DB2 11 Application Compatibility (APPLCOMPAT)
What you Need to Know

Christopher J. Crone
cjc@us.ibm.com
DB2 11 Application Compatibility (APPLCOMPAT)
What you Need to Know

Christopher J. Crone

© 2014 IBM Corporation
Disclaimer/Trademarks

© Copyright IBM Corporation 2015. All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

THE INFORMATION CONTAINED IN THIS DOCUMENT HAS NOT BEEN SUBMITTED TO ANY FORMAL IBM TEST AND IS DISTRIBUTED AS IS. THE USE OF THIS INFORMATION OR THE IMPLEMENTATION OF ANY OF THESE TECHNIQUES IS A CUSTOMER RESPONSIBILITY AND DEPENDS ON THE CUSTOMER’S ABILITY TO EVALUATE AND INTEGRATE THEM INTO THE CUSTOMER’S OPERATIONAL ENVIRONMENT. WHILE IBM MAY HAVE REVIEWED EACH ITEM FOR ACCURACY IN A SPECIFIC SITUATION, THERE IS NO GUARANTEE THAT THE SAME OR SIMILAR RESULTS WILL BE OBTAINED ELSEWHERE. ANYONE ATTEMPTING TO ADAPT THESE TECHNIQUES TO THEIR OWN ENVIRONMENTS DO SO AT THEIR OWN RISK.

ANY PERFORMANCE DATA CONTAINED IN THIS DOCUMENT WERE DETERMINED IN VARIOUS CONTROLLED LABORATORY ENVIRONMENTS AND ARE FOR REFERENCE PURPOSES ONLY. CUSTOMERS SHOULD NOT ADAPT THESE PERFORMANCE NUMBERS TO THEIR OWN ENVIRONMENTS AS SYSTEM PERFORMANCE STANDARDS. THE RESULTS THAT MAY BE OBTAINED IN OTHER OPERATING ENVIRONMENTS MAY VARY SIGNIFICANTLY. USERS OF THIS DOCUMENT SHOULD VERIFY THE APPLICABLE DATA FOR THEIR SPECIFIC ENVIRONMENT.

Trademarks
IBM, the IBM logo, ibm.com, DB2, and z/OS are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.
Agenda

- Overview of the Problem
  - Why is DB2 Introducing this new capability?
- Solution Approach
- Solution Detail
- How to Identify Potential Issues
- Summary and Questions
Abstract

DB2 11 for z/OS introduces the concept of Application Compatibility to simplify migration.

This session will introduce the concept and provide the information you need to know to exploit this new capability to simply your migration from DB2 10 to DB2 11.

We will also cover how to exploit this new capability to identify changes you need to make to applications that may be affected by incompatible changes in DB2 11.
Overview of the Problem
Problem

• Changes to SQL and XML behavior are problematic to adopt
  – It is difficult to understand the impact of change to thousands of applications
  – Change needs to be controlled at the application level
• Service stream incompatibilities cause maintenance issues, especially when key PTFs pre-req the PTFs that cause the changed behavior
• Changes at a release boundary allow for some planning, but it is difficult to sync application changes with release migration plans
• When a change surfaces varies
  – On Migration?
  – When New Function is available?
  – BIND/Prepare?
  – Combination of the above
Problem - Standards

- There are IBM and industry standards for SQL that DB2 for z/OS must be compliant with.

- DB2 for z/OS may be out of compliance because of defect or incomplete implementation
  - Fixing the compliance issue introduces an incompatible change that may break existing applications
  - These incompatible changes are saved up and introduced on DB2 release boundary

- Example of incompatibility introduced in DB2 10
  - CHAR function results (also for VARCHAR and CAST of these data types)
    - Leading zeroes no longer returned when there is a decimal point
      - Originally PMR raised to report as error
      - Working as designed to conform to SQL Standards
    - Advice in manuals was to change application code to accommodate new behaviour
Problem – New Function

- New SQL Function can create incompatibilities
- Example - DB2 10 introduced implicit casting
  - Implicit casting enables characters to be implicitly converted to numbers and numbers to be implicitly to characters
    
    CREATE TABLE T1 C1 INTEGER;
    INSERT INTO T1 VALUES (‘123’) ;– Fails in DB2 9, works in DB2 10
  - Function resolution rules, and the functions themselves have been changed to allow both character and number interchangeably
    
    SELECT 123 || ‘abc’ FROM T1; -- Fails in DB2 9, works in DB2 10
- By Definition – most new function is incompatible in some way with existing behavior
  - Most common case is removal of an SQLCODE
Solution Requirements

- Do not force application changes to address incompatible SQL changes on a release boundary
- Allow changes to be introduced at the application (package) level
- Provide more warning and time to customers for incompatible changes to be addressed
- Provide a mechanism to identify applications that need to be analyzed for incompatible behavior
- Provide a 'fence' so DB2 can address errors vs. new incompatible change
Solution Approach
Proposed Solution – Separate Release Migration and Application Migration to new release function

- Limit SQL DML and XML incompatibilities when possible
- Provide a mechanism to identify applications affected by DML and XML changes
- Provide a mechanism to make changes at an application (package) level
  - DB2 11 will be the initial deployment of this capability
  - DB2 10 will be the lowest level of compatibility supported
  - This mechanism will enable support for up to two back level releases (N-2). For Example:
    - DB2 11 supports DB2 10 and DB2 11
    - DB2 12 supports DB2 10 and DB2 11 and DB2 12
    - DB2 _3* supports DB2 11 and DB2 12 and DB2 _3*
    - DB2 _4* supports DB2 12 and DB2 _3* and DB2 _4*
- Provide a mechanism to support skip release migration if such support is needed for a future release migration.

*DB2 _3, and DB2 _4 are theoretical releases used in the above example strictly for demonstrative purposes. This is not a commitment by IBM to actually produce any of these releases.*
DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability

Solution Detail
Solution Details

• New ZPARM APPLCOMPAT for DEFAULT of bind option
  – Will be set to V10R1 on Migration to V11
    • Since this is a new bind option for V11, previously bound packages will not have specified this parm. If a package is bound in V10 or below, APPLCOMPAT(V10R1) will be assumed.
    • The first time a package is bound in V11, it will pick up the DEFAULT.
    • In CM only V10R1 may be specified
  – V11R1 for new install
  – Default on subsequent migrations will be the down level release - e.g. On migration to V_2, default will be V11R1

• REBIND picks up the current value or default
  – Customers can override rebind with any “valid” value
Solution Details

- New PACKAGE bind (BIND/REBIND) option (APPLCOMPAT) and special register
  - Applies to anything that has a package (including things like SPs, UDFs, Triggers…).
    - Trigger Packages are tied to an OBJECT, but compatibility is tied to the package bind options.
  - BIND option applies to static SQL, and the default for the special register.
  - The CURRENT APPLICATION COMPATIBILITY special register applies to dynamic SQL.
    - The default behavior for this special register in a SP or UDF is to not inherit this special register.
    - Special Register need to be set in properties/ini file for (ODBC/JDBC/.NET) drivers.
  - DSN_PROFILE tables instead of, or in addition to, a SPECIAL REGISTER may be used to control remote applications.
BIND/REBIND

- BIND/REBIND PACKAGE
  
  Syntax
  
  **BIND/REBIND**
  
  >--------------------------------------------------------------------->
  
  '--------- APPLCOMPAT---(--V10R1--)---------'
  
  (--V11R1--)

- REBIND TRIGGER
  
  Syntax
  
  **REBIND TRIGGER**
  
  >--------------------------------------------------------------------->
  
  '--------- APPLCOMPAT---(--V10R1--)---------'
  
  (--V11R1--)

---

**DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile**

Industry-leading performance, security, scale and reliability
csect-name ATTEMPT TO USE COMMAND OR OPTION command WHEN THE APPLICATION COMPATIBILITY IS SET FOR A PREVIOUS LEVEL

Explanation
A new command or command option was issued when application compatibility bind option is set to a prior DB2 release. Another possibility is when a command or command option introduced in release n was used but the application compatibility bind option is lower than n.

command The command or command option that can be used only when the application compatibility bind option is set to the compatible release.

System action
The command or command option is not processed.

System programmer response
To use the command or command option, specify the application compatibility bind option to at least the release that introduces this bind option or command.
CREATE/ALTER Routines

- CREATE/ALTER PROCEDURE (SQL native):
  Syntax
  >>>-CREATE/ALTER PROCEDURE ---------------------------------------->
  option-list:
  >------------------------------------------------------------------------<
  |---APPLCOMPAT V10R1--|  
  +---APPLCOMPAT V11R1---+

- CREATE/ALTER FUNCTION (SQL native):
  Syntax
  >>>-CREATE/ALTER FUNCTION ---------------------------------------->
  option-list:
  >------------------------------------------------------------------------<
  |---APPLCOMPAT V10R1--|  
  +---APPLCOMPAT V11R1---+
# Updated Catalog Columns

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Column Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSPACKAGE</td>
<td>APPLCOMPAT</td>
<td>VARCHAR(10)</td>
<td>The DB2 release level that the SQL statements in this package is compatible with.</td>
</tr>
<tr>
<td>SYSPACKCOPY</td>
<td>APPLCOMPAT</td>
<td>VARCHAR(10)</td>
<td>The DB2 release level that the SQL statements in this package is compatible with.</td>
</tr>
</tbody>
</table>
Solution Details (cont)

• ALL SQL DML and XML changes are fenced by these options. This includes:
  – New SQL function
  – Incompatible SQL changes such as:
    • Function Resolution rules
    • Result type changes
    • SQLCODE changes
    • BIF changes
    • …
SQLCODE -4743

-4743
ATTEMPT TO USE A FUNCTION WHEN THE APPLICATION COMPATIBILITY IS SET FOR A PREVIOUS LEVEL

Explanation
Functions that this release of DB2® introduces cannot be used when application compatibility setting is for a prior DB2 release. An attempt was made to execute one of these functions. Another possibility is when a function introduced in release n was used but the application compatibility setting is lower than n.
To use the new functions that have been introduced in this release of DB2, the application compatibility bind option or special register CURRENT APPLICATION COMPATIBILITY must be set to the current release. An attempt to use functions introduced in release n requires the application compatibility bind option or special register CURRENT APPLICATION COMPATIBILITY value to be at least n.

System action
The statement cannot be processed.

Programmer response
Either delay running your program until the application compatibility bind option (for static SQL) or special register CURRENT APPLICATION COMPATIBILITY (for dynamic SQL) has been set to at least the release that introduces this function, or restrict your program to functions that are allowed in a lower release only.

SQLSTATE
56038
Interaction with New Function Availability

• New Function Availability is intended to fence application and system behavior in a migration/co-existance environment
  – Ensure that the “n-m”* level of code can process objects created in the “n” level of the code
  – Ensure that applications bound on level “n” can run (after autobind) on level “n-m”*

• The APPLCOMPAT bind option is intended to ensure application behavior is consistent across releases of DB2
  – As such, APPLCOMPAT for the current release “n” may not be specified until New Function is available mode
  – Once New Function is available, either APPLCOMPAT(n) or APPLCOMPAT(n-1) may be specified (see the following charts). In a future release APPLCOMPAT(n-2) may be specified

* n-m where m = 1 or 2. DB2 10 is the lowest level that may be specified
Normal Migration  DB2 10 → DB2 11

DB2 10 New Function Mode (NFM) With SPE

CATMAINT UPDATE (DSNTIJTC)

DB2 10 Catalog

DB2 10 Libraries

DB2 11 Libraries

DB2 10 Conversion Mode (CM10)

CATENFM START (DSNTIJEN)

DB2 11 Conversion Mode (ENFM10)

CATENFM COMPLETE (DSNTIJNF)

DB2 11 New Function Mode (NFM)

DB2 11 Catalog

Data Sharing Coexistence

1 – 2 months

2 hours

1 week

Bind with APPLCOMPAT(V10R1) option only

Bind with APPLCOMPAT(V10R1) or APPLCOMPAT(V11R1)
Probable Normal Migration  DB2 11 → DB2 12

DB2 11 New Function Mode (NFM) With SPE

CATMAINT UPDATE (DSNTIJTC)

DB2 12 New Function Not Available

-ACTIVATE NEW FUNCTION

DB2 12 Catalog

DB2 11 Catalog

1 – 2 months

1 week

Data Sharing Coexistence

DB2 11 Libraries

DB2 12 Libraries

Bind with APPLCOMPAT(V10R1) or APPLCOMPAT(V11R1)

Bind with APPLCOMPAT(V10R1), APPLCOMPAT(V11R1), APPLCOMPAT(V12R1)
Probable Migration  DB2 12 → DB2 _3

DB2 12 New Function Available
With migration SPE

CATMAINT UPDATE (DSNTIJTC)

DB2 _3 New Function Not Available

-ACTIVATE NEW FUNCTION

DB2 _3 New Function available

DB2 12 Catalog

1 – 2 months

1 week

Data Sharing Coexistence

DB2 12 Libraries

Bind with APPLCOMPAT(V11R1) or APPLCOMPAT(V_2R1)

Bind with APPLCOMPAT(V11R1), APPLCOMPAT(V12R1), APPLCOMPAT(V_3R1)

DB2 _3 Libraries

* Packages bound with APPLCOMPAT(V10R1) are inoperative

DB2 _3 Catalog
Interaction with New Function Availability

- What about New DDL and Authorization syntax and semantics in an application bound with APPLCOMPAT(n-1)?
  - DDL and Authorization are fenced by New Function Availability and are not affected by the APPLCOMPAT level
    - For example – You can use SPUFI- bound with APPLCOMPAT(V10R1)
      - “CREATE” a GLOBAL VARIABLE (new DB2 11 function) – “gv”
      - “GRANT” authority to a user to “SET” the GLOBAL VARIABLE – “gv”
      - “SET” “gv” fails because “SET” is DML
New Reserved Words

• Every release, DB2 adds new reserved words. For example: DB2 11 added ARRAY_EXISTS (a new reserved word for a predicate).

• DB2 has made great strides in enhancing the parser to make new reserved words only be reserved in context. This reduces but does not eliminate impact to customers as they migrate from release to release.
New Reserved Words Example

In V10 and V11, these statements work fine because ARRAY_EXISTS is softened to identifier:

```sql
CREATE TABLE T1 (ARRAY_EXISTS INT);  -- no syntax error
INSERT INTO T1 VALUES (11);

CREATE FUNCTION UDF2 () RETURNS INT
BEGIN
    DECLARE X INT;
    SELECT ARRAY_EXISTS INTO X FROM T1;  -- no syntax error
    RETURN X * 2;
END!

SELECT UDF2() FROM SYSIBM.SYSDUMMY1!  -- returns 22
```
However, the following works in V10 but the syntax in red fails in V11

CREATE FUNCTION ARRAY_EXISTS (PARM1 CHAR(3), PARM2 INT) -- no error in V10 or V11
    RETURNS CHAR(1)
BEGIN
    RETURN SUBSTR(PARM1, PARM2, 1);
END!
COMMIT!
CREATE FUNCTION UDF1 () RETURNS INT
BEGIN
    DECLARE X CHAR(3) DEFAULT 'ABC';
    IF ARRAY_EXISTS(X, 1) = 'A' THEN  -- OK in V10, -104 in V11
        RETURN 1;
    ELSE
        RETURN 2;
    END IF;
END!

SELECT UDF1() FROM SYSIBM.SYSDUMMY1;

New Reserved Words Example (cont)
V10/V11 Reserved Word Solution

• Each DB2 release has only one parser grammar
  – Using the APPLCOMPAT capability doesn’t work in this case because to dual path the incompatible change would require two parsers (essentially the DB2 10 and DB2 11 grammar)

• To avoid the cost of shipping two parsers, we will add code (in an APAR) to DB2 10 to identify where DB2 11 keywords would cause a failure and issue IFCID 366 to alert customers of this issue.
  – Since new reserved words are only reserved in context, we expect customers will identify few or no issues, but the function allows customers to move forward with confidence.

• APAR PM84769/UK94459 closed in 2Q2013
  – Three words are checked, in context ARRAY_EXISTS, CUBE, and ROLLUP

• Other directions possible in the future
How to Identify Potential Issues
Solution Detail (cont)

- Accounting Summary Field (WPKTINCOMPAT)
- Trace records IFCID 366/376 will be provided for all incompatible code paths
  - The traces will be in the down-level code paths to illuminate the path until the down-level code is removed (in release N+3)
  - For example if we change the SQLCODE -123 to SQLCODE -456 in DB2 V11
    - New trace with function code yyyy will be cut in the paths where the old SQLCODE is being returned for packages bound with APPLCOMPAT(V10)
    - The trace will not be cut if APPLCOMPAT(V11R1), or above is specified
    - The trace will be cut in DB2 11 and DB2 _2 in the down level path (APPLCOMPAT(V10R1)), but will be removed in DB2 _3 when the compatibility code is removed
IFCID 366/376

**IFCID 366/376** identify the SQL statements with potential incompatible changes when switching to the new V11R1 application behavior.

IFCID 366 is enhanced to identify any new type (TBD) of incompatible application changes so that customers can identify subject packages prior to switching to the new V11R1 application behavior.

IFCID 376 This trace record is written once for each unique dynamic cached statement and static statement if it was bound on V10 NFM or later. For static statements that are bound before V10 NFM, this trace record will still be externalized once per unique combo of plan, pkg ID, statement number.

**IFCID 376 is the strategic solution**
ICCID 376

QW0376 DSECT
QW0376FN DS F The value '1' indicates that
* the DB2 for z/OS Version 9
* SYSIBM.CHAR(decimal-expr)
* function has been executed.
* 
* The value '3' indicates
* unsupported character string
* representation of a timestamp
* 
* The value of '1101' indicates inserting into an XML column
* without XMLDOCUMENT function
* 
* The value of '1102' indicates V10 Xpath evaluation was in effect
* 
* The value of '1103' indicates V10 RLF
* reactive governing behavior for dynamic SQL was in effect
...

DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability
IFCID 376

QW0376SN DS F Statement number of the query
QW0376PL DS CL8 Plan name for this query
QW0376TS DS CL8 Timestamp for this query
QW0376SI DS CL8 Statement Identifier
QW0376TY DS XL2 Statement information
QW0376DY EQU X'8000' Statement is dynamic
QW0376SC EQU X'4000' Statement is static
QW0376SE DS H Section number
QW0376PC_Off DS H Offset from QW0366 to Package
  * Collection ID
QW0376PN_Off DS H Offset from QW0366 to
  * Program name
QW0376VL DS H Version length
QW0376VN DS CL64 Version
  *
QW0376PC_D DSECT
QW0376PC_Len DS H Length of Package Collection ID
QW0376PC_Var DS 0CL128 %U Package Collection ID
  *
QW0376PN_D DSECT
QW0376PN_Len DS H Length of Program Name
QW0376PN_Var DS 0CL128 %U Program Name
Updated IFCID 106

IFCID 106 is updated to trace the new ZPARM

QWP4RMDB DS CL8 REORG_MAPPING_DATABASE N0239R3
QWP4DM1636 DS CL8 (s) DM1636
QWP4MIMTS DS F MAXSORT_IN_MEMORY N4504r5
QWP4MUSE DS XL2 (s) N4504r5
QWP4IXCU DS H INDEXCLEANUP_THREADS n0010r5
QWP4DEGD DS F PARAMDEG_DPSI n231r5
QWP4APCO DS 0H APPLCOMPAT n8195r5
QWP4APCO_LEN DS H length of APPLCOMPAT setting
QWP4APCO_VAR DS CL10 APPLCOMPAT setting
DS CL32 UNUSED LI872

The IFCID 106 formatter stored procedures, SYSPROC.DSNWZP and SYSPROC.ADMIN_INFO_SYSPARM are updated to report the APPL_COMPAT setting.
On Migration

- Premigration jobs will flag as a warning any packages bound with APPLCOMPAT(V10R1) or APPLCOMPAT() on migration to DB2.
- Premigration jobs will flag as a serious-warning (error) any packages bound with APPLCOMPAT(V10R1) or APPLCOMPAT() on migration to DB2.
  - These packages will be **NOT** be autobound in DB2.
  - These packages will be inoperative in DB2.
What isn’t addressed

- Changes to SQL function because of customer reported issues PMRs/APARs
  - Customer initiated APARs can result in differences in SQL behavior. While these changes almost always correct a defect in the product, the changed behavior can affect results that other customers see.
Summary and Questions
Summary

• With the APPLCOMPAT bind option, we are striving to separate release migration from application changes that may be necessary to migrate to a new release.
• DB2 is providing two releases of backward compatible behavior to enable customers to migrate faster to a new release with confidence that applications will be have the same in the new release as they have in the previous releases.
• Customers can use the tracing capabilities that have been provided to identify applications that potentially need to change.
Follow us in the Digital World!

- Join “The World of DB2” global community
- DB2 for z/OS home page
DB2 11 Application Compatibility (APPLCOMPAT) – What you Need to Know