Total Environment Simulation

Workload Replay in an agile world, using

ContinuousDelivery DeploymentCheck for Db2 z/OS

Roy Boxwell

SOFTWARE ENGINEERING

Session code: [V4]

Tuesday, November 6th, 2018 - 11:30-12:30 AM

Db2 z/OS

Agenda

- Testing, virtualizing and simulating – the aspects of reliable quality assurance
- Db2 database/object cloning – what's state of the art and what's beyond
- XML commander – the comprehensive automation of flexibility, covering
  - FTP/routing
  - JCL
  - ISPF file tailoring, panels, messages
  - Db2 commands
- Different flavors of (pro-active) testing and how it can be automated:
  - Anomaly alerting based on Incompatibility Change Indicators (ICIs)
  - Dynamic/static access path change detection e.g. Plan Management
  - Cloning exploiting Backup System
  - Workload-KPI verification using SQL replay and KPI comparison
- Real world experience highlighting the benefits of automated testing

The aspects of reliable quality assurance

Agile development requires near-time delivery

- Continuous Delivery (CD) is an approach to produce software in short cycles
- CD ensures that changes can be released at any time, considering building, testing and releasing faster and more frequently
- Key is a focus on more incremental updates
- CD requires a straightforward and repeatable deployment

The aspects of reliable quality assurance

- Regression testing is the critical piece to keep up production
  - Include critical as well as custom processes
  - Scale up testing to cover workloads peaks
  - Build a (performance) baseline for comparison and trending
  - Only move on after tests completed to your satisfaction
  - Postpone maintenance as opposed to forcing in
  - Practice fallback scenarios for individual Applications as well as the system
  - Make fallback strategies part of the (pre-production) testing and practice them
- Always assure yourself a back door or Plan B
- Minimize change and use of new function

The aspects of reliable quality assurance

ContinuousDelivery DeploymentCheck for Db2 z/OS is able to:

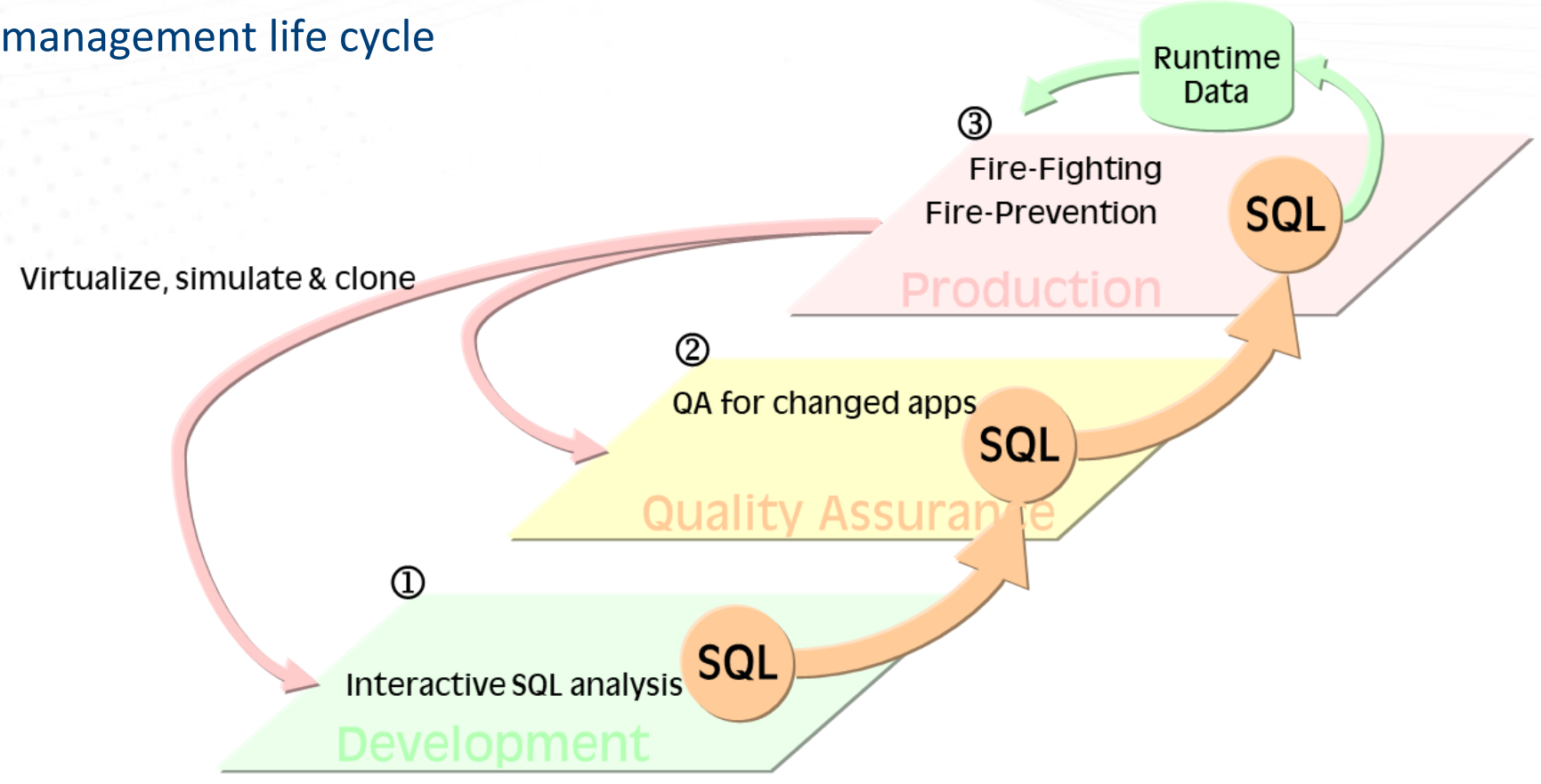
- Test changes (application, statistics, Db2 code, hardware, …),
  or a combination of those flexibly and reliably
- Speed up and fully automate testing to keep up with code drops being continuously released
- Keep sets of representative workload to easily and reliably detect anomalies
- Its cloning component speeds up and fully automates testing for an entire environment
- Its capturing component covers Workload/Replay to include and automate application workload tests

The aspects of reliable quality

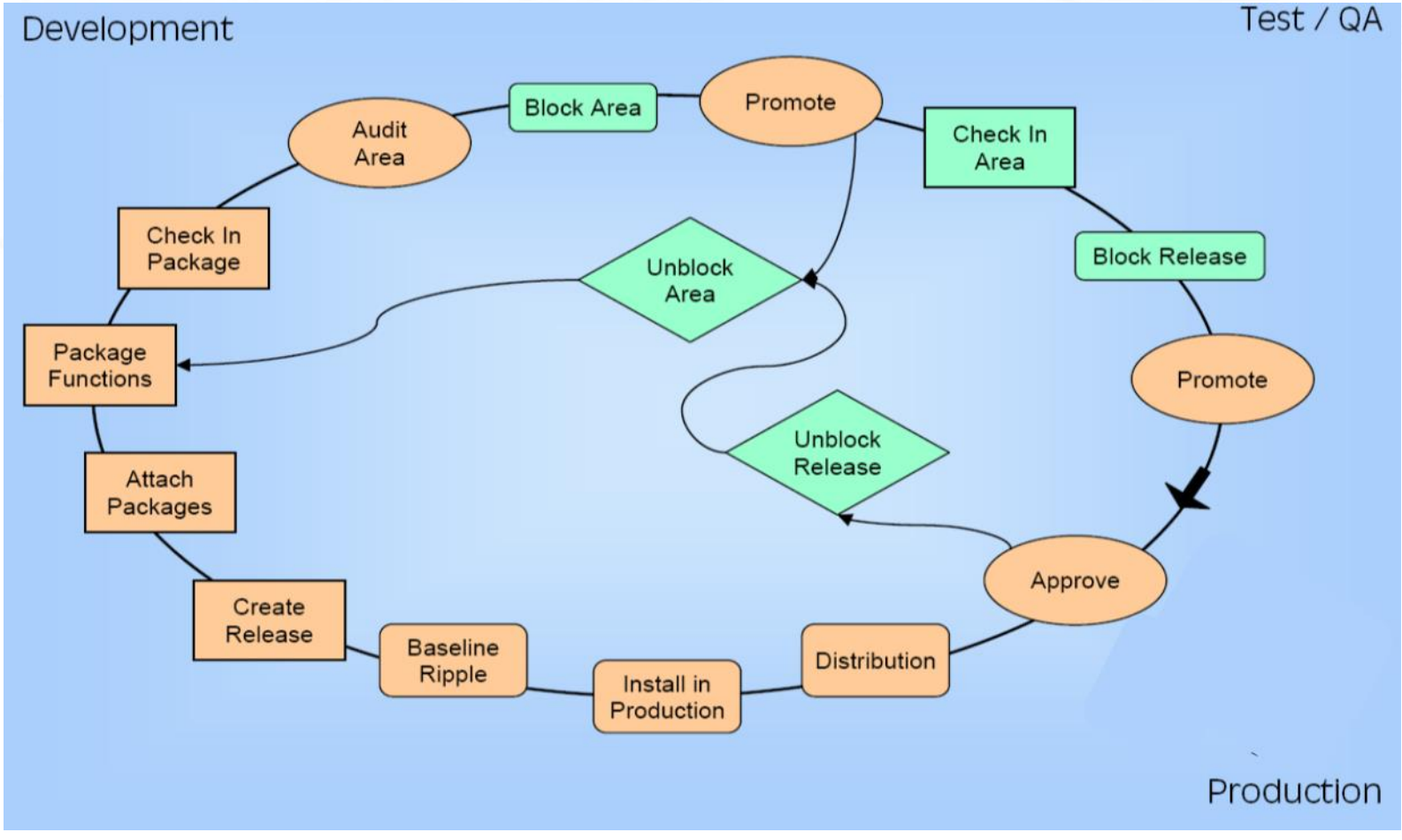
- Capture *all* Dynamic and Static SQL running in the Plex
- Use production backups, DDL, statistics and hardware metrics to build simulation environments instantaneously
  - Propagate production DDL and statistics down the environment chain (Prod -> QA, Prod -> Dev)
  - Simulate productions hardware, ZPARMS, and BUFFERPOOLS in QA/Dev
  - Comparing workload metrics is todays approach to keep up with ongoing changes, huge amounts of workload and less staff
  - Tune any new SQL, or SQL that got worse
    - Pick the „low hanging fruit“
  - Stage level by level (Dev -> QA, QA -> Prod)

The aspects of reliable quality assurance

SQL management life cycle



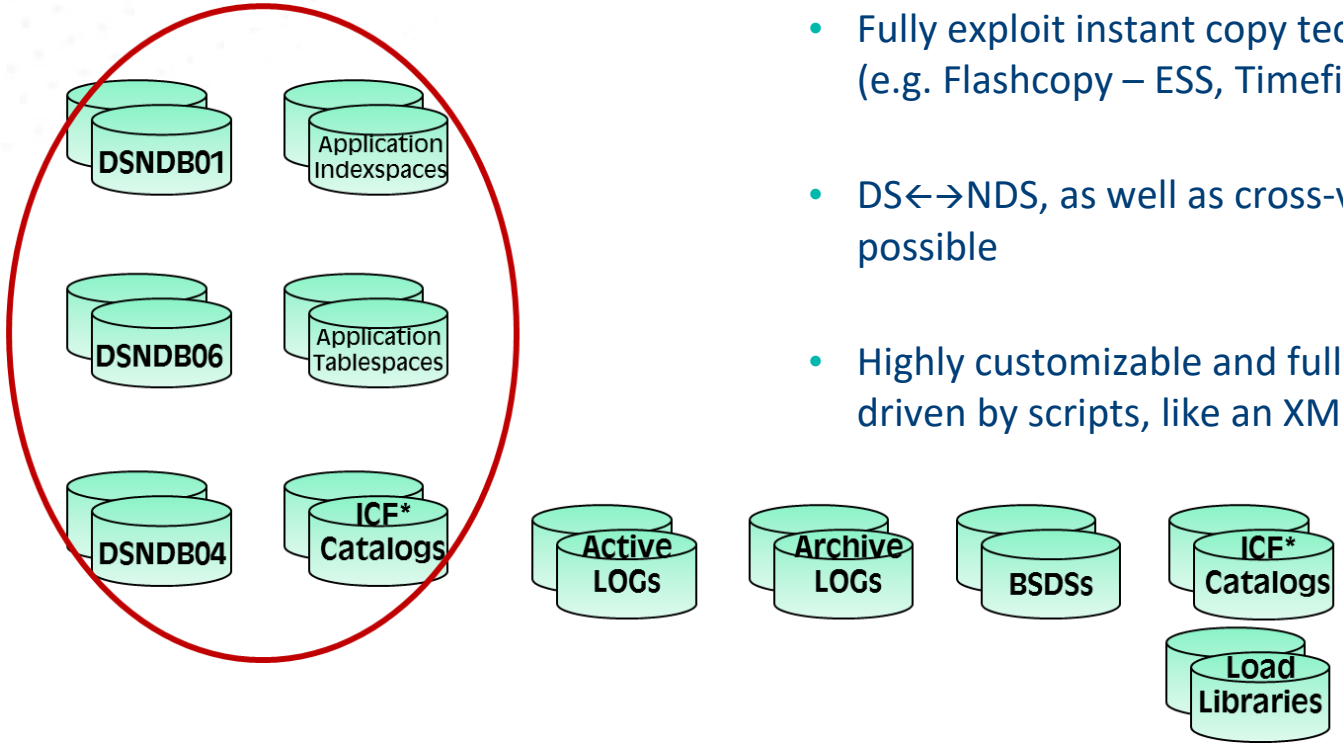
The aspects of reliable quality assurance



Db2 database/object cloning

Instant Cloning for clone based code level checks:

Scope of Cloning: Subsystem level



- Fully exploit instant copy technology (e.g. Flashcopy – ESS, Timefinder, Snapshot)
- DS←→NDS, as well as cross-version cloning possible
- Highly customizable and fully automated if driven by scripts, like an XML scenario scheme

Db2 database/object cloning

- XML controlled cloning
  - Due to its nature XML is a fully flexible, human- and machine-readable language
  - It may, or may not have elements and/or attributes, but has to be well-formed only
  - Since a complex cloning procedure may, or may not have individual steps, including some very customer specific tasks, XML is a perfect choice to drive a cloning scenario

```
        <description>Check installation specific datasets</description>
    </menuitem>
  - <menuitem>
        <name>Gather information</name>
        <description>Get all needed information</description>
    </menuitem>
  - <menuitem>
        <name>Stop DB2</name>
        <description>Stop target DB2</description>
    </menuitem>
  - <menuitem>
        <name>Restore</name>
        <description>Restore volumes</description>
    </menuitem>
```

Db2 database/object cloning

- FTP and Routing
  - Usually source and target subsystem reside on different LPARs, machines, or even locations
  - A fundamental part of cloning automation is to take care of routing and transferring required data, no matter if being logged in on source, or target
    - Temporary/Workfile datasets are shared via FTP
    - Commands are routed by adding the system name
      `ROUTE LPRS,/F DB2S,STOP DB2`
    - Jobs can be route either via
      `ROUTE XEQ LPRSNJE`, or `SCHENV=LPRSDB2S`

Automated testing – anomaly allerting

- In recent versions, IBM has modified the behavior of certain Db2 functionality
  - Built-in Functions (BiFs)
  - Reserved Words
  - SQL Return Codes
  - Deprecated Functionality

- It is important that you are aware of, and track, the incompatibilities that may cause issues/problems… why?
  - Applications no longer function
  - Applications function differently
  - The results of your SQL SELECT statements can change

Automated testing – anomaly allerting

ICI = Incompatibility Change Indicator

- Db2 can track when your applications use incompatible functionality
  - If you start the right traces/IFCIDs

Automated testing – anomaly allerting

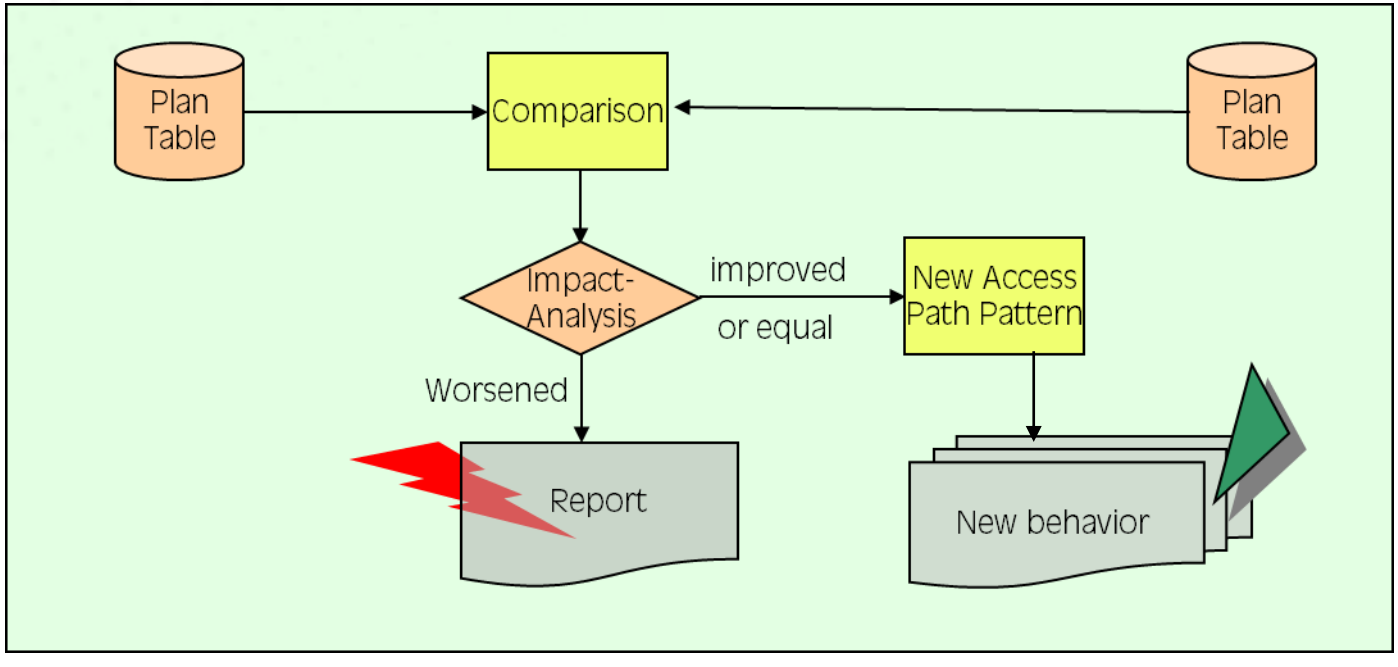
- 1: V9 version of CHAR(DEC) executed
- 2: V9 version of VARCHAR(DEC) executed
- 3: Unsupported character string representation of a TIMESTAMP
- 4: V10 default SQL path used instead of V11
- 7: SQLCODE -301 from a DB2 11 server
- 8: Stored procedure data types
- 9: TIMESTAMP TIMEZONE from DRDA
- 10: V9 version of LTRIM, RTRIM, STRIP executed
- 1101: INSERT into XML column w/o XMLDOCUMENT function
- 1102: XPATH evaluation resulted in error
- 1103: Dynamic SQL ASUTIME limit RLF issue
- 1104: CLIENT_ACCTNG longer than supported length pre-V11
- 1105: CLIENT_APPLNAME longer than supported length pre-V11
- 1106: CLIENT_USERID longer than supported length pre-V11
- 1107: CLIENT_WORKSTNNAME longer than supported length pre-V11
- 1108: CLIENT register longer than supported used for RLF
- 1109: CAST(STRING AS TIMESTAMP) using invalid string lengths as of V11
- 1110: Argument for SPACE function greater than 32764
- 1111: Optional integer argument of VARCHAR greater than 32764
- 1112: Empty XML element

Automated testing – anomaly allerting

- Db2 11 introduces APPLCOMPAT DSNZPARM and BIND option
- Provides support for up to two back level releases of Db2
  - Db2 11 supports V10R1 and V11R1
  - Db2 12 supports 10, 11, and VvvRr**Mmmm**
  - Next+1… 10 support goes away
- So you can put things off until Db2 13
  - Disclaimer: IBM has not stated when there will actually be a version 13, but they said there will be one.

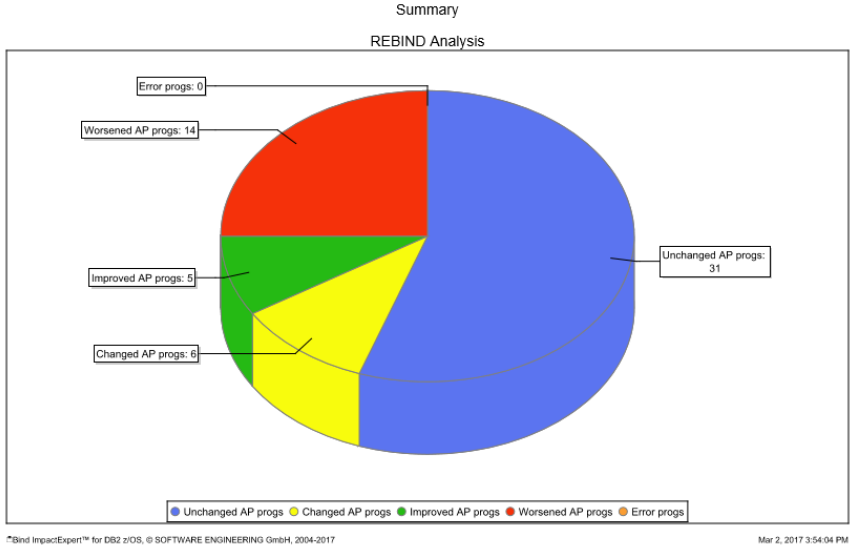
Access path change detection

Automatically and reliably check access path changes



Access path change detection

Access Path Check: Static & Dynamic SQL



STMT No.	Section Number	DSC STMT ID	Impact	Serviceunits Old	Serviceunits New	Milliseconds Old			
2547	13	2547	DEGRADED	85	1	30			
796	2	796	DEGRADED	152	1	54			
804	3	804	DEGRADED	152	1	54			
812	4	812	DEGRADED	152	1	54			
820	5	820	DEGRADED	152	1	54			
671	2	671	DEGRADED	152	1	54			
679	3	679	DEGRADED	152	1	54			
687	4	687	DEGRADED	152	1	54			
695	5	695	DEGRADED	152	1	54			
3	3	3	DEGRADED	130	1	46	1	6	8
4	4	4	DEGRADED	95	2	34	1	4	8
5	5	5	DEGRADED	82	1	29	1	4	8
6	6	6	DEGRADED	149	1	53	1	4	8
968	7	968	IMPROVED	171	16	60	6	8	5
1167	9	1167	IMPROVED	174	10	62	4	10	5
235	1	235	IMPROVED	178	10	63	4	8	6
1194	2	1194	IMPROVED	150	191	53	67	12	1
2409	7	2409	IMPROVED	21	23	8	9	5	1
1194	2	1194	IMPROVED	175	191	62	67	12	1

Workload KPI verification at a glance

Workload Capture/Replay with KPI verification:

→ Workload Capture/Replay adds application level testing and automates executing sets of captured workload.

- Highly efficient IFCID (OPx) capturing to catch SQL for automated execution in the isolated, cloned environment.
- Workload sets can be saved to represent quarter's end, year's end and other specific workload patterns.
- Tested workload is automatically compared on a KPI level (e.g. # of getpages, rows returned, rows processed…) to report only anomalies.

Workload KPI verification

- Counters
  - More than 100 KPIs, like
    - executions, getpages, IX/TS scans, rows processed/examined, …
- Zero-Counters
  - Failure indicators that should always be zero, like
    - RID list overflow, RID list append, RID pool failure, …
- Timings
  - More than 80 KPIs, like
    - CPU/elapsed time, claim/lock/latch wait time, thread read/write
- Identification
  - About 20 IDs, like
    - SQL ID, end user, workstation, transaction, …
- Environmental
  - Metadata, like collection ID, currentdata, isolation level, …

Workload KPI verification

- SQL workload that has been captured can be divided into two groups:
  1. Re-executable statements
     - SELECT A, B, C FROM MYTABLE WHERE B = 'B'
  2. Non re-executable statements
     - SELECT A, B, C FROM MYTABLE WHERE B = ?
     - SELECT A, B, C FROM MYTABLE WHERE B = :B
- SQL with literals and host variables needs to be prepared for re-execution

Workload KPI verification

To make non re-executable statements executable, we replace parameter markers and host variables with real values, considering the characteristics of the affected column.

```
SELECT A, B, C FROM MYTABLE WHERE B = ?
```

```
SELECT A, B, C FROM MYTABLE WHERE B = 'B’
```

A simple shot is a character = 'A' and numeric = 5 replacement.

A more sophisticated solution checks catalog statistics for more solid replacements.

Workload KPI verification

- The captured statements can (and should) represent various workloads to cover a representative scope
  - Month's end processing
  - Quarter's end processing
  - Year's end processing
  - Typical OLTP
  - Typical batch
- … and can be bundled in workload sets to be individually chosen for testing.

Workload KPI verification

- Executing the captured statements dynamically doesn't require the associated applications
  - There are typically no static programs/packages as part of the Db2 clone
- Consider parameters allowing to chose the number of executions per statement
- Ignore certain SQL errors/warnings, like
  - Duplicate key
  - Object exists
  - Grantee already has the permission
  - +100

Workload KPI verification

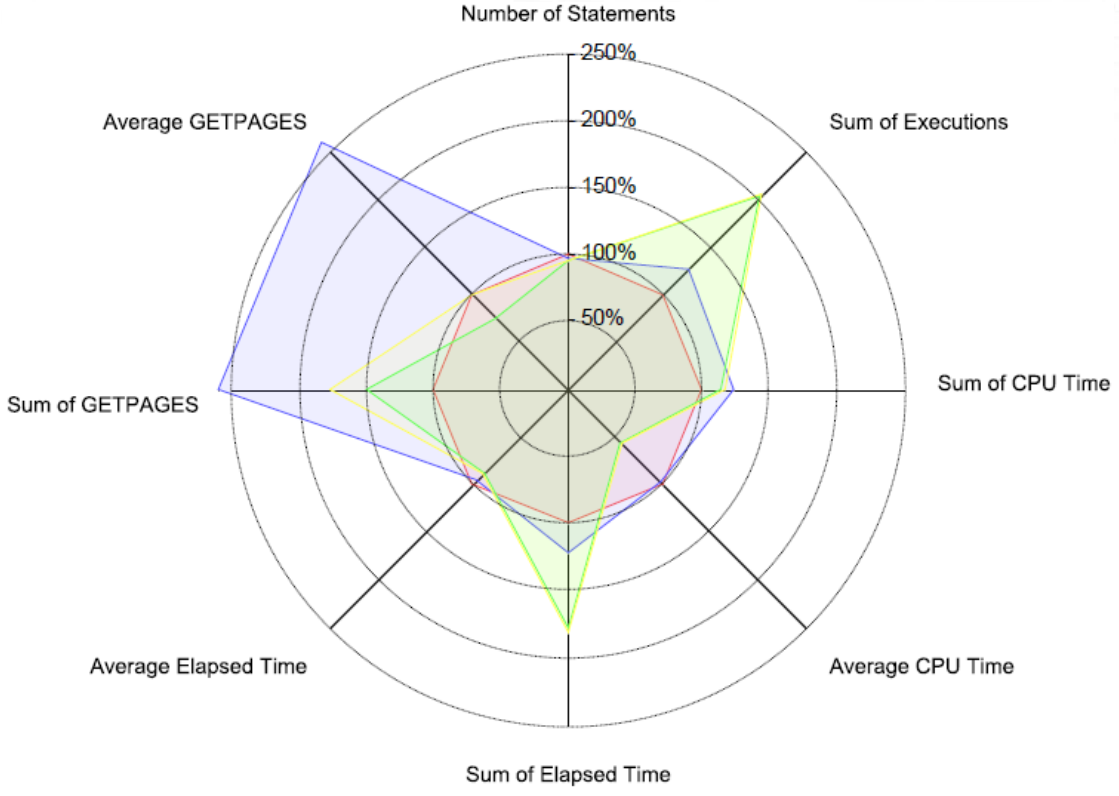
- Before starting the mass execution, verify to be in our own cloned and encapsulated environment!

- The entire execution needs to be monitored (using OPx based, highly-efficient capturing technology) to gather comparison metrics and KPIs, like
  - CPU consumption
  - Access path pattern
  - Rows processes/examined
  - …

  for a before and after comparison of changes, like:
  - New application release
  - System changes
  - Db2 APAR/PTF/new Db2 12 modification level
  - Environment/hardware changes

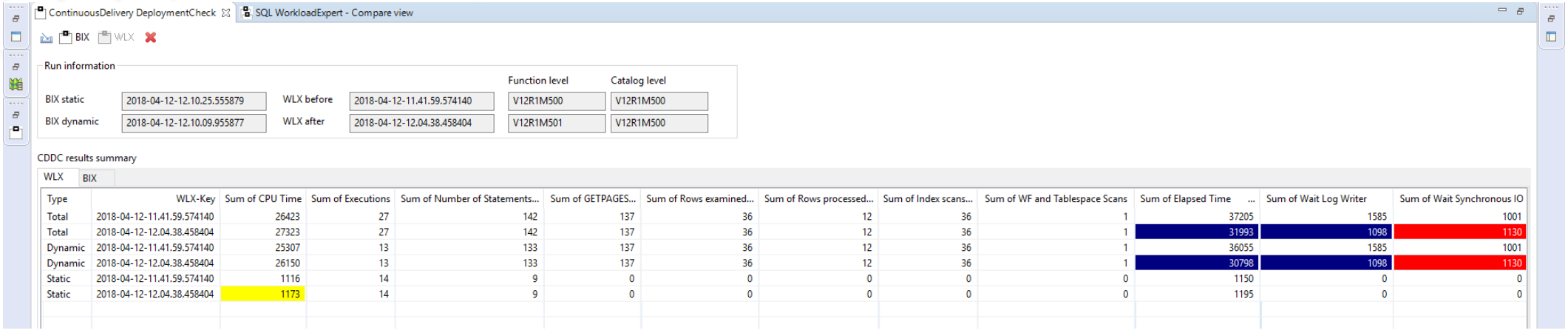
Workload KPI verification

The big benefit of KPI verification is the ability to handle massive amounts of testing without the overhead of having to review individual executions.



Workload KPI verification

Drill down capabilities allow looking into details, when anomalies are detected



ContinuousDelivery DeploymentCheck | SQL WorkloadExpert - Compare view

BIX | WLX

Run information

				Function level	Catalog level
BIX static	2018-04-12-12.10.25.555879	WLX before	2018-04-12-11.41.59.574140	V12R1M500	V12R1M500
BIX dynamic	2018-04-12-12.10.09.955877	WLX after	2018-04-12-12.04.38.458404	V12R1M501	V12R1M500

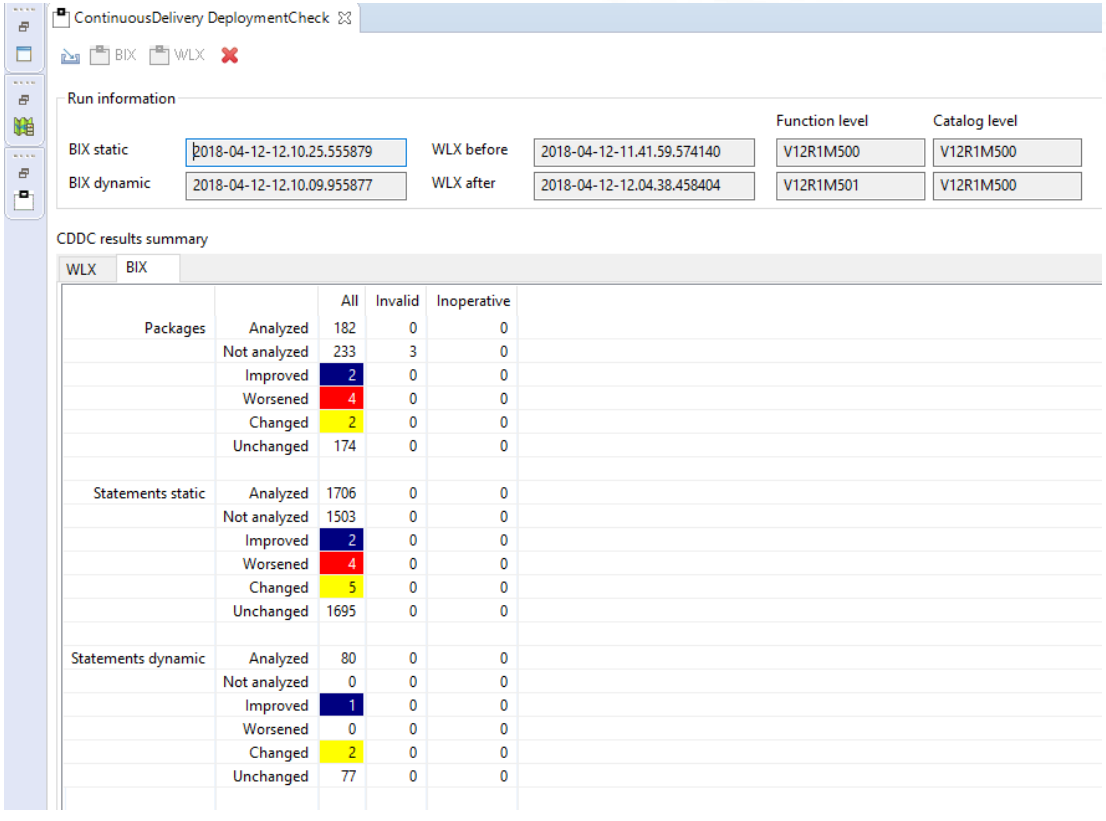
CDDC results summary

WLX | BIX

Type	WLX-Key	Sum of CPU Time	Sum of Executions	Sum of Number of Statements...	Sum of GETPAGES...	Sum of Rows examined...	Sum of Rows processed...	Sum of Index scans...	Sum of WF and Tablespace Scans	Sum of Elapsed Time ...	Sum of Wait Log Writer	Sum of Wait Synchronous IO
Total	2018-04-12-11.41.59.574140	26423	27	142	137	36	12	36	1	37205	1585	1001
Total	2018-04-12-12.04.38.458404	27323	27	142	137	36	12	36	1	31993	1098	1130
Dynamic	2018-04-12-11.41.59.574140	25307	13	133	137	36	12	36	1	36055	1585	1001
Dynamic	2018-04-12-12.04.38.458404	26150	13	133	137	36	12	36	1	30798	1098	1130
Static	2018-04-12-11.41.59.574140	1116	14	9	0	0	0	0	0	1150	0	0
Static	2018-04-12-12.04.38.458404	1173	14	9	0	0	0	0	0	1195	0	0

Workload KPI verification

Drill down capabilities allow looking into details, when anomalies are detected



ContinuousDelivery DeploymentCheck

BIX WLX

Run information

				Function level	Catalog level
BIX static	2018-04-12-12.10.25.555879	WLX before	2018-04-12-11.41.59.574140	V12R1M500	V12R1M500
BIX dynamic	2018-04-12-12.10.09.955877	WLX after	2018-04-12-12.04.38.458404	V12R1M501	V12R1M500

CDDC results summary

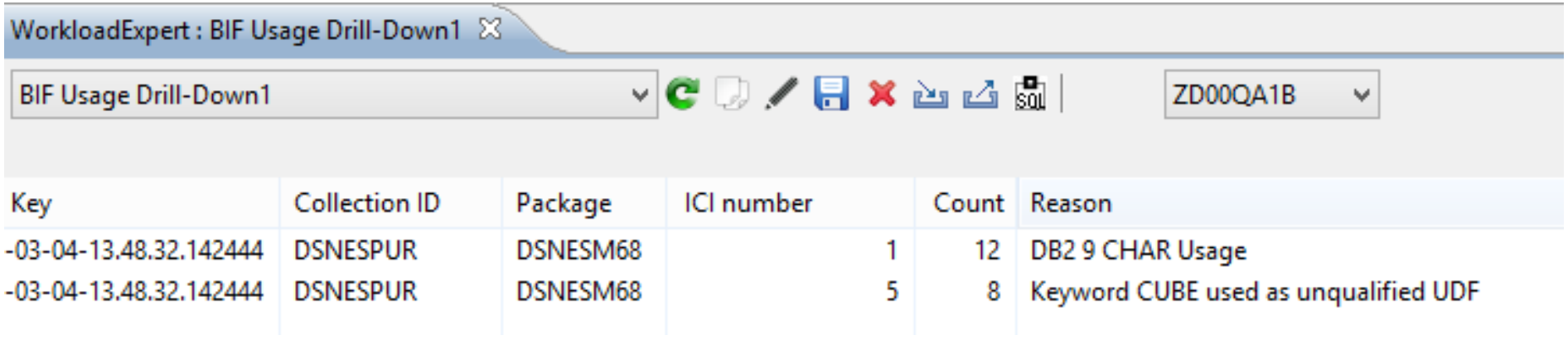
WLX BIX

		All	Invalid	Inoperative
Packages	Analyzed	182	0	0
	Not analyzed	233	3	0
	Improved	2	0	0
	Worsened	4	0	0
	Changed	2	0	0
	Unchanged	174	0	0
Statements static	Analyzed	1706	0	0
	Not analyzed	1503	0	0
	Improved	2	0	0
	Worsened	4	0	0
	Changed	5	0	0
	Unchanged	1695	0	0
Statements dynamic	Analyzed	80	0	0
	Not analyzed	0	0	0
	Improved	1	0	0
	Worsened	0	0	0
	Changed	2	0	0
	Unchanged	77	0	0

Workload KPI verification

BIF Usage is a major area of concern and occurs quite often.

The ability to test multiple sets of workloads, detects even quarters-end, or years-end query issues before they occur in production.



Key	Collection ID	Package	ICI number	Count	Reason
-03-04-13.48.32.142444	DSNESPUR	DSNESM68	1	12	DB2 9 CHAR Usage
-03-04-13.48.32.142444	DSNESPUR	DSNESM68	5	8	Keyword CUBE used as unqualified UDF

CSV Success Story

Get rid of the sh**

Securely freeing up resources

by

eliminating relics

Project margin growth – do more with less

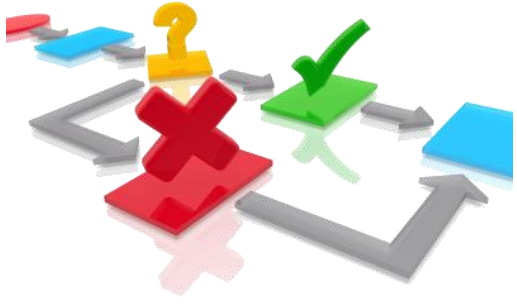
Goals:

- Cap system growth
- Reduce maintenance efforts
- Simplify environments
- Free up system and human resources
- Infrastructure modernization and consolidation

The CCV approach

Requirements:

- Reliably discover unused objects, applications and data without harming the production environment
  - system is needed that behaves exactly like production
    1. Fast, uninterupted cloning procedure
    2. Resource friendly workload capturing
    3. Interactive, but highly automated testing opportunities



→ Clone – Cap – Verify

CDDC by SEGUS – Success Story

SEGUS offers a common product that comes with the right components for our CCV approach:

1. Fast, non-disruptive cloning
   Clones Db2 based on Db2 Backup System
2. Highly efficient workload capturing
   Collects any amount of workload
3. Powerful workload replay
   Replays individual applications and automatically verifies anomalies

→ Step by step unused applications, data and objects are discovered, removed and verified

CDDC by SEGUS – Success Story

Findings:

- Various Db2 objects not used by our current workload
  - Hundreds of tables dropped, freeing up space and maintenance
  - Several unused indexes discovered and relaibly verified before production removal
- Significant number of OOS packages and statements
  - Long-term workload capturing technology of SEGUS opens up great insight into the effective workload!

→ Clean up was reliably verified before touching production

ContinuousDelivery DeploymentCheck at a glance

- Fully automates the process of setting up a production clone for testing
  - Exploiting non-disruptive, resource friendly flashcopies
  - Simulating CPU and storage by virtualization technology
  - Comprehensive process, including routing, ftp, operating commands
  - Dynamically supports a variety of source (production systems) with a single one time setup
  - Fully flexible in regards of customization via XML scenario scripts
  - Capture/Replay covers the entire production workload and includes
- Inconsistency checks
  - Access path comparison
  - SQL workload result verification
  - Resource monitoring

→ All results are automatically analyzed to generate multi level reports

Roy Boxwell
SOFTWARE ENGINEERING
r.Boxwell@seg.de

Session code: [V4]