DB2 Web Query Dynamic Runtime Support

Many IBM i customers have multiple DB2 Web Query reporting environments that contain different sets of data that are identical in name and format. Examples of this are

- Development and production environments that have different data sets
- A state database that has different data sets for each county in the state
- A SAaS (Software As a Service) Provider that has different data sets for each of their clients

Each environment can have files and formats that are identical to other environments – the only thing that is different is the data. Each environment is typically set up and controlled by setting up different libraries and adding those libraries to the user portion of their Runtime. IBM i customers are accustomed to having the ability to quickly and easily switch from one environment to another. Usually this is controlled by a menu option in the application so that the user can quickly swap from say CUSTOMER_ABC'S library list and data to CUSTOMER_XYZ's library and data. When this option is taken, the application issues the CHGLIBL command to change the library list so that the library(s) holding data for CUSTOMER_XYZ is present and/or first in the user's library list. This way, whenever an unqualified reference is made to a file in this library(s), CUSTOMER_XYZ's data is located first by the application and displayed to the user.

DB2 Web Query users need this same kind of easy, quick, and flexible ability to dynamically change their library lists. Today this can be handled by signing off from Web Query and signing on as a different user (which has a different library list defined in the job description). While this works, it is not an acceptable long term solution for many reasons: it requires more (seemingly unnecessary Web Query licenses), is difficult to maintain, and it is not quick, flexible, or easy for the end user.

This document introduces DB2 Web Query Dynamic Runtime Environments (RTEs). RTEs are first supported in 5733WQX group PTF level 4 and provide a way for DB2 Web Query users to quickly, easily, and dynamically change the user portion of the library list prior to retrieving the data from the database.

Assumptions:

The following assumptions are made for this specific implementation:

- Tables, views, and files are identical in name and format across the various libraries
- Supports only those reports that are based on synonyms created using DB2 CLI adapter
- *LOCAL connection support only. No remote (DRDA) database support.
- Synonyms are set up with one part naming (no explicit library qualification)

Terminology

Runtime Environment – Attributes that are executed during report execution before the SQL statement is submitted to the DB2 Engine. Current examples include the user portion of the library list and an exit program name and library.

Active Runtime Environment – The environment that was activated for a specific user profile. Activation can be set by an administrator or it can be overridden by the user by selecting a different Runtime Environment from the DB2 Web Query BI Portal.

Administration

To set up a DB2 Web Query Runtime Environment, sign into a 5250 session as a DB2 Web Query administrator user profile (**member of QWQADMIN group profile)** and issue the following command:
WRKWQRTE
No parameters

The Work with DB2 Web Query Runtime Environments screen is presented. This interface allows to define RTEs and assign users to them. You will see the following screen:

```
                    Work with DB2 Web Query Runtime Environments


 Type options, press Enter.
   2=Change  3=Copy  4=Delete  5=Display  8=Assign Users


 Opt Env Name   Environment Description
   _    *NONE



 F3=Exit    F5=Refresh     F6=Add          F10=View Library Lists
 F11=Work with users       F12=Cancel
```

Adding, changing, copying, and deleting Runtime Environments
Press F6 to add a new DB2 Web Query Runtime Environment

You will see the following screen:

```
                Add DB2 Web Query Runtime Environment

  Runtime Environment Name . .  _____
  Runtime Env Description   . .  _____
  Library List . . . . . . . .  _____
  _____
  _____
  _____   Name, *NONE
   (Type up to 25 library names, separated by a space)

  Exit program to call . . . .   *NONE_____     Name, *NONE
    Library  . . . . . . . . .   _____     Name
  Press Enter to continue.

  F3=Exit    F12=Cancel
```

Specify the following information and press Enter:

- Environment name – Specify the name of the runtime environment. Environment names are 10 characters in length and must be unique.
- Environment description – Specify a description for the runtime environment. This is an optional field but you are encouraged to use it. This value will appear as the "hover over" value for that RTE when user is presented with the list of RTEs to select in the BI Portal.
- Library List – Specify up to 25 library names (in the order you want them). Each library should be separated by a space. Specify *NONE if you do not want to change the library list.
- Exit program and library – Specify the name and library of the exit program you want called. The exit program is one that you develop and is called prior to each DB2 Web Query request (before the library list is changed). Specify *NONE if you do not want to call an exit program.

```
               Change DB2 Web Query Runtime Environment

  Runtime Environment Name . .  CUST_XYZ__
  Runtime Env Description   . .  Runtime environment for customer XYZ data_____
  Library List . . . . . . . .  COBBG QTEMP QGPL QWQBLD210 QRY400DSC WQCRTSYN_____
  _____
  _____
  _____   Name, *NONE
   (Type up to 25 library names, separated by a space)

  Exit program to call . . . .   *NONE_____     Name, *NONE
    Library  . . . . . . . . .   _____     Name
```

```
             Work with DB2 Web Query Runtime Environments

Type options, press Enter.
  2=Change  3=Copy  4=Delete  5=Display  8=Assign Users

Opt Env Name   Environment Description
  _   CUST_ABC   Runtime environment for customer ABC
  _   CUST_ACME  Runtime environment for customer ACME
  _   CUST_FRED  Runtime environment for customer FRED
  _   CUST_JOE   Runtime environment for customer JOE
  _   CUST_XYZ   Runtime environment for customer XYZ
```

If you need to change the Runtime Environment (add/remove a library from the library list or specify a different exit program), use option 2.

If you want to delete the runtime environment, specify option 4.

To copy an existing runtime environment definition, specify option 3 and give the runtime environment a new name.

Assigning Users to Runtime Environments

Next you need to assign users to the runtime environments. Alternatively you could assign runtime environments to users (more on that later).

Since we are at the "Work with Runtime Environments" screen, let's assign users to environments first. Type option 8 (Assign users) next to the runtime environment you want to assign users to and press Enter.

```
             Work with DB2 Web Query Runtime Environments

Type options, press Enter.
  2=Change  3=Copy  4=Delete  5=Display  8=Assign Users

Opt Env Name   Environment Description
  _   CUST_ABC   Runtime environment for customer ABC
  _   CUST_ACME  Runtime environment for customer ACME
  _   CUST_FRED  Runtime environment for customer FRED
  _   CUST_JOE   Runtime environment for customer JOE
  8   CUST_XYZ   Runtime environment for customer XYZ
```

A list of all registered DB2 Web Query users (including runtime groups) is presented.

```
            Assign Users to DB2 Web Query Library List Environment

Runtime Environment Name . . CUST_XYZ
Runtime Env Description  . . Runtime Environment for customer XYZ

Type options, press Enter.
  1=Assign as user's active environment    2=Assign user to environment

Opt User          Opt User          Opt User          Opt User          Opt User
  _   *ALL           _  MUCOBBGRPP*   _  RMR
  =   BESTGEN        _  MUDB2WBQRY*    _  SALES*
  _   COBBG          _  OZZIE         _  WANGYUYU
  _   COBBG2         _  QRYUSR01      _  ZHUYU
  _   COBBG3         _  QRYUSR02
  _   COBBG4         _  QRYUSR03
  _   HBEDOYA        _  QRYUSR04
  _   JIM            _  QRYUSR05
  _   JIMBAINB       _  QRYUSR06
  _   KRS            _  QRYUSR07
  _   MACKD          _  QRYUSR08
  _   MUADVLAB02     _  RDANDREW
```

Notes:
• *ALL is always the first entry on the list
• Runtime groups are denoted by an asterisk (*) on the right side of the profile name.

To assign a profile to a runtime environment, simply type a 2 next to the profile name and press Enter.

If you want this environment to be the active (default) runtime environment and be automatically activated, then specify a '1' next to that profile.

To "unassign" (remove) a user from a runtime environment, simply blank out the 1 or the 2 option next to that profile name.

When you a user is assigned to a runtime environment, that environment shows up on the user's runtime environment drop down list on the DB2 Web Query BI Portal. The user can then select that runtime environment to make it the Active one.

```
            Assign Users to DB2 Web Query Library List Environment

Runtime Environment Name . . CUST_XYZ
Runtime Env Description  . . Runtime Environment for customer XYZ


Type options, press Enter.
  1=Assign as user's active environment    2=Assign user to environment


Opt User          Opt User          Opt User          Opt User          Opt User
  _   *ALL          _   MUCOBBGRPP*   _   RMR
  2   BESTGEN       =   MUDB2WBQRY*   _   SALES*
  1   COBBG         2   OZZIE         _   WANGYUYU
  _   COBBG2        _   QRYUSR01      _   ZHUYU
  _   COBBG3        _   QRYUSR02
  _   COBBG4        _   QRYUSR03
  _   HBEDOYA       _   QRYUSR04
  _   JIM           _   QRYUSR05
  _   JIMBAINB      _   QRYUSR06
  2   KRS           _   QRYUSR07
  _   MACKD         _   QRYUSR08
  _   MUADVLAB02    _   RDANDREW
```

In the above example, users BESTGEN, COBBG, OZZIE, and KRS have been assigned to the CUST_XYZ
runtime environment.

For user COBBG, notice that we specified option 1 to make it the "Active" environment for that user.
This runtime environment becomes the default active one for COBBG and is "activated" automatically.
This means that when COBBG logs into DB2 Web Query, that runtime environment will be executed
automatically prior to running every DB2 Web Query procedure (report, chart, dashboard). When this
happens, the library list is changed to the one defined in CUST_XYZ and the exit program defined in
CUST_XYZ will be called prior to the report's data retrieval process.

Because option 1 is used and the environment is made the active one by default, COBBG does not have
to activate CUST_XYZ manually it by selecting it from the runtime Environment drop down list in the DB2
Web Query BI portal.

> **Note:** A user can have only one Active runtime environment.

Notice that the first user in the list is one named *ALL. This special value is provided as a "convenience"
feature and can be used to assign all DB2 Web Query users to an environment. This way, you do not
have to assign all of the individual users and groups to a runtime environment.

Also, as mentioned previously, runtime group profiles are also included in this list. So if assigned to a runtime environment, any users that are members of that group, can select that runtime environment from the DB2 Web Query BI portal.

So what happens if you assign different Runtime Environments to *ALL, one or more group profiles that a user profile is a member of, and the user profile itself? Which one wins and becomes the active one for that user profile? The following ordered list is used to resolve this:

1. Specific user profile – A specific user profile match always wins
2. Group profile - if the user is a member of multiple groups that are assigned to different runtime environments, then the first one alphabetically is selected
3. *ALL

Assigning Runtime Environments to Users

If you prefer, you could also make these assignments from a user perspective. To do this, return to the **Work with DB2 Web Query Runtime Environments** panel. Notice at the bottom there is a function key **F11=Work users**. Press this function key.

```
            Work with DB2 Web Query Runtime Environments

 Type options, press Enter.
   2=Change  3=Copy  4=Delete  5=Display  8=Assign Users

 Opt Env Name    Environment Description
   _   CUST_ABC    Runtime environment for customer ABC
   _   CUST_ACME   Runtime environment for customer Acme
   _   CUST_FRED   Runtime environment for customer Fred
   _   CUST_JOE    Runtime environment for customer Joe
   _   CUST_XYZ    Runtime environment for customer XYZ


 F3=Exit  F5=Refresh  F6=Add  F10=View Lib lists   F11=Work users  F12=Cancel
```

A list of all registered DB2 Web Query users is presented. Again, runtime groups are included and denoted by an asterisk to the right of the profile name.

```
              Work with DB2 Web Query Runtime Environment Users

 Type options, press Enter.
  8=Assign Runtime Environments

 Opt User            Opt User           Opt  User            Opt User
  _    *ALL           _   OZZIE
  _    BESTGEN         _   QRYUSR01
  _    COBBG           _   QRYUSR02
  _    COBBG2          _   QRYUSR03
  _    COBBG3          _   QRYUSR04
  _    COBBG4          _   QRYUSR05
  _    CSMITH          _   QRYUSR06
  _    HBEDOYA         _   QRYUSR07
  _    JIM             _   QRYUSR08
  _    JIMBAINB        _   QRYUSR09
  _    KRS             _   QRYUSR10
  _    MACKD           _   RDANDREW
  _    MUADVLAB02      _   RMR
  _    MUCOBBGRPP*     _   WANGYUYU
  _    MUDB2WBQRY*     _   ZHUYU

F3=Exit    F5=Refresh    F12=Cancel
```

To assign environments to a specific user, type option 8 next to that user and press Enter.

The *Assign DB2 Web Query Runtime Environments to Users* is presented.

```
              Assign DB2 Web Query Runtime Environments to Users

 User profile . . . . . . . . . . .   COBBG

 Type options, press Enter.
   1=Assign as active environment to user    2=Assign environment to user

 Opt Env Name      Environment Description
  _    *NONE
  _    CUST_ABC     Runtime environment for customer FRED
  _    CUST_ACME    Runtime environment for customer FRED
  _    CUST_FRED    Runtime environment for customer FRED
  _    CUST_JOE     Runtime environment for customer FRED
  _    CUST_XYZ     Runtime environment for customer FRED
```

This displays a list of DB2 Web Query Runtime Environments that can be assigned to the user.

Just as before, you can select option 2 to assign the Runtime Environment to the user or you can select option 1 to make it the Active Runtime Environment for that user.

```
                Assign DB2 Web Query Runtime Environments to Users

User profile . . . . . . . . . .    COBBG


Type options, press Enter.
  1=Assign as active environment to user    2=Assign environment to user


Opt Env Name     Environment Description
  _   *NONE
  2  CUST_ABC     Runtime environment for customer FRED
  2  CUST_ACME    Runtime environment for customer FRED
  1  CUST_FRED    Runtime environment for customer FRED
  2  CUST_JOE     Runtime environment for customer FRED
  2  CUST_XYZ     Runtime environment for customer FRED
```

In the above example, the administrator has assigned a total of five RTEs to user profile COBBG. Notice that the active one is CUST_FRED and is highlighted using reverse image display attribute.

> **Note:** A user can have only one Active runtime environment.

So what is the difference between assigning users to RTEs and assigning RTEs to users? The answer is nothing. The idea is the same, it is just working with the data from a different perspective. You can either assign users to a specific Runtime Environment, or you can assign Runtime Environments to a specific user. In the end, you are just making an association between the two so that whenever the user logs into DB2 Web Query, a list of valid Runtime Environments is presented to that user and one can be selected as the RTE to activate.

RTE Exit Program
The RTE exit program is a custom written program that can be used to address specific pre-processing DB2 Web Query requirements. This program will be called before EVERY Web Query request to the database (run reports, sample data in Synonym Editor, Live Preview in InfoAssist, etc.). The program can be written in any supported IBM i OPM or ILE language such as RPG, RGPLE, CL, and CLLE.

Parameters:

- User Profile – Input - CHAR(10)
  The user profile making the request.

- Report fullpath - Input - CHAR(256)
  The full path of the report that is being requested for execution.

- Return Code – Output – Integer
  Return code to pass back. 0=Successful, Non Zero =not successful.
  At this time DB2 Web Query does not evaluate the return code value so there is no difference in behavior for any value returned.

The configured exit program will execute before the configured library list is changed and before the SQL statement is submitted to DB2 for execution. An example is shown below.

The configured exit program must exist as an IBM i *PGM object prior to its specification on this panel.

Note:
If you are using the exit program to change the library list, be aware that the program is called **BEFORE** the value specified in the RTE library list. This means that if you have a value in the RTE library list *other than* *NONE, the CHGLIBL command will be issued with this library list after the exit program invocation – consequently any library list manipulation you performed in the exit program will be overridden/lost.

Exit program implementation ideas:

- Issue STRDBMON for an isolated DB2 Web Query environment or request. Keep in mind, it is called EVERY time a query request is sent to DB2.

- Specify QAQQINI settings for an isolated DB2 Web Query environment or request

- Call a program to set up a more complex library list environment (based on conditions like day of week, current values in database tables, etc). Remember to specify *NONE for the RTE library list if you choose this implementation.

- Carry out whatever environment that you need to set up prior to executing the DB2 request – but **BE CAREFUL** because this program is called prior to every DB2 Web Query database request to DB2.

DB2 Web Query metadata considerations

In order for the library list portion of the RTE to be effective, DB2 Web Query must know that the library list should be used when a query is accessing the table or view. Just as with other methods of accessing system objects on the IBM i, this means that you cannot explicitly qualify the object reference with a specific library name. In DB2 Web Query, the location of this object reference can be found in the synonym (or metadata) of the data source. Specifically, it is located in the synonym's access file (.ACX extension).

For RTE support, the following metadata restrictions exist:

- Support for synonyms created using DB2 CLI adapter only
- *LOCAL connection support only. No remote (DRDA) database support.

Creating new synonyms

In order to instruct DB2 Web Query to use the library list, you must select the **One-part name** setting in the Create Synonym process. One-part name simply means to only include the name of the table (or view) and to exclude the library name in the synonym definition. By default, **One-part name** is NOT selected. This means you must manually make this selection on the Create Synonym panel. This is shown in the following screen.

Alternatively, you can use the CRTWQSYN command from a 5250 interface to create the synonym. Specify *NO for the QUALLIB (*Include library qualification*) parameter to exclude the library from the synonym definition. *YES is the default value, so just as before you must manually override this default value. Here is an example:

CRTWQSYN FILE(ORDERS) SCHEMA(QWQCENT) APPFLR(CENTURY) **QUALLIB(*NO)**

Editing existing synonyms

What if you already have existing synonyms and you want to use RTEs and the library list feature? If you have many synonyms, you probably do not want to recreate them and lose any customizations you have made to them over time. In these cases, you can edit the synonyms to remove the library qualification. To do this, log into DB2 Web Query as either an administrator or folder DBA and open the metadata console as shown in the following screen:
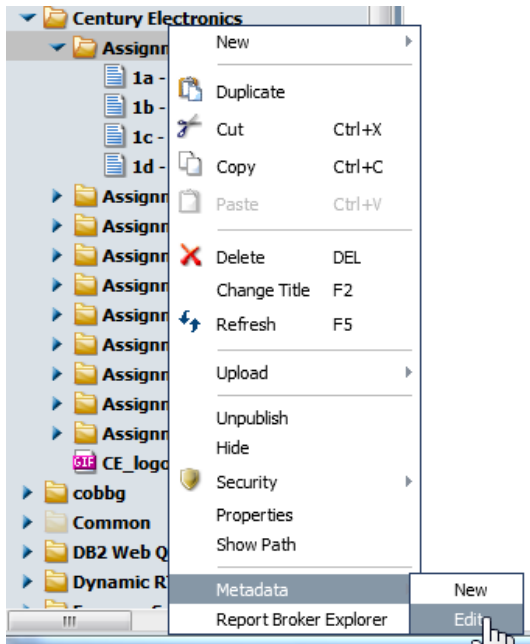
Find and open the synonym you want to change as shown below:
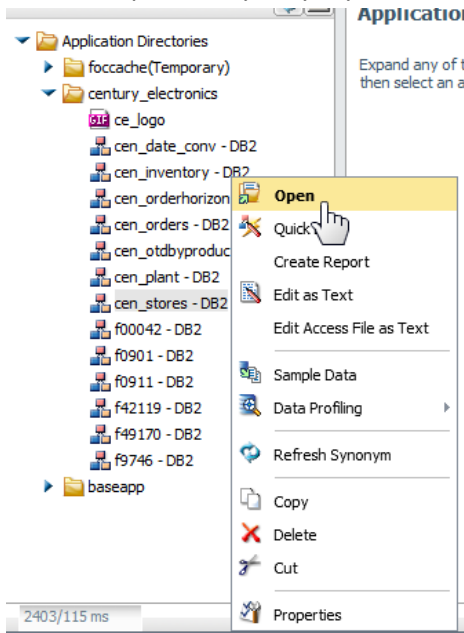
Right click on the synonym segment name and select **Properties**. This will display the Property View and the TABLENAME property. If has a two-part name, you will see this value in the format library_name/table_name as shown in the following example:
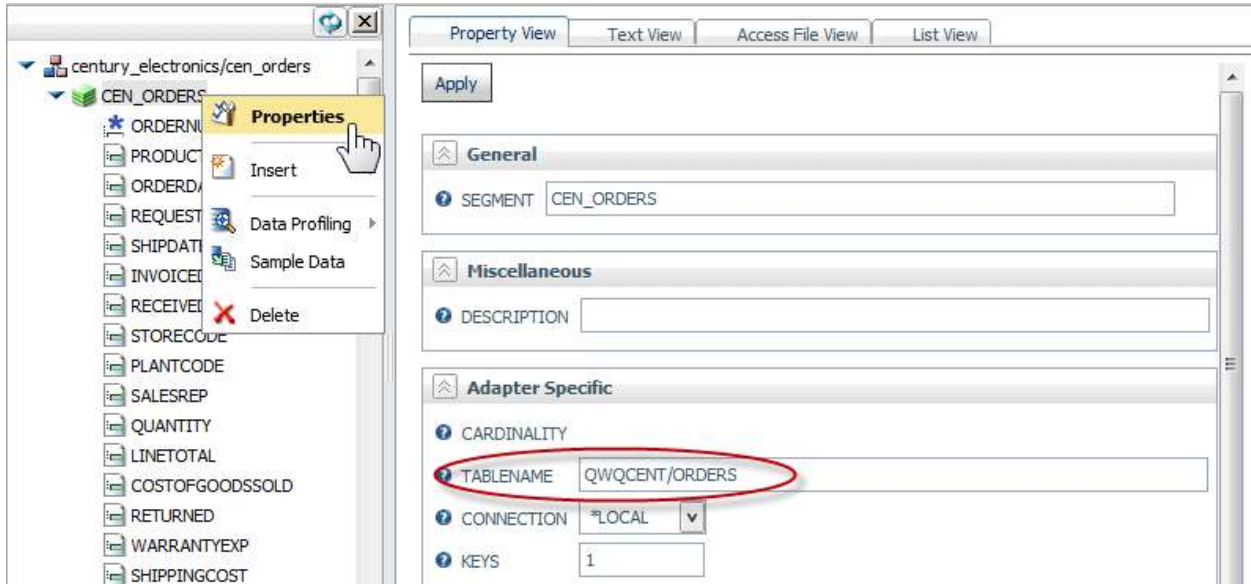
Carefully remove the library name and the slash. When finished, you should only see the table name as shown:
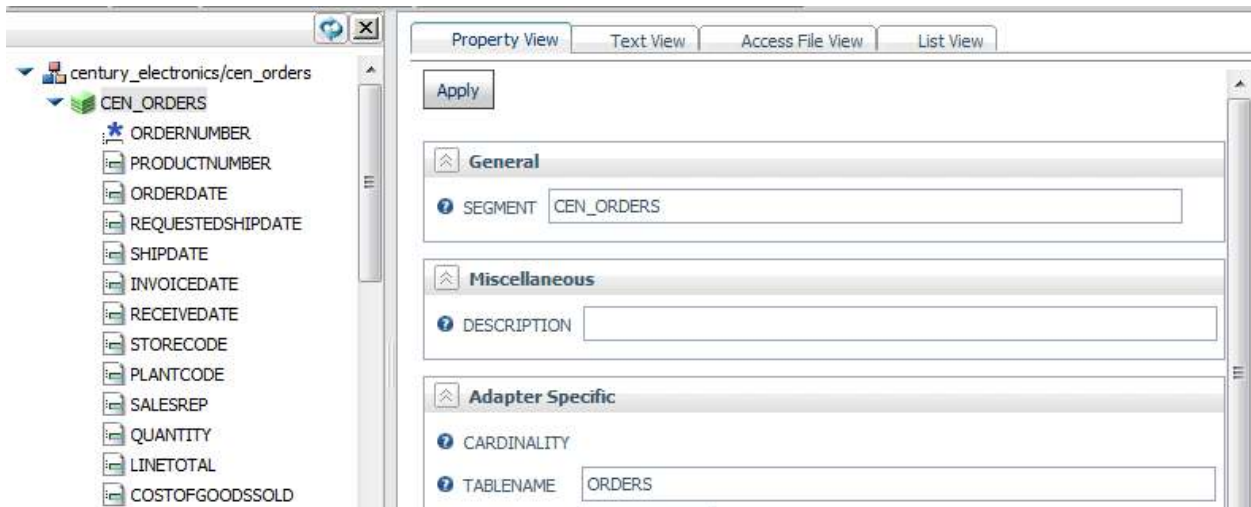
Click the **Apply** button and save the synonym.

Usage

Now that Runtime Environments have been set up and synonyms have one-part naming in place, your end users are ready to use the new feature. Once logged into the DB2 Web Query BI Portal, users can see and change their active RTE by selecting the **Tools→Runtime Environments** menu option as shown below.
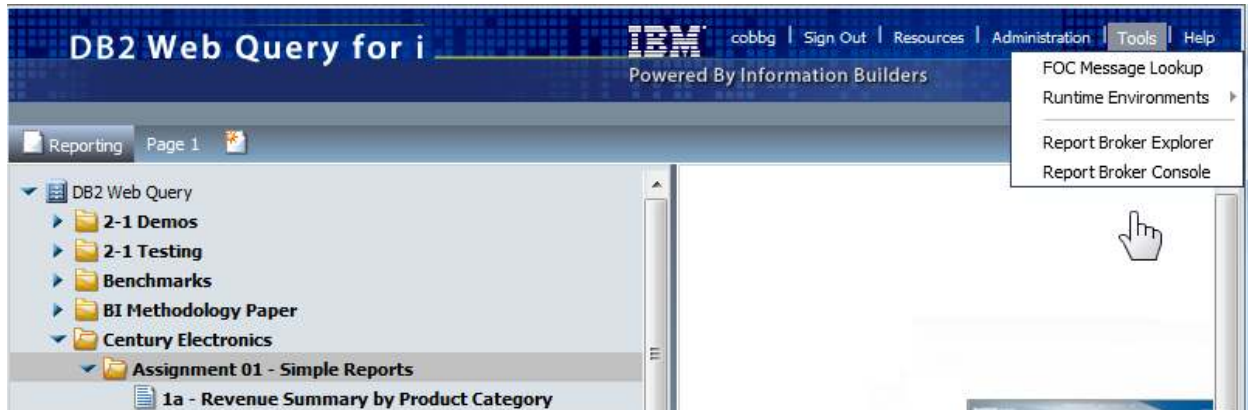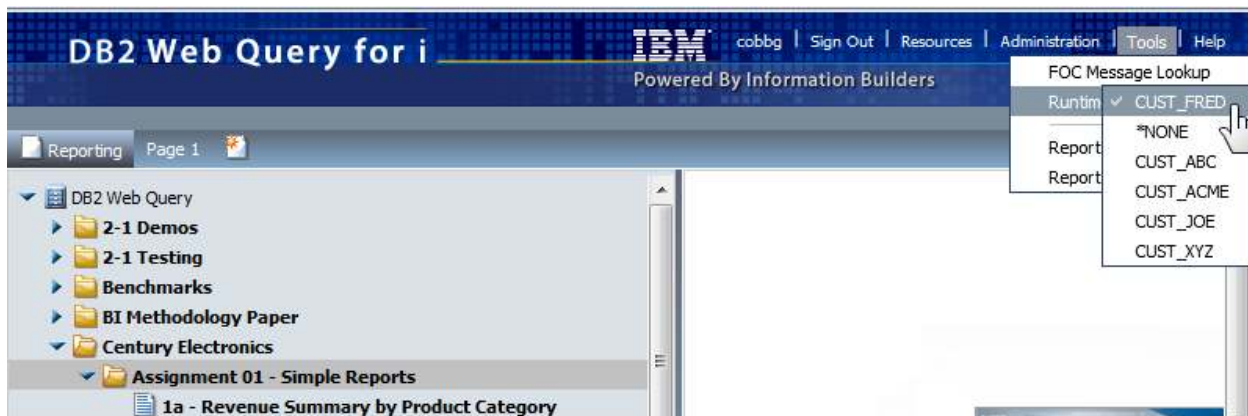

The list of RTEs assigned to the user is presented:

Here are some important points regarding this interface:

- The "Active" RTE always appears first on the list and is checked.
- *NONE always appears on list and can be selected. This simply means no RTE is activated or executed (nothing happens)
- From this list the user can change the active RTE by selecting it.
- The active RTE is persistent and remains in effect for that user throughout subsequent logoff/logon requests and other Web Query interfaces such as the Application Extension and web service function calls. It will remain the active one until the user or an administrator (using the WRKWQRTE command) changes it.

With an active RTE in place, when the user runs a report based on a one part name synonym, the RTE is executed and calls the exit program (if not *NONE) and changes the library list (if not *NONE) prior to

report execution. If the tables referenced in the synonym (that report is based upon) are found in the library list, the report will return data as shown. Notice this sample report returned two years' worth of data:

To change the active RTE, simply select a different one from the list:
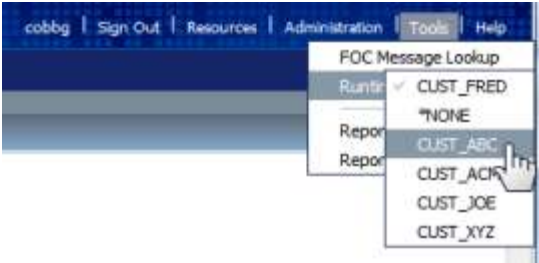
When user runs that same report again, tables in a different library are found (based on the different library list defined in the RTE) and subsequently different results are returned. Notice this report has four years' worth of data:

DB2 Web Query
- 2-1 Demos
- 2-1 Testing
- Benchmarks
- BI Methodology Paper
- Century Electronics
  - Assignment 01 - Simple Reports
    - 1a - Revenue Summary by Product Category
    - 1b - Top 10 Products by Margin
    - 1c - Revenue Summary with Subtotals
    - 1d - Revenue and Returns by Type
  - Assignment 02 - Implementing Various Date Functions in Report
    - 2a - Monthly Orders by Country, Region, Store
    - 2a - Weekly Orders by Country, Region
    - 2c - Seasonal Orders by Country, Region
  - Assignment 03 - Other Report Features and Format
  - Assignment 04 - Charting
  - Assignment 05 - Adding Filters to Reports
  - Assignment 06 - Detail and Print Reports
  - Assignment 07 - Implementing OLAP
  - Assignment 08 - Building Documents and Dashboards
  - Assignment 09 - Exploring and comparing other Dashboard optic

Seasonal Orders by Country, Region, Store

Country	Region	Store Name	DC_SEASON	Number Of Orders 2009	2010	2011	2012
Canada	Eastern Canada	ABC Electronics	Autumn	144	143	144	143
			Spring	312	312	312	312
			Summer	182	194	182	194
			Winter	206	206	206	206
		Planete Digitale	Autumn	10	10	10	10
			Spring	20	20	20	20
			Summer	10	5	10	5
			Winter	5	5	5	5
		Winnipeg Audio	Spring	31	26	31	26
			Summer	11	5	11	5
			Winter	31	31	31	31
	Western Canada	Home Audio Outfitters	Autumn	40	45	40	45
			Spring	35	35	35	35
			Summer	45	32	45	32
			Winter	63	63	63	63
France	Ile de France	Images et Son	Autumn	29	26	29	26
			Spring	22	22	22	22
			Summer	41	41	41	41
			Winter	6	6	6	6
	Nord-Pas-De-Calais	Videomarche	Autumn	5	11	5	11

Troubleshooting

This section contains information about how to troubleshoot RTE problems.

Logging

Each RTE has a logging feature flag. If the flag is turned on, each DB2 Web Query request that uses the RTE will insert a row to a log table. If you think something is wrong with RTEs in your environment you can turn on logging and examine the contents of the log table.

The values for this flag are as follows

- '0'= Off (shipped/default value)
- '1'=On.

Currently there is no interface in WRKWQRTE to turn on/off. In order to turn on logging for a specific RTE an administrator must execute the following SQL statement:

```
UPDATE QWEBQRY/QWQARTMENV
   SET WQRTELOGFL = '1'
   WHERE WQRTENAME = 'RTE_NAME'
```

To turn on logging for all RTE's, use this SQL statement:

```
UPDATE QWEBQRY/QWQARTMENV
   SET WQRTELOGFL = '1'
```

The logging table is QWEBQRY/QWQARTELOG and has the following format:

Column Name	Data Type	Length	Scale	Heading
WQQUERYID	CHAR	12	.	Query ID – Unique ID of query execution
WQQRYTIM	TIMESTMP	10	.	Query run timestamp
WQUSER	CHAR	10	.	User Profile making request
WQFULLPATH	CHAR	256	.	Full path of report
WQRTEID	INTEGER	4	0	The ID of the RTE that was used. To get the name of the RTE, join to table QWEBQRY/QWQARTMENV using this column as the join field.
WQRTERC	INTEGER	4	0	Return Code
WQRTEMSGID	CHAR	7	.	Message ID
WQRTEMSG	CHAR	80	.	Message

Here is an example log table:

WQQUERYID	WQQRYTIM	WQUSER	WQFULLPATH	WQRTEID	WQRTERC	WQRTEMSGID	WQRTEMSG
QWQ000078156	2013/03/19 16:53:51.285583	COBBG	IBFS:/WFC/Repository/RTEtesting/System_column_report.fex	4	-5	CPF2110	Library ZZZ not found.
QWQ000078159	2013/03/19 16:58:19.859977	COBBG4	IBFS:/WFC/Repository/RTEtesting/System_column_report.fex	6	0	CPC2101	Success! RTE found and library list changed.
QWQ000078160	2013/03/19 16:59:37.551662	COBBG4	IBFS:/WFC/Repository/RTEtesting/System_column_report.fex	6	0	CPC2101	Success! RTE found and library list changed.
QWQ000078161	2013/03/19 17:00:32.469692	COBBG4	IBFS:/WFC/Repository/RTEtesting/System_column_report.fex	6	0	CPC2101	Success! RTE found and library list changed.

The first row was for an unsuccessful attempt of changing the library list. In this case the library named ZZZ (defined in the RTE) was deleted. Consequently an attempt by the RTE to change the library list to one that included this library failed and message CPF2110 was thrown. The other rows contain information of successful RTE CHGLIBL requests.

Hanging Reports

If you have RTEs set up and begin to experience "hanging reports" (report requests that never seem to end and return data), there are steps you can take to determine if the problem is caused by an RTE.

RTE's provide two capabilities:
1. Change the library list
2. Call an exit program

To change the library list, the RTE issues the CHGLIBL command. Any errors returned by this command are monitored for and captured in the RTE logging table (if logging is turned on). Consequently a hanging report should not be caused by a failed CHGLIBL command request.

A poorly written exit program, on the other hand, can and will cause a report request to hang. This is the inherent risk of exit points and exit programs: unmonitored errors in the program that are awaiting a response to an inquiry message in QSYSOPR message queue. To determine if this is happening, you should first check the message queue using the following command:
DSPMSG MSGQ(QSYSOPR)

Here is an example of what this might look like:

```
                          Display Messages
                                              System:     ERPSF4
Queue . . . . . :    QSYSOPR            Program . . . . :    *DSPMSG
  Library . . . :      QSYS               Library . . . :
Severity  . . . :    60                  Delivery  . . . :    *HOLD


Type reply (if required), press Enter.
  The call to COBBG/WQEXIT1 ended in error (C G D F).
    Reply . . .    _
```

To determine the reason this inquiry message was thrown, you must find the job and examine the joblog. Because DB2 Web Query uses server side CLI for database access, all database requests

(including the RTE stored procedure call) occur in the QSQSRVR jobs started in the QWEBQRY21 subsystem. So this is where you should look for your hanging job. Issue the following command and look for QSQSRVR job in MSGW status:
WRKSBSJOB SBS(QWEBQRY21)

```
                          Work with Subsystem Jobs                      ERPSF4
                                                           03/20/13   09:35:37
 Subsystem  . . . . . . . . . . . :     QWEBQRY21


 Type options, press Enter.
   2=Change   3=Hold    4=End    5=Work with   6=Release   7=Display message
   8=Work with spooled files    13=Disconnect


 Opt  Job          User        Type     -----Status-----  Function
  __   QP0ZSPWP    QWQADMIN    BATCHI   ACTIVE            JVM-SCHSchedul
  __   QSQSRVR     QUSER       PJ       ACTIVE
  __   QSQSRVR     QUSER       PJ       ACTIVE
       QSQSRVR     QUSER       PJ       ACTIVE
  __   QSQSRVR     QUSER       PJ       MSGW
  __   QSQSRVR     QUSER       PJ       ACTIVE
  __   STRBROKER   QWQADMIN    BATCH    ACTIVE            CMD-QSH
  __   TSCOM3      QWQADMIN    BATCHI   ACTIVE            PGM-TSCOM3
                                                                      More
```

Take option 5 to work with this job and then option 10 to display the joblog. An example joblog is shown below:

```
                          Display Job Log
                                                    System:     ERPSF4
 Job . . :   QSQSRVR          User . . :   QUSER         Number . . . :    465184


     Job 465184/QUSER/QSQSRVR started on 03/18/13 at 15:48:12 in subsystem
       QWEBQRY21 in QWEBQRY. Job entered system on 03/18/13 at 15:48:12.
     ACGDTA for 465184/QUSER/QSQSRVR not journaled; reason 1.
     Printer device PRT01 not found.
     Errors on CHGJOB command for job 465184/QUSER/QSQSRVR.
     Printer device PRT01 not found.
     Job changed successfully; however errors occurred.
     ACGDTA for 465184/QUSER/QSQSRVR not journaled; reason 1.
     User Profile = COBBG
     SERVER MODE CONNECTING JOB IS 465502/QWQADMIN/TSCOM3.
     The following special registers have been set: CLIENT_ACCTNG: COBBG    ,
       CLIENT_APPLNAME: DB2 WebQuery, CLIENT_PROGRAMID: DB2WBQRY,
       CLIENT_USERID: COBBG    , CLIENT_WRKSTNNAME: 465502/QWQADMIN/TSCOM3
     Total parameters passed does not match number required.
                                                             More...

 Press Enter to continue.
```

In this case the administrator or developer has written an exit program that did not contain the correct parameter list. Consequently the RTE exit program request failed.

Displaying Active RTE information in reports

You may find it useful or necessary to include information about a user's Active RTE in a DB2 Web Query report. To provide this information, the product ships with the following stored procedure:

qwebqry.qwq_display_runtime_environment (current_user CHAR(10))

This stored procedure returns information about the current active Runtime Environment and the actual library list for the current user. It can be used to validate/test results of the changing the active RTE.

A seven column result set is returned. Each row in the result set represents an individual library in the user's actual library list. Consequently, the number of rows returned is equivalent to the number of libraries in the user's actual library list. The columns returned in the result set are as follows:

Column Name	Data Type	Length	Heading
RTE_Name	CHAR	10	Name of Active RTE
RTE_Description	CHAR	50	Description of Active RTE
RTE_ExitProgram	CHAR	21	Name and library of Active RTE Exit Program
RTE_LibraryList	CHAR	275	Library list of Active RTE
Actual_LIBL_seqnum	INT		Sequence number (order) of library in actual library list
Actual_LIBL_libname	CHAR	10	Name of library in actual library list
Actual_LIBL_libtype	CHAR	3	Type of library in actual library list • SYS – System portion of library list • PRD – Production library • CUR – Current library • USR – User portion of library list

A sample result set is provided below:

RTE_NAME	RTE_DESCRIPTION	RTE_EXITPROGRAM	RTE_LIBRARYLIST	ACTUALLIBL_SEQNUM	ACTUALLIBL_LIBNAME	ACTUALLIBL_LIBTYPE
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	1	QSYS	SYS
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	2	QSYS2	SYS
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	3	QHLPSYS	SYS
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	4	QUSRSYS	SYS
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	5	QIWS	PRD
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	6	QWQCENT	USR
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	7	QWQCENTDEM	USR
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	8	QTEMP	USR
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	9	QGPL	USR
CUST_FRED	Runtime environment for customer FRED ...	*NONE	QWQCENT QWQCENTDEM QTEMP QGPL FLGHT400C ...	10	FLGHT400C	USR

To use this stored procedure in a report, create a DB2 Web Query synonym over it. Name the synonym **QWQ_DISPLAY_RUNTIME_ENVIRONMENT**. Next create a report based on that synonym. A sample of the FEX source is provided below:

```
TABLE FILE QWQ_DISPLAY_RUNTIME_ENVIRONMENT
BY QWQ_DISPLAY_RUNTIME_ENVIRONMENT.ANSWERSET1.ACTUALLIBL_LIBTYPE
BY QWQ_DISPLAY_RUNTIME_ENVIRONMENT.ANSWERSET1.ACTUALLIBL_SEQNUM
BY QWQ_DISPLAY_RUNTIME_ENVIRONMENT.ANSWERSET1.ACTUALLIBL_LIBNAME
WHERE QWQ_DISPLAY_RUNTIME_ENVIRONMENT.INPUT.USER_PROFILE EQ '&FOCUSER';
ON TABLE SUBHEAD
"Active RTE: <QWQ_DISPLAY_RUNTIME_ENVIRONMENT.ANSWERSET1.RTE_NAME<+0> "
" "
"RTE Library List: "
"<QWQ_DISPLAY_RUNTIME_ENVIRONMENT.ANSWERSET1.RTE_LIBRARYLIST<+0> "
" "
"User: <+0>&FOCUSER"
ON TABLE PCHOLD FORMAT HTML
ON TABLE NOTOTAL
ON TABLE SET CACHELINES 100
ON TABLE SET PAGE-NUM NOLEAD
ON TABLE SET SQUEEZE ON
ON TABLE SET HTMLCSS ON
ON TABLE SET HTMLENCODE OFF
ON TABLE SET EMPTYREPORT ON
ON TABLE SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/javaassist/intl/EN/combine_templates/ENgray_theme.sty,$
TYPE=REPORT, TITLETEXT=&WF_TITLE.QUOTEDSTRING, SUMMARY=&WF_SUMMARY.QUOTEDSTRING, ORIENTATION=LANDSCAPE, $
TYPE=TABHEADING, LINE=1, JUSTIFY=LEFT, $
TYPE=TABHEADING, LINE=1, ITEM=1, OBJECT=TEXT, SIZE=14, STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=1, ITEM=1, OBJECT=FIELD, SIZE=14, COLOR=RGB(0 0 255), STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=1, ITEM=2, OBJECT=TEXT, SIZE=14, COLOR=RGB(0 0 255), STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=2, JUSTIFY=LEFT, $
TYPE=TABHEADING, LINE=2, ITEM=1, OBJECT=TEXT, SIZE=12, STYLE=NORMAL, $
TYPE=TABHEADING, LINE=3, JUSTIFY=LEFT, $
TYPE=TABHEADING, LINE=3, ITEM=1, OBJECT=TEXT, SIZE=14, STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=4, JUSTIFY=LEFT, $
TYPE=TABHEADING, LINE=4, ITEM=1, OBJECT=FIELD, SIZE=14, COLOR=RGB(0 0 255), STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=4, ITEM=1, OBJECT=TEXT, SIZE=14, COLOR=RGB(0 0 255), STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=5, JUSTIFY=LEFT, $
TYPE=TABHEADING, LINE=5, ITEM=1, OBJECT=TEXT, SIZE=12, STYLE=NORMAL, $
TYPE=TABHEADING, LINE=6, JUSTIFY=LEFT, $
TYPE=TABHEADING, LINE=6, ITEM=1, OBJECT=TEXT, SIZE=14, STYLE=BOLD+ITALIC, $
TYPE=TABHEADING, LINE=6, ITEM=2, OBJECT=TEXT, SIZE=14, COLOR=RGB(0 0 255), STYLE=BOLD+ITALIC, $
ENDSTYLE
END
```

Example output from running this report: