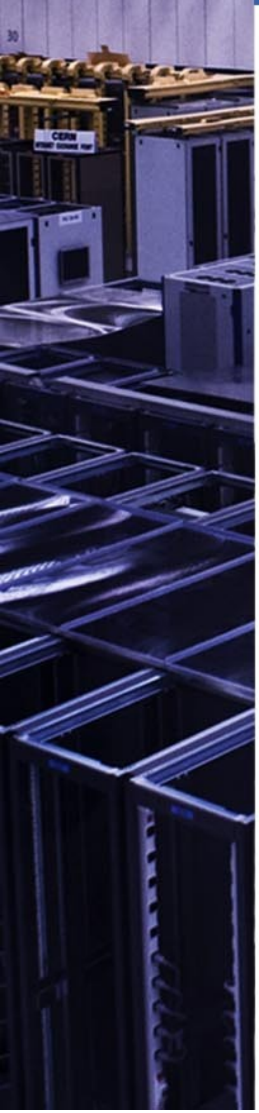


Application Validation for Upgrades

Jacek Wojcieszuk

CERN/IT-DB, November 16th, 2010

Outline

- Validation:
  - Why?
  - When?
  - What?
  - How?
- Oracle Real Application Testing
- Status of 10.2.0.5 validation

Few good reasons to validate changes

- Information systems get increasingly complicated
  - It is harder and harder to predict consequences of even small changes
- Information systems get increasingly important
  - Reliability is one of the most important properties
- Credibility is one of the things very easy to loose and very difficult to recuperate

Proper validation of applications is essential

Types of changes requiring validation/testing

- Application changes:
  - Schema changes
  - Workflow changes
  - Leveraging new DB features
  - Significant query changes
- DB client software changes
- RDBMs changes:
  - Software upgrades
  - Configuration changes
  - Hardware changes

Status Quo

- Majority of DB applications deployed at CERN used to miss comprehensive validation
  - Typically only functionality was being checked
  - Some types of changes often not validated at all
  - Reasons:
    - Difficulties to generate realistic/representative workload
    - Lack of dedicated validation environment
    - Lack of manpower
    - Validation doesn't give direct benefits
    - Changes in different places are relatively frequent
    - It is impossible to validate an application without help from its developers/maintainers

Few real-life examples of possible consequences

- Inability to run

  E.g. PSU April introduced a bug resulting in spikes of load when certain DB features are heavily used.

  - Not caught during validation
  - Patch had to be rolled back

- Logical data corruption

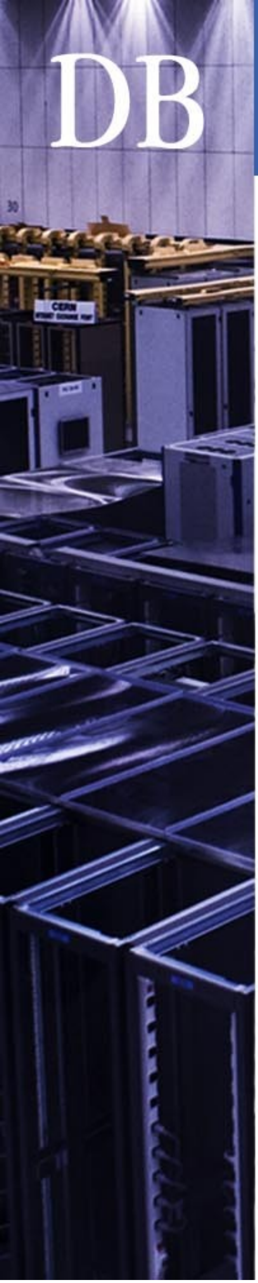
  E.g. 10.2.0.2 patchset introduced a bug due to which Oracle could mix up cursors executed against different schemas.

  - Not caught during validation
  - few prod schemas corrupted

- Degraded performance

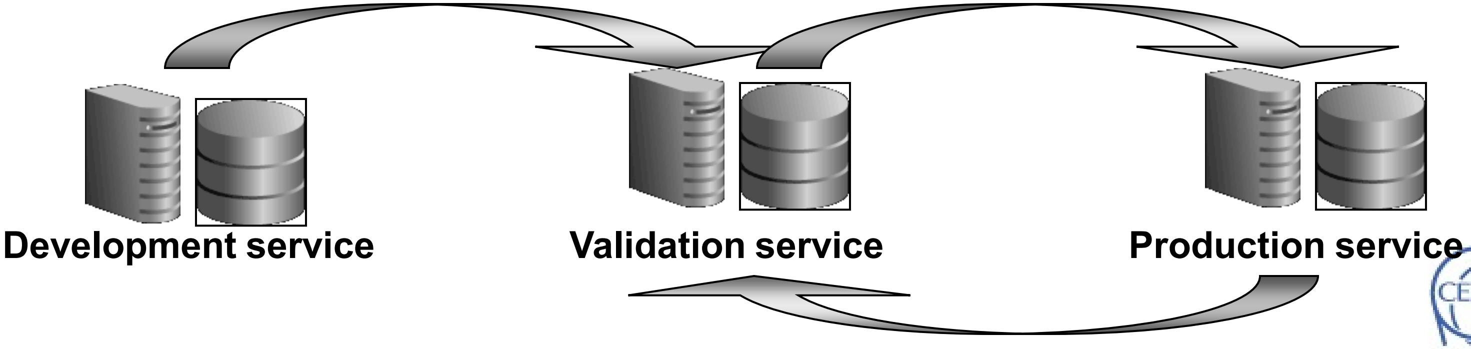
  E.g. Populating summary tables using triggers in one of online applications caused serious locking issues and severe performance degradation

  - In single user mode worked beautifully
  - Application could not cope with the load



Validation principles

- Validation should be considered an integral part of software lifecycle
  - It requires attention, resources and a lot of effort
- Should cover all essential areas:
  - Functionality
  - Stability
  - Performance
  - Scalability
- Should cover all relevant access patterns:
  - OLTP & batch
  - Single user & concurrent



Load generation

- Providing representative load is a key for successful validation
  - Never easy to achieve, sometimes almost impossible
  - Especially tricky in case of Web applications
  - The goal should be to stay as close to real workload as possible
- Automatic, repeatable load generators are the best to feed validation process
- Understanding the usage pattern of the application is important
  - Analyzing application logs and gathering statistics on application usage may help
- Custom workload capture and replay sometimes feasible

Validation environment

- Dedicated validation environment
  - Saves time
  - Guarantees repeatability of results
  - Suitable for all types of tests
- Shared environment
  - Can be sufficient in many cases
  - Can be a good compromise

Analysis

- Was the application running correctly?
- Has single-thread performance changed?
  - To better or to worse?
  - Do all SQL statement have acceptable execution/response times?
- Are there any concurrency issues?
  - Has the overall throughput improved/degraded?
- Does the application scale well?
  - Has the scalability improved/degraded?
- What's the database footprint of the tests?

What DBAs cannot do

- Prepare applications for validation
- Run validation
- Enforce validation
- Assess if performed validation was comprehensive enough
- Assess the impact of the validated change on the application

What DBAs can do

- Provide test/integration DB services
  - IT/DB group maintains several test/integration databases
  - Deployed on hardware similar to prod and following the same configuration
  - Typically patched much in advance before production
- Move data
  - Production schema can be copied to test/integration on demand
  - An effort is made to automate it as much as possible
- Analyze validation runs from database perspective
- Consult

Examples - CMS PhEDEx validation environment

- Four-level development/test/validation environment:
  1. Development service:
     - for testing new ideas
  2. Testbed - a small number of dedicated client machines to run stress-tests agains an integration database
     - ~50 fake clients continuously deployed and ready to be used
     - Extra clients deployed on borrowed hardware for large scale tests
     - PhEDEx software used + a set of scripts generating fake transfer requests
     - Comprehensive web-based monitoring
     - Key utility to ensure smooth changes both on the application and database level

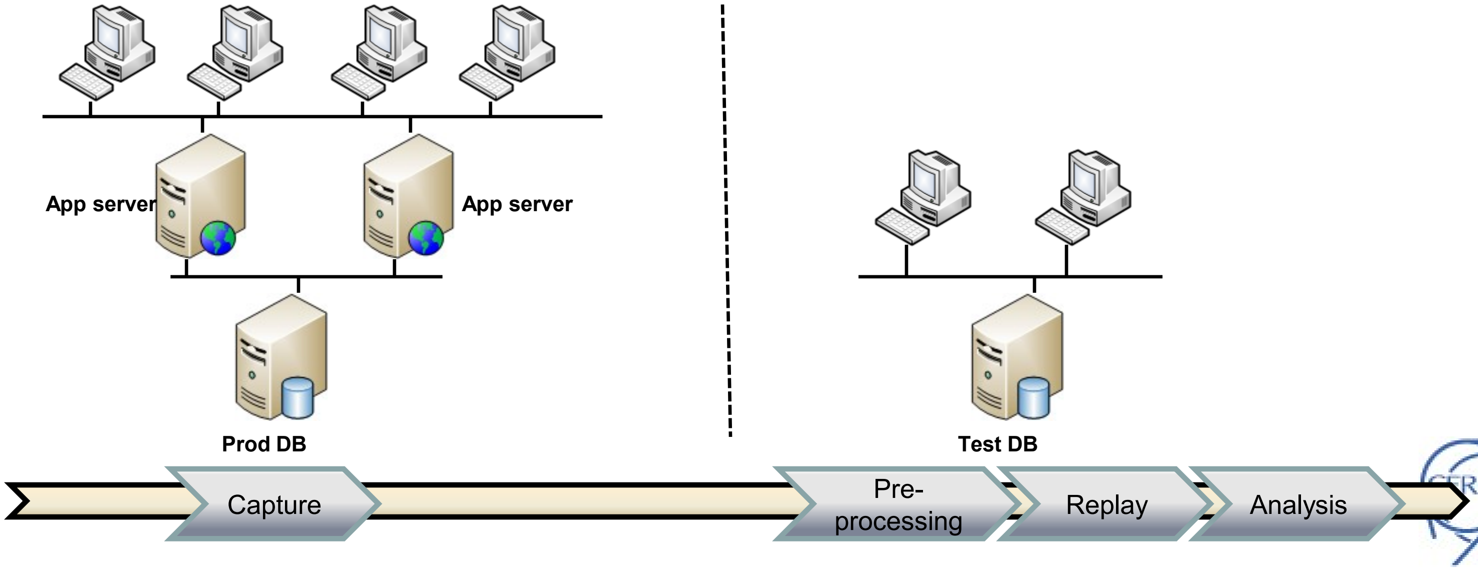
  3,4. Two more layers to test and debug changes at the application layer
     - Clients deployed at production sites
     - Debug instance completely mimics production environment
- Majority of problems catched at level 1 and 2

COOL validation

- Consist of a comprehensive set of unit test
  - Deployed on dedicated hardware
  - Using a test RAC database
  - Run automatically every night
  - Generate artificial load but exercising all DB features levereged by the application
- Chosen nightly tests can be run on demand concurrently to stress the DB
  - Set of scripts simlifing it
  - Deployed on AFS
- No real scallability tests
- Enough to intercept majority of possible issues
- Very handy for reproducing problems caused by RDBMS software bugs

Oracle Real Application Testing

- A feature of Oracle 11g RDBMS
- Consists of load capture and load reply engines
- Capture:
  - Allows for capturing and storing in files of database load
  - Output files in Oracle’s proprietary format
  - Many filtering options
- Replay:
  - Re-executes captured load
  - Can be done using separate client hardware
  - Can be done against a database of the same or higher version
  - Several ways to impact replay intensity



Oracle Real Application Testing – known issues

- Is an extra-paid option
- Relatively new software – still in the process of becoming mature
- Capturing load is not always straightforward
  - Database restart sometimes needed to get a clear capture start point
- Replay requires that the database is at the same state as at the beginning of load capture
- Issues when replying OLTP workload
- Can simplify only a limited set of validation cases
  - E.g. It is useless for validating changes in the application
- Still it has potential to become a very handy complementary validation utility

10.2.0.5 patchset validation

- 10.2.0.5 is a ‘thick’ patchset
  - The binaries are 1.2 GB big
  - Includes several hundreds of bug fixes
  - Most likely many functional changes especially in the optimizer code
- Validation of all critical applications is essential
  - 2011 run will be very important for experiments
- Decission concerning upgrade of production DBs and detailed schedule expected in the middle of December
- Validation in progress since middle of October
  - So far positive
  - To be concluded in 2-3 weeks
- It is a useful exercise before validating 11.2 release

Status of 10.2.0.5 validation

Experiment	Application	Validated for 10.2.0.5
Alice	PVSS	In progress
ATLAS	PVSS	In progress
	Panda	pending
	DDM	pending
	PVSS2COOL	In progress
	COOL/T0 processing	pending
	TAGS	pending
CMS	PVSS	pending
	Storage Manager	✓
	Conditions/FronTier	pending
	PhEDEx	In progress
	T0AST	In progress
	DBS	In progress
LHCb	PVSS, RunDB	In progress
WLCG	Dashboards	pending
	FTS	pending
	LFC	pending
	SAM	pending

Summary

- Application validation is a necessity
- It is a costly process but sooner or later it pays off
- Lack of proper validation may have serious consequences
  - even this year we had some examples
- Oracle Real Application Testing can potentially simplify validation of RDMBS upgrades
- Validation of 10.2.0.5 patchset is progressing; still quite some work ahead
  - Even more validation ahead due to upgrade to 11.2 planned for 2012