET CURRENT PATH = SYSCURRENT, SYSIBM;
RESULT OF SQL STATEMENT:
DSNT400I  SQLCODE = 000,  SUCCESSFUL EXECUTION
SET      SUCCESSFUL
--
***INPUT STATEMENT:
SELECT  SUBSTR (CHAR (R.RVTR_NA)           , 2, 2)  AS RVTR_NA
        , SUBSTR (CHAR (R.RENTE_VAID_SL), 2, 2)  AS RENTE_VAID_SL
FROM    QDQP.TKVR030_RENTE      R
FETCH FIRST 2 ROWS ONLY WITH UR;
+-----+
! RVTR_NA ! RENTE_VAID_SL !
+-----+
1_!  0  !      !
2_!  0  !      !
+-----+
SUCCESSFUL RETRIEVAL OF          2 ROW(S)
```

# Modified SQL Statement

## BIF\_COMPATIBILITY=V9\_DECIMAL\_VARCHAR



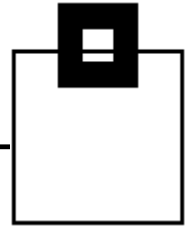
```
--#SET MAXERRORS 0
-- AUSZUG JOB #724328
SET CURRENT PATH = SYSCURRENT,SYSIBM;
SET SUCCESSFUL
--
***INPUT STATEMENT:
SELECT LPAD(R.RYTB_NA, 2, '0') AS RYTB_NA
,LPAD(R.RENTE_VAID_SL, 2, '0') AS RENTE_VAID_SL
FROM DBSP.TKNAD20_RENTE R
FETCH FIRST 2 ROWS ONLY WITH UR;
+-----+
! RYTB_NA ! RENTE_VAID_SL !
+-----+
1_ 70      ! 01      !
2_ 70      ! 01      !
+-----+
```

SUCCESSFUL RETRIEVAL OF 2 ROW(S)



# SQL WorkloadExpert (WLX) for DB2 z/OS

---



- The SQL WorkloadExpert architecture comes in two parts:
  - Mainframe DB2 Workload collector and processing engine
  - Workstation analysis and reporting engine
- DSC and SSC were selected and the data is stored in WLX tables
- Processing and display of the data in Datastudio 4.1.1 or Eclipse on the workstation



# Evaluation with SQL WorkloadExpert (1)

The screenshot displays the SQL WorkloadExpert application window. The title bar shows three tabs: 'Eigenschaften', 'SQL-Ergebnisse', and 'SQL WorkloadExpert'. The main interface is divided into a left-hand navigation pane and a right-hand content area. The navigation pane lists several analysis categories, with 'BiF Nutzung' (Build-in-Funktion Nutzungsanalyse) highlighted by a red box. The content area on the right shows a dropdown menu with 'ANWENDUNG\_ERGEBNISSE' selected, also highlighted by a red box. The overall interface is in German.

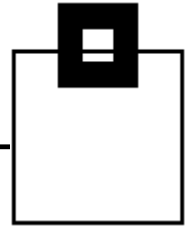
- Eigenschaften
- SQL-Ergebnisse
- SQL WorkloadExpert

Navigation pane items:

- Anwendungs-Workload  
Detaillierte Analyse des Anwendungs-Workload
- BiF Nutzung**  
Build-in-Funktion Nutzungsanalyse
- CPU intensive SQLs  
CPU intensive SQL Statements
- DSC Flush Raten  
Berechnung der DSC Flush Raten.
- Objekt Ruhezeiten  
Objekt Ruhezeiten
- Skalierung von Workloads  
Auf- und Abskalierung von Workloads.
- SELECT ONLY Ermittlung  
Finde heraus welche Tabellen nur von SELECT SQL Statements benutzt werden.
- SQL gleich, mehr. Qual.  
Gleiche SQL Statements, die sich nur in den Qualifizierungen (Schema) der Objekte unterscheiden.
- Verzögerungsermittlung  
Finde heraus welche SQLs ungewöhnlich hohe Verzögerungen haben und finde verwandte SQL Statements um dies zu verhindern.
- WLX KPIs und Summaries  
WLX Leistungskennzahlen (Key Performance Indicators) und Zusammenfassungen

Content area dropdown: ANWENDUNG\_ERGEBNISSE

# Evaluation with SQL WorkloadExpert (2)



BIF Nutzung

Beschreibung: BIF Test1

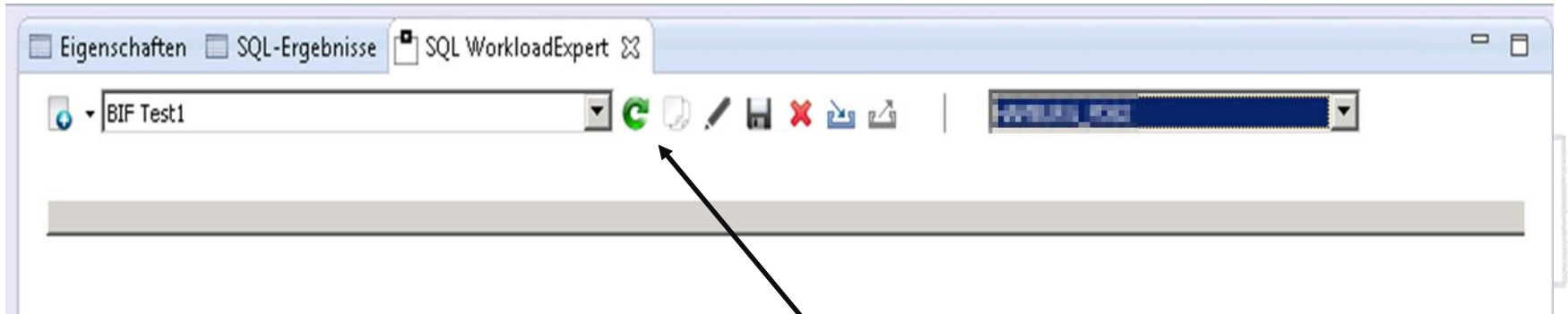
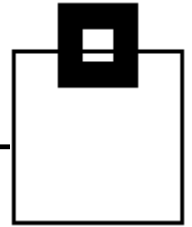
Projektion | Selektion | Sortierung

Bezeichnung	Beschreibung
WLX Key	Der WorkloadE...
Collection ID	Die Collection I...
Package	Das Package de...
Anzahl	Anzahl

Bezeichnung	Operator	Wert	Beschreibung
WLX Key	=	neuester	Der WorkloadExpert Key für diesen Wo...

OK Abbrechen

# Evaluation with SQL WorkloadExpert (3)

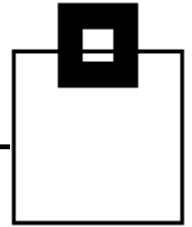


Run query





# Evaluation with SQL WorkloadExpert (4)

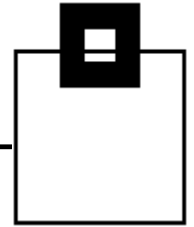


The screenshot shows the SQL WorkloadExpert application window. The title bar reads 'Eigenschaften' and 'SQL WorkloadExpert BIF Test1'. The main area displays a table with the following data:

WLX Key	Collection ID	Package	Anzahl
2015-05-05-10.39.37.293082	NULLID	SYSLN200	77
2015-05-05-10.39.37.293082	IQA_COLLECTIO...	IQADB...	8
2015-05-05-10.39.37.293082	DSNTEP4	DSNŞEP4L	1



# Evaluation with SQL WorkloadExpert (5)



The screenshot shows the SQL WorkloadExpert interface. The window title is "SQL WorkloadExpert BIF Test1". The main area displays a table with the following data:

WLX Key	Collection ID	Package	Anzahl
2015-05-05-10.39.37.293082	NULLID	SYSLN200	77
2015-05-05-10.39.37.293082	IQA_COLLECTIO...	IQADB...	8
2015-05-05-10.39.37.293082	DSNTEP4	DSNSEP4L	1

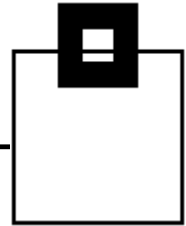
A context menu is open over the third row, with a blue arrow pointing to the "Report" option. The menu items are:

- Report
- Drill down anzeigen
- Compare View anzeigen

Select



# Evaluation with SQL WorkloadExpert (6)

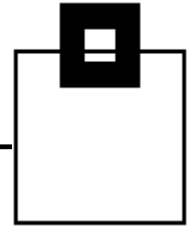


The screenshot shows the SQL WorkloadExpert interface. The window title is "SQL WorkloadExpert neueAbfrage1". The main area displays a table with the following data:

WLX Key	Collection ID	Package	ICI-Nummer	Anzahl	Grund
2015-05-05-10.39.37.293082	DSNTEP4	DSN\$EP4L		1	DB2 9 CHAR benutzt



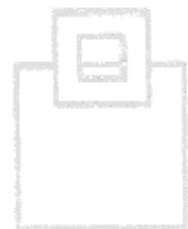
# Evaluation with SQL WorkloadExpert (7)



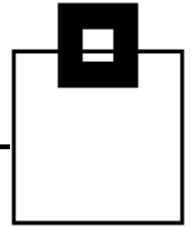
The screenshot shows the SQL WorkloadExpert interface. The table below is selected, and a context menu is open over it. A blue arrow labeled "Select" points from the text to the context menu.

WLX Key	Collection ID	Package	ICI-Nummer	Anzahl	Grund
2015-05-05-10.39.37.293082	DSNTEP4	DSN\$EP4L		1	DB2 9 CHAR benutzt

- Report
- Drill down anzeigen
- Compare View anzeigen



# Evaluation with SQL WorkloadExpert (8)

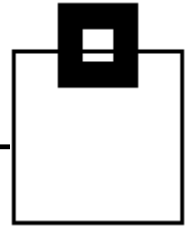


The screenshot shows the SQL WorkloadExpert application window. The title bar contains several tabs: 'Eigenschaften', 'SQL-Ergebnisse', 'SQL WorkloadExpert BIF Test1', 'SQL WorkloadExpert neueAbfra...', and 'SQL WorkloadExpert neueAbfra...'. The main area displays a table with the following data:

WLX Key	Collection ID	Package	ICI-Nummer	Grund	WLX DB2 SSID	Statement ID	Statement
2015-05-05-10.39.37.293082	DSNTEP4	DSNSEP4L	1	DB2 9 CHAR benutzt	DB2	1.462.008	



# Evaluation with SQL WorkloadExpert (9)



The screenshot shows the SQL WorkloadExpert application interface. At the top, there are several tabs: 'Eigenschaften', 'SQL-Ergebnisse', 'SQL WorkloadExpert BIF Test1', 'SQL WorkloadExpert neueAbfra...', and 'SQL WorkloadExpert neueAbfra...'. Below the tabs is a toolbar with various icons. The main area displays a table with the following data:

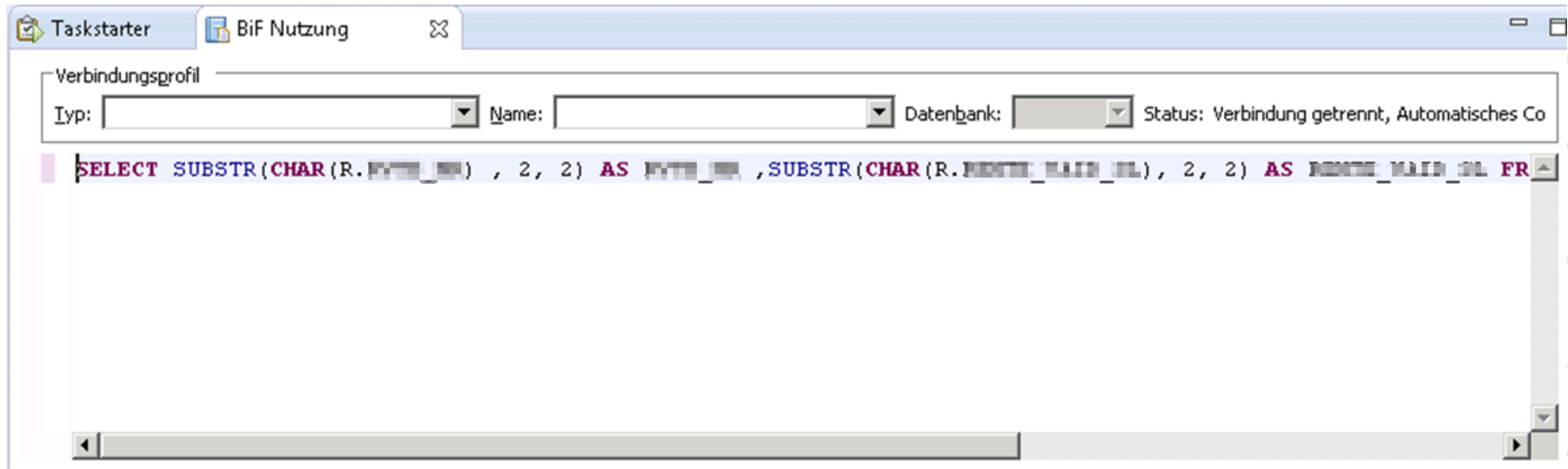
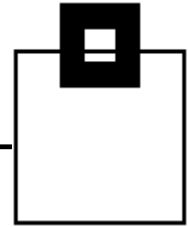
WLX Key	Collection ID	Package	ICI-Nummer	Grund	WLX DB2 SSID	Statement ID	Statement
2015-05-05-10.39.37.293082	DSNTEP4	DSNSEP4L	1	DB2 9 CHAR benutzt		1.462.008	

A context menu is open over the first row of the table, with the word 'Select' and a blue arrow pointing to the menu. The menu options are:

- Report
- SQL Anweisung in Zwischenablage kopieren
- Compare View anzeigen
- SQL Text anzeigen



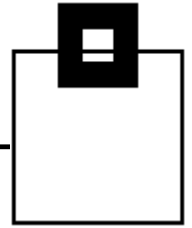
# Evaluation with SQL WorkloadExpert (10)



```
SELECT SUBSTR(CHAR(R.RYTB_NR) , 2, 2) AS RYTB_NR
      , SUBSTR(CHAR(R.RENTE_VAID_SL) , 2, 2) AS RENTE_VAID_SL
FROM DB2P.TKVR020_RENTE R
FETCH FIRST 2 ROWS ONLY WITH UR;
```

# Customers Goal

---



To already redress in DB2 10 all incompatibilities before the migration to DB2 11 and to set the ZPARM **BIF\_COMPATIBILITY**, still in DB2 10, to be **CURRENT**.

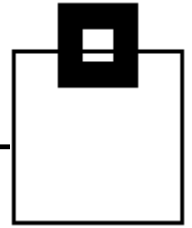




# What we have learnt

---

- In contrast to the previously performed examinations using Omegamon, with WLX you need not collect SMF data for more than 12 months before the migration and analyze them.
- With WLX you do not even need SMF data. The WLX straight-forward approach saves enormous time, manpower, and storage.
- “Where is the BiF?” Is answered directly and the SQL is categorized and shown – Even for dynamic SQL you get the entire SQL text displayed.

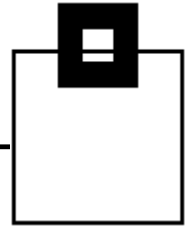


# Where's the BiF?

---

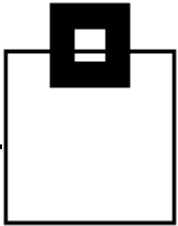
As you have seen BiFs can cause you problems if you are **\*not\*** aware of their usage:

- Start a project plan **\*now\*** for finding the BiF
- Action and Approve any and all SQL Changes
- Store all results for comparison next time...
- Stay aware and ready!



# Where's the BiF?

You do not want your next migration looking like this:



# Where's the BiF?

