White Paper

DB2 Workload Auditing

Simple. Affordable. Reliable.

Siegfried Fürst - SOFTWARE ENGINEERING GmbH

Introduction

Much of the news in 2014 was fraught with tales of cyber-attacks, leaks, data privacy breaches, and employee data theft. Within the first five days of 2015, news of personal data from customers of a large financial institution being posted online made the rounds.

Whether the results are masses of personal data being exposed, or the news that an executive doesn't approve of a certain celebrity, no company wants to see their name associated with this type of publicity.

Companies of all sizes are tasked with protecting against threats from modern day thieves and security is only as strong as the weakest link.

But what about the unintended consequences from simple "mistakes"? How can you find irregularities if you can't see them? How do you know what that ERP package is really doing? Why does your external team block your whole system every day at a certain time?

Background

Most large mainframe sites these days use DB2 z/OS to store their critical data in tables. Data may also be stored in a distributed environment that interacts with the mainframe via various middleware. How is it possible to capture those interactions and show all SQL access to the DB2 z/OS tables? How is it possible to capture those interactions with virtually no overhead?

Many companies use the DB2 Audit Trace to capture and review SQL accesses. However, in order to capture all SQL access to DB2 objects, it must run constantly. The overhead cost to run this can be astronomical and, even then, certain information may not be recorded:

- Trace records don't show input variables.
- Trace records only show the first time an SQL statement was executed.

There are tools available on the market today to log access information in separate DB2 tables, which can then be evaluated and reported on. However, they use hooks into DB2 and may, in and of themselves, inadvertently pose a risk for system availability. This type of approach could potentially cause issues in IBM-related support.

Other down-sides are:

- Incomplete data
- Delayed turn-around on key performance metrics
- Inability to meet auditing requirements

SEGUS Inc has come up with a better solution.

The Solution

We have developed an SQL Workload Warehouse called SQL WorkloadExpert™ for DB2 z/OS, (or "WLX") to capture virtually all[1] SQL access to DB2 z/OS tables via both static and dynamic SQL.

The data is stored in DB2 tables and is immediately available for querying via a GUI interface or even Mainframe SPUFI.

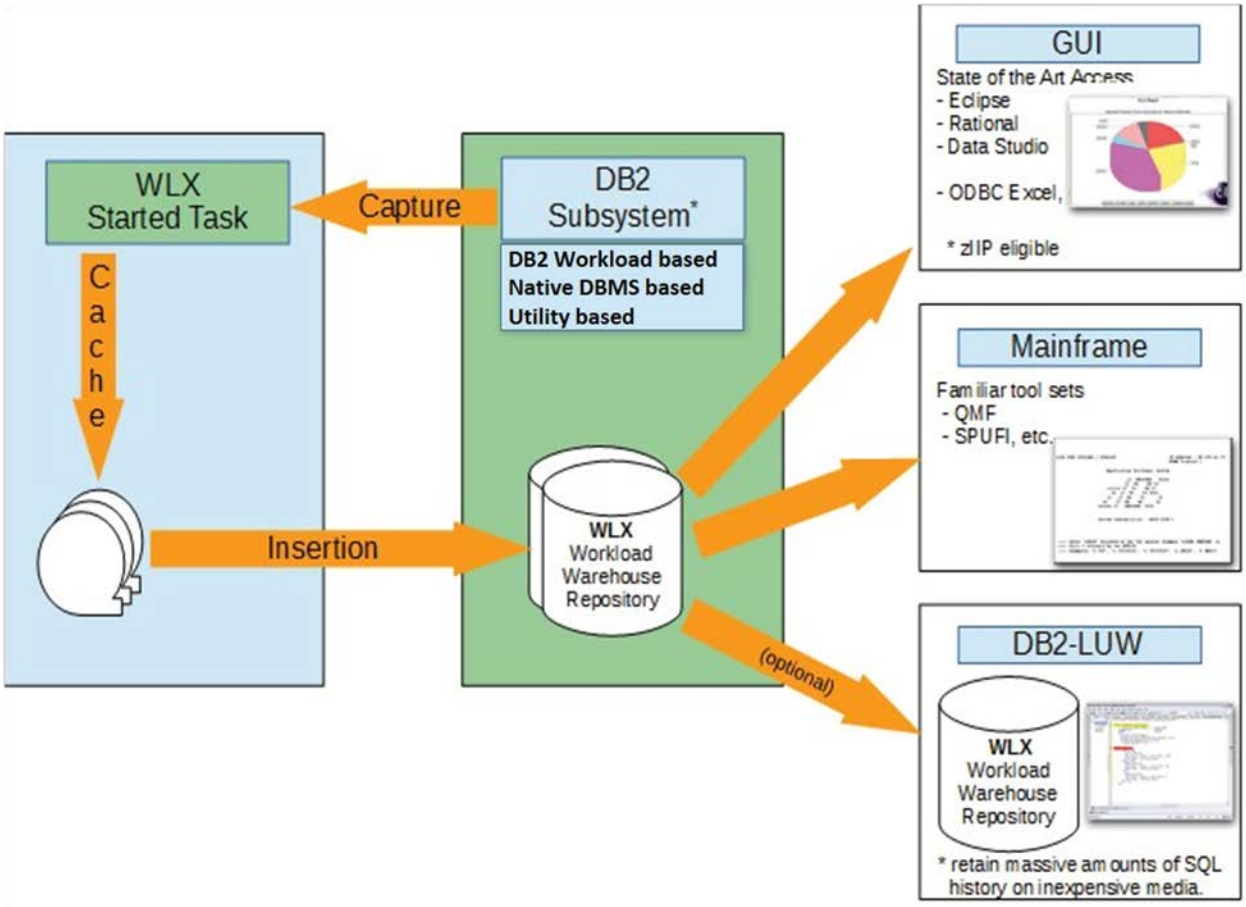


Figure 1

Of course you could also develop a similar solution in-house, but do you really have the staff necessary to maintain a solution like this? And make changes every time DB2 is updated? (How about ten years from now?)

1 Some exclusions apply: WLX is available for DB2 10 NFM and above. The collection of Static SQL data requires a BIND or REBIND in at least DB2 10 NF. As WLX exploits the DB2 DSC and EDM Pool, certain (uncommon) DB2 restrictions may affect the results, e.g. if an access path is PRUNED, the data is removed by the DB2 Optimizer and the object usage counter is not obtained.

So now that you can inexpensively collect all of this data, how can you quickly make sense of it and start looking for anomalies?

WLX is a complete tool that can be used to capture SQL within the enterprise. To make the product more utilitarian, we have designed it with a series of discrete "Use Cases". The Audit group very quickly shows who did what, when and where.

- Who is updating the payroll tables?
- Who is accessing any table from outside the data center?
- Is anyone using ODBC to select from production tables?
- Where did certain updates come from?
- How many user ids have accessed the DB2 systems?
- Has any SYSADM-enabled user id done anything today?
- Show me the SQL text for a certain SQL

What is collected?

1. Virtually all SQL executed in the Plex
2. The time and date of execution
3. Relevant information regarding the origination
4. Various cost factors

Examples

Example 1: Who is updating the Payroll tables?

Statistics Updated	Transaction name ...	End User ID ...	Workstation name ...	Qualifier	Package	Query type	Table creator	Table name	Object type
-09.43.47.088663	WLXNEWWL ...	H	DB2CALL	H	IQADBACP	INSERT	IQA0610	IQATW001	T
-09.43.35.842770	WLXNEWWL ...	H	DB2CALL	H	IQADBACP	UPDATE	IQA0610	IQATW001	T
-09.43.48.599040	WLXNEWWL ...	H	DB2CALL	IQAEXP09	IQADBACP	INSERT	IQA0610	IQATW001	T
-17.00.30.257119	WLX1QA1B ...	H	DB2CALL	H	IQADBACP	INSERT	IQA0610	IQATW001	T
-17.00.31.610052	WLX1QA1B ...	H	DB2CALL	H	IQADBACP	UPDATE	IQA0610	IQATW001	T
-09.53.14.119720	WLX1QA1B ...	H	DB2CALL	IQAEXP09	IQADBACP	INSERT	IQA0610	IQATW001	T
-10.01.01.126154	WLX1QA1B ...	H	DB2CALL	IQAEXP09	IQADBACP	INSERT	IQA0610	IQATW001	T

Figure 2

Figure 2 above shows who was INSERTing to a specific table (Object type "T" on the far right.)

Examples 2 & 3: Who is accessing any Table from outside the data center? And is anyone using ODBC to select from production tables?

SQL WorkloadExpert Audit

Audit | ZD00QA1B

Transaction name	End User ID	Workstation name	Primary Authorization ID	Package	Collection ID	Min. INSERT timestamp	Max. UPDATE timestamp	Number of Statements
db2jcc_application	boxwell	w213a07	BOXWELL	n/a	n/a	2014-12-05-12.01.22.778020	2014-12-05-12.04.50.349654	29
db2jcc_application		192.168.222.105		n/a	n/a	2014-12-03-11.15.14.310453	2014-12-09-15.17.36.337347	27
db2jcc_application		192.168.222.104		n/a	n/a	2014-12-03-11.45.46.614905	2014-12-09-17.00.25.651804	150
db2jcc_application		192.168.1.241		n/a	n/a	2014-12-04-15.54.13.438945	2014-12-04-16.08.40.639459	25
db2jcc_application		192.168.1.244		n/a	n/a	2014-12-04-19.14.01.543110	2014-12-04-21.47.36.829843	12
db2jcc_application		192.168.1.246		n/a	n/a	2014-12-04-03.09.51.148869	2014-12-04-22.27.29.762663	30
AUDITRPT		BATCH		n/a	n/a	2014-12-04-16.15.58.137378	2014-12-04-16.56.49.752497	9
BOXWELL	BOXWELL	DB2CALL	BOXWELL	n/a	n/a	2014-12-03-09.45.19.878414	2014-12-05-13.04.22.359929	92
BOXWELL	BOXWELL	TSO	BOXWELL	n/a	n/a	2014-12-03-08.35.33.555289	2014-12-04-10.52.30.475696	17
		DB2CALL		n/a	n/a	2014-12-04-09.52.29.962204	2014-12-09-15.01.56.697489	179
		DB2CALL		n/a	n/a	2014-12-03-11.59.41.995957	2014-12-09-16.19.43.373154	9
EXCEL.EXE		N172A08		n/a	n/a	2014-12-04-17.09.20.560220	2014-12-08-15.46.42.764461	3

Figure 3

The Workstation name, in Figure 3 above, shows you the Internet or the Intranet address thus enabling you to very quickly see any external addresses.

Further, the Transaction Name shows, in this case, EXCEL.EXE so someone was using an ODBC connection from a PC up to the host.

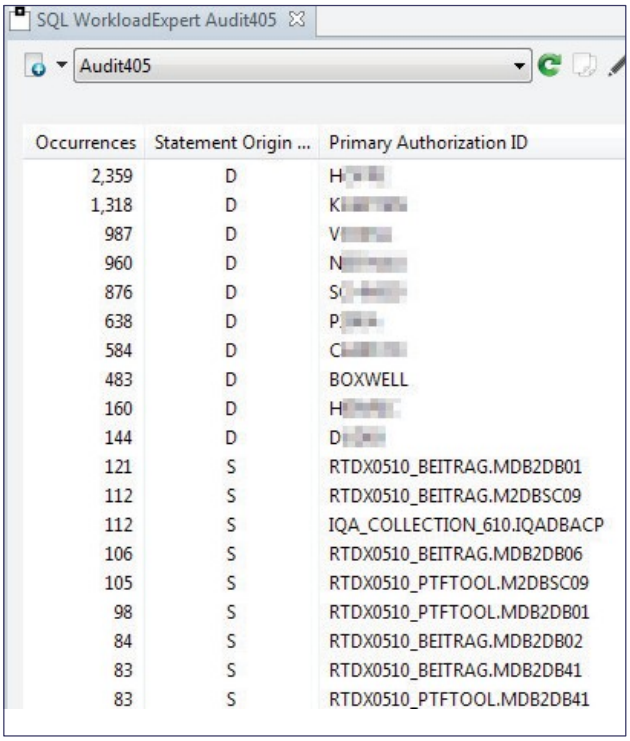
Example 4: Where did certain updates come from?

Primary Authorization ID	Qualifier	Package	Query type	Table creator	Table name	Object type
H	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	USERTAB	T
S	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	USERTAB	T
S	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	PTFTIN01	I
P	P	IQADBACP	INSERT	IQA0610	IQATY007	T
S	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	USERTAB	T
S	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	PTFTIN01	I
S	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	USERTAB	T
S	PTFADMIN	DSNREXX	UPDATE	PTFADMIN	PTFTIN01	I
S	PTFADMIN	DSNREXX	INSERT	PTFADMIN	PTF	T

Figure 4

The "Query type" column shows whether a Table ("T" in Object type column), or Index ("I" in Object type column), was UPDATEd.

Example 5: How many user ids have accessed the DB2 systems?



Occurrences	Statement Origin ...	Primary Authorization ID
2,359	D	H
1,318	D	K
987	D	V
960	D	N
876	D	S
638	D	P
584	D	C
483	D	BOXWELL
160	D	H
144	D	D
121	S	RTDX0510_BEITRAG.MDB2DB01
112	S	RTDX0510_BEITRAG.M2DBSC09
112	S	IQA_COLLECTION_610.IQADBACP
106	S	RTDX0510_BEITRAG.MDB2DB06
105	S	RTDX0510_PTFTOOL.M2DBSC09
98	S	RTDX0510_PTFTOOL.MDB2DB01
84	S	RTDX0510_BEITRAG.MDB2DB02
83	S	RTDX0510_BEITRAG.MDB2DB41
83	S	RTDX0510_PTFTOOL.MDB2DB41

Figure 5

Figure 5 displays the "real" users (Primary Authorization ID), but also shows that static queries were executed. "D" denotes access was via Dynamic SQL and "S" denotes Static.

Example 6: Has any SYSADM-enabled user id done anything today?

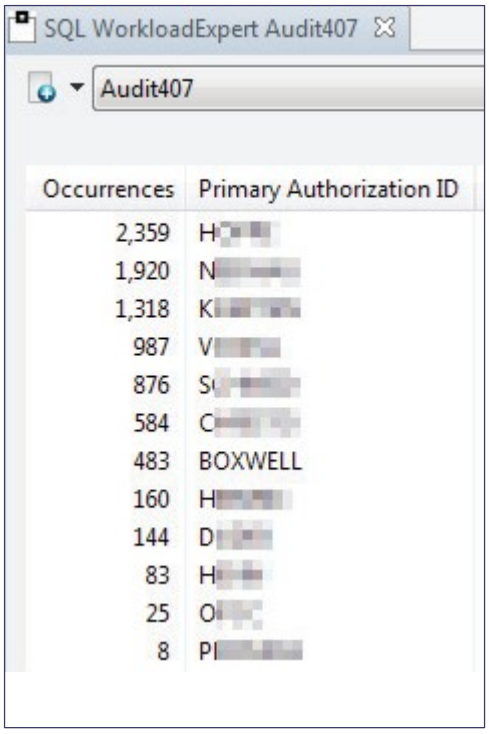


Figure 6

WLX

- Catch all SQL running on a typical system, without the overhead of a trace or utilizing monitors.
- Analyze SQL Workload by aggregate, or in detail.
- Comparatively analyze access paths and SQL.
- Report and chart metrics via a standard Eclipse interface—a free and commonly-accepted desktop tool.

Figure 6 above shows all the user ids with SYSADM privilege and how many SQLs that Userid performed.

Example 7: Show me the SQL text for a certain SQL

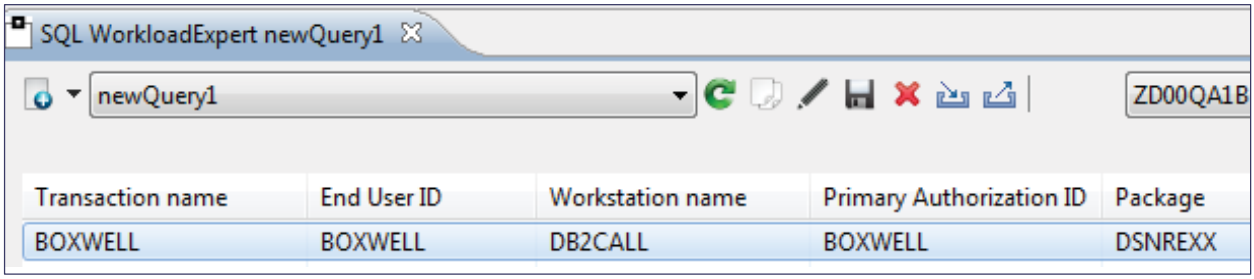


Figure 7

Using filters, it is easy to find, for example, that a certain user ran a REXX program.

By double-clicking on the appropriate name in the End User ID column…

... The actual SQL text is then shown

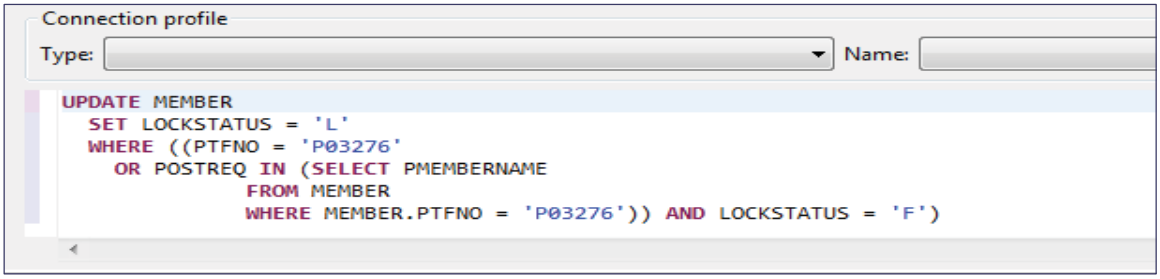


Figure 8

Conclusion

With data security at the forefront of everybody's minds, companies are doing their utmost to protect their sensitive data from threats of all types. Inadvertent changes and simple "mistakes" due to lack of transitory data can also cause significant problems. The truth is: unless the cause of a problem is known, it can't be rectified.

WLX easily and inexpensively leverages data, that DB2 10 NFM (or higher) makes available, in order to create a Workload Warehouse. It provides a solid basis for quick problem resolution and displays information that may otherwise be extremely difficult—or expensive—to obtain.

The Workload Warehouse can be simply ported down to an LUW DB2 server, where it is less expensive to store, to provide a full record of all details over the years.

Benefits

- Complete SQL chronicle for the enterprise
- Low to no overhead
- Appeals to a wide audience with various skill levels
- A must-have tool for anyone involved with SQL
- Quick to learn and easy to use.

Requirements

- For Dynamic SQL DB2 10 NFM or higher.
- For Static SQL the package must have been rebound in DB2 10 NFM or higher.
- The GUI requires Eclipse Indigo or Data Studio 4.1 as a minimum.

SEGUS Inc
14151 Park Meadow Dr
Chantilly, VA 20151
USA

(800) 327-9650
www.segus.com
info@segus.com

SOFTWARE ENGINEERING GMBH
Heinrichstraße 83-85
40239 Düsseldorf
Germany

+49-211-961-49-0
www.seg.de
info@seg.de