

Rexx/SQL Design Changes

2024 International Rexx Symposium
Brisbane, QLD, Australia

Mark Hessling
5 March 2024



Rexx/SQL Design Changes

- What Rexx/SQL does
 - Consistent interface to access SQL databases
 - Same code different databases
 - Access to multiple database types in one program



Rexx/SQL Design Changes

- How Rexx/SQL currently works
 - Naming conventions
 - General
 - External Functions
 - Global variables
 - Artefacts built
 - Example code



How Rexx/SQL currently works

- General Naming Conventions
 - SQL...
 - Generic name; database access dependent on how built
 - [DB]...
 - where [DB] is one of LITE, MY, DB2, ODBC, etc.
- External Functions
 - SQLLoadFuncs, SQLConnect, etc.
 - Generic name; database access dependent on how built
 - [DB]LoadFuncs, [DB]Connect, etc.
 - where [DB] is one of LITE, MY, DB2, ODBC, etc.



How Rexx/SQL currently works

- Global variables
 - SQLCA. - compound variable with error messages, and other details of current connection and transaction
 - Generic name; database access dependent on how built
 - Eg SQLCA.SQLERRM - last error message text
 - [DB]CA. - compound variable with error messages, and other details of current connection and transaction
 - where [DB] is one of LITE, MY, DB2, ODBC, etc.
 - Eg SQLCA.SQLERRM - last error message text



How Rexx/SQL currently works

- Artefacts built
 - Linux/Unix like systems
 - rexxsql, rexx[DB], librexxsql.so, librexx[DB].so
 - macOS
 - rexxsql, rexx[DB], librexxsql.dylib, librexx[DB].dylib
 - Windows/OS/2
 - rexxsql.exe, rexx[DB].exe, rexxsql.dll, rexx[DB].dll
- rexxsql and librexxsql.so are specific to database



How Rexx/SQL currently works

- Contents of librexxsql.so:
 - SQLLoadFuncs()
 - Registers all other SQL*() functions
 - SQLConnect()...
 - Database API calls
- Contents of librexx[DB].so: (librexxlite.so)
 - LITELoadFuncs()
 - Registers all other LITE*() functions
 - LITEConnect()...
 - SQLite3 database API calls



How Rexx/SQL currently works

MyProg.rexx

Call RxFuncAdd 'SQLLoadFuncs', 'rexssql', 'SQLLoadFuncs'

Call SQLLoadFuncs

rcode = SQLConnect(...)

librexssql.so

SQLLoadFuncs()

SQLConnect()

SQLDisconnect()



How Rexx/SQL currently works

- Example Code
 - sql-connect-old.rexx lite|my
 - Same code but different databases
 - [DB]-connect-old.rexx
 - (lite-connect-old.rexx) specific to SQLite3
 - lite-my-connect-old.rexx
 - Multiple database connections



Rexx/SQL Design Changes

- How Rexx/SQL (3.0) will work
 - Naming conventions
 - General
 - External Functions
 - Global variables
 - Artefacts built
 - Example code



How Rexx/SQL will work

- General Naming Conventions
 - SQL...
 - Generic name; database access dependent on how built
 - [DB]...
 - where [DB] is one of LITE, MY, DB2, ODBC, etc.
- External Functions
 - SQLLoadFuncs, SQLConnect, etc.
 - Generic name; database access dependent on how built
 - [DB]LoadFuncs, [DB]Connect, etc.
 - where [DB] is one of LITE, MY, DB2, ODBC, etc.



How Rexx/SQL will work

- Global variables
 - SQLCA. - compound variable with error messages, and other details of current connection and transaction
 - Generic name; same variables for all databases
 - Eg SQLCA.SQLERRM - last error message text



How Rexx/SQL will work

- Artefacts built
 - Linux/Unix like systems
 - rexxsql, rexx[DB], librexxsql.so, librexx[DB].so
 - macOS
 - rexxsql, rexx[DB], librexxsql.dylib, librexx[DB].dylib
 - Windows/OS/2
 - rexxsql.exe, rexx[DB].exe, rexxsql.dll, rexx[DB].dll
- rexxsql and librexxsql.so are a translation layer



How Rexx/SQL will work

- Contents of librexssql.so:
 - SQLLoadFuncs([DB])
 - If [DB] not supplied, registers all other SQL*() functions
 - Else dynamically loads [DB]*() functions from librex[DB].so into lookup table
 - SQLConnect(...[DB])...
 - Finds [DB]Connect() in lookup table and calls [DB]Connect()
- Contents of librex[DB].so: (librexlite.so)
 - LITELoadFuncs()
 - Registers all other LITE*() functions
 - LITEConnect()...
 - SQLite3 database API calls



How Rexx/SQL will work

myprog.rexx

Call RxFuncAdd 'SQLLoadFuncs', 'rexxsql', 'SQLLoadFuncs'

Call SQLLoadFuncs 'MY'

Call SQLLoadFuncs 'LITE'

rcode = SQLConnect(..., 'MY')

rcode = SQLConnect(..., 'LITE')

librexxsql.so

SQLLoadFuncs()

SQLConnect()

librexxlite.so

LITELoadFuncs()

LITEConnect()

librexxmy.so

MYLoadFuncs()

MYConnect()



How Rexx/SQL will work

- Example Code
 - `lite-mysql-connect-new.rexx`
 - `dbclient.rexx`
 - Similar to `mysql`, `sqlplus`



How Rexx/SQL will work

- Advantages
 - Change database by changing SQLLoadFuncs() and SQLConnect() calls rather than copying different rexxsql.dll/.so/.dylib which may be installed in system areas
 - Makes packaging simpler; no duplicate rexxsql/rexxsql.exe, rexxsql.dll/.so/.dylib with different functionality



How Rexx/SQL will work

- Disadvantages
 - Slower; an extra call level involved for each external function call; SQLConnect() -> [DB]Connect()



Questions

