RDX

REXX Db2 eXtensions for z/OS

- **Embed SQL/XML statements and XQuery and XPath expressions natively within REXX execs**
- **Use static SQL syntax or the dynamic SQL syntax employed by DSNREXX**
- **Run existing DSNREXX applications without source changes**
- **Publish (create) new XML data from existing data sources**
- **Flow SQL/XML query results into REXX stemmed variables in a single operation**
- **Issue Db2 commands, IFI requests and RRSAF / CAF services – natively from REXX**
- **Learn pureXML with immediate execution and powerful debugging**

Overview

REXX **Db2** e**X**tensions (**RDX**) lets you code SQL and SQL/XML statements natively within REXX execs. RDX execs that combine SQL statements, XQuery and XPath expressions, Db2 commands, IFI requests and RRSAF/CAF services can be developed rapidly and tested immediately. Without compile and link edit steps. Without preprocess and bind delays. You can edit and test RDX execs directly within ISPF Edit and get instant feedback.

What's more, there's a minimal learning curve. z/OS developers can leverage their skills with REXX, SQL and ISPF to become productive almost immediately with pureXML and Large Objects (LOBs). RDX harnesses the expressiveness and ease of use of REXX to get you started right away with the pureXML capabilities of Db2 for z/OS.

In addition to being a superb training tool, RDX is well suited to prototyping high volume systems and developing tools and small applications quickly – all with immediate feedback, extensive diagnostics and context sensitive help.

The combination of REXX and RDX facilitates learning and using pureXML to such a degree, *every* organization should consider RDX in their enablement plans for Db2 for z/OS.

Benefits

- Leverage expertise in REXX, SQL and Db2
- Provide valuable "getting started" help with pureXML and Db2 for z/OS in terms of training, usage and debugging
- Provide Db2 for z/OS developers with the expressive power and instant feedback of REXX
- Streamline development with powerful debugging facilities and immediate execution
- Develop applications with *far less* coding than compiled and assembled languages
- Improve productivity and reduce the cost of training and development
- Exploit multi-row FETCH to develop REXX applications that run faster then their compiled, static SQL counterparts
- Explore and prototype new Db2 SQL with REXX, before programming as static SQL

Riverview Historic Plaza • 33 Newark Street • Hoboken NJ 07030 USA • Tel: +1 201 420-0400 • www.relarc.com

Relational Architects International

Features

- Fully supports SQL/XML as implemented by the most current versions of Db2 for z/OS
- Supports all Db2 data types including XML, LOB, LOB_FILE and LOB_LOCATOR
- Converts automatically between Db2 and REXX datatypes
- Supports host variables within SQL and XQuery and XPath expressions
- Provides granular access to XML document content
- Extends DSNREXX with full support for SQL/XML, Large Objects and multi-row FETCH
- Provides extensive debugging and diagnostic facilities that are virtually absent from DSNREXX
- Provides REXX functions for UNICODE conversion as well as between ASCII and EBCDIC
- Augments the XML Guide for Db2 for z/OS with an extensive set of sample execs and RDX User Guide that parallel the Db2 XML Guide. They get you started as quickly as possible with pureXML and Db2 for z/OS.
- A formatted SQLCA provides feedback after each SQL statement executes
- Formatted SQLDAs describe query result columns and the result sets returned by stored procedures

Who Will Use REXX Db2 eXtensions?

Application Developers	✓
End users	✓
DBAs	✓
Operations	✓
Technical Support	✓
Systems Programmers	✓

Prerequisites

RDX is easily installed and administered.
It runs with all releases of z/OS and
Db2 for z/OS whose IBM support is current

RDX Sample Exec

```
Address RDX                                                          (1)

"EXECSQL DECLARE C1 CURSOR FOR",                                     (2)
   "SELECT CID,",
   "XMLQUERY(",
     "'declare default element namespace",
     '"http://posample.org";',
     "/customerinfo/phone[@type = $type]'",
     'passing INFO, CAST(:phone_type AS VARCHAR(16)) as "type")',
     'AS "PHONE FROM INFO"',
  "FROM MYCUSTOMER",
  "WHERE CID between :low_value and :high_value"

If SQLCODE \= 0 then return SQLCODE                                  (3)
"EXECSQL OPEN C2"
"EXECSQL FETCH C2 into :CID, :info"

Do while sqlcode = 0
   SAY cid info
   "EXECSQL FETCH C2 into :CID, :info"                               (4)
End

"EXECSQL CLOSE C2"
Return SQLCODE
```

Sample execs reference the XML Guide for Db2 for z/OS

(1) ADDRESS RDX directs REXX to route host commands to RDX for execution.

(2) The SQL/XML DECLARE statement (preceded by EXECSQL) is simply embedded within the REXX exec. The standard colon prefix denotes host variable references within SQL clauses and XQuery and XPath expressions.

(3) RDX updates all the host variables comprising the SQLCA to provide feedback about the most recently executed SQL/XML statement. For example, you can reference the current value of SQLCODE by name.

(4) RDX FETCHes values directly thru memory into host variables shared by RDX, REXX and ISPF. Thus, the host variables referenced by SQL/XML statements are automatically REXX variables as well as dialog variables accessible to ISPF.

Harness the power of Db2 for z/OS and its pureXML capabilities

Call the REXX/Db2 experts at 800 776-0771

Riverview Historic Plaza • 33 Newark Street • Hoboken NJ 07030 USA • Tel: +1 201 420-0400 • www.relarc.com

Relational Architects International