Information Management

IBM

Enabling Applications from Oracle to DB2 9.7 the Easy Way

September 15, 2010

Agenda

- Why should I?
- Why haven't I done it?
- What's changed in the new DB2®?
- How do I do it?
- What's suitable?

Why Should I move to DB2?

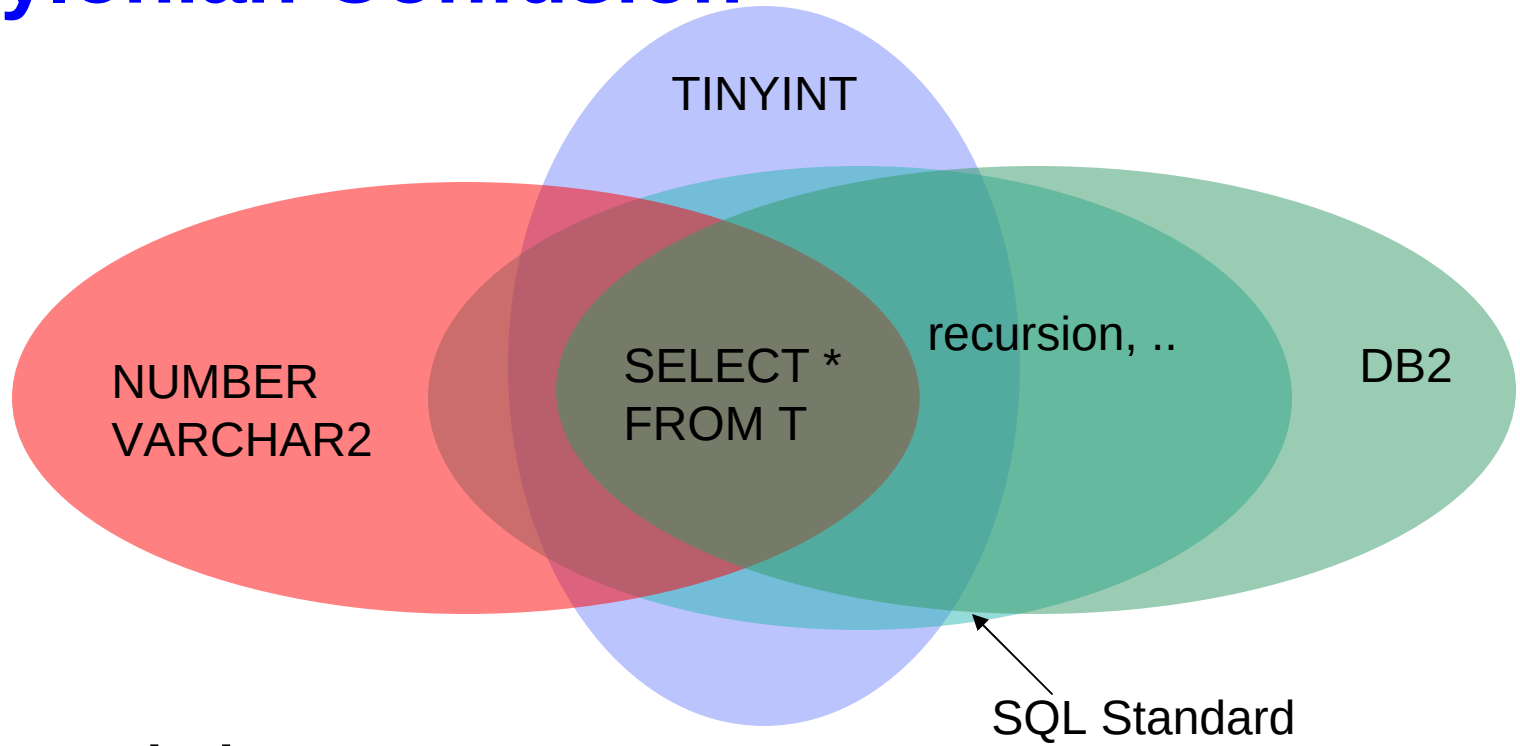
- **I'm a customer and**
  - My current DBMS doesn't scale to my needs
  - I have no leverage when negotiating prices with one vendor
  - I think DB2 provides better value for my money

- **I'm a vendor and**
  - I want to expand my customer base
  - I want to get enabled on a "blue stack"
  - My DBMS vendor is also my competitor
  - I think DB2 provides better value for my money

Why have I not moved to DB2 yet?

- **I'm a customer and I think**
  - It may be risky to move my critical apps
  - The return on investment may be too far out
- **I'm a vendor and I think**
  - Porting to and maintaining another platform may be too expensive
  - The return on investment may be too far out

Babylonian Confusion



- **Your choices**
  - Pick one dialect and stick (or be stuck) with it.
  - For ISVs: Pick two or all and have dual/triple maintenance
  - Pick intersect only and escape to Java®, Hibernate etc…
- **You lose either way!**

What did DB2 do to bridge the gap pre 9.7?

Traditionally

1. **Implement SQL Standard features**
2. **Extend SQL standard with popular features**
3. **Map proprietary SQL using Migration Tool Kit (MTK)**

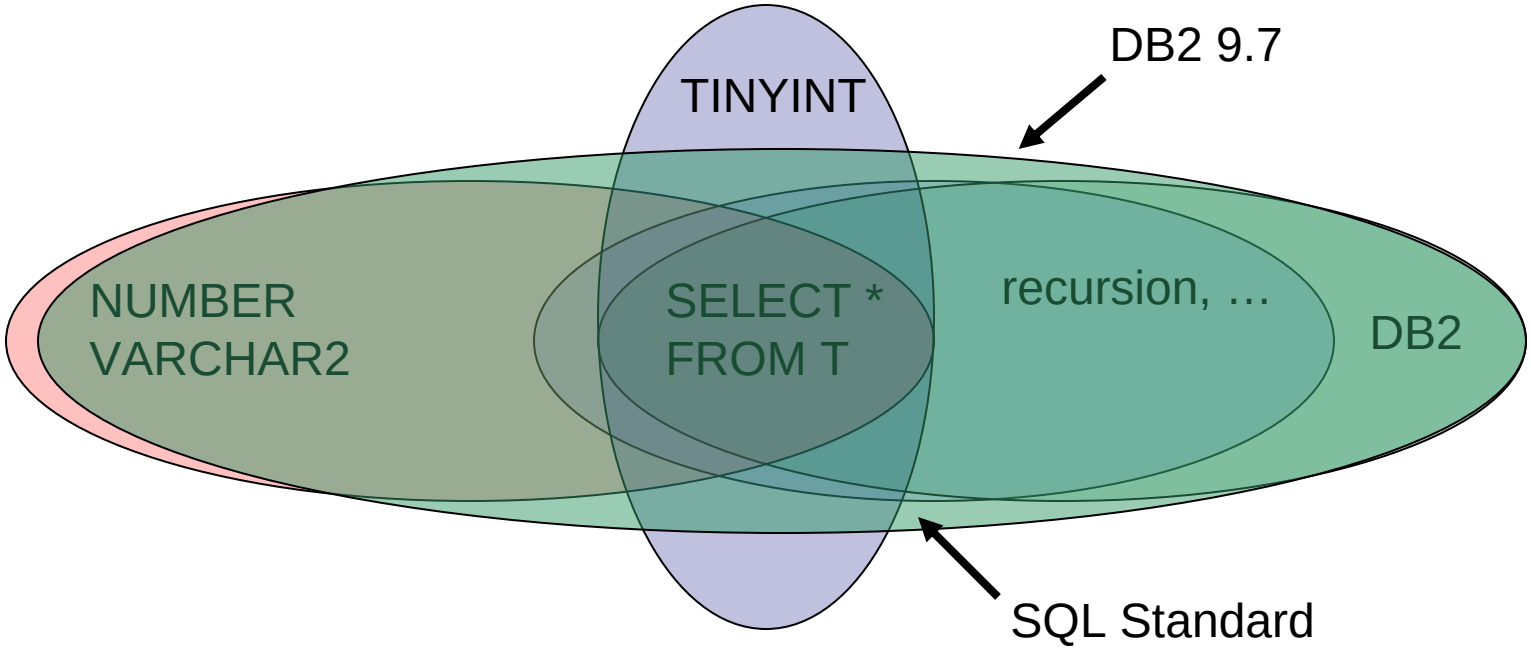
Problem

- **MTK can only help with data base objects and DDL e.g. procedures, triggers, functions, tables**
- **..but SQL is littered throughout the application code!**
- **Impedance mismatch can cause performance issues**

A new approach

- Syntax toleration
  Where no semantic conflicts exist
- Add infrastructure
  Eliminate impedance mismatch
- Compatibility modes
  Overcome clashes between DB2 and other SQL dialects
- Out of the box compatibility for your app
  - **Client side**
  - **Server side**

Bridging the gap in DB2 9.7



Early feedback

- **More than 300 participants in the Early Access**
  - Some participants with multiple applications
  - Applications that currently use Oracle
  - Many participants enabled their applications to use DB2 in a few days
  - 7.5M LOC PL/SQL for biggest beta customer
- **Detailed analysis reveals that:**
  - On average, participants experience **90% less effort** to enable DB2
  - Some participants need only change the connect string!!!

A new reality

Oracle		DB2
Concurrency Control	→	Native support
SQL dialect	→	Native support
PL/SQL	→	Native support
PL/SQL Packages	→	Native support
Built-in package library	→	Native support
JDBC extensions	→	Native support
OCI	→	Native support
Forms	→	Through *realease*
SQL*Plus Scripts	→	Native support
RAC	→	pureScale

Changes are the exception. Not the rule.

THIS IS WHY WE CALL IT ENABLEMENT AND NOT PORT !

Concurrency Control

- Pre DB2 9.7
- DB2 9.7
- Architecture

Concurrency and DB2 9.7

- **Oracle default**
  - Statement level snapshot

blocks	Reader	Writer
Reader	No	No
Writer	No	Yes

- **DB2 default prior to 9.7**
  - Cursor stability

blocks	Reader	Writer
Reader	No	Maybe
Writer	Yes	Yes

Enabling Oracle application to DB2 required significant effort to re-order table access to avoid deadlocks

- **DB2 default with DB2 9.7**
  - Currently Committed

blocks	Reader	Writer
Reader	No	No
Writer	No	Yes

Concurrency Control in DB2 9.7

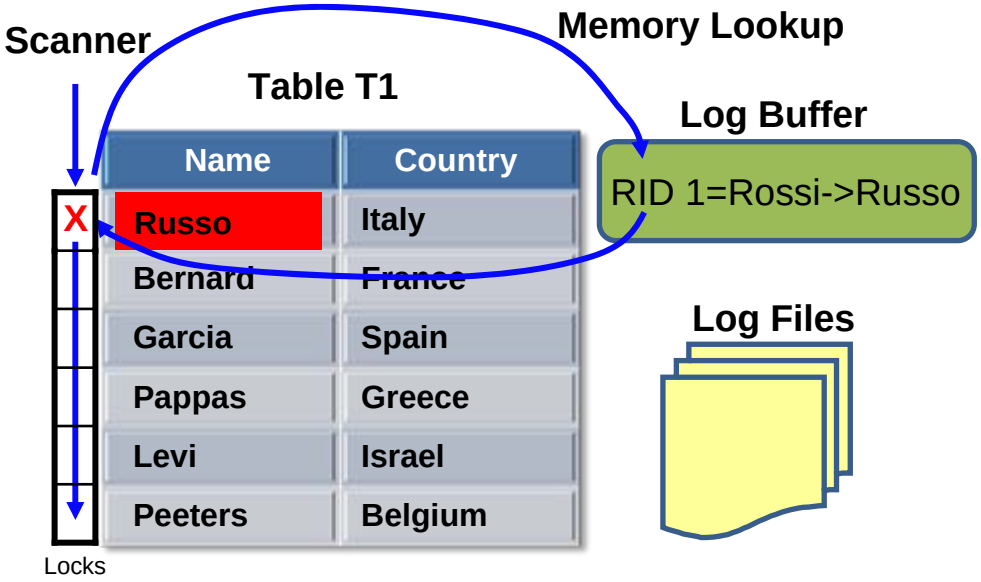
- Log based
  - No management overhead
  - No performance overhead
  - No wasted memory/storage (no undo tablespace)

User 1:

```
update T1 set name = 'Russo'
where country='Italy'
```

User 2:

```
select * from T1
```



Currently Committed Advantage

- **Only incur added processing when a reader and writer are working on the same row**
- **No added overhead for a “just in case” collision**
  - With Oracle past images are stored in the undo table space just in case there is a collision
- **DB2 uses existing log infrastructure to retrieve currently committed data in flight**
  - Better performance
  - Lower overhead
  - Simplified management

Oracle SQL Dialect support

- Data types
- Casting
- Function library
- SQL constructs

New types in DB2 9.7

Type	Comment
NUMBER	Exploits P6 hardware accelerated DECFLOAT
VARCHAR2	‘’ is NULL, trailing blank sensitive collation
TIMESTAMP(n)	0 (date + time) <= N <= 12 (date + time + picoseconds)
“DATE”	Year to seconds, SYSDATE
BOOLEAN	In procedural code
Hash tables	Associative “INDEX BY” arrays in procedural code
VARRAY	Regular arrays in procedural code
Row Type	In procedural code, ARRAY, hash tables
Cursor type	Allows passing, and predefining of “ref-cursor”

Weak Typing in DB2

- **DB2 has been following strict SQL standard typing rules**
  - The trend is towards weak typing (PERL, RUBY, PHP, …)
- **New behavior**
  - Implicit conversion between numerics and strings on:
    - `SET salary = ‘52000’`
    - `WHERE salary > ‘52000’`
    - `‘salary: ‘ || 52000`
  - Substitutability between DATE and TIMESTAMP
    - Save storage for a DATE column by using SQL Standard DATE without changing an Oracle “DATE” based application
  - More aggressive usage of NULL and parameter markers
    - Function invocation with untyped parameter markers and NULL `values foo(?, NULL)`

New functions in DB2 9.7

Function	
Conversion and Formatting	**TO_CHAR, TO_DATE, TO_TIMESTAMP, TO_NUMBER, TO_CLOB**
Date arithmetic	**EXTRACT, ADD_MONTHS, …**
String manipulation	**INITCAP, RPAD, LPAD, INSTR, SUBSTRB, …**
Misc	**DECODE, NVL, LEAST, GREATEST, BITAND**

New SQL in DB2

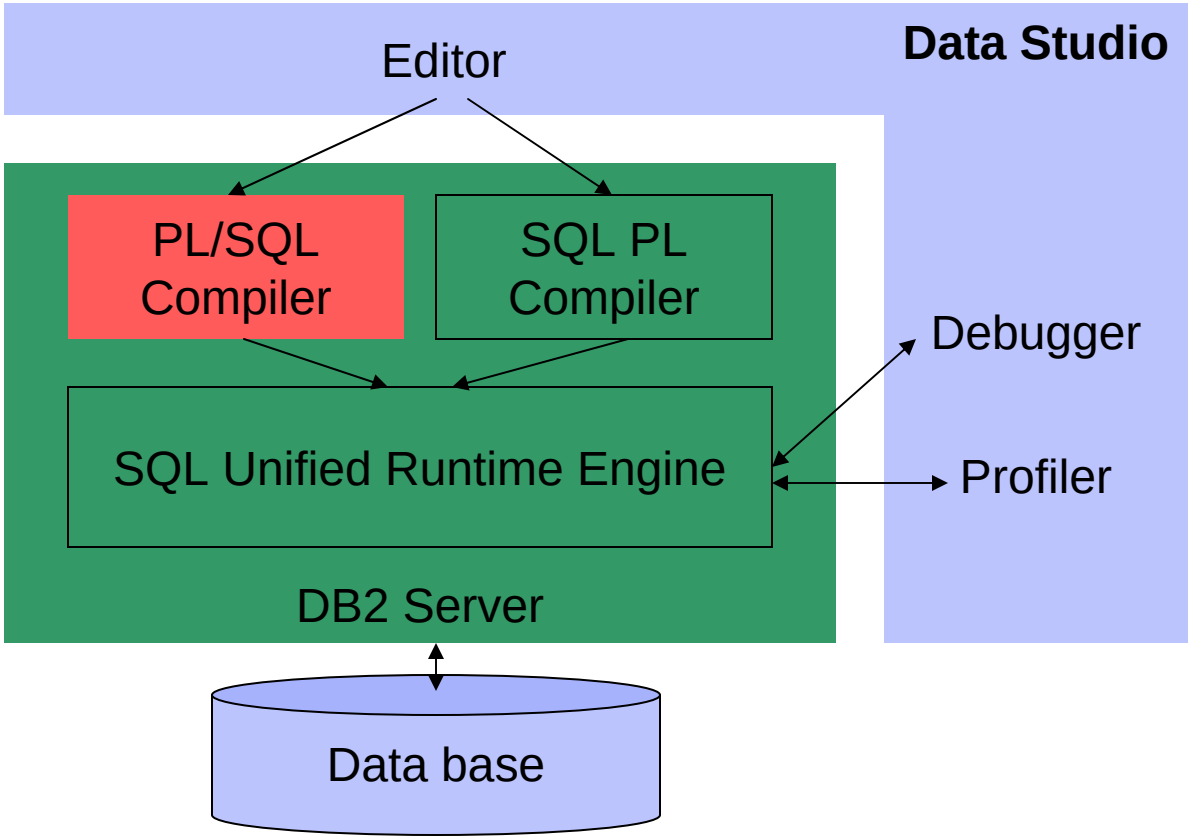
Feature	
CONNECT BY	Tree walk recursion, includes helper functions (9.5)
(+)-join	Old style OUTER JOIN syntax (9.5)
DUAL	Dummy one row, one column table (9.5)
ROWNUM	Pseudo column for ROWNUMBER() (9.5)
SELECT INTO FOR UPDATE	Get update lock without a cursor
ROWID	Pseudo column syntax for RID_BIT()
AUTONOMOUS TX	Independent transactions for auditing
TRUNCATE table	Quickly delete a table without logging
Public synonym	For table, sequence, module
CREATEd temp table	Catalogued, private temporary table
…and more…	

PL/SQL in DB2 9.7

- Architecture
- Debugging etc…
- Depth and breadth of support

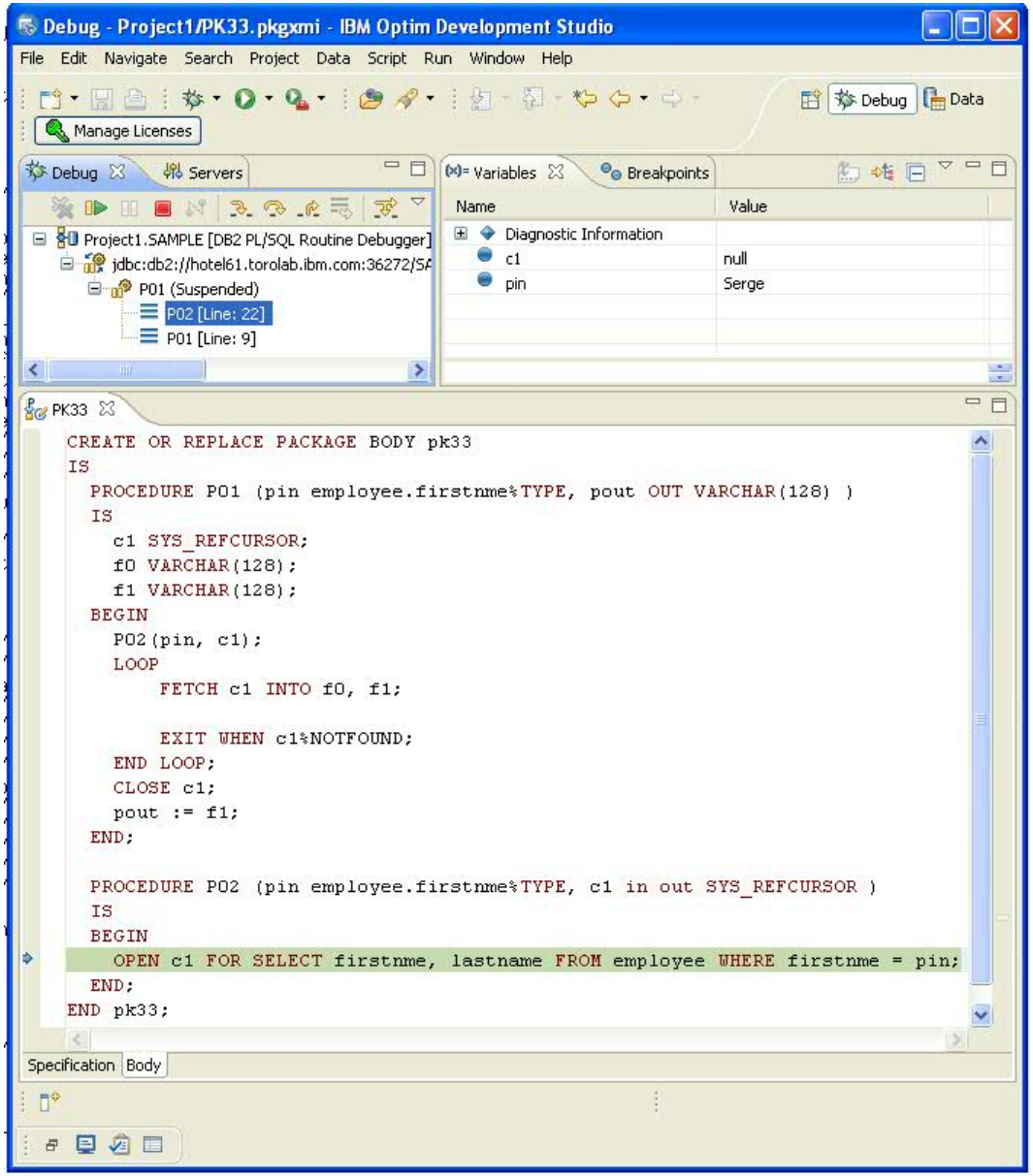
Native PL/SQL support in DB2

- Source level debugging and profiling
- Built-in
- Proven



- QA exit criteria: PL/SQL execution at >95% of SQL PL

Optim Development Studio PL/SQL Debugger



Supported PL/SQL constructs

Function	
All logic	IF, WHILE, :=, etc..
EXCEPTION	Try/catch handling
Constant variables	Variables that cannot be set
FOR over range	Step through numbers
over SELECT	Step through result set of query
over cursor	Step through result set of cursor
User Defined Exceptions	Define conditions with or without SQLCODEs
%TYPE	Anchored scalar data types
%ROWTYPE	Anchored row types
#PRAGMA AUTONOMOUS	Private commit scope
FORALL/BULK COLLECT	New in FP1
Conditional compilation	New in FP1

PL/SQL in DB2

Area	
Anonymous block	**Server side scripting** **New also in SQL PL dialect**
Scalar function	**Including OUT parameters (FP1)**
Procedure	**Including DEFAULT,** **and named parameters**
Package	**Known as MODULE in SQL PL**
Trigger	**Row level, Before or After**

PL/SQL Packages in DB2

Feature	
CREATE PACKAGE	Define prototypes and public objects
CREATE PACKAGE BODY	Define content and private objects
Replace package body	Replace body without loosing prototypes or public objects
PKG [BODY] VARIABLE	Public/private variables
CURSOR	Public/private cursors
TYPE	Public/private types
EXCEPTION	User defined exceptions
FUNCTION	
PROCEDURE	User defined exceptions
SYNONYM ON PACKAGE	Public synonyms

Built-in package library

Library	
DBMS_OUTPUT	“print debugging” and simple reporting
UTL_FILE	Server side I/O API
DBMS_ALERT	Cross session semaphoring
DBMS_PIPE	Cross session data pipe
DBMS_JOB	Job scheduler
DBMS_LOB	Alternate API to DB2 native LOB functions
DBMS_SQL	Alternate API to PREPARE/EXECUTE
DBMS_UTILITY	Misc functions and procedures
UTL_MAIL	Server API to email
UTL_SMTP	Server API to SMTP

Using SQL*Plus scripts in DB2

CLPPlus

Compatible commands

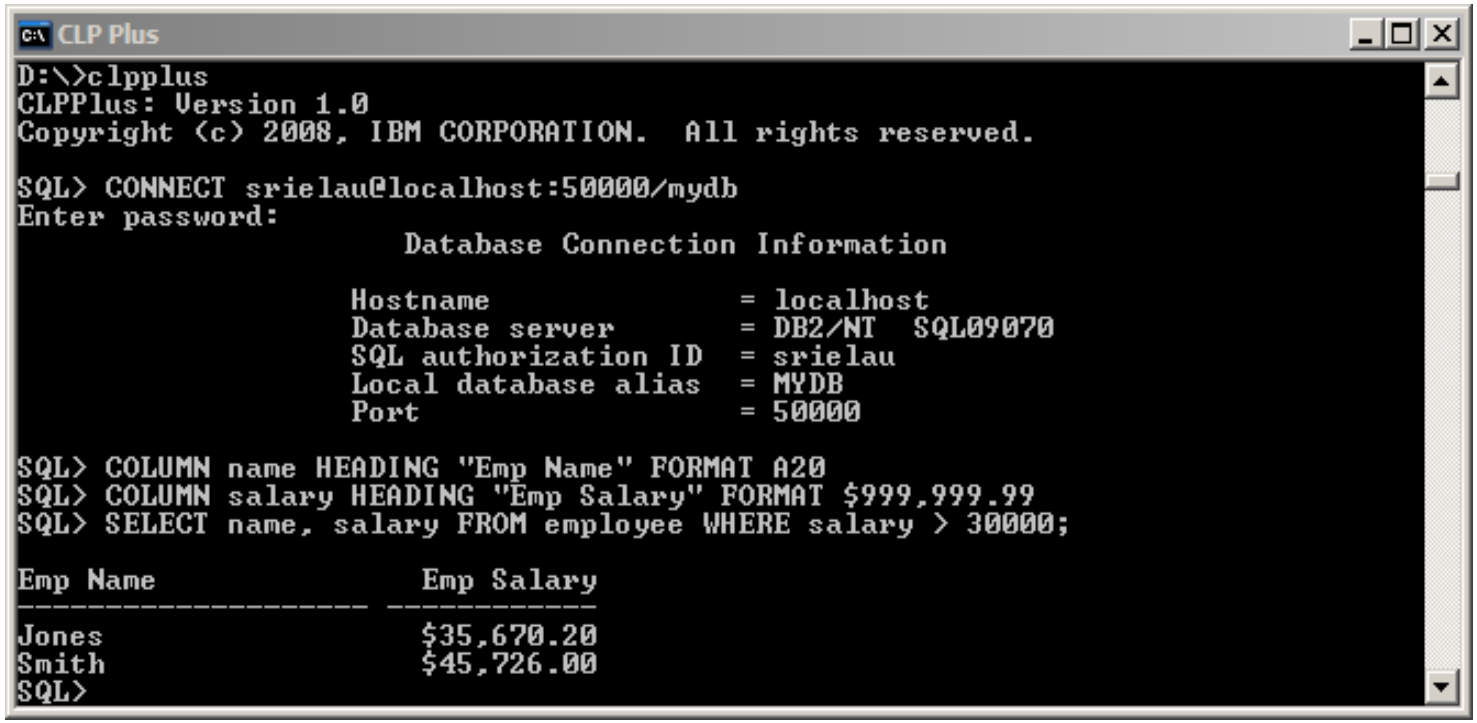
Compatible options

Variable substitution

Column formatting

Simple reporting

Control variables



Ongoing focus

- New in FP1
  - OCI client support
  - Misc. function refinements (SUBSTRB, SUBSTR, TO_CHAR ..)
  - OUT parameters for functions
  - Improved multi partition environment support (DPF)
  - Conditional compilation for increased “one-source” capabilities
  - Improved BOOLEAN support
  - FORALL and BULK COLLECT
- New in FP2
  - WRAPPED routines, triggers and views (IP obfuscation)
  - NCHAR, NVARCHAR2, etc
  - DEFAULT and named parameters in functions
  - NUMBER and BULK COLLECT performance enhancements
- FP3 and DB2 next in development

Steps of Enablement

1. Identify and size-up candidate using the MEET DB2 Tool
2. Enable application using IBM Data Movement Tool
3. Test, Test, Test
4. Parallel production
5. Cut-Over

Best targets for enablement to DB2

- **Cross vendor standardized client APIs**
  - JDBC
  - ODBC
  - .NET
  - OCI
- **Isolated/no dependencies on extended options or features**
  - Spatial
  - Streams (AQ)
  - Text Search
  - …

DB2 9.7 MEET Report Case Study

MEET DB2 Report - IBM Confidential

Migration Enablement Evaluation Tool for DB2
Knowledgebase version 00.20
Knowledgebase for DB2 version 9.7

Send any comments to meetdb2@torolab.ibm.com

Executive Summary

99.0% of statements immediately transferable to IBM DB2.

MEET DB2 has estimated that **99.0% of statements** and **99.2% of objects** are immediately transferable to IBM DB2. The technical report below is provided to detail exact instances and locations of potential issues to simplify the migration process.

99.0% of statements
immediately transferable to IBM DB2

MEET Report

- **Rapid assessment of application**
- **Early confirmation of compatibility**
- **Lists details and source code line number for exceptions**

Object Type	Total Number	Number That Require Attention	Percent That Require Attention
package	0	0	0 %
anonymous	0	0	0 %
table	0	0	0 %
type	0	0	0 %
function	97	0	0 %
procedure	158	2	1 %
Total Objects	255	2	0.8 %
Statements	13292	2	0 %

PL/SQL Summary

Object Type	Total Number	Number That Require Attention	Percent That Require Attention
Triggers	0	0	0 %
Anonymous blocks	8	0	0 %
Functions	31	7	23 %
Packages	6	2	33 %
Procedures	59	27	46 %
Total Objects	104	36	34.6 %
Statements	967	105	10.9 %

DDL Summary

The DDL reports assume that the IBM Data Movement Tool will be used to migrate DDL to DB2.

Statement Type	Total Number	Number That Require Attention	Percent That Require Attention
Create type	0	0	0 %
Create view	3	0	0 %
Create sequence	91	0	0 %
Create table	227	0	0 %
Create index	374	0	0 %
Other DDL	1034	0	0 %
Statements	1729	0	0 %

Click here to show the detailed technical report

Tools – MEET

Feature	Description	#
CREATE DIRECTORY	CREATE DIRECTORY is not supported.	1
SYS_CONTEXT	SYS_CONTEXT function is not supported.	2
V$ VIEWS	V$* or V_$* VIEWS are not supported.	52
REF/SYSREF CURSOR TYPE	Ref and Sysref cursor declarations are supported only in packages.	1
AUTHID CURRENT_USER/DEFINER	AUTHID CURRENT_USER functionality is not supported.	1
DBMS_SQL changes	Some DBMS_SQL procedures have different names in DB2.	7
XMLDOM	XMLDOM functionality is not supported.	5
FUNCTION PARAMETERS WITH DEFAULT	Defaults on Function Parameters are not supported.	1
ALTER SESSION STATEMENTS	Dynamic ALTER SESSION statements are not supported.	2
SYS SCHEMA/DATA DICTIONARY	Sys schema calls and Data Dictionary views are not supported	22
DBMS_LOCK	DBMS_LOCK package is not supported.	8
DBMS_APPLICATION_INFO	DBMS_APPLICATION_INFO package is not supported.	1
SYSTIMESTAMP	SYSTIMESTAMP function is not supported.	2

Tools - MEET

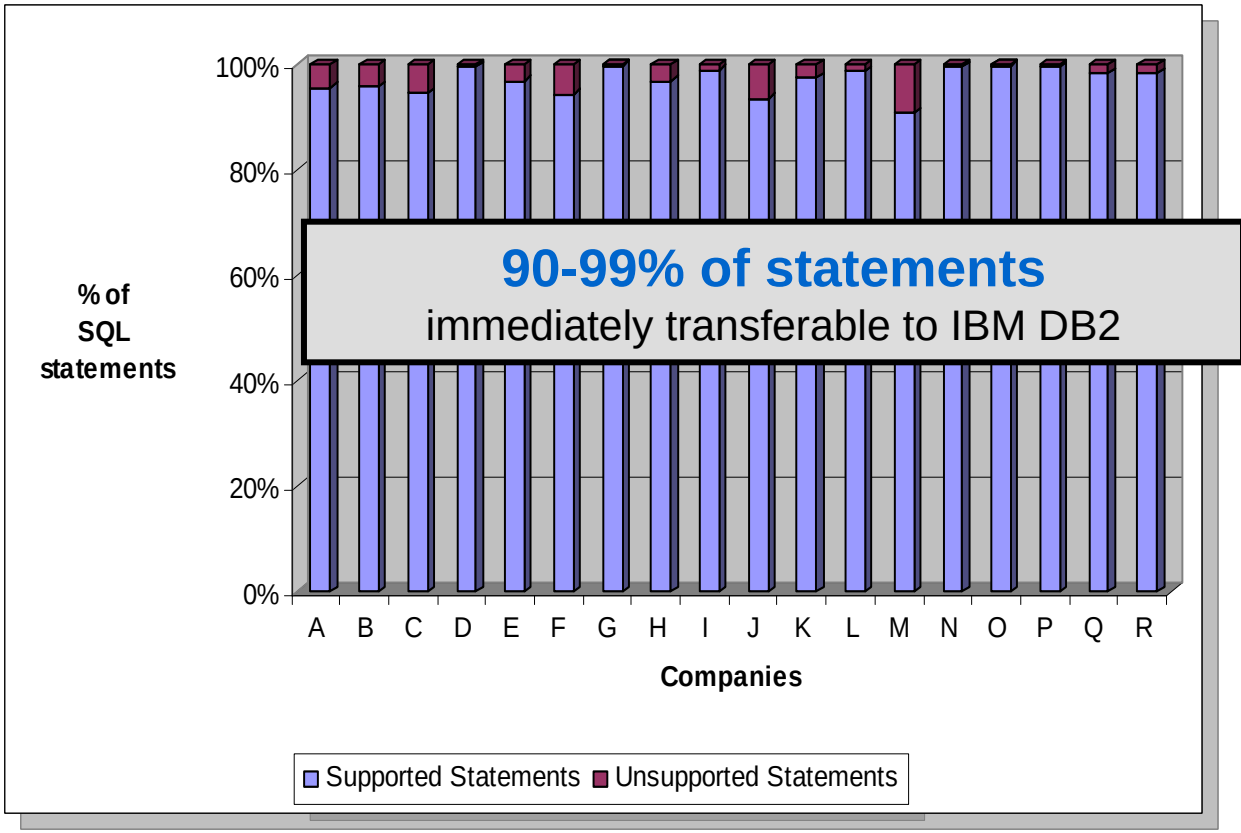
Feature	Description	#
CREATE DIRECTORY	CREATE DIRECTORY is not supported.	1
Line 14932	[procedure] "ORACLE_OCM"."MGMT_CONFIG_UTL".create_replace_dir_obj	execute immediate 'CREATE OR REPLACE DIRECTORY ORACLE_OCM_CONFIG_DIR AS ''' || l_ocm_dir_path || '''';
Solution:	Use UTL_DIR.CREATE_DIRECTORY statement.	
V$ VIEWS	V$* or V_$* VIEWS are not supported.	52
Line 11431	[procedure] "BETRYOUT_DBA"."DBLOG".insertMessage	from v$session
Line 14916	[procedure] "ORACLE_OCM"."MGMT_CONFIG_UTL".create_replace_dir_obj	SELECT platform_id INTO pfid FROM v$database;
Line 14945	[package] "ORACLE_OCM"."MGMT_DB_LL_METRICS"	g_dbID v$database.DBID%TYPE := NULL;
Line 14946	[package] "ORACLE_OCM"."MGMT_DB_LL_METRICS"	g_db_version v$instance.version%TYPE := NULL;
Line 14955	[function] "ORACLE_OCM"."MGMT_DB_LL_METRICS".get_version_category	l_db_version v$instance.version%TYPE;
Line 14956	[function] "ORACLE_OCM"."MGMT_DB_LL_METRICS".get_version_category	l_temp_version v$instance.version%TYPE;

DB2 Early Access Program Study

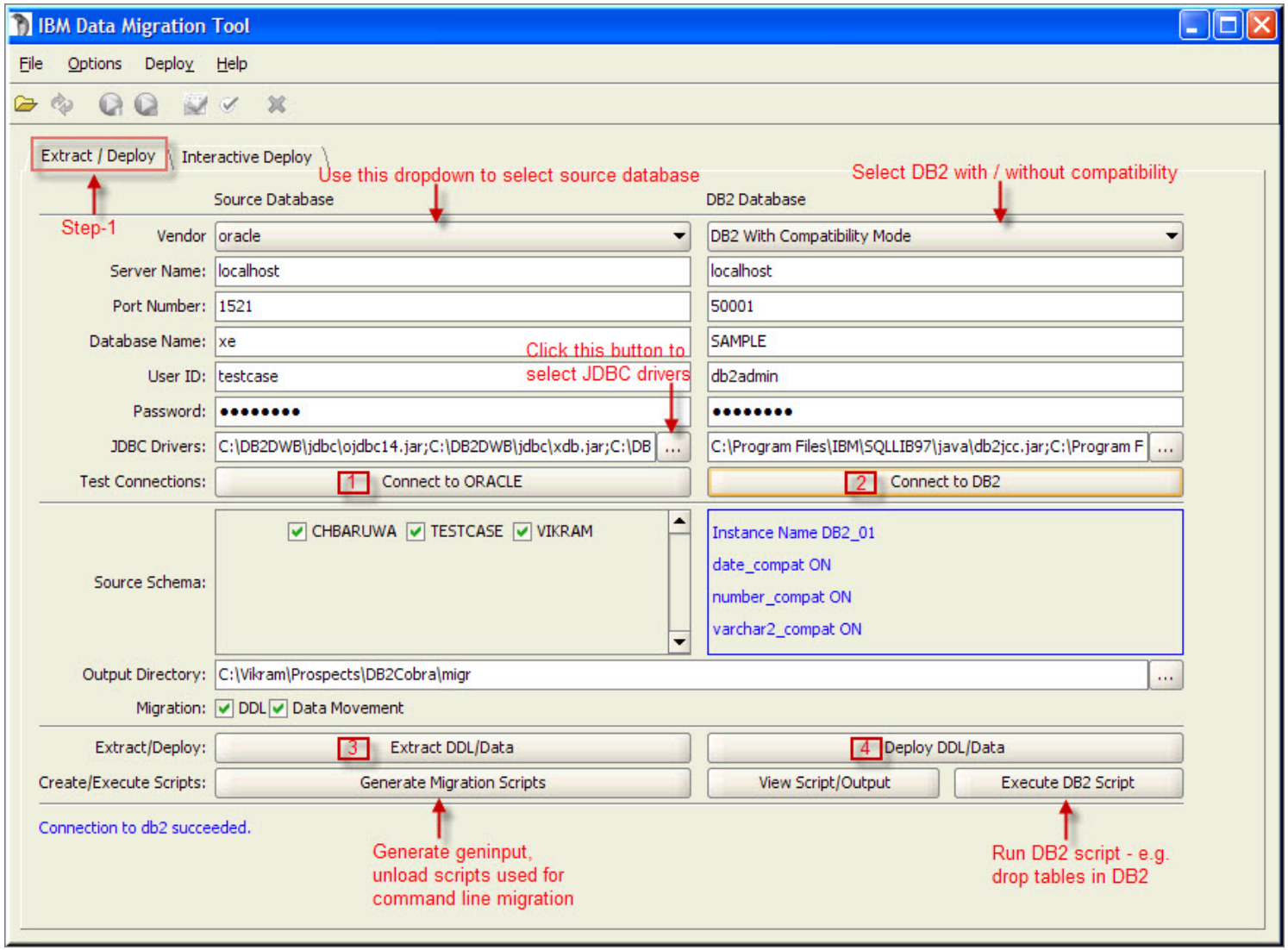
> **"The compatibility level that DB2 9.7 achieved is amazing."**
> *—Masato Kudo, Works Applications*

Code from 18 EAP Participants analyzed

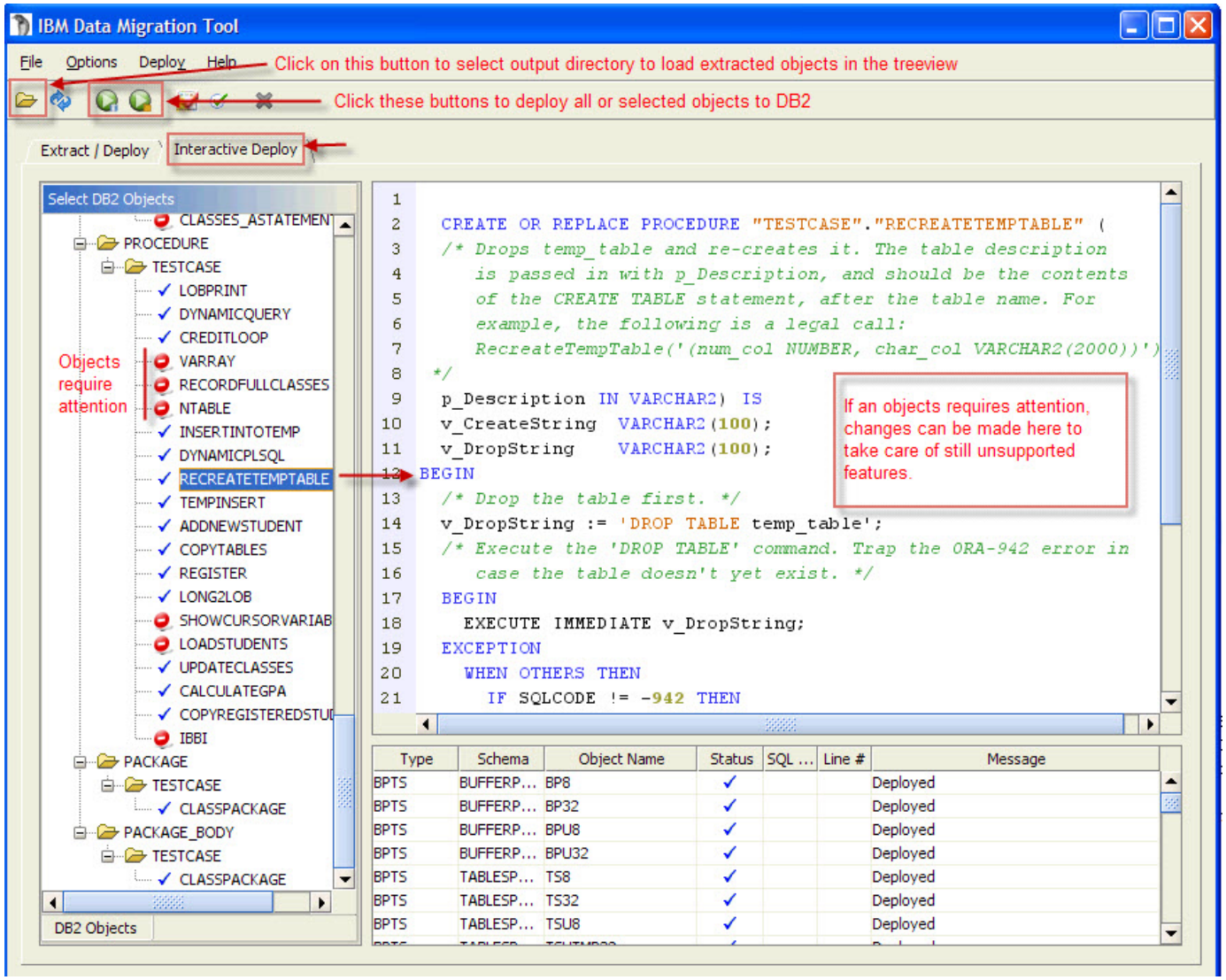
- Range of industries, solutions, countries
- Range of sizes: 2,000 – 185,000 SQL statements
- Over 750,000 lines tested
- 90-99% of lines immediately transferable to DB2



IBM Data Movement Tool – Setup



Data Movement Tool – Drag, Drop, Patch



Conclusion

- **DB2 breaks with traditional approach to application migration**
  - Re-use existing skills
  - PL/SQL and SQL PL have similar performance characteristics
    - Remain in your dialect of choice
  - Significantly simplified enablement

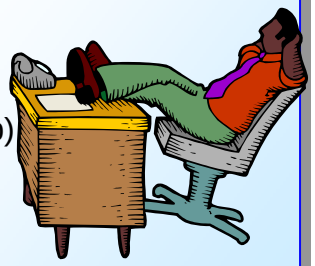
openbravo to DB2 9.7

DB2 9.5: ~2 YEARS

1. **Map schema and data types**
   - *Different datatype behavior. Eg: DATE*
   - *Different DDL*
2. **Move data**
3. **Translate PL/SQL code (Triggers/Procedures/Functions)**
   - *Missing built-in functions & packages*
   - *Missing functionality: REF CURSORs, %TYPE, …*
4. **Translate SQL in application logic (manual)**
   - *Untyped expressions (thousands of occurences)*
5. **Debugging**
   - *Hard to fix concurrency problems*
6. **Test and Tuning**

DB2 9.7: 1 WEEK

1. **Map schema and data types**
   - **Native support**
   - **99% of SQL – no change**
   - **156 of 158 procedures- no change**
   - **PL/SQL – no change**
2. **Move data**
3. **Run the shared code**
   - **Native support – NO EMULATION!!**
   - Minor adjustments (triggers needed tune up)
4. **Test and Tuning**



→ ***No porting effort for future releases of the application***

Reactions to DB2 9.7

Two years … One week!

"To move our application to DB2 9.5 would have taken an estimated two-year effort. We were thrilled to see it took only one week to move it to DB2 9.7. This represents a terrific opportunity to expand our international community of users, partners and developers, and we're very excited to partner with IBM to make new deployment options available."
Paolo Juvara, CTO of Openbravo (Spain)

Significantly lower overall costs

"These features drastically reduce the time required for migration efforts and significantly lower overall costs."
Axel Puerner, Managing consultant, Puerner Unternehmensberatung

Paradigm shift

"The new IBM DB2 offers true ENABLEMENT and not mere PORTING.
This feature is a paradigm shift in the very concept of database migration!"
Godson Retna, Senior Architect, Cognizant Technology Solutions

Reactions

Amazing!

“DB2's PL/SQL compatibility is excellent. We’re looking forward to integrating the current dual source code base into a single one. This will increase our development and testing productivity. In addition, SQL compatibility is significantly improved. We ran an Oracle Database program as is on DB2, and the test result was more than we expected.
The compatibility level that DB2 9.7 achieved is also amazing. We can integrate a lot of incompatible queries into the same one. Now we can stop our program’s different behaviors, which comes from DBMS’s differences, and this will help us improve the quality of our package.”
Masato Kudo, Developer for Platform Development Group, Works Applications

Porting time 1/6 the original estimate

“As we expand, we consistently see a requirement *to support DB2 within large government departments. We specifically chose to take part in the IBM DB2 early access program because of the program's goal to run much of Oracle Database applications without modification.* ***This allows us to reduce the time to port our stored produce persistence layer from Oracle Database to DB2 from 450 days down to 75 days.*** *With what we regard as excellent support from the IBM DB2 team, we believe that IBM has achieved these goals.*”
David Moody - Senior Vice President of Product and Founding Director, Lagan Technologies Ltd.