

Database Applications Lifecycle Management

Marcelo F. Ochoa
Oracle Ace Member
Fac. Cs. Exactas - UNICEN



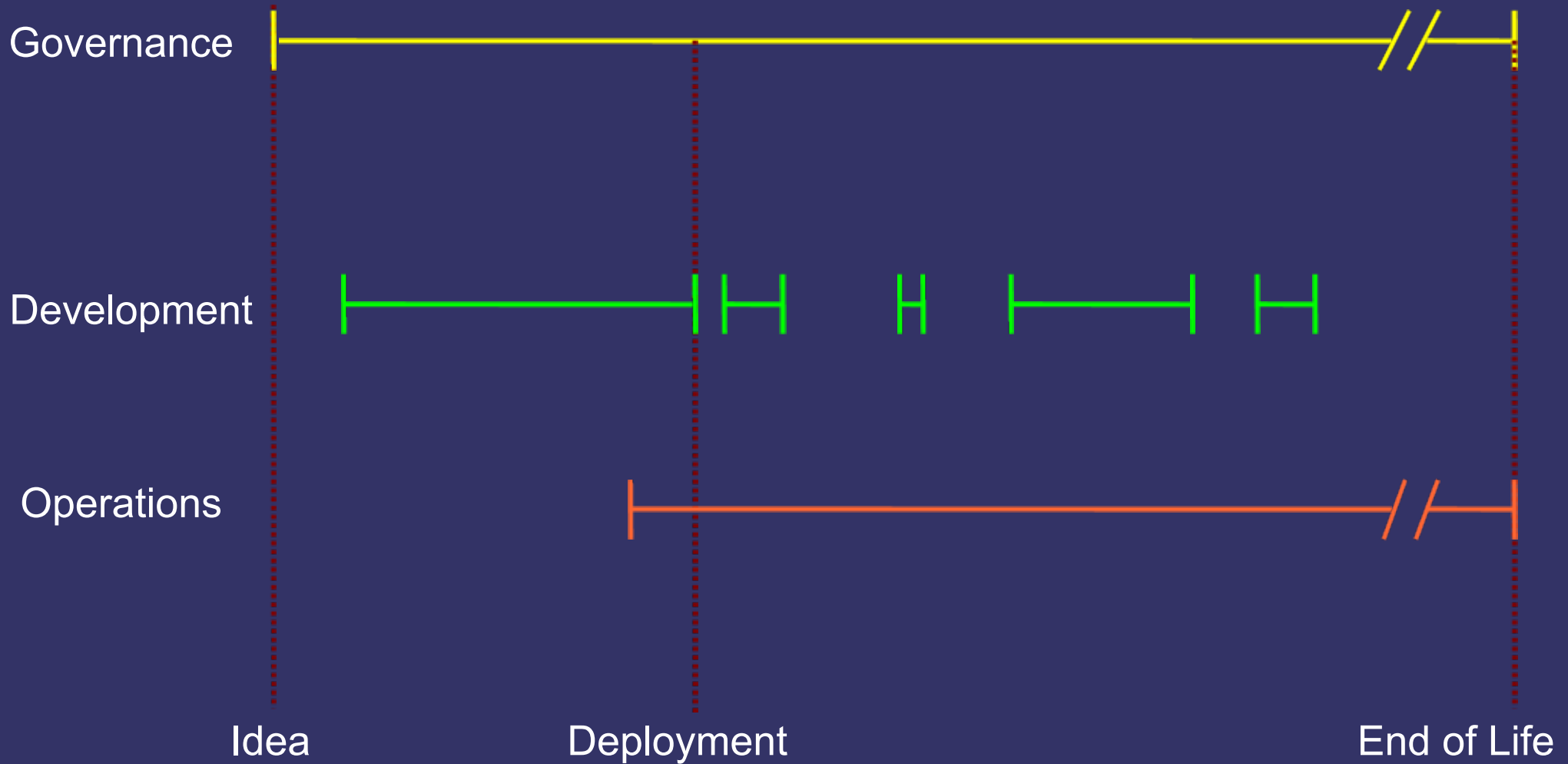
What is ALM?

Application lifecycle management (ALM) is the marriage of business management to software engineering made possible by tools that facilitate and integrate requirements management, architecture, coding, testing, tracking, and release management.

[Source Wikipedia](#)



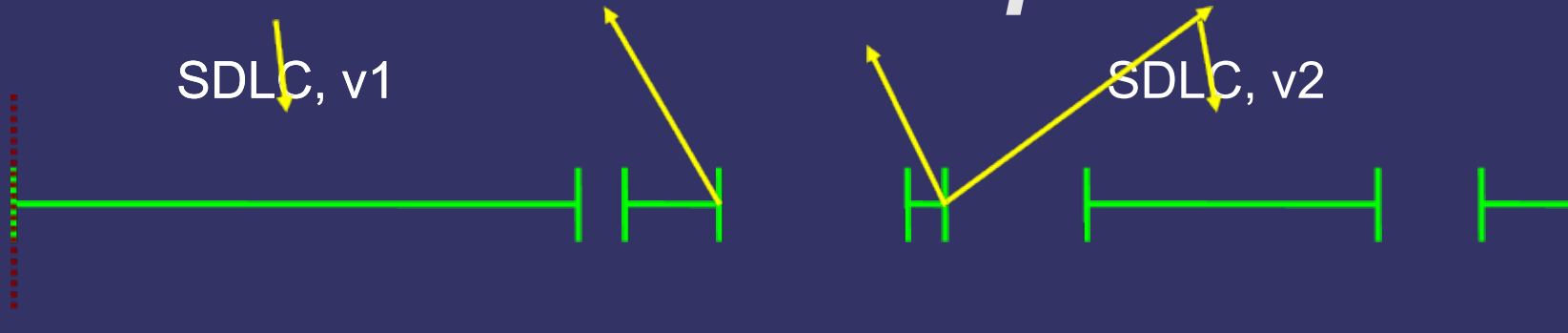
ALM Areas



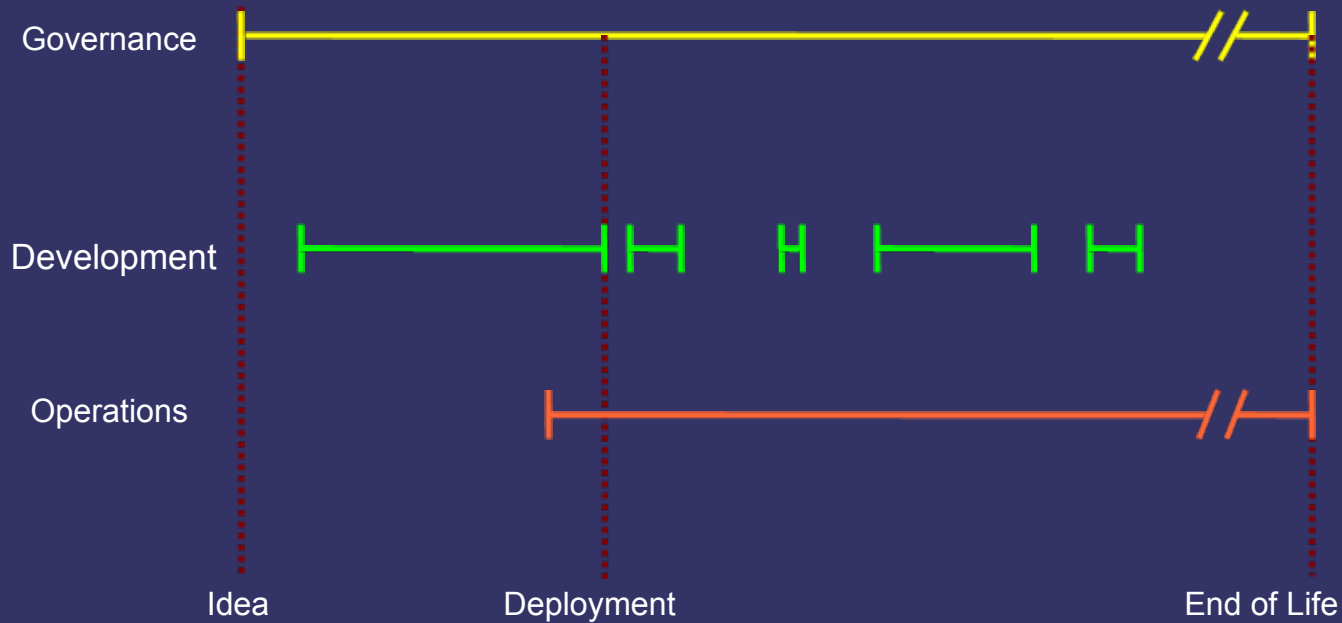
Source [What is ALM--Chappell.pdf](#)



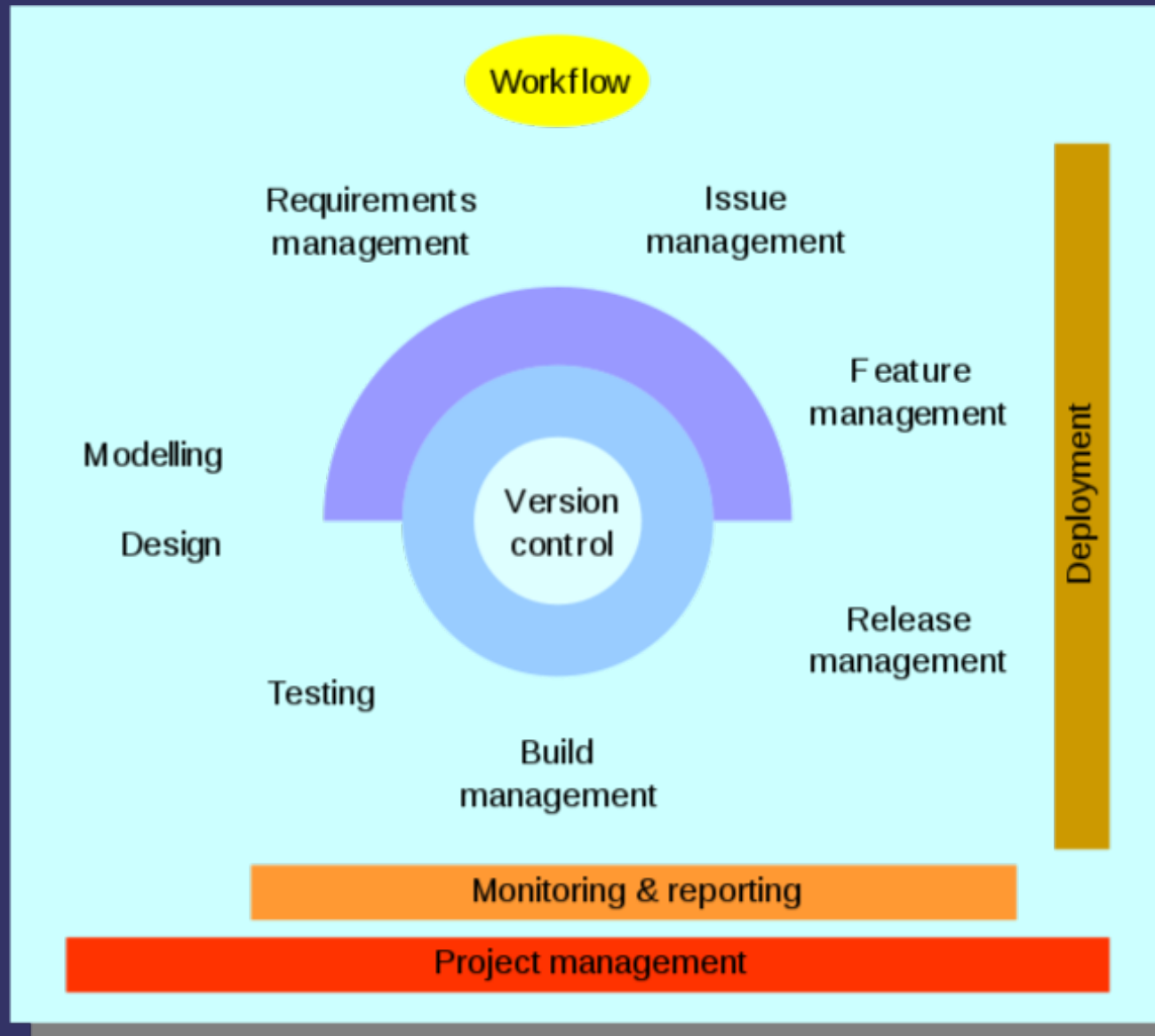
ALM: Development



Maintenance



ALM Tools



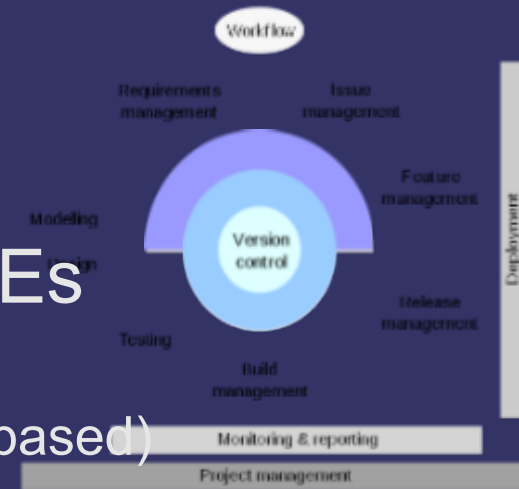
Server Side Components

- PL/SQL
 - Packages
 - Procedures and Functions
 - Triggers
 - Object types
- Tables, Indexes and other
- XMLDB repository objects
- Java Code
 - Package implementations
 - Procedure and Function Implementations
 - Trigger implementations
 - Object Type implementations
 - Common libraries

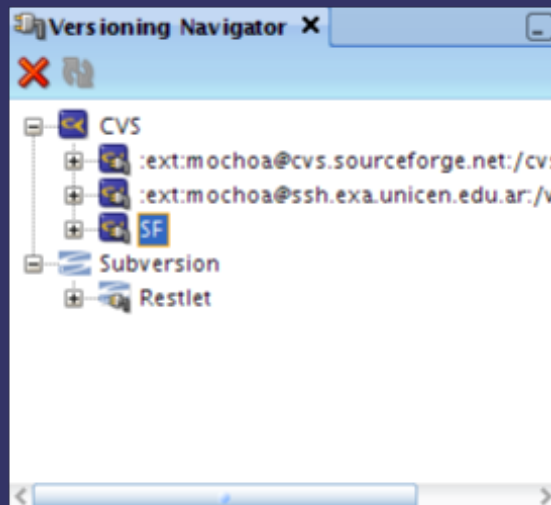
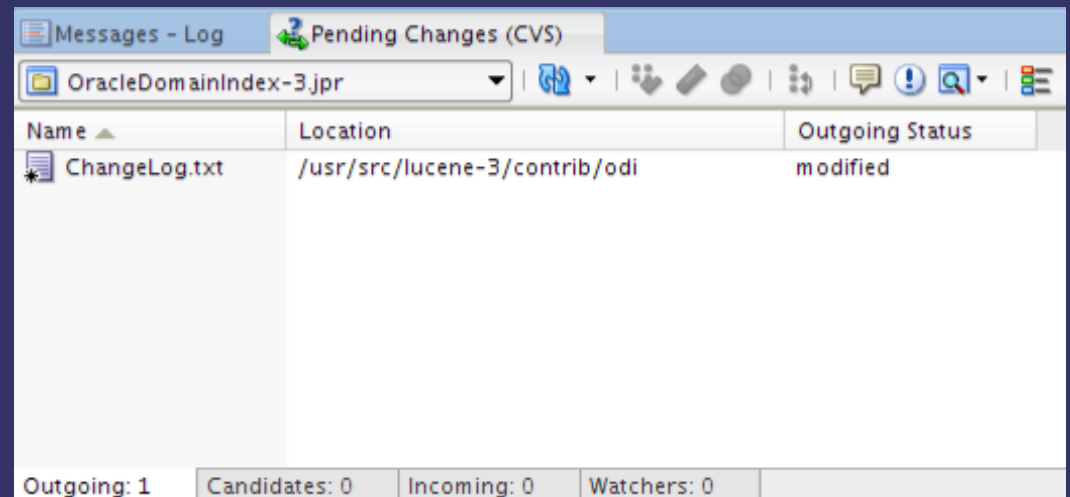
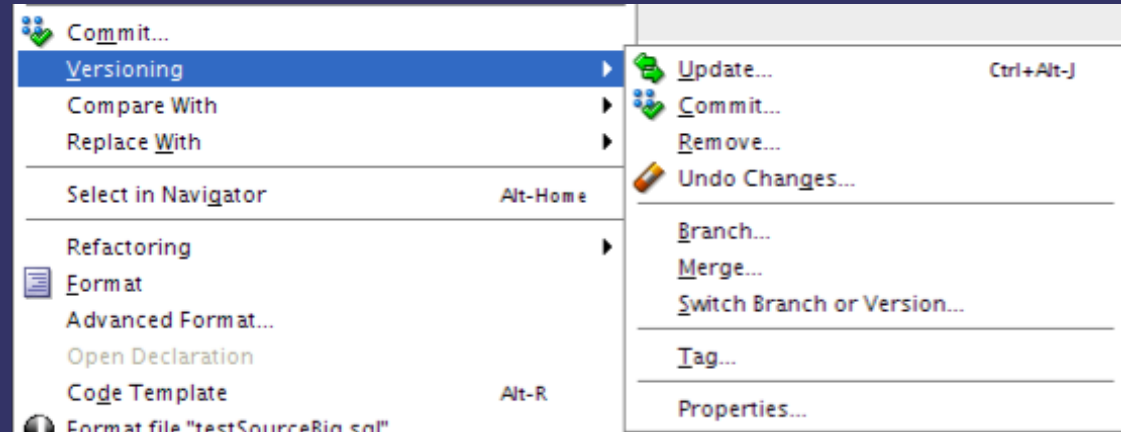
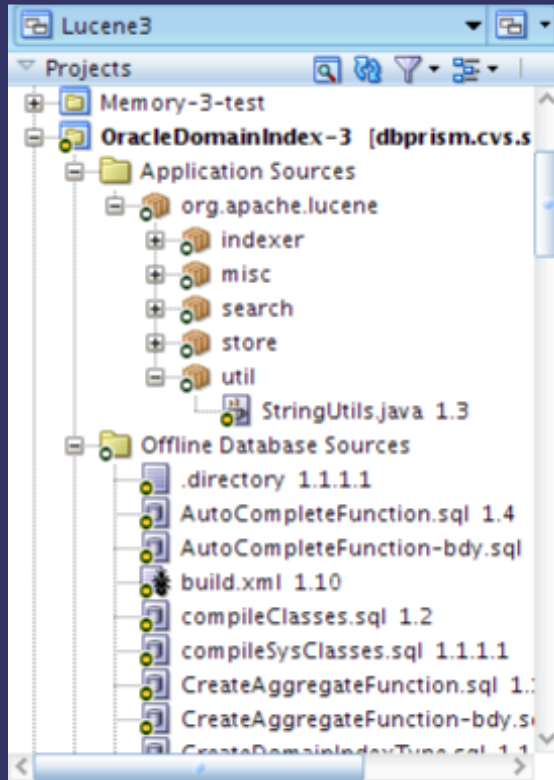


SCM tools

- Popular SCMs are integrated on all IDEs
- JDeveloper includes by default
 - CVS (password based or public-private key based)
 - SVN
 - Dimension
 - Perforce
 - Team System VCS
- SQLDeveloper is based on JDeveloper core so it enables similar list of SCM tools support
- Oracle Team Productivity Center is a good add-in for free
 - ALM tool requires that everybody on dev. team use JDeveloper



JDeveloper SCM tool in action



JDeveloper SCM tool in action

Revision Filter: All

Date	Tags	Author	State	Lines	Revision	Description
Aug 24, 2010 3:59:27 PM					Local	Load External State
Apr 20, 2010 11:44:48 AM		mochoa	Exp	+22 -30	1.2	Added IndexOnRam functionality
Apr 7, 2010 10:55:04 AM		mochoa	Exp		1.1	Initial revision
Apr 7, 2010 10:55:04 AM	start	mochoa	Exp	+0 -0	1.1.1.1	Imported sources

Revision 1.1 - odi/db/LuceneDomainIndex-bdy.sql,v (View-Only)

```
LAST_MODIFIED TIMESTAMP,  
FILE_SIZE INTEGER,  
DATA BLOB,  
DELETED CHAR(1)  
) LOB (DATA) STORE AS SECUREFILE ('||lobStorageParam||');  
end if;  
EXECUTE IMMEDIATE  
'create index '||prefix||'$DI on '||prefix||'$T (DELETED)';  
end createTable;  
  
static procedure dropTable(prefix VARCHAR2) is  
begin  
EXECUTE IMMEDIATE 'DROP TABLE '||prefix||'$T FORCE';  
end dropTable;  
  
-- This function is not available on 10g  
static function ResourceExists(path varchar2)  
return boolean as  
result number;  
begin
```

File on Disk - LuceneDomainIndex-bdy.sql (Editable)

```
LAST_MODIFIED TIMESTAMP,  
FILE_SIZE INTEGER,  
DATA BLOB,  
DELETED CHAR(1)  
) LOB (DATA) STORE AS SECUREFILE ('||lobStorageParam||');  
end if;  
if (instr(v_version,'Enterprise')>0) then  
EXECUTE IMMEDIATE  
'create bitmap index '||prefix||'$DI on '||prefix||'$T (DELETED)';  
else  
EXECUTE IMMEDIATE  
'create index '||prefix||'$DI on '||prefix||'$T (DELETED)';  
end if;  
end createTable;  
  
static procedure dropTable(prefix VARCHAR2) is  
begin  
EXECUTE IMMEDIATE 'DROP TABLE '||prefix||'$T FORCE';  
end dropTable;
```

Source | SQL Worksheet | History

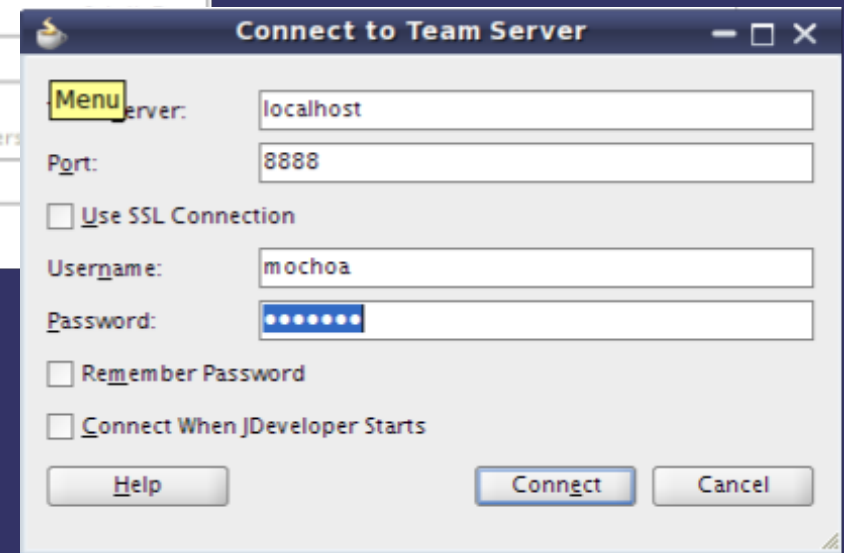
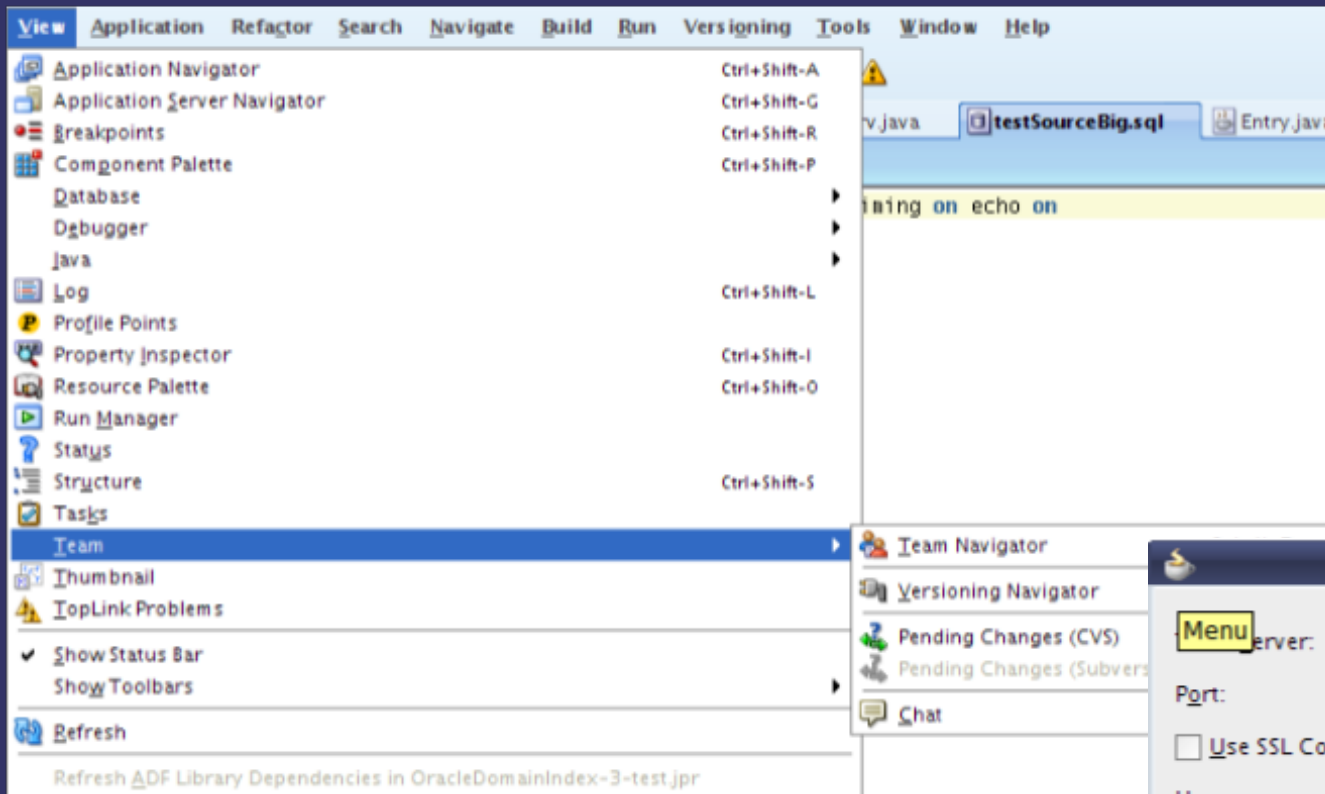


Oracle Team Productivity

- Simple
- Free
- Integrated with JDeveloper
- Ready to use connectors:
 - Bugzilla – defect tracking system
 - JIRA – bug tracking, issue tracking and project management
 - Microsoft Project Server – task management
 - Tasks – built in Task repository to create and track team tasks
- Support hierarchies of teams and users
- Messaging with the team using XMPP
- Extensible framework and sample connector



Team Productivity Center - login



Team Productivity Center - admin

Team Administration

Users Teams Repositories Roles

Users: +X

- Administrator TPC
- Gastón Martini
- Marcelo Ochoa
- Usuario Invitado

First Name: Marcelo Last Name: Ochoa

Login: mochoa Status: Active

Email: mochoa@lab-exa.unicon.edu.ar Password:

User Permissions:

- Create new teams Administrator
- Create new users

User Teams:

Name	Role
LDI	Team Administrator

Help OK Cancel

Team Administration

Users Teams Repositories Roles

Teams: +X

Team Details Team Repositories

Name: LDI

Parent Team: Status: Active

Description: Lucene Domain Index dev. group

Team Members: +X

Name	Role	Status
Marcelo Ochoa	Team Administrator	Active
Usuario Invitado	Team Member	Active
Gastón Martini	Team Member	Active

Help OK Cancel

Team Administration

Users Teams Repositories Roles

Repositories: +X

- Work Item
- Lucene
- LDI Tasks
- Document
- Versioning

Connector: Jira

Name: Lucene

Description: Lucene

Repository Servers: +X

Name	Description
Apache	Apache Servers

Repository Parameters:

Name	Value
Server URL	https://issues.apache.org/jira/

Help OK Cancel

Team Administration

Users Teams Repositories Roles

Roles: +X

Name: Team Administrator

Description: Users can manage their teams and artifacts

Role Privileges:

Privilege	Description
<input checked="" type="checkbox"/> Manage Team Queries	Create and modify team queries
<input checked="" type="checkbox"/> Manage Team Tags	Create and modify team tags
<input checked="" type="checkbox"/> Manage Team Sources	Create and modify team source or versioning
<input checked="" type="checkbox"/> Administer Teams	Add, modify, and remove team members and roles
<input type="checkbox"/> Gain Privilege Over Child Teams	Gain privilege over child teams
<input checked="" type="checkbox"/> Manage Team Documents	Create and modify team documents

Help OK Cancel



Team Productivity Center my tasks

Application x | create-lucene-role.sql | OJVMIndexInput.java | IndexScanImpl.java | IndexScanServ.java | testSourceBig.sql | My Tasks | Assign localhost to the list of perm...

LDI

- Team Members
 - Administrator TPC
 - Marcelo Ochoa
 - Usuario Invitado
- Work Items
 - Active WorkItem none
 - LDI Tasks
 - Team Queries
 - My Queries
 - My Tasks
 - Lucene
 - Team Queries
 - My Queries
 - Lucene 3.0.2 bugs

Search Criteria

Assigned To: [mochoa] Equals

Match All Match Any

Search

Result: 1 matches found. Aug 24, 2010 5:34:02 PM

Task Id	Task Name	Assigned To	Priority	Status	Start Date	End Date	Created By
1	Assign localhost to...	Marcelo Ochoa	High	Done	Mon Aug 23 09:38:	Thu Aug 26 09:38 4...	Marcelo Ochoa

Make Active

Detail * Indicates required field

Relationships

Tags

Changes

Task Name: * Assign localhost to the list of permitted server

Task Id: 1

Team: * LDI

Priority: High

Start Date: 2010-08-23 09:38:45

Created By: Marcelo Ochoa

Assigned To: Marcelo Ochoa

Status: Done

End Date: 2010-08-26 09:38:45

Description: Modify create-lucene-role.sql adding localhost to the list permitted server connections

Make Active

Detail

Revision	Comm Date	Commit By	Comments
LDI Tasks_1_2010.08.23.16.33.145	8/23/2010 16:33:26	Marcelo Ochoa (Offline)	add localhost to the list of permitted conecciones

Changes

Included Files

Name	Version
odi/db/create-lucene-role.sql	1.2



Team Productivity Center my bugs

LDI

Team Members
Administrator TPC
Marcelo Ochoa
Usuario Invitado

Work Items
Active Work Item none
LDI Tasks
Team Queries
My Queries
My Tasks
Lucene
Team Queries
My Queries
Lucene 3.0.2 bugs

Search Criteria

Project: Lucene - Java
Affects Versions: 3.0.2

Match All Match Any

Search More Actions

Result: 30 matches found. Aug 24, 2010 5:35:59 PM

Type	Key	Project	Summary	Assignee	Reporter	Priority	Status	Created	Updated	Due Date
Bug	LUCENE-2299	Lucene - Java	if you open an NRT reader while addIndexes* is running...	mikemcca...	mikemcca...	Minor	Closed	Sun Mar 0...	Fri Jun 18 ...	
Bug	LUCENE-2307	Lucene - Java	Spurious exception in TestIndexWriter	-1	earwin	Major	Open	Tue Mar 0...	Tue May 0...	
Improvement	LUCENE-2368	Lucene - Java	stopword files should be versioned; accessor for default...	-1	hossman	Major	Open	Mon Apr 0...	Fri Jun 25 ...	
New Feature	LUCENE-2454	Lucene - Java	Nested Document query support	markh	markh	Minor	Open	Mon May 1...	Fri Jul 16 1...	
Bug	LUCENE-2465	Lucene - Java	QueryParser should ignore double-quotes if mid-word	-1	itamar	Major	Resolved	Sat May 15...	Mon May 1...	
Bug	LUCENE-2467	Lucene - Java	IndexWriter memory leak when large docs are indexed	mikemcca...	mikemcca...	Major	Closed	Mon May 1...	Fri Jun 18 ...	
Improvement	LUCENE-2506	Lucene - Java	A Stateful Filter That Works Across Index Segments	-1	karthick	Major	Open	Mon Jun 2...	Tue Jun 29...	
New Feature	LUCENE-2515	Lucene - Java	Expose ExecutorService in ParallelMultiSearcher	-1	rocco ver	Major	Closed	Fri Jun 25	Fri Jun 25	
Improvement	LUCENE-2518	Lucene - Java	Make check of BooleanClause.Occur[] in MultiFieldQuer...	-1	itamar	Minor	Open	Mon Jun 2...	Mon Jun 2...	
Improvement	LUCENE-2535	Lucene - Java	update FieldSelectorResult with a BREAK only result opti...	-1	woody an...	Minor	Open	Tue Jul 13 ...	Sun Jul 18 ...	
Bug	LUCENE-2536	Lucene - Java	Rollback doesn't preserve integrity of original index	mikemcca...	markh	Major	Resolved	Tue Jul 13 ...	Tue Jul 13 ...	

Make Active

Detail * Indicates required field

Relationships
Tags
Changes

Key: LUCENE-2465

Project: * Lucene - Java

Type: * Bug

Summary: * QueryParser should ignore double-quotes if mid

Priority: Major

Due Date:

Components: QueryParser

Affects Versions: 9.2,2.9.3,3.0, Flex Branch,3.0.1,3.0.2,3.1,4.0

Fix Versions:

Assignee: *

Reporter: itamar

Environment:

Current implementation of Lucene's QueryParser char, even if it is mid-word. For example, the stri and one PhraseQuery ("bar test"). This behavior i double quotes as defined by http://lucene.apach say double-quotes will also tokenize the input. A surrounded by whitespaces.



Design Tools

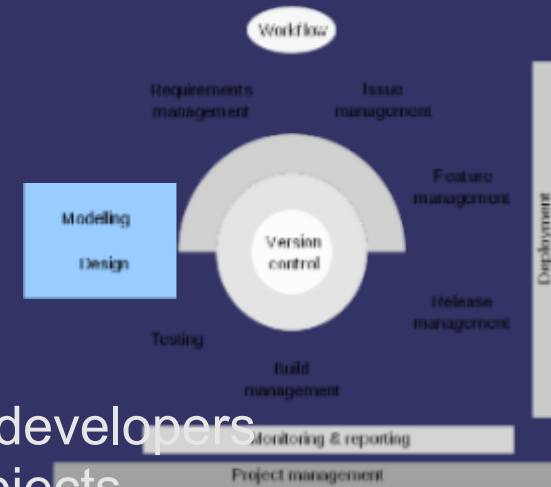
- JDeveloper

- Benefits

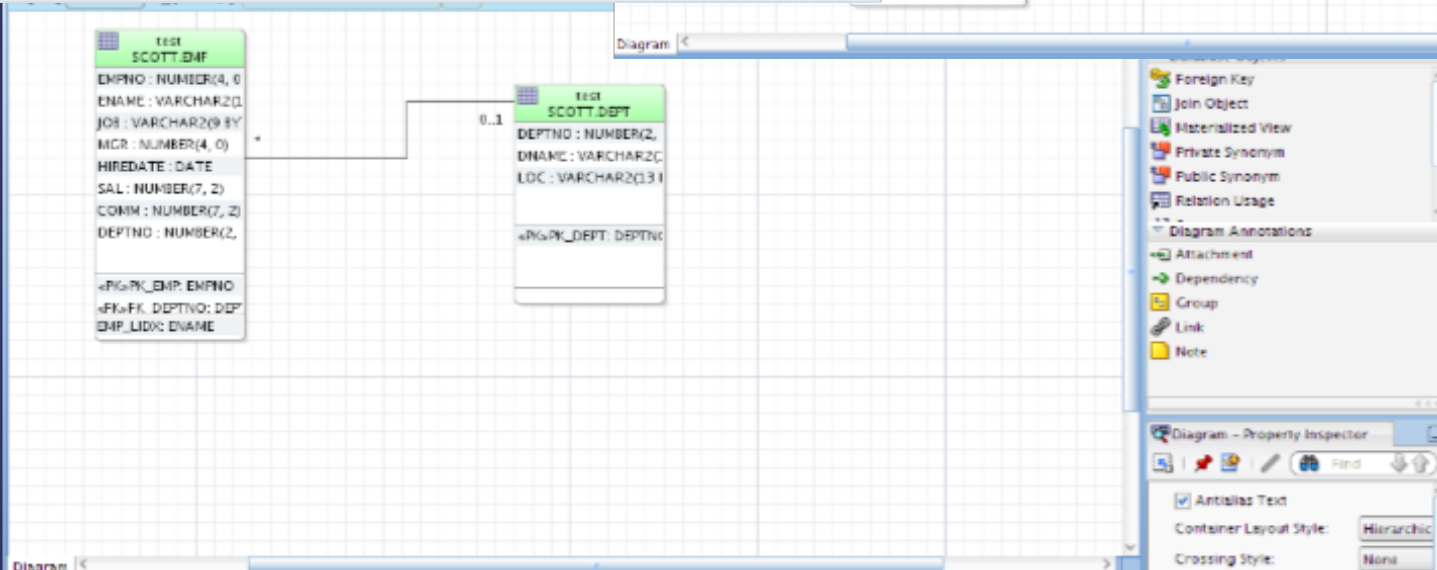
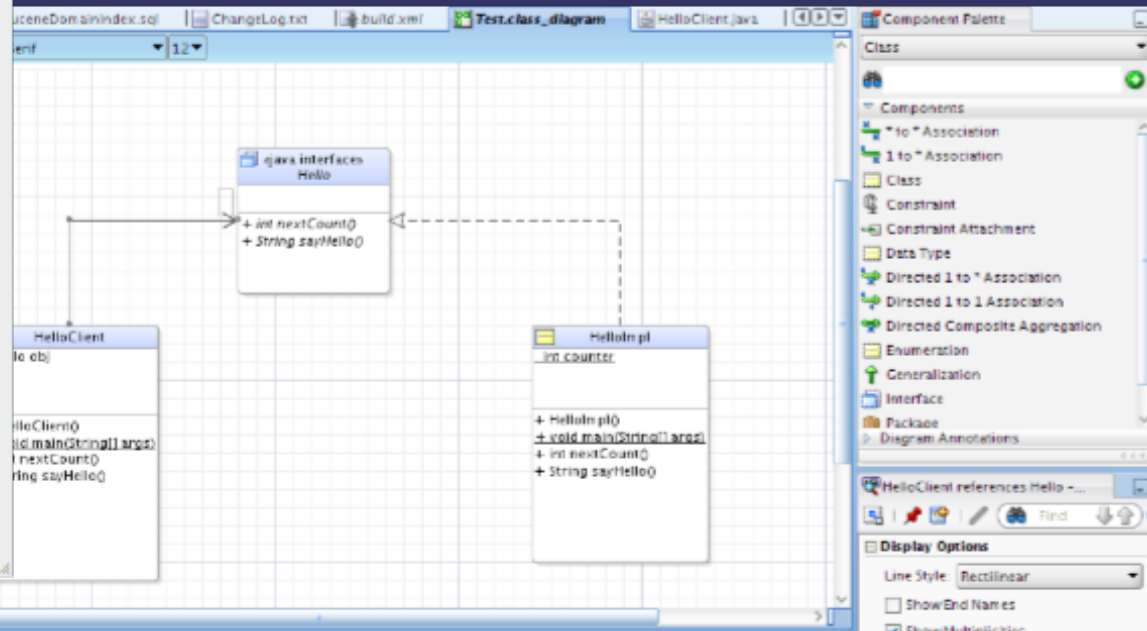
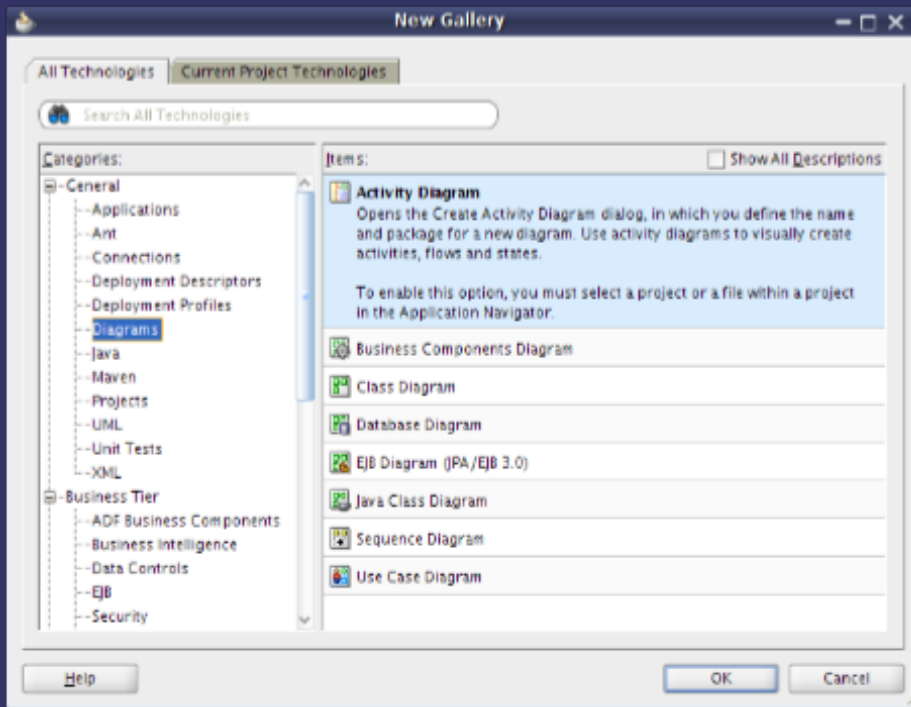
- Is good from the developer point of views
 - Covers most of design objects required by developers
 - Provides reverse engineering of schema objects
 - Has a two way editor (Class Diagram <-> Java Source)

- Drawbacks

- Is not a powerful UML tool
 - Is not possible to create live documents with diagram embedded
 - Does not include all UML diagrams



Sample JDeveloper Design tools



Development tools

- Editing tool support
 - Code insight
 - Code completion
 - Refactoring
 - Code assist
 - Live code auditing
 - Quick JavaDoc
- Debugger
- Profiling
- JUnit, Maven integration
- Database
 - Forward and reverse modeling
 - DB browsing and manipulation
 - Execution, auto-trace and explain-plan

Important: For Java server side compile always using **\$ORACLE_HOME/jdk**



Compile, build, deploy

- Most of the IDE does this task graphically
 - Is good for novel developers
 - Can do typical tasks
 - Not good for off-shoring development
 - Not good for cloud computing scenery
- Command line tools solved above problems
 - Can do almost all tasks
 - DBA and Sys admin loves this during deploy
 - Is good for remote deployment
 - Only a text terminal is necessary to run
 - Ant and Maven are the kind of these tools



Ant

- Tool designed to replace old tools such as make
- Multi-platform – Java based
- Extensible
- Tons of tasks to do everything you need
- Cover all the steps during:
 - Compile
 - Build
 - Test
 - Release management
 - Deploy
- Also you can work with PLSQL and SQL*Plus



Ant commons tasks

- Define your *ORACLE_HOME*

```
<property name="env" environment="env" value="env"/>  
<property name="ORACLE_HOME" value="{env.ORACLE_HOME}"/>
```

- Using *SQL*Plus*

```
<property name="SQLPlus" value="sqlplus"/>  
.....  
<target name="create-lucene-role">  
<exec executable="{ORACLE_HOME}/bin/{SQLPlus}" dir="./db" failonerror="true">  
<arg value="{dba.usr}/{dba.pwd}@{db.str} as sysdba"/>  
<arg value="@create-lucene-role"/>  
</exec>  
</target>
```

- Other Oracle's external tools (using tasks)

- loadjava, dropjava
- NCOMP (10g)
- Retro-translator (JDK 1.5 Libraries running in 10g)



Ant examples

- Uploading Log4J to the RDBMS

```
<target name="load-extra-libs" depends="get-junit-jar,get-codec-jar">
<loadjava-with-public-synonym
login="${db usr}/${db.pwd}@${db.str}"
file="lib/junit-3.8.2.jar"/>
<loadjava-with-public-synonym
login="${db usr}/${db.pwd}@${db.str}"
file="lib/commons-codec-1.4.jar"/>
</target>
```

- NCOMPing 10g code (Java -> assembler)

```
<target name="ncomp-runtime-retrotranslator-sys-code"
description="NCOMP Retrotranslator Runtime Java code - translate Java to assembler" >
<delete file="exact_class_list.txt" />
<delete file="*ClassListAndTCDumper.java" />
<ncomp
login="${dba usr}/${dba.pwd}@${db.str}"
file="bin/bcel-5.2.jar"
dir="tmp"
libName="bcel_ncomp_odi" />
.....
</target>
```



Maven

- Ant superset
- A project management framework
- Based on a Project Object Model (POM)
- Provided compile, *test*, assemble, install
- Maven's principles:
 - Convention over configuration
 - Standard directory layout for projects
 - A single Maven producing a single output
 - Standard naming convention
 - Declarative execution (POM)
 - Reuse build logic (*plugins*)
 - Coherence organization of dependencies
- You can not build a component if the test stage fails
- If there isn't a *plugin* for your need Ant tasks can do the job

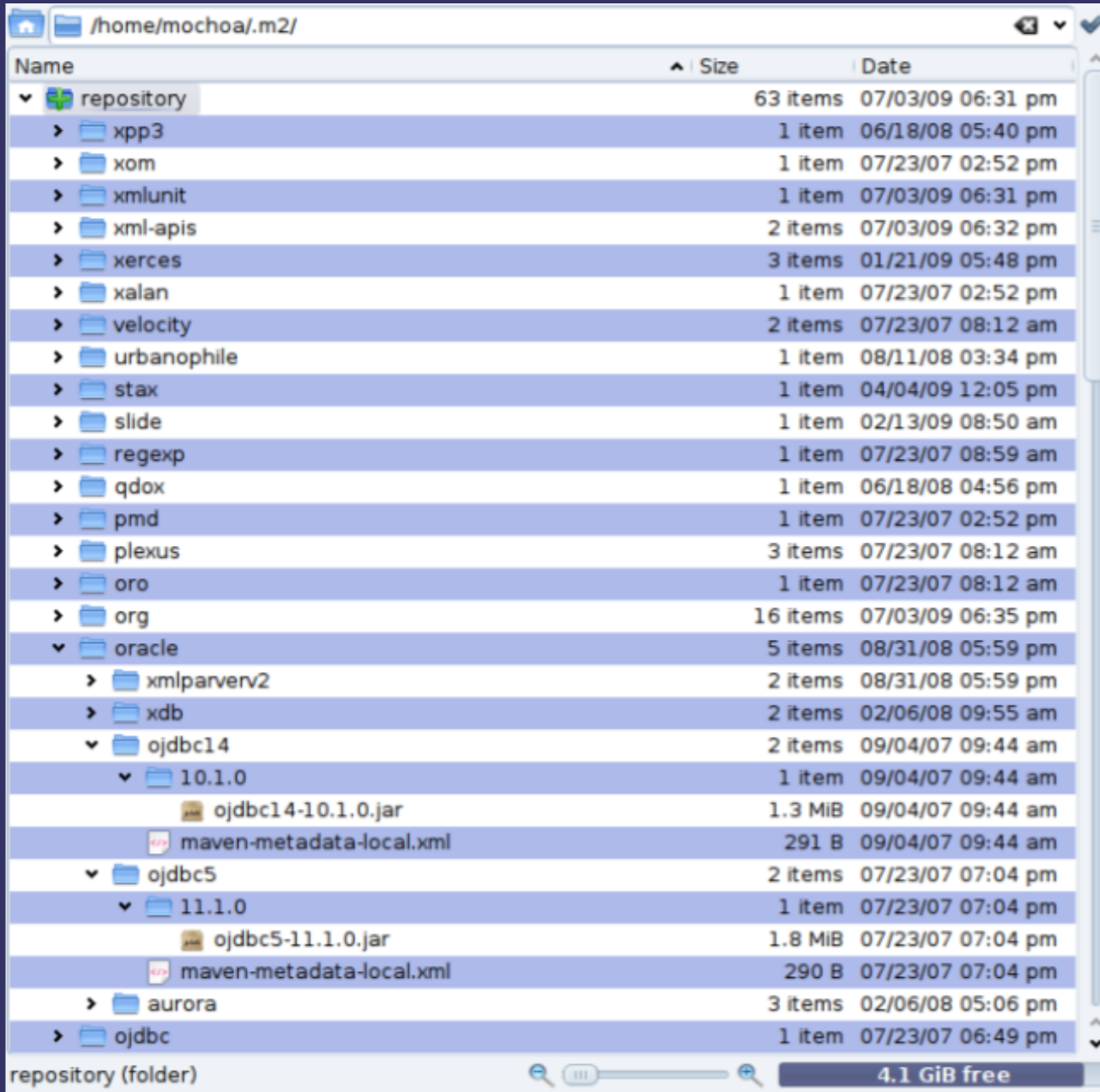


Maven's project example

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
<modelVersion>4.0.0</modelVersion>
<groupId>com.noelios.restlet.test</groupId>
<artifactId>com.noelios.restlet.test.xdb</artifactId>
<packaging>jar</packaging>
<version>1.2-SNAPSHOT</version>
<name>com.noelios.restlet.test.xdb</name>
<url>http://maven.restlet.org/</url>
<repositories>.....</repositories>
<dependencies>
<dependency>
<groupId>oracle</groupId>
<artifactId>ojdbc5</artifactId>
<version>11.1.0</version>
</dependency>....
</dependencies>
<build>
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-compiler-plugin</artifactId>
<configuration>
<source>1.5</source>
<target>1.5</target>
<debug>>true</debug>
</configuration>
</plugin>...
</plugins>
</build>...
</project>
```



Maven's local repository



Name	Size	Date
repository	63 items	07/03/09 06:31 pm
> xpp3	1 item	06/18/08 05:40 pm
> xom	1 item	07/23/07 02:52 pm
> xmlunit	1 item	07/03/09 06:31 pm
> xml-apis	2 items	07/03/09 06:32 pm
> xerces	3 items	01/21/09 05:48 pm
> xalan	1 item	07/23/07 02:52 pm
> velocity	2 items	07/23/07 08:12 am
> urbanophile	1 item	08/11/08 03:34 pm
> stax	1 item	04/04/09 12:05 pm
> slide	1 item	02/13/09 08:50 am
> regexp	1 item	07/23/07 08:59 am
> qdox	1 item	06/18/08 04:56 pm
> pmd	1 item	07/23/07 02:52 pm
> plexus	3 items	07/23/07 08:12 am
> oro	1 item	07/23/07 08:12 am
> org	16 items	07/03/09 06:35 pm
> oracle	5 items	08/31/08 05:59 pm
> xmlparverv2	2 items	08/31/08 05:59 pm
> xdb	2 items	02/06/08 09:55 am
> ojdbc14	2 items	09/04/07 09:44 am
> 10.1.0	1 item	09/04/07 09:44 am
ojdbc14-10.1.0.jar	1.3 MiB	09/04/07 09:44 am
maven-metadata-local.xml	291 B	09/04/07 09:44 am
> ojdbc5	2 items	07/23/07 07:04 pm
> 11.1.0	1 item	07/23/07 07:04 pm
ojdbc5-11.1.0.jar	1.8 MiB	07/23/07 07:04 pm
maven-metadata-local.xml	290 B	07/23/07 07:04 pm
> aurora	3 items	02/06/08 05:06 pm
> ojdbc	1 item	07/23/07 06:49 pm



Testing, testing, testing...

- Include testing from the early stage of the project
- Do client side test
- Do server side test
 - PL/SQL code
 - Server side (requires code installed at server side):
 - SQLDeveloper
 - utPLSQL
 - PLUTO
 - Client side (no server side requirements)
 - SQLUnit (can be integrated with continuous integration servers)
 - Ad-hoc (SQL files executed using SQL*Plus)
 - Java Code
 - JUnit
 - Client Side
 - Server Side
 - Using real server side JDBC drivers
 - Closed Security Manager



Server side JUnit

- Installing JUnit

```
<target name="load-extra-libs" depends="get-junit-jar,get-codec-jar">  
<loadjava-with-public-synonym  
login="${db.usr}/${db.pwd}@${db.str}"  
file="lib/junit-3.8.2.jar"/>  
<loadjava-with-public-synonym  
login="${db.usr}/${db.pwd}@${db.str}"  
file="lib/commons-codec-1.4.jar"/>  
</target>
```

- Permission to use JUnit at server side

```
dbms_java.grant_permission('SCOTT','SYS:java.io.FilePermission', '/junit.properties', 'read' );  
dbms_java.grant_permission('SCOTT','SYS:java.lang.RuntimePermission', 'getClassLoader', " ");  
dbms_java.grant_permission('SCOTT','SYS:java.lang.RuntimePermission', 'accessDeclaredMembers', " ");
```

- Runner classes:

- junit.runner.OJVMTestSuiteLoader
- junit.textui.OJVMRunner
- junit.textui.XMLPrinter



Server side JUnit, running tests

- Define regression tests

```
create table lucene_regression_tests (  
method_or_class CHAR(1),  
class_name VARCHAR2(400),  
elapsed_time NUMBER(6,3),  
result_status VARCHAR2(32),  
result_count NUMBER,  
failure_count NUMBER,  
error_count NUMBER,  
error_msg CLOB  
)  
insert into lucene_regression_tests values ('c','org.apache.lucene.store.TestWindowsMMap',0.0,'-',0,0,0,"");
```

- Running the test

```
declare  
result XMLType;  
begin  
for c in (select * from lucene_regression_tests  
where class_name like 'org.apache.lucene.%'  
order by class_name) loop  
result := XMLType(OJVMRunnerAsString(c.method_or_class,c.class_name));  
commit;  
update lucene_regression_tests  
set elapsed_time = result.extract('/junitreport/time/text()').getNumberVal(),  
result_status = result.extract('/junitreport/result/@status').getStringVal(),  
result_count = result.extract('/junitreport/result/@run_count').getNumberVal(),  
failure_count = result.extract('/junitreport/result/failures/@failure_count').getNumberVal(),  
error_count = result.extract('/junitreport/result/failures/@error_count').getNumberVal(),  
error_msg = result.extract('/*').getClobVal()  
where class_name = c.class_name;  
commit;  
end loop;  
end;
```



SQLUnit (PL/SQL Testing)

- Define an Ant task

```
<taskdef name="sqlunit" classname="net.sourceforge.sqlunit.ant.SqlunitTask">  
<classpath>  
<fileset dir="bin">  
<include name="jdom-1.0.jar" />  
<include name="log4j-1.2.16.jar" />  
<include name="xercesImpl-2.9.1.jar" />  
<include name="sqlunit-5.0.jar" />  
</fileset>  
</classpath>  
</taskdef>
```

- Calling a test suite

```
<target name="test-odi" description="Run Test on Domain Index functionality"  
depends="init">  
<sqlunit haltOnFailure="false" debug="false" logformat="default"  
logfile="../../build/contrib/odi/test/sqlunit-test-odi.txt">  
<fileset dir="db">  
<include name="test*.xml" />  
</fileset>  
</sqlunit>  
</target>
```



SQLUnit (test suite example)

```
<sqlunit>
<connection extern="db/sqlunit.properties" />
<setup>
<include file="db/schemaSourceSmall.sql" />
</setup>
<test name="Optimize index">
<call>
<stmt>{call LuceneDomainIndex.optimize(?)}</stmt>
<param id="1" type="VARCHAR">SOURCE_SMALL_LIDX</param>
</call>
<result/>
</test>
<test name="Count for type">
<call>
<stmt>{call
declare
cn NUMBER;
begin
select count(*) into cn from test_source_small where lcontains(text,?)>0;
if cn &lt; 60 then
raise_application_error(-20101, 'Expected less than 60 rows found: ' || cn);
end if;
end
}</stmt>
<param id="1" type="VARCHAR" inout="in">type</param>
</call>
<result/>
</test>
....
<teardown>
<sql>
<stmt>drop table test_source_small</stmt>
</sql>
</teardown>
</sqlunit>
```




Continuous integration

- Agile development requires short cycles
- Typically an Integration server talks with the SCM server and does daily a compile, test, build process
- Using Hudson CI
 - Easy to execute: `java -jar hudson.war`
 - Easy to configure using a web browser
 - User friendly
 - Multiple executions options
 - Jobs triggering
 - Multiple Notifications options
 - Auto testing (Nunit, JUnit, Maven)
 - Extensible +150 plugins



Hudson examples



Hudson search

Hudson

- New Job
- Manage Hudson
- People
- Build History

Build Queue
No builds in the queue.

Build Executor Status

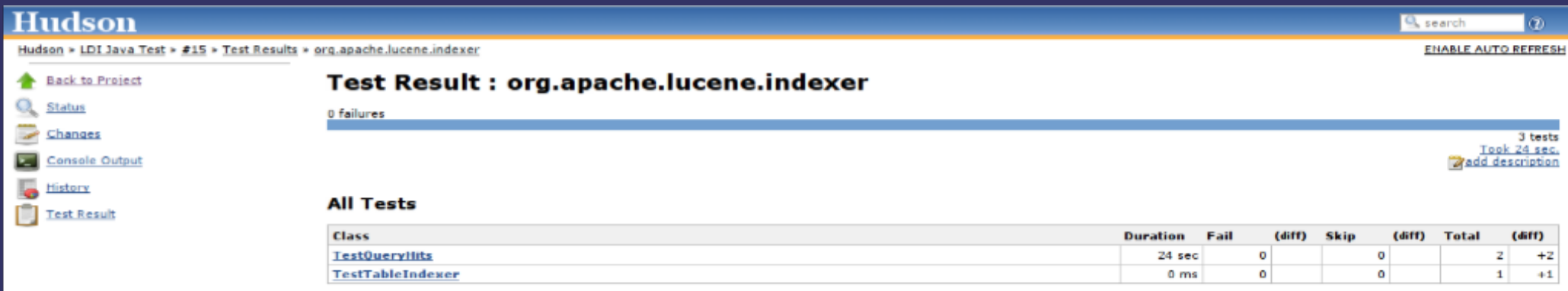
#	Status
1	Building LDI SQLUnit test #7
2	Idle

All +

S	W	Job	Last Success	Last Failure	Last Duration	
		LDI Build	11 days (#2)	N/A	13 sec	
		LDI Java Test	7 days 16 hr (#15)	N/A	8 min 9 sec	
		LDI SQLUnit test	1 day 12 hr (#6)	N/A	43 sec	
		Test LDI	7 days 15 hr (#12)	N/A	2 min 30 sec	

Icon:

Legend for all for failures for just latest builds



Hudson search

Hudson > [LDI Java Test](#) > #15 > [Test Results](#) > [org.apache.lucene.indexer](#)

[Back to Project](#)

- Status
- Changes
- Console Output
- History
- Test Result

Test Result : org.apache.lucene.indexer

0 failures

3 tests
Took 29 sec.
[add description](#)

All Tests

Class	Duration	Fail	(diff)	Skip	(diff)	Total	(diff)
TestQueryHits	24 sec	0		0		2	+2
TestTableIndexer	0 ms	0		0		1	+1



Hudson examples

Hudson search ENABLE AUTO REFRESH add description

[New Job](#)
[Manage Hudson](#)
[People](#)
[Build History](#)

Build Queue
No builds in the queue.

Build Executor Status

#	Status
1	Idle
2	Idle

All +

S	W	Job	Last Success	Last Failure	Last Duration	
		LDI Build	11 days (#2)	N/A	13 sec	
		LDI Java Test	7 min 57 sec (#17)	1 min 34 sec (#18)	4 min 15 sec	
		LDI SQLUnit test	14 min (#7)	N/A	50 sec	
		Test LDI	7 days 15 hr (#12)	N/A	2 min 30 sec	

Icon: [S](#) [M](#) [L](#)

[Legend](#) [for all](#) [for failures](#) [for just latest builds](#)

Hudson search ENABLE AUTO REFRESH edit description

Hudson > LDI Java Test

[Back to Dashboard](#)
[Status](#)
[Changes](#)
[Workspace](#)
[Build Now](#)
[Delete Project](#)
[Configure](#)

Build History (trend) [for all](#) [for failures](#)

- #18 [Sep 2, 2010 9:30:51 AM](#)
- #17 [Sep 2, 2010 9:29:28 AM](#)

Project LDI Java Test

Performs all Java Test on LDI project

- [Workspace](#)
- [Recent Changes](#)
- [Latest Test Result \(0 failures / +0\)](#)

Permalinks

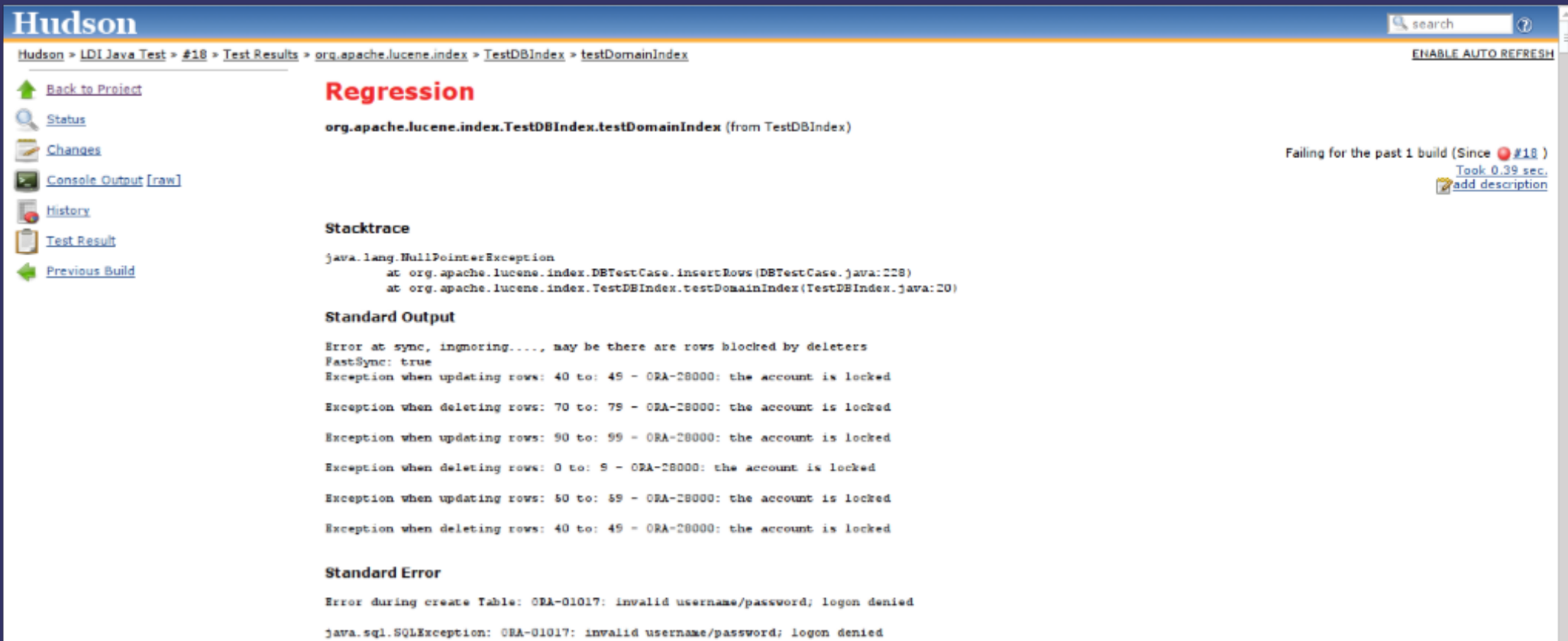
- [Last build \(#18\), 2 min 11 sec ago](#)
- [Last stable build \(#17\), 0 min 34 sec ago](#)
- [Last successful build \(#17\), 0 min 34 sec ago](#)
- [Last failed build \(#18\), 2 min 11 sec ago](#)
- [Last unsuccessful build \(#18\), 2 min 11 sec ago](#)

Test Result Trend

(just show failures) [enlarge](#)



Hudson examples



The screenshot displays the Hudson web interface for a failed build. The breadcrumb trail is: Hudson > LDI Java Test > #18 > Test Results > org.apache.lucene.index > TestDBIndex > testDomainIndex. The page title is "Regression" in red. The build name is "org.apache.lucene.index.TestDBIndex.testDomainIndex (from TestDBIndex)". A status indicator shows "Failing for the past 1 build (Since #18)" with a red circle icon, and it took 0.39 seconds. There is a link to "add description".

Stacktrace

```
java.lang.NullPointerException
    at org.apache.lucene.index.DBTestCase.insertRows(DBTestCase.java:228)
    at org.apache.lucene.index.TestDBIndex.testDomainIndex(TestDBIndex.java:20)
```

Standard Output

```
Error at sync, ignoring...., may be there are rows blocked by deleters
FastSync: true
Exception when updating rows: 40 to: 49 - ORA-28000: the account is locked
Exception when deleting rows: 70 to: 79 - ORA-28000: the account is locked
Exception when updating rows: 90 to: 99 - ORA-28000: the account is locked
Exception when deleting rows: 0 to: 9 - ORA-28000: the account is locked
Exception when updating rows: 50 to: 59 - ORA-28000: the account is locked
Exception when deleting rows: 40 to: 49 - ORA-28000: the account is locked
```

Standard Error

```
Error during create Table: ORA-01017: invalid username/password; logon denied
java.sql.SQLException: ORA-01017: invalid username/password; logon denied
```



Debugging

- Client side debugging
- Server side debugging
 - Attaching to JPDA

```
java -client -agentlib:jdwp=transport=dt_socket,server=y,address=<port> <other Java options>  
YourMainClass <program arguments>
```

- Listening for JPDA

Remote	
Protocol	Listen for JPDA
Port	4000
Timeout	0
Local address	

```
SQL> grant DEBUG CONNECT SESSION, DEBUG ANY PROCEDURE, JAVADEBUGPRIV to SCOTT;
```

```
preparedstatement = this.conn  
.prepareCall("{ call dbms_debug_jdwp.connect_tcp(?,?) }");  
preparedstatement.setString(1, "localhost");  
preparedstatement.setInt(2, 4000);  
preparedstatement.execute();
```

```
preparedstatement = this.conn  
.prepareCall("{ call dbms_debug_jdwp.disconnect }");  
preparedstatement.execute();
```



Logging

- Two options are for server side Java Code
 - JDK Java Util Logging API
 - Simple
 - Included by default into the OJVM
 - No extra jars
 - Only one grant to use by regular users

```
dbms_java.grant_permission('SCOTT','SYS:java.util.logging.LoggingPermission', 'control', " ");
```

Apache Log4J

- Powerful
- Requires log4j.jar uploaded

```
loadjava -u JLIB_SCHEMA/JLIB_SCHEMA -r -v -s -g PUBLIC log4j-1.8.2.jar  
loadjava -u JLIB_SCHEMA/JLIB_SCHEMA -r -v -g PUBLIC log4j.properties
```

- Minimum grant required

```
dbms_java.grant_permission( 'LOGROLE', 'SYS:java.lang.RuntimePermission', 'getClassLoader', " );  
dbms_java.grant_permission( 'LOGROLE', 'SYS:java.io.FilePermission', 'log/a.log', 'write' );  
dbms_java.grant_permission( 'LOGROLE', 'SYS:java.io.FilePermission', 'log/b.log', 'write' );
```



Log4J complex example

- Using Log4J to get unified logging
 - Components:
 - Server Side logging
 - Middle tier application logging
 - Unix syslogd

/etc/sysconfig/syslogd

```
# See syslogd(8) for more details
SYSLOGD_OPTIONS="-r"
```

/etc/rsyslog.conf

```
# $AllowedSender - specifies which remote systems are allowed to send syslog messages to rsyslog
# -----
$AllowedSender UDP, 127.0.0.1, 192.0.2.0/24
```



Log4J complex example

grants

```
dbms_java.grant_permission( 'CMSUSERROLE', 'SYS:java.net.SocketPermission', 'localhost', 'resolve' );
dbms_java.grant_permission( 'CMSUSERROLE', 'SYS:java.net.SocketPermission', 'localhost:1024-', 'listen,resolve' );
dbms_java.grant_permission( 'CMSUSERROLE', 'SYS:java.net.SocketPermission', 'loghost:514', 'connect,resolve' );
```

log4j.properties:

```
# A1 should be a ConsoleAppender.
log4j.appender.A1=org.apache.log4j.ConsoleAppender
# A1's layout is a PatternLayout
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.layout.ConversionPattern=%-4r [%t] %-5p %c %x - %m%n

# Set options for appender named "A2".
# Appender "A2" will be a SyslogAppender
log4j.appender.A2=org.apache.log4j.net.SyslogAppender
log4j.appender.A2.SyslogHost=localhost

# Only warnings or above will be logged for the logger "SECURITY.access".
# Output will go to A1.
log4j.logger.SECURITY.access=WARN
# uncomment this if you want DBPrism debugging
log4j.logger.com.prism=DEBUG,A2
log4j.additivity.com.prism=false
```



Release management

- Always use release number on jar-files
 - Follow Maven default three-digit convention (e.g. 2.0.44)
- loadjava tool extract all .class from jar and the relation with the version is missing
- PL/SQL procedures and packages have no release number inside the RDBMS.
- Try to use a dev. environment with same:
 - Architecture
 - OS version
 - RDBMS versionas the production environment



Backup, Restore

Upgrade, Downgrade

- Traditional backup-restore are not useful with server side code
 - A full backup will not work to downgrade an application
 - Storing only code in one schema can be an option to use traditional backup-restore tools
- XMLDB repository objects requires a custom backup
- A set of classes uploaded with *loadjava* can be downgraded using:
 - *dropjava myapp-jar-1.1.0.jar*
 - *loadjava myapp-jar-1.0.0.jar*but you need a repository like Maven to keep track all jar version-ed
- If you need to know which version of PLSQL code is currently used you could use SCM keywords such as

■ \$Revision\$, \$Log\$, \$Author\$, \$Date\$



Summary

- From the developer perspective ALM requires us
 - Choose the right tool for the precise job
 - Use SCM to keep the changes of everything
 - Develop, test and deploy using command line tools
 - Follow release number for libraries
 - Integrate logging to follow the application running
 - Go Agile using CI
 - Automatize all test

Remember:

- Write a test case is not a waste of time, a test is a time-saving machine



Q & A



Links

[Wikipedia information about ALM](#)

[Oracle JDeveloper](#)

[Oracle Team Productivity Center](#)

[Apache Ant](#)

[Apache Maven](#)

[JUnit](#)

[SQLUnit](#)

[Hudson CI](#)

[Log4J](#)

[Lucene Domain Index](#)

[Temperies](#)



Thank You

