

DB2 Web Query New Features

Release 2.2.0

Active Technologies, EDA, EDA/SQL, FIDEL, FOCUS, Information Builders, the Information Builders logo, iWay, iWay Software, Parlay, PC/FOCUS, RStat, Table Talk, Web390, WebFOCUS, WebFOCUS Active Technologies, and WebFOCUS Magnify are registered trademarks, and DataMigrator and Hyperstage are trademarks of Information Builders, Inc.

Adobe, the Adobe logo, Acrobat, Adobe Reader, Flash, Adobe Flash Builder, Flex, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Due to the nature of this material, this document refers to numerous hardware and software products by their trademarks. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2016, by Information Builders, Inc. and iWay Software. All rights reserved. Patent Pending. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

Contents

- 1. Web Query Version 2.2.0 - April 2016.....7**
- DB2 Web Query for i Enhancements.....8
 - Microsoft Windows 10 Support.....8
 - Business Intelligence Portal Enhancement.....8
- InfoAssist Enhancements.....9
 - Rearranging HOLD File Components.....9
 - New Options on the Home Tab.....10
 - File & Printer Format DB2 Append Option.....10
 - Options for Aligning a Page Footer.....11
 - Color of Line Indicator in Query Design Pane Based on Theme.....11
 - Addition of Calendar Feature.....11
 - Support for Microsoft Excel 2007 Added to Chart and Document Mode.....11
 - Implementation of Dynamic Grouping.....11
 - Addition of Field List Search Functionality.....12
- Work With DB2 Web Query Command (WRKWEBQRY) Enhancement13
- QWQREPOS and QWQCEN User Libraries Enhancement.....13
- Developer Workbench Enhancements.....14
 - Using the Autosize Children Option.....14
 - Designing Content for Smartphones in the HTML Canvas.....15
 - Saving Control Selections in a Browser Session.....17
 - Previewing HTML Output Across Different Browsers.....17
 - Short Last Modified Date Format.....18
- Reporting Language Enhancements.....19
 - Embedded Images in Microsoft Excel (XLSX) Workbooks.....19
 - Web Query Client Amper Autoprompt Internal Processing.....20
 - Simplified Character Functions.....20
 - Simplified Date and Date-Time Functions.....39
 - Adding a Value Test to RESTRICT=NOPRINT.....47
 - Using a SYNONYM or FOCEXEC for ACCEPT Values in a Master File.....47

Using Multiple Prefix Operators on the Same Measure in SUBTOTAL.....	49
Report Broker Enhancements.....	50
Addition of Internal Variables.....	50
Showing Only Folders with Content.....	50
Creating Schedules With Parameterized Settings.....	51
View Logs and Traces for Daily and On-Demand Log Purge and Library Expiration Jobs.....	51
Updating Global FTP Password for All Schedules.....	52
Viewing Trace Information in the Report Broker Console.....	52
Versioning Added for Scheduler.log Files.....	52
SFTP With Multi-Factor Authentication.....	52
Web Query Client Administration and Security Enhancements.....	53
New ZIP All Button Captures Traces.....	53
Centralized Validation of Product Variables.....	53
Reporting Server Enhancements.....	54
Reporting on the Impact of Changing Adapter Connection Names.....	54
Setting Offline Print Attributes on IBM i.....	54
Adapter Enhancements.....	55
New Adapters.....	55
SQL Adapters.....	56
Adapter for DB2.....	60
Adapter for Microsoft SQL Server.....	61
Sequential and Indexed Adapters.....	62
Known Issues.....	63
Business Intelligence Portal.....	63
InfoAssist.....	64
Developer Workbench.....	64
JD Edwards Adapters.....	65
Report Broker.....	65
Spreadsheet Client.....	66
National Language Support.....	66
DataMigrator/Data Management Console.....	67
Metadata.....	67
Documentation	68

Web Browser Support.....70
Mobile Browser Support.....74

1 Web Query Version 2.2.0 - April 2016

This documentation describes new features, known issues, web browser support, and mobile support for the April 2016 - 2.2.0 release.

This document is intended for all levels of users, including application developers, administrators, and end users. It is also intended to serve as a quick reference for users upgrading from a prior version.

Topics:

- ❑ DB2 Web Query for i Enhancements
- ❑ InfoAssist Enhancements
- ❑ Work With DB2 Web Query Command (WRKWEBQRY) Enhancement
- ❑ QWQREPOS and QWQCENT User Libraries Enhancement
- ❑ Developer Workbench Enhancements
- ❑ Reporting Language Enhancements
- ❑ Report Broker Enhancements
- ❑ Web Query Client Administration and Security Enhancements
- ❑ Reporting Server Enhancements
- ❑ Adapter Enhancements
- ❑ Known Issues
- ❑ Web Browser Support
- ❑ Mobile Browser Support

DB2 Web Query for i Enhancements

In this section:

Microsoft Windows 10 Support

Business Intelligence Portal Enhancement

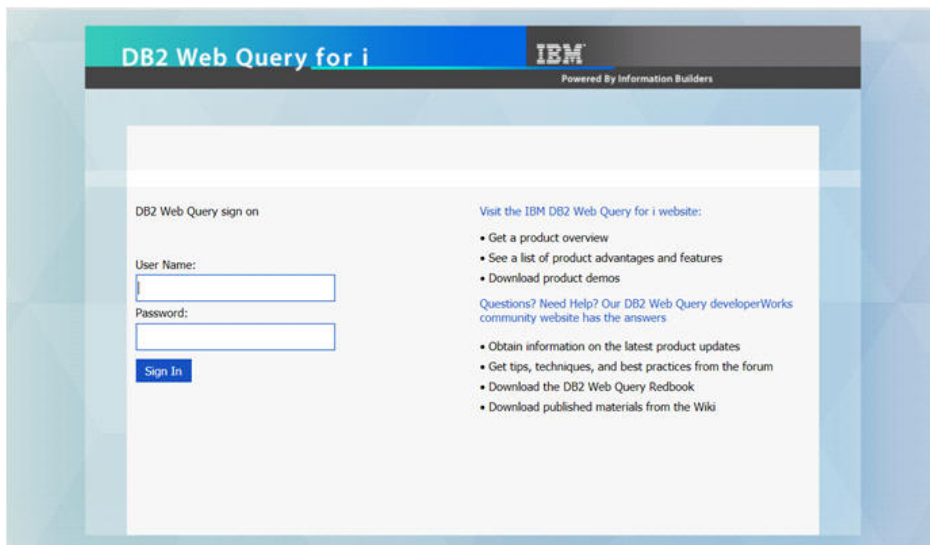
The following new features apply to DB2 Web Query for i.

Microsoft Windows 10 Support

Microsoft® Windows® 10 is now supported with DB2 Web Query and Developer Workbench Release 2.2.0.

Business Intelligence Portal Enhancement

The DB2 Web Query for i sign on page and banner have been redesigned, as shown in the following image. The banner has been enhanced to accommodate wide screen monitors.



InfoAssist Enhancements

In this section:

Rearranging HOLD File Components

New Options on the Home Tab

File & Printer Format DB2 Append Option

Options for Aligning a Page Footer

Color of Line Indicator in Query Design Pane Based on Theme

Addition of Calendar Feature

Support for Microsoft Excel 2007 Added to Chart and Document Mode

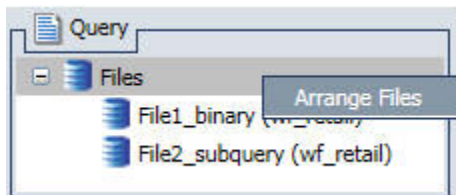
Implementation of Dynamic Grouping

Addition of Field List Search Functionality

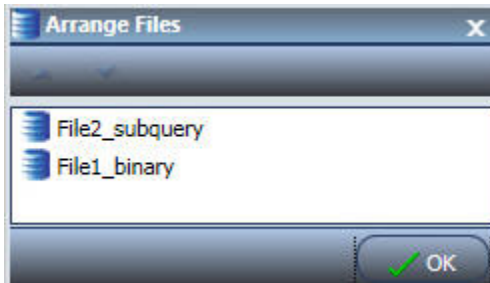
The following new features apply to InfoAssist reports, charts, and documents.

Rearranging HOLD File Components

When working with multiple HOLD files or subqueries, you can change the order in which they are referenced. When you right-click *Files* in the Query pane, the Arrange Files option displays, as shown in the following image.



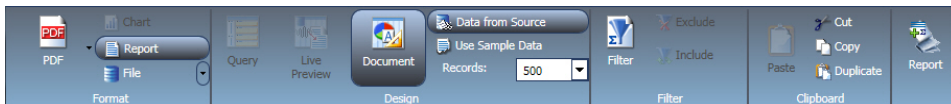
When you click *Arrange Files*, the Arrange Files dialog box displays, as shown in the following image.



You can select a file to move up or down using the arrow keys that are enabled upon file selection.

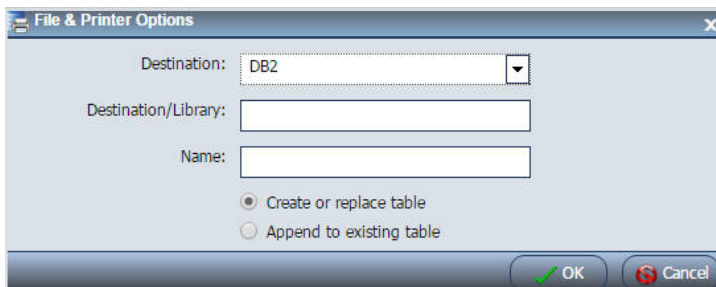
New Options on the Home Tab

When working in Document Mode, in InfoAssist, a new group, Clipboard, has been added to the Home tab. Functions in this group enable you to cut, copy, paste, and duplicate items in your document. The Clipboard group is shown in the following image.




File & Printer Format DB2 Append Option

On the InfoAssist Home tab, the File & Printer Format option to direct report output to DB2 has been enhanced to support an append option, as shown in the following image.



By default, report output to DB2 will create a new table. Select the *Append to existing table* option if you wish to add records to an existing table.

Options for Aligning a Page Footer

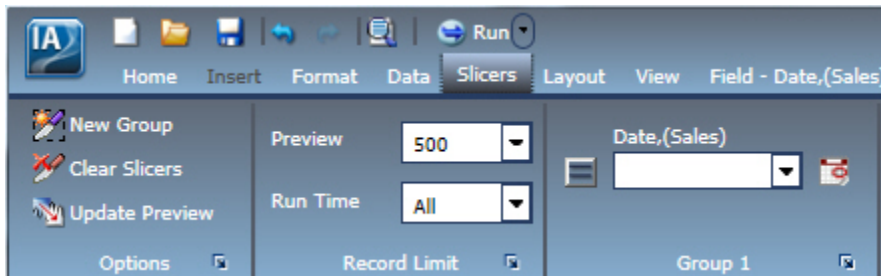
The Additional alignment options icon  , which is located in the Page Footer section of the Header & Footer dialog box, provides options for aligning your page footer. You can place the footing that you specify at the bottom of your report relative to the data or to the page.

Color of Line Indicator in Query Design Pane Based on Theme

When clicking and dragging fields within the Query Design pane, a line indicator displays. The color of this line is determined by the theme.

Addition of Calendar Feature

When working with full date fields while using Traffic Lights, Filtering, or Slicers, a calendar icon now displays in development mode or at run time. The calendar feature enables you to select a date using a calendar control. The calendar icon, as it displays for a Slicer on a date field, is shown in the following image.



Support for Microsoft Excel 2007 Added to Chart and Document Mode

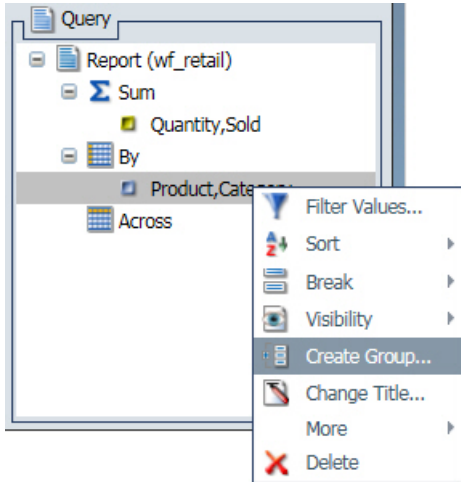
Microsoft Excel 2007 output format has been added to the Chart and Document Modes in InfoAssist. This enables the Excel (xlsx) and Excel Formula (xlsx) formats in areas of InfoAssist where an output format can be selected.

Implementation of Dynamic Grouping

Dynamic grouping allows you to create groups of elements based on the field data type that you select. Once you define a new group, a higher-level field is created that contains the selected elements. For example, in the Century database, there are a number of brands of televisions. Using the dynamic grouping functionality, you can create groups based on the popularity of a particular brand. The first group might include top sellers such as LG and Sony. The second group might contain the remaining brands (Panasonic, GPX, Supersonic, Tivax, and Audiovox). This would allow you to group top sellers into one group, and the remaining brands into another group.

Note: The Create Group option is only available for dimension fields of non-numeric format or attribute.

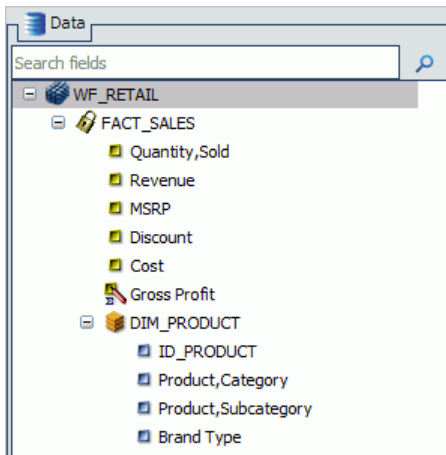
You can create a dynamic group using the Create Group option on the shortcut menu in the Query pane, as shown in the following image.



Note: You can also access the Create Group option from the shortcut menu in the Data pane.

Addition of Field List Search Functionality

You can use the field list search functionality to search for specific fields within a tree or list in the Resources panel. Searching occurs as you type in the Search fields box, which is located at the top of the Data pane, as shown in the following image.



Work With DB2 Web Query Command (WRKWEBQRY) Enhancement

The Work With DB2 Web Query (WRKWEBQRY) panel has been enhanced to show the Processor Core license usage and the active Web Query edition, as shown in the following image. For more detailed license information, use the Work with License Information (WRKLICINF) command.

```

4/08/16 15:01:51 Work with DB2 Web Query UT30P33
-----Usage Count-----
DB2 Web Query status: Active License Information Max Local All
Port Status Named Users *NOMAX 3 3
12331 Active Runtime Groups 0 0 0
12332 Active Dev Workbench users *NOMAX 0 0
12333 Active Processor Cores 2 2 2
11331 Active
12335 Active Product ID/Version . . . 5733WQX V2R2M0
12336 Active Active Edition . . . . Standard
12338 Active Latest group PTF level .
12339 Active All prerequisite met . . Yes

Type options, press Enter.
 1=End DB2 Web Query 4=End immediately 5=Work with Runtime Environments

F3=Exit F5=Refresh F12=Cancel
MA A MW 20/004

```

QWQREPOS and QWQCENT User Libraries Enhancement

The Web Query repository library, QWQREPOS, and the sample Century Electronics database library, QWQCENT, are changed to user libraries. They will be saved with SAVLIB *ALLUSR and will no longer be saved by SAVLIB *IBM.

Developer Workbench Enhancements

In this section:

- Using the Autosize Children Option
- Designing Content for Smartphones in the HTML Canvas
- Saving Control Selections in a Browser Session
- Previewing HTML Output Across Different Browsers
- Short Last Modified Date Format

Developer Workbench has a brand new look and feel. This release makes use of the common design paradigm seen in many Windows desktop products: the ribbon, panels, and development canvases.

The following new features apply to Developer Workbench.

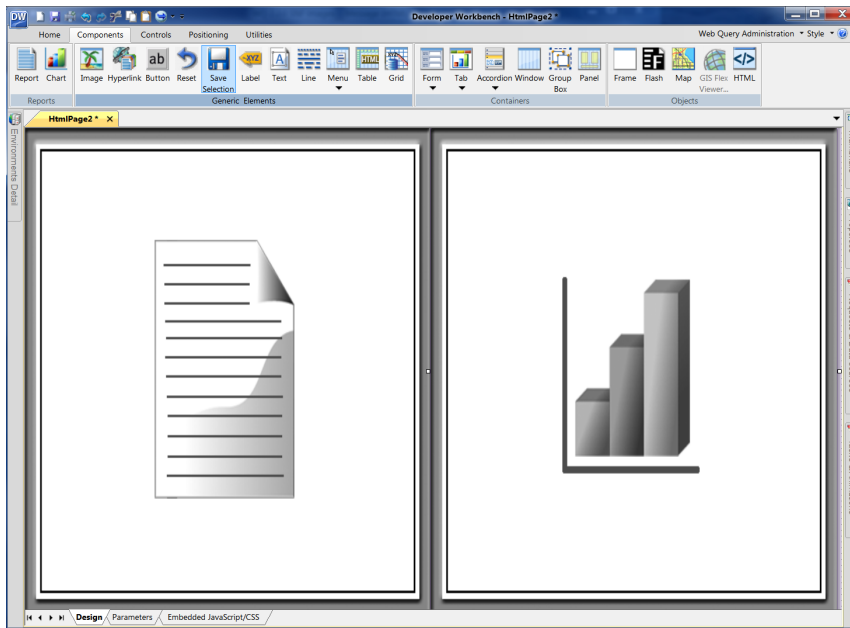
Using the Autosize Children Option

The Autosize Children option resizes all components on an HTML page to automatically fit the maximum available screen space on any monitor or device where it might be viewed. For example, a single component will take up one hundred percent of the available space, while two components will take up fifty percent each.

This option is used so that when your HTML page is run on different devices, components and controls do not appear off the screen and have to be scrolled to in order to use them. To accommodate this, the orientation of components may be different on different devices. For example, a report and chart that appear side-by-side when run on a desktop monitor, may appear stacked one above the other when run on a tablet.

You can prevent the component from becoming too small, using the Autosize: min width and Autosize: min height options. Type the minimum number of pixels for the width, height, or both, in the appropriate fields. When a component reaches the minimum width or height that you specified, a scroll bar appears and the size of the component does not decrease further.

The following image shows a report component and a chart component on the HTML canvas where Autosize children is enabled. Both components share the available space equally.



Designing Content for Smartphones in the HTML Canvas

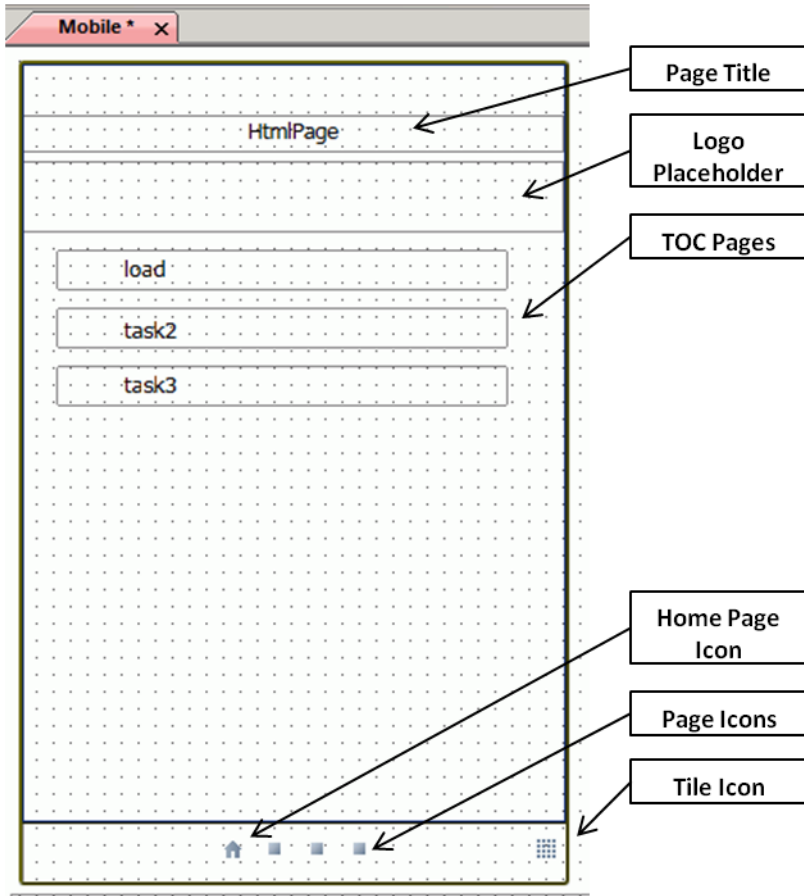
You can design content specifically to be used on smartphones, in the HTML canvas, using the new mobile layout options and enhancements. Content designed in this way is automatically sized, and takes full advantage of the page-swipe navigation and touch-screen capabilities of smartphones.

There are two different layouts for designing HTML output for smartphones, Basic and Advanced.

Basic Mobile Layout. Enables you to build simple smartphone pages that you can navigate using graphical icons and page-swipe. The Basic Mobile Layout displays individual frames from your HTML canvas on individual mobile pages, by default. You can change this as needed.

Advanced Mobile Layout. Offers the same features as the Basic, but also gives you the additional ability to add a table of contents with or without a logo, and add a logo on each page.

You can use the options that are available with these two layouts to modify and view your mobile layout. In addition, you can switch between the two layouts to determine which design fits your application requirements. Both layouts can be applied to new or existing HTML pages.



Saving Control Selections in a Browser Session

When working with controls on HTML pages, you can link the controls so that they automatically default to the same selected value when you switch between multiple pages in the same browser session. To link the controls, you need to assign a common value to the *Global name* property for each control that you want to link, as this field enables you to save control selections within a browser session.

This functionality enables you to link controls based on commonality. For example, you may have two HTML pages that contain controls with information that relates to regional sales. You can assign a value to the *Global name* property, such as *Region*, to each of the controls that you want to link. When you run those pages in the same browser session, and choose a value from one of the linked controls, such as *Southeast*, the controls on the other pages will refresh and display the information for the Southeast, by default.

When you select a control on an HTML page, you can access the *Global name* field in the Properties panel, under Miscellaneous. Once you have defined values for the *Global name* property for the controls that you want to link, save the HTML page.

Note: A selected value is retained as the default only during a single browser session. The value is not retained after you close the browser.

Previewing HTML Output Across Different Browsers

You can now preview your HTML output in different browsers using the icons that are available in the Preview group on the Utilities tab. This allows you to review how your output will display across different browsers, and verify that your content displays correctly.

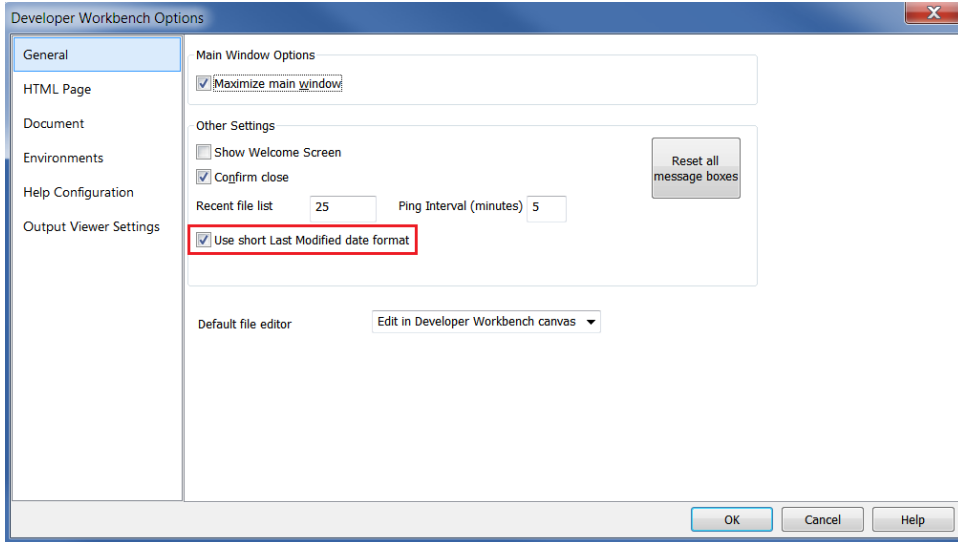
The Preview group contains browser icons based on those that you have installed locally. For example, if you do not have a local installation of Firefox, the Firefox browser icon is unavailable (grayed out) in the Preview group.

Easy access to the three standard browsers (Internet Explorer, Chrome, and Firefox) makes it easy to compare the display of your HTML output, as shown in the following image.

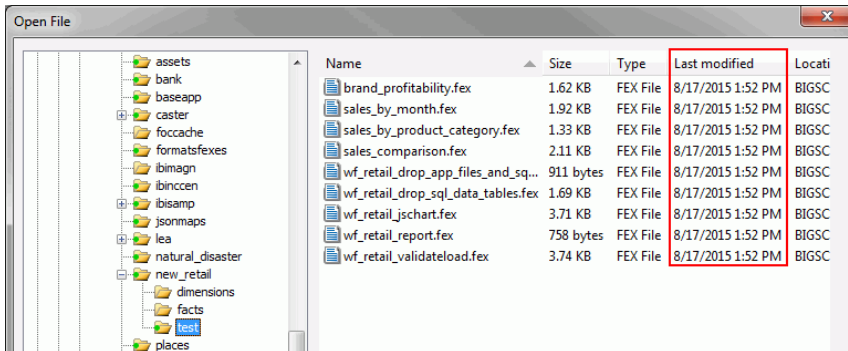


Short Last Modified Date Format

A new option, *Use short Last Modified date format*, is selected in the Developer Workbench Options dialog box, by default, as shown in the following image.



This option displays the last modified date information for a file or folder, in a short format, *mm/dd/yyyy hh:mm AM/PM*. This date information is available in the Open File dialog box and the Environments Detail panel, as shown in the following image.



Reporting Language Enhancements

In this section:

Embedded Images in Microsoft Excel (XLSX) Workbooks

Web Query Client Amper Autoprompt Internal Processing

Simplified Character Functions

Simplified Date and Date-Time Functions

Adding a Value Test to RESTRICT=NOPRINT




Using a SYNONYM or FOCEXEC for ACCEPT Values in a Master File

Using Multiple Prefix Operators on the Same Measure in SUBTOTAL

The following new features apply to the reporting language.

Embedded Images in Microsoft Excel (XLSX) Workbooks

The XLSX workbooks can contain embedded images in fixed positions in each of the Web Query report areas including headers, footers, and data cells. Additionally, graphs can be placed on individual worksheets within compound workbooks, as shown in the following image.

Product Sales: Midwest				
	Actual		Budget	
Category	Dollars	Units	Dollars	Units
Coffee	\$4,178,513	332,777	\$4,086,032	335,526 
Food	\$4,338,271	341,414	\$4,220,721	339,263 
Gifts	\$2,883,881	230,854	\$2,887,620	232,318 
ALL Midwest	\$11,400,665	905,045	\$11,194,373	907,107

Web Query Client Amper Autoprompt Internal Processing

The performance and security of the Web Query Release 2.2.0 Amper Autoprompt facility has been improved because the Client no longer uses XSLT templates to create the JavaScript for the Autoprompt HTML page.

The Web Query Client uses the XML returned by the Reporting Server to create the JavaScript for the Autoprompt HTML page.

Simplified Character Functions

In this section:

CHAR_LENGTH: Returning the Length in Characters of a String

DIGITS: Converting a Number to a Character String

LPAD: Left-Padding a Character String

LOWER: Returning a String With All Letters Lowercase

LTRIM: Removing Blanks From the Left End of a String

POSITION: Returning the First Position of a Substring in a Source String

RTRIM: Removing Blanks From the Right End of a String

SUBSTRING: Extracting a Substring From a Source String

RPAD: Right-Padding a Character String

TOKEN: Extracting a Token From a String

TRIM_: Removing Leading Characters, Trailing Characters, or Both From a String

UPPER: Returning a String With All Letters Uppercase

New character functions have been developed that make it easier to understand and enter the required arguments. These functions have streamlined parameter lists, similar to those used by SQL functions. In some cases, these simplified functions provide slightly different functionality than previous versions of similar functions.

The simplified functions do not have an output argument. Each function returns a value that has a specific data type.

When used in a request against a relational data source, these functions are optimized (passed to the RDBMS for processing).

Note: The simplified character functions are supported in Dialogue Manager.

CHAR_LENGTH: Returning the Length in Characters of a String

How to:

Return the Length of a String in Characters

The CHAR_LENGTH function returns the length, in characters, of a string. In Unicode environments, this function uses character semantics, so that the length in characters may not be the same as the length in bytes. If the string includes trailing blanks, these are counted in the returned length. Therefore, if the format source string is type A_n , the returned value will always be n .

Syntax: How to Return the Length of a String in Characters

```
CHAR_LENGTH(source_string)
```

where:

source_string

Alphanumeric

Is the string whose length is returned.

The data type of the returned length value is Integer.

Example: Returning the Length of a String

The following request against the EMPLOYEE data source creates a virtual field named LASTNAME of type A15V that contains the LAST_NAME with the trailing blanks removed. It then uses CHAR_LENGTH to return the number of characters.

```
DEFINE FILE EMPLOYEE
LASTNAME/A15V = RTRIM(LAST_NAME);
END
TABLE FILE EMPLOYEE
SUM LAST_NAME NOPRINT AND COMPUTE
NAME_LEN/I3 = CHAR_LENGTH(LASTNAME);
BY LAST_NAME
ON TABLE SET PAGE NOPAGE
END
```

The output is:

LAST_NAME	NAME_LEN
BANNING	7
BLACKWOOD	9
CROSS	5
GREENSPAN	9
IRVING	6
JONES	5
MCCOY	5
MCKNIGHT	8
ROMANS	6
SMITH	5
STEVENS	7

DIGITS: Converting a Number to a Character String

How to:

Convert a Number to a Character String

Reference:

Usage Notes for DIGITS

Given a number, DIGITS converts it to a character string of the specified length. The format of the field that contains the number must be Integer.

Syntax: How to Convert a Number to a Character String

`DIGITS(number, length)`

where:

number

Integer

Is the number to be converted, stored in a field with data type Integer.

length

Integer between 1 and 10

Is the length of the returned character string. If *length* is longer than the number of digits in the number being converted, the returned value is padded on the left with zeros. If *length* is shorter than the number of digits in the number being converted, the returned value is truncated on the left.

Example: Converting a Number to a Character String

The following request against the WF_RETAIL_LITE data source converts -123.45 and ID_PRODUCT to character strings:

```
DEFINE FILE WF_RETAIL_LITE
MEAS1/I8=-123.45;
DIG1/A6=DIGITS(MEAS1,6) ;
DIG2/A6=DIGITS(ID_PRODUCT,6) ;
END
TABLE FILE WF_RETAIL_LITE
PRINT MEAS1 DIG1
ID_PRODUCT DIG2
BY PRODUCT_SUBCATEG
WHERE PRODUCT_SUBCATEG EQ 'Flat Panel TV'
ON TABLE SET PAGE NOPAGE
END
```

The output is:

Product Subcategory	MEAS1	DIG1	ID Product	DIG2
Flat Panel TV	-123	000123	4012	004012
	-123	000123	4017	004017
	-123	000123	4018	004018
	-123	000123	4017	004017
	-123	000123	4017	004017
	-123	000123	4018	004018
	-123	000123	4018	004018
	-123	000123	4017	004017
	-123	000123	4014	004014
	-123	000123	4016	004016
	-123	000123	4016	004016
	-123	000123	4018	004018
	-123	000123	4017	004017
	-123	000123	4018	004018
	-123	000123	4018	004018
	-123	000123	4017	004017
	-123	000123	4016	004016
	-123	000123	4018	004018
	-123	000123	4016	004016
	-123	000123	4018	004018
	-123	000123	4017	004017
	-123	000123	4018	004018
	-123	000123	4017	004017
	-123	000123	4017	004017
	-123	000123	4014	004014
	-123	000123	4018	004018

Reference: Usage Notes for DIGITS

- ❑ Only I Format numbers will be converted. D, P, and F formats generate error messages and should be converted to I before using the DIGITS function. The limit for the number that can be converted is 2GB.
- ❑ Negative integers are turned into positive integers.
- ❑ Integer formats with decimal places are truncated.
- ❑ DIGITS is not supported in Dialogue Manager.

LPAD: Left-Padding a Character String**How to:**

Pad a Character String on the Left

Reference:

Usage Notes for LPAD

LPAD uses a specified character and output length to return a character string padded on the left with that character.

Syntax: How to Pad a Character String on the Left

`LPAD(string, out_length, pad_character)`

where:

string

Fixed length alphanumeric

Is a string to pad on the left side.

out_length

Integer

Is the length of the output string after padding.

pad_character

Fixed length alphanumeric

Is a single character to use for padding.

Example: Left-Padding a String

In the following request against the WF_RETAIL data source, LPAD left-pads the PRODUCT_CATEGORY column with @ symbols:

```
DEFINE FILE WF_RETAIL
LPAD1/A25 = LPAD(PRODUCT_CATEGORY,25,'@');
DIG1/A4 = DIGITS(ID_PRODUCT,4);
END
TABLE FILE WF_RETAIL
SUM DIG1 LPAD1
BY PRODUCT_CATEGORY
ON TABLE SET PAGE NOPAGE
ON TABLE SET STYLE *
TYPE=DATA, FONT=COURIER, SIZE=11, COLOR=BLUE, $
END
```

The output is:

Product Category	DIG1	LPAD1
Accessories	5005	@@@@@@@@@@@@@@@@@@@@Accessories
Camcorder	3006	@@@@@@@@@@@@@@@@@@@@Camcorder
Computers	6016	@@@@@@@@@@@@@@@@@@@@Computers
Media Player	1003	@@@@@@@@@@@@@@@@@@@@Media Player
Stereo Systems	2155	@@@@@@@@@@@@@@@@@@@@Stereo Systems
Televisions	4018	@@@@@@@@@@@@@@@@@@@@Televisions
Video Production	7005	@@@@@@@@@@@@@@@@@@@@Video Production

Reference: Usage Notes for LPAD

- ❑ To use the single quotation mark as the padding character, you must double it and enclose the two single quotation marks within single quotation marks (LPAD(COUNTRY, 20,'''')). You can use an amper variable in quotation marks for this parameter, but you cannot use a field, virtual or real.
- ❑ Input can be fixed or variable length alphanumeric.
- ❑ Output, when optimized to SQL, will always be data type VARCHAR.

- ❑ If the output is specified as shorter than the original input, the original data will be truncated, leaving only the padding characters. The output length can be specified as a positive integer or an unquoted &variable (indicating a numeric).

LOWER: Returning a String With All Letters Lowercase

How to:

Return a String With All Letters Lowercase

The LOWER function takes a source string and returns a string of the same data type with all letters translated to lowercase.

Syntax: How to Return a String With All Letters Lowercase

```
LOWER(source_string)
```

where:

source_string

Alphanumeric

Is the string to convert to lowercase.

The returned string is the same data type and length as the source string.

Example: Converting a String to Lowercase

In the following request against the EMPLOYEE data source, LOWER converts the LAST_NAME field to lowercase and stores the result in LOWER_NAME:

```
TABLE FILE EMPLOYEE
PRINT LAST_NAME AND COMPUTE
LOWER_NAME/A15 = LOWER(LAST_NAME);
ON TABLE SET PAGE NOPAGE
END
```

The output is:

LAST_NAME	LOWER_NAME
-----	-----
STEVENS	stevens
SMITH	smith
JONES	jones
SMITH	smith
BANNING	banning
IRVING	irving
ROMANS	romans
MCCOY	mccoy
BLACKWOOD	blackwood
MCKNIGHT	mcknight
GREENSPAN	greenspan
CROSS	cross

LTRIM: Removing Blanks From the Left End of a String

How to:

Remove Blanks From the Left End of a String

The LTRIM function removes all blanks from the left end of a string.

Syntax: How to Remove Blanks From the Left End of a String

`LTRIM(source_string)`

where:

source_string

Alphanumeric

Is the string to trim on the left.

The data type of the returned string is AnV, with the same maximum length as the source string.

Example: Removing Blanks From the Left End of a String

In the following request against the MOVIES data source, the DIRECTOR field is right-justified and stored in the RDIRECTOR virtual field. Then LTRIM removes leading blanks from the RDIRECTOR field:

```
DEFINE FILE MOVIES
RDIRECTOR/A17 = RJUST(17, DIRECTOR, 'A17');
END
TABLE FILE MOVIES
PRINT RDIRECTOR AND
COMPUTE
TRIMDIR/A17 = LTRIM(RDIRECTOR);
WHERE DIRECTOR CONTAINS 'BR'
ON TABLE SET PAGE NOPAGE
END
```

The output is:

RDIRECTOR	TRIMDIR
-----	-----
ABRAHAMS J.	ABRAHAMS J.
BROOKS R.	BROOKS R.
BROOKS J.L.	BROOKS J.L.

POSITION: Returning the First Position of a Substring in a Source String**How to:**

Return the First Position of a Substring in a Source String

The POSITION function returns the first position (in characters) of a substring in a source string.

Syntax: How to Return the First Position of a Substring in a Source String

```
POSITION(pattern, source_string)
```

where:

pattern

Alphanumeric

Is the substring whose position you want to locate. The string can be as short as a single character, including a single blank.

source_string

Alphanumeric

Is the string in which to find the pattern.

The data type of the returned value is Integer.

Example: Returning the First Position of a Substring

In the following request against the EMPLOYEE data source, POSITION determines the position of the first capital letter I in LAST_NAME and stores the result in I_IN_NAME:

```
TABLE FILE EMPLOYEE
PRINT LAST_NAME AND COMPUTE
I_IN_NAME/I2 = POSITION('I', LAST_NAME);
ON TABLE SET PAGE NOPAGE
END
```

The output is:

LAST_NAME	I_IN_NAME
-----	-----
STEVENS	0
SMITH	3
JONES	0
SMITH	3
BANNING	5
IRVING	1
ROMANS	0
MCCOY	0
BLACKWOOD	0
MCKNIGHT	5
GREENSPAN	0
CROSS	0

RTRIM: Removing Blanks From the Right End of a String

How to:

Remove Blanks From the Right End of a String

The RTRIM function removes all blanks from the right end of a string.

Syntax: How to Remove Blanks From the Right End of a String

```
RTRIM(source_string)
```

where:

```
source_string
```

Alphanumeric

Is the string to trim on the right.

The data type of the returned string is AnV, with the same maximum length as the source string.

Example: Removing Blanks From the Right End of a String

The following request against the MOVIES data source creates the field DIRSLASH, that contains a slash at the end of the DIRECTOR field. Then it creates the TRIMDIR field, which trims the trailing blanks from the DIRECTOR field and places a slash at the end of that field:

```
TABLE FILE MOVIES
PRINT DIRECTOR NOPRINT AND
COMPUTE
DIRSLASH/A18 = DIRECTOR | '/';
TRIMDIR/A17V = RTRIM(DIRECTOR) | '/';
WHERE DIRECTOR CONTAINS 'BR'
ON TABLE SET PAGE NOPAGE
END
```

On the output, the slashes show that the trailing blanks in the DIRECTOR field were removed in the TRIMDIR field:

DIRSLASH		TRIMDIR
-----		-----
ABRAHAMS J.	/	ABRAHAMS J./
BROOKS R.	/	BROOKS R./
BROOKS J.L.	/	BROOKS J.L./

SUBSTRING: Extracting a Substring From a Source String**How to:**

Extract a Substring From a Source String

The SUBSTRING function extracts a substring from a source string. If the ending position you specify for the substring is past the end of the source string, the position of the last character of the source string becomes the ending position of the substring.

Syntax: **How to Extract a Substring From a Source String**

`SUBSTRING(source_string, start_position, length_limit)`

where:

source_string

Alphanumeric

Is the string from which to extract the substring. It can be a field, a literal in single quotation marks ('), or a variable.

start_position

Integer

Is the starting position of the substring in *source_string*. If the position is 0, it is treated as 1. If the position is negative, the starting position is counted backward from the end of *source_string*.

length_limit

Integer

Is the limit for the length of the substring. The ending position of the substring is calculated as *start_position* + *length_limit* - 1. If the calculated position beyond the end of the source string, the position of the last character of *source_string* becomes the ending position.

The data type of the returned substring is AnV.

Example: **Extracting a Substring From a Source String**

In the following request, POSITION determines the position of the first letter I in LAST_NAME and stores the result in I_IN_NAME. SUBSTRING then extracts three characters beginning with the letter I from LAST_NAME, and stores the results in I_SUBSTR.

```
TABLE FILE EMPLOYEE
PRINT
COMPUTE
I_IN_NAME/I2 = POSITION('I', LAST_NAME); AND
COMPUTE
I_SUBSTR/A3 =
SUBSTRING(LAST_NAME, I_IN_NAME, I_IN_NAME+2);
BY LAST_NAME
ON TABLE SET PAGE NOPAGE
END
```


The output is:

LAST_NAME	I_IN_NAME	I_SUBSTR
-----	-----	-----
BANNING	5	ING
BLACKWOOD	0	BL
CROSS	0	CR
GREENSPAN	0	GR
IRVING	1	IRV
JONES	0	JO
MCCOY	0	MC
MCKNIGHT	5	IGH
ROMANS	0	RO
SMITH	3	ITH
	3	ITH
STEVENS	0	ST

RPAD: Right-Padding a Character String

How to:

Pad a Character String on the Right

Reference:

Usage Notes for RPAD

RPAD uses a specified character and output length to return a character string padded on the right with that character.

Syntax: How to Pad a Character String on the Right

RPAD(string, out_length, pad_character)

where:

string

Alphanumeric

Is a string to pad on the right side.

out_length

Integer

Is the length of the output string after padding.

pad_character

Alphanumeric

Is a single character to use for padding.

Example: Right-Padding a String

In the following request against the WF_RETAIL data source, RPAD right-pads the PRODUCT_CATEGORY column with @ symbols:

```
DEFINE FILE WF_RETAIL
  RPAD1/A25 = RPAD(PRODUCT_CATEGORY,25,'@');
  DIG1/A4 = DIGITS(ID_PRODUCT,4);
END
TABLE FILE WF_RETAIL
  SUM DIG1 RPAD1
  BY PRODUCT_CATEGORY
  ON TABLE SET PAGE NOPAGE
  ON TABLE SET STYLE *
  TYPE=DATA, FONT=COURIER, SIZE=11, COLOR=BLUE, $
END
```

The output is:

Product Category	DIG1	RPAD1
Accessories	5005	Accessories@@@@@@@@@@@@@@@@
Camcorder	3006	Camcorder@@@@@@@@@@@@@@@@
Computers	6016	Computers@@@@@@@@@@@@@@@@
Media Player	1003	Media Player@@@@@@@@@@@@
Stereo Systems	2155	Stereo Systems@@@@@@@@
Televisions	4018	Televisions@@@@@@@@
Video Production	7005	Video Production@@@@

Reference: Usage Notes for RPAD

- ❑ The input string can be data type AnV, VARCHAR, TX, and An.
- ❑ Output can only be AnV or An.

- When working with relational VARCHAR columns, there is no need to trim trailing spaces from the field if they are not desired. However, with An and AnV fields derived from An fields, the trailing spaces are part of the data and will be included in the output, with the padding being placed to the right of these positions. You can use TRIM or TRIMV to remove these trailing spaces prior to applying the RPAD function.

TOKEN: Extracting a Token From a String

How to:

Extract a Token From a String

The TOKEN function extracts a token (substring) based on a token number and a delimiter character.

Syntax: How to Extract a Token From a String

`TOKEN(string, delimiter, number)`

where:

string

Fixed length alphanumeric

Is the character string from which to extract the token.

delimiter

Fixed length alphanumeric

Is a single character delimiter.

number

Integer

Is the token number to extract.

Example: Extracting a Token From a String

TOKEN extracts the second token from the PRODUCT_SUBCATEG column, where the delimiter is the letter P:

```
DEFINE FILE WF_RETAIL_LITE
TOK1/A20 =TOKEN(PRODUCT_SUBCATEG, 'P', 2);
END
TABLE FILE WF_RETAIL_LITE
SUM TOK1 AS Token
BY PRODUCT_SUBCATEG
ON TABLE SET PAGE NOPAGE
END
```

The output is:

Product Subcategory	Token
Blu Ray	
Boom Box	
CRT TV	
Charger	
DVD Players	layers
DVD Players - Portable	layers -
Flat Panel TV	anel TV
Handheld	
Headphones	hones
Home Theater Systems	
Portable TV	ortable TV
Professional	rofessional
Receivers	
Smartphone	hone
Speaker Kits	eaker Kits
Standard	
Streaming	
Tablet	
Universal Remote Controls	
Video Editing	
iPod Docking Station	od Docking Station

TRIM_: Removing Leading Characters, Trailing Characters, or Both From a String

How to:

Remove Leading Characters, Trailing Characters, or Both From a String

The TRIM_ function removes all occurrences of a single character from either the beginning of a string, the end of a string, or both.

Syntax: How to Remove Leading Characters, Trailing Characters, or Both From a String

```
TRIM_(trim_where, trim_character, source_string)
```

where:

trim_where

Keyword

Defines where to trim the source string. Valid values are:

- LEADING, which removes leading occurrences.
- TRAILING, which removes trailing occurrences.
- BOTH, which removes leading and trailing occurrences.

trim_character

Alphanumeric

Is a single character, enclosed in single quotation marks ('), whose occurrences are to be removed from *source_string*. For example, the character can be a single blank (' ').

source_string

Alphanumeric

Is the string to be trimmed.

The data type of the returned string is AnV.

Example: Trimming a Character From a String

In the following request, TRIM_ removes leading occurrences of the character 'B' from the DIRECTOR field:

```
TABLE FILE MOVIES
PRINT DIRECTOR AND
COMPUTE
TRIMDIR/A17 = TRIM_(LEADING, 'B', DIRECTOR);
WHERE DIRECTOR CONTAINS 'BR'
ON TABLE SET PAGE NOPAGE
END
```

The output is:

DIRECTOR	TRIMDIR
-----	-----
ABRAHAMS J.	ABRAHAMS J.
BROOKS R.	ROOKS R.
BROOKS J.L.	ROOKS J.L.

UPPER: Returning a String With All Letters Uppercase

How to:

Return a String With All Letters Uppercase

The UPPER function takes a source string and returns a string of the same data type with all letters translated to uppercase.

Syntax: How to Return a String With All Letters Uppercase

```
UPPER(source_string)
```

where:

source_string

Alphanumeric

Is the string to convert to uppercase.

The returned string is the same data type and length as the source string.

Example: Converting Letters to Uppercase

In the following request, LCWORD converts LAST_NAME to mixed case. Then UPPER converts the LAST_NAME_MIXED field to uppercase:

```
DEFINE FILE EMPLOYEE
LAST_NAME_MIXED/A15=LCWORD(15, LAST_NAME, 'A15');
LAST_NAME_UPPER/A15=UPPER(LAST_NAME_MIXED) ;
END
TABLE FILE EMPLOYEE
PRINT LAST_NAME_UPPER AND FIRST_NAME
BY LAST_NAME_MIXED
WHERE CURR_JOBCODE EQ 'B02' OR 'A17' OR 'B04';
ON TABLE SET PAGE NOPAGE
END
```

The output is:

LAST_NAME_MIXED	LAST_NAME_UPPER	FIRST_NAME
-----	-----	-----
Banning	BANNING	JOHN
Blackwood	BLACKWOOD	ROSEMARIE
Cross	CROSS	BARBARA
Mccoy	MCCOY	JOHN
Mcknight	MCKNIGHT	ROGER
Romans	ROMANS	ANTHONY

Simplified Date and Date-Time Functions**In this section:**

DTADD: Incrementing a Date or Date-Time Component

DTDIFF: Returning the Number of Component Boundaries Between Date or Date-Time Values

DTPART: Returning a Date or Date-Time Component in Integer Format

DTRUNC: Returning the Start of a Date Period for a Given Date

New date and date-time functions have been developed that make it easier to understand and enter the required arguments. These functions have streamlined parameter lists, similar to those used by SQL functions. In some cases, these simplified functions provide slightly different functionality than previous versions of similar functions.

The simplified functions do not have an output argument. Each function returns a value that has a specific data type.

When used in a request against a relational data source, these functions are optimized (passed to the RDBMS for processing).

Standard date and date-time formats refer to YYMD and HYYMD syntax (dates that are not stored in alphanumeric or numeric fields). Dates not in these formats must be converted before they can be used in the simplified functions. Literal date-time values can be used with the DT function.

All arguments can be either literals, field names, or amper variables.

DTADD: Incrementing a Date or Date-Time Component

How to:

Increment a Date or Date-Time Component

Reference:

Usage Notes for DTADD

Given a date in standard date or date-time format, DTADD returns a new date after adding the specified number of a supported component. The returned date format is the same as the input date format.

Syntax: **How to Increment a Date or Date-Time Component**

DTADD(date, component, increment)

where:

date

Date or date-time

Is the date or date-time value to be incremented.

component

Keyword

Is the component to be incremented. Valid components (and acceptable values) are:

- YEAR (1-9999)
- QUARTER (1-4)
- MONTH (1-12)
- WEEK (1-53). This is affected by the WEEKFIRST setting.
- DAY (of the Month, 1-31)
- HOUR (0-23)

- ❑ MINUTE (0-59)
- ❑ SECOND (0-59)

increment

Integer

Is the value (positive or negative) to add to the component.

Example: Incrementing the DAY Component of a Date

The following request against the WF_RETAIL data source adds three days to the employee date of birth:

```
DEFINE FILE WF_RETAIL
NEWDATE/YYMD = DTADD( DATE_OF_BIRTH, DAY, 3 );
MGR/A3 = DIGITS( ID_MANAGER, 3 );
END
TABLE FILE WF_RETAIL
SUM MGR NOPRINT DATE_OF_BIRTH NEWDATE
BY MGR
ON TABLE SET PAGE NOPAGE
END
```

The output is:

MGR	Date of Birth	NEWDATE
001	1985/01/29	1985/02/01
101	1982/04/01	1982/04/04
201	1976/11/14	1976/11/17
301	1980/05/15	1980/05/18
401	1975/10/19	1975/10/22
501	1985/04/11	1985/04/14
601	1967/02/03	1967/02/06
701	1977/10/16	1977/10/19
801	1970/04/18	1970/04/21
901	1972/03/29	1972/04/01
999	1976/10/21	1976/10/24

Reference: Usage Notes for DTADD

- ❑ Each element must be manipulated separately. Therefore, if you want to add 1 year and 1 day to a date, you need to call the function twice, once for YEAR (you need to take care of leap years) and once for DAY. The simplified functions can be nested in a single expression or created and applied in separate DEFINE or COMPUTE expressions.
- ❑ With respect to parameter validation, DTADD will not allow anything but a standard date or a date-time value to be used in the first parameter.
- ❑ The increment is not checked, and the user should be aware that decimal numbers are not supported and will be truncated. Any combination of values that increases the YEAR beyond 9999 returns the input date as the value, with no message. If the user receives the input date when expecting something else, it is possible there was an error.

DTDIFF: Returning the Number of Component Boundaries Between Date or Date-Time Values

How to:

Return the Number of Component Boundaries

Given two dates in standard date or date-time formats, DTDIFF returns the number of given component boundaries between the two dates. The returned value has integer format for calendar components or double precision floating point format for time components.

Syntax: How to Return the Number of Component Boundaries

`DTDIFF(end_date, start_date, component)`

where:

end_date

Date or date-time

Is the ending date in either standard date or date-time format. If this date is given in standard date format, all time components are assumed to be zero.

start_date

Date or date-time

Is the starting date in either standard date or date-time format. If this date is given in standard date format, all time components are assumed to be zero (0).

component

Keyword

Is the component on which the number of boundaries is to be calculated. For example, QUARTER finds the difference in quarters between two dates. Valid components (and acceptable values) are:

- YEAR (1-9999)
- QUARTER (1-4)
- MONTH (1-12)
- WEEK (1-53). This is affected by the WEEKFIRST setting.
- DAY (of the Month, 1-31)
- HOUR (0-23)
- MINUTE (0-59)
- SECOND (0-59)

Example: Returning the Number of Years Between Two Dates

The following request against the WF_RETAIL data source calculates employee age when hired:

```
DEFINE FILE WF_RETAIL
YEARS/I9 = DTDIFF(START_DATE, DATE_OF_BIRTH, YEAR);
END
TABLE FILE WF_RETAIL
PRINT START_DATE DATE_OF_BIRTH YEARS AS 'Hire, Age'
BY EMPLOYEE_NUMBER
WHERE EMPLOYEE_NUMBER CONTAINS 'AA'
ON TABLE SET PAGE NOPAGE
END
```

The output is:

Employee Number	Start Date	Date of Birth	Hire Age
AA100	2008/11/14	1991/06/04	17
AA12	2008/11/19	1985/07/13	23
AA137	2013/01/15	1988/12/24	25
AA174	2013/01/15	1980/08/30	33
AA195	2013/01/15	1977/12/11	36
AA427	2008/12/23	1969/08/08	39
AA820	2013/10/29	1983/11/27	30
AA892	2013/10/27	1981/04/24	32

DTPART: Returning a Date or Date-Time Component in Integer Format

How to:

Return a Date or Date-Time Component in Integer Format

Given a date in standard date or date-time format and a component, DTPART returns the component value in integer format.

Syntax: How to Return a Date or Date-Time Component in Integer Format

`DTPART(date, component)`

where:

date

Date or date-time

Is the date in standard date or date-time format.

component

Keyword

Is the component to extract in integer format. Valid components (and values) are:

- ❑ YEAR (1-9999).

- ❑ QUARTER (1-4).
- ❑ MONTH (1-12).
- ❑ WEEK (of the year, 1-53). This is affected by the WEEKFIRST setting.
- ❑ DAY (of the Month, 1-31).
- ❑ DAY_OF_YEAR (1-366).
- ❑ WEEKDAY (day of the week, 1-7). This is affected by the WEEKFIRST setting.
- ❑ HOUR (0-23).
- ❑ MINUTE (0-59).
- ❑ SECOND (0-59.)
- ❑ MILLISECOND (0-999).
- ❑ MICROSECOND (0-999999).

Example: Extracting the Quarter Component as an Integer

The following request against the WF_RETAIL data source extracts the QUARTER component from the employee start date:

```
DEFINE FILE WF_RETAIL
QTR/I2 =DTPART(START_DATE, QUARTER);
END
TABLE FILE WF_RETAIL
PRINT START_DATE QTR AS Quarter
BY EMPLOYEE_NUMBER
WHERE EMPLOYEE_NUMBER CONTAINS 'AH'
ON TABLE SET PAGE NOPAGE
END
```

The output is:

Employee Number	Start Date	Quarter
AH118	2013/01/15	1
AH288	2013/11/11	4
AH42	2008/11/13	4
AH928	2009/04/11	2

DTRUNC: Returning the Start of a Date Period for a Given Date

How to:

Return the First Date of a Date Period

Given a date or timestamp and a component, DTRUNC returns the first date within the period specified by that component.

Syntax: How to Return the First Date of a Date Period

```
DTRUNC(date_or_timestamp, date_period)
```

where:

date_or_timestamp

Date or date-time

Is the date or timestamp of interest.

date_period

Is the period whose starting date you want to find. Can be one of the following:

- DAY, returns day of the month (1-31).
- YEAR, returns year (1-9999).
- MONTH, returns month (1-12).
- QUARTER, returns quarter (1-4).

Example: Returning the First Date in a Date Period

In the following request against the WF_RETAIL data source, DTRUNC returns the first date of the quarter given the start date of the employee:

```
DEFINE FILE WF_RETAIL
QTRSTART/YYMD = DTRUNC(START_DATE, QUARTER);
END
TABLE FILE WF_RETAIL
PRINT START_DATE QTRSTART AS 'Start,of Quarter'
BY EMPLOYEE_NUMBER
WHERE EMPLOYEE_NUMBER CONTAINS 'AH'
ON TABLE SET PAGE NOPAGE
END
```

The output is:

Employee Number	Start Date	Start of Quarter
AH118	2013/01/15	2013/01/01
AH288	2013/11/11	2013/10/01
AH42	2008/11/13	2008/10/01
AH928	2009/04/11	2009/04/01

Adding a Value Test to RESTRICT=NOPRINT

Under prior releases, a RESTRICT=NOPRINT DBA restriction displayed all values or only default values (blank, zero (0), or MISSING).

This optional extension to RESTRICT=NOPRINT enables you to use a VALUE=expression clause in the RESTRICT command. The expression will be evaluated, and the value will display only if the expression evaluates to true for that value. Any value for which the expression evaluates to false will be replaced on the output by one of the default values.

Therefore, a DBA command that includes the following restriction will only display the true value of SEATS when COUNTRY has the value 'ENGLAND'. Otherwise, default values are displayed:

```
RESTRICT=NOPRINT, NAME=SEATS, VALUE= COUNTRY EQ 'ENGLAND';,$
```

Using a SYNONYM or FOCEXEC for ACCEPT Values in a Master File

How to:

Use ACCEPT = FOCEXEC in a Master File

Use ACCEPT = SYNONYM in a Master File

The functionality of ACCEPT in a Master File has been extended. When placed on a FIELD declaration, it can be used to control the values that show up in a filter (WHERE) dialogue box. When used with a global amper variable in the Master File, it can be used to control the values displayed by the Amper Auto-Prompting facility.

The ACCEPT attribute supports the following types of operations:

- ❑ ACCEPT = value1 OR value2 ...

This option is used to specify one or more acceptable values.

- ❑ ACCEPT = *value1* TO *value2*

This option is used to specify a range of acceptable values.

- ❑ ACCEPT = FIND

This option is used to validate incoming transaction data against a value from a Web Query data source when performing maintenance operations on another data source. FIND is only supported for FOCUS data sources and does not apply to OLAP-enabled synonyms.

- ❑ ACCEPT = DECODE

This option is used to supply pairs of values for auto amper-prompting. Each pair consists of one value that can be looked up in the data source and a corresponding value for display.

- ❑ ACCEPT = FOCEXEC

This option is used to retrieve lookup and display field values by running a FOCEXEC. Each row in the output must include one value for lookup and a corresponding value for display. These values can be anywhere in the row, in any order. The FOCEXEC can return other columns as well.

- ❑ ACCEPT = SYNONYM

This option is used to look up values in another data source and retrieve a corresponding display value. The lookup field values must exist in both data sources, although they do not need to have matching field names. You supply the name of the synonym, the lookup field name and the display field name.

Syntax: **How to Use ACCEPT = FOCEXEC in a Master File**

`ACCEPT=FOCEXEC(lookup_field AS display_field IN lookup_focexec)`

where:

lookup_field

Is the field returned by the FOCEXEC whose value will be used in the filter (WHERE dialogue) or by the amper autoprompt facility that will be compared with the field that has the ACCEPT attribute.

display_field

Is the field returned by the FOCEXEC, whose value will be displayed for selection in the filter dialogue or amper autoprompt drop-down list.

lookup_focexec

Is the name of the FOCEXEC that returns the lookup and display field values, in any order. This FOCEXEC can return other field values as well.

Syntax: How to Use ACCEPT = SYNONYM in a Master File

`ACCEPT=SYNONYM(lookup_field AS display_field IN lookup_synonym)`

where:

lookup_field

Is the field in the *lookup_synonym* whose value will be used in the filter (WHERE dialogue) or by the amper autoprompt facility that will be compared with the field that has the ACCEPT attribute.

display_field

Is the field in the *lookup_synonym*, whose value will be displayed for selection in the filter dialogue or amper autoprompt drop-down list.

lookup_synonym

Is the name of the synonym that describes the lookup data.

Using Multiple Prefix Operators on the Same Measure in SUBTOTAL

You can now reference a field with multiple prefix operators in a summary command using the prefix operator to differentiate between the fields with multiple operators.

Report Broker Enhancements

In this section:

Addition of Internal Variables

Showing Only Folders with Content

Creating Schedules With Parameterized Settings

View Logs and Traces for Daily and On-Demand Log Purge and Library Expiration Jobs

Updating Global FTP Password for All Schedules

Viewing Trace Information in the Report Broker Console

Versioning Added for Scheduler.log Files

SFTP With Multi-Factor Authentication

The following new features apply to Report Broker.

Addition of Internal Variables

Two new internal variables have been added. The first variable, `&DSTSCHEID`, is an ID that uniquely identifies the schedule. The second variable, `&DSTSCHEDDESC`, is the title (description) of the schedule.

Showing Only Folders with Content

In the Report Broker Explorer, two new icons have been added to control the display of folders. The *Show only folders with content* option allows you to display only folders that contain content. This option is shown in the following image.

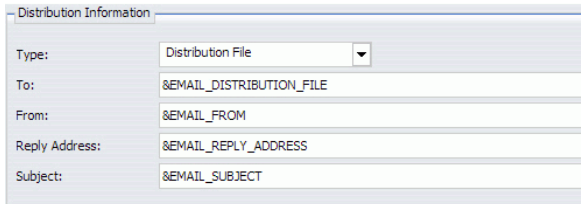


If you have selected the option to Show only folders with content and subsequently want to restore the display of all folders, click *Show all folders*, as shown in the following image.



Creating Schedules With Parameterized Settings

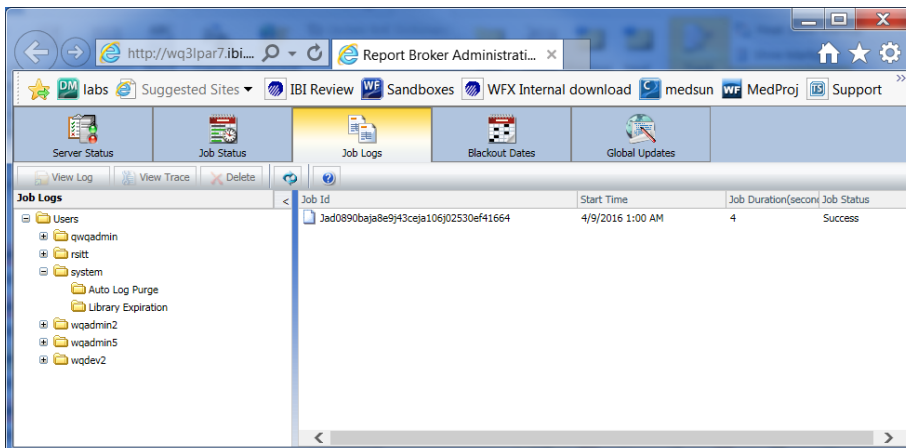
You can use parameters (amper variables) to populate fields in the Basic Scheduling tool. The following image shows an example of the email distribution option. These include, but are not limited to: Save As, Email Zip file name, and Zip file name.



Note: You can use parameters (amper variables) when specifying settings in a schedule. These parameters can be system variables, such as &DATE or any amper variable for which a value is returned by the Reporting Server when the scheduled procedure is run. If a value for a parameter specified in a schedule is not returned by the Reporting Server when the procedure or procedures run, the schedule will fail with a *No report to distribute* error.

View Logs and Traces for Daily and On-Demand Log Purge and Library Expiration Jobs

You can view log and trace information for the daily and on-demand Log Purge and Library Expiration jobs. You can view purge logs and traces using the Job Logs tab in the Report Broker Console. You can view log and trace information, open job logs, delete job logs, refresh job logs, or access related help for job logs on the Job Logs tab. To access a listed job for which you want to view logs or traces, click the *Auto Log Purge* folder or the *Library Expiration* folder, which are located in the system folder. Once you select a job, click *View Log* or *View Trace*, as shown in the following image.

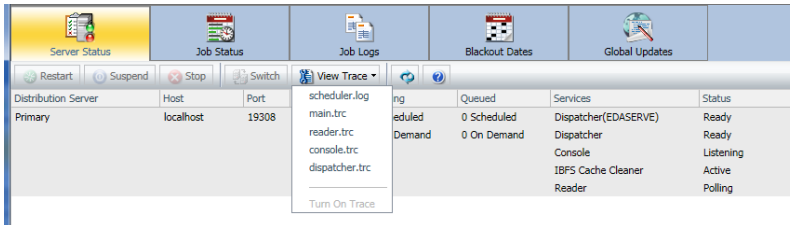


Updating Global FTP Password for All Schedules

Changing the password of an FTP user applies to all schedules that use that user ID for that FTP server and schedule owner.

Viewing Trace Information in the Report Broker Console

A View Trace button has been added to the Server Status, Job Status, and Job Logs tabs in the Report Broker Console. This function enables administrators to view trace information in the scheduler.log, main.trc, reader.trc, console.trc, and other trace files.



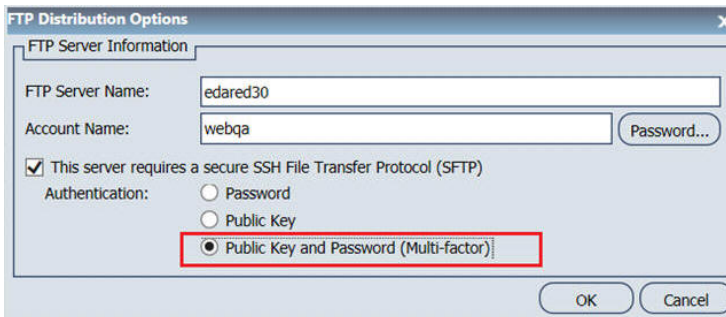
Versioning Added for Scheduler.log Files

When multiple scheduler.log files are created on the same day, Report Broker creates a unique file for each instance, using a date and time stamp. The following format is used: scheduler_DD-MM-YY_HH-MM-SS.

The scheduler.log trace files are always created in the /qibm/UserData/qwebqry/base80/ReportCaster/log directory.

SFTP With Multi-Factor Authentication

In the Distribution dialog box, Multi-factor authentication has been added for FTP Server distribution. When customizing settings for FTP Server information, you can select the Public Key and Password (Multi-factor) authentication option. This option indicates that the server requires authentication with a user ID and password.



Web Query Client Administration and Security Enhancements

In this section:

- New ZIP All Button Captures Traces
- Centralized Validation of Product Variables

The following new features apply to Web Query Client administration and security.

New ZIP All Button Captures Traces

The ZIP All button appears when you select the *All Clients*, *Client Connection*, *MR Deferred Ticket*, *Cleanup Utility*, or *WF Servlet* pages from the Traces folder of the Diagnostics menu on the Administration Console. This button saves copies of all trace files on display into a single zip file, as shown in the following image.

<input type="checkbox"/>	audit.log_2015-08-17.log	41 KB
<input type="checkbox"/>	audit.log_2015-08-18.log	45 KB
<input type="checkbox"/>

Centralized Validation of Product Variables

Web Query provides enhanced protection against SQL injection and cross-site scripting attacks by using a centralized filter to validate all product variables by URI. When a request fails the validation test, the request is not validated and a generic error message is displayed to the user. Blocked requests are logged for administrator review and violations are aggregated into a list that can be used to develop new filters.

Reporting Server Enhancements

In this section:
 Reporting on the Impact of Changing Adapter Connection Names
 Setting Offline Print Attributes on IBM i

The following new features apply to the Reporting Server.

Reporting on the Impact of Changing Adapter Connection Names

If an administrator changes a connection name for an adapter, synonyms and FOCXECs that were created using the old connection name will be invalid. You can generate a report on the impact of changing an adapter connection name.

To generate the impact analysis report, right-click a configured connection on the Adapters page and select *Impact Analysis* from the context menu. A report similar to the following image is generated. The report provides links to the files and lists such information as the connection names, type of file, usage, and application name.

Impact Analysis for wfretail

Found In Application	Found In	Found In Type	Connection	Usage	Adapter	Line	Procedure Description
bisamp	 stategpop	Synonym	wfretail	CONNECTION=	MS SQL Server	1	
bisamp	 wf_product	Synonym	wfretail	CONNECTION=	MS SQL Server	1	
bisamp	 wf_sales	Synonym	wfretail	CONNECTION=	MS SQL Server	1	
doc77x	 retail_script	Synonym	wfretail	CONNECTION=	MS SQL Server	1	
doc77x	 retail_script3	Synonym	wfretail	CONNECTION=	MS SQL Server	1	
wfretail/dimensions	 wf_retail_age	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Age Dimension
wfretail/dimensions	 wf_retail_currency	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Currency Dimension
wfretail/dimensions	 wf_retail_customer	Synonym	wfretail	CONNECTION=	MS SQL Server	1	Customer Dimension
wfretail/dimensions	 wf_retail_discount	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Discount Dimension
wfretail/dimensions	 wf_retail_education	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Education Dimension
wfretail/dimensions	 wf_retail_employee	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Employee Dimension
wfretail/dimensions	 wf_retail_geography	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Geography Dimension
wfretail/dimensions	 wf_retail_income	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Income Dimension
wfretail/dimensions	 wf_retail_industry	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Industry Dimension
wfretail/dimensions	 wf_retail_labor_rate	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Labor Rate Dimension
wfretail/dimensions	 wf_retail_marital_status	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Marital Status Dimension
wfretail/dimensions	 wf_retail_occupation	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Occupation Dimension
wfretail/dimensions	 wf_retail_product	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Product Dimension
wfretail/dimensions	 wf_retail_shipping_options	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Shipping Options Dimension
wfretail/dimensions	 wf_retail_store	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Store Dimension
wfretail/dimensions	 wf_retail_time	Synonym	wfretail	CONNECTION=	MS SQL Server	2	Time Dimension

Setting Offline Print Attributes on IBM i

You can set native IBM i QPRINT and SPLUSRDTA spool system values for spooled OFFLINE print files. The print spool job will pick up the attributes of the specified QPRINT file and associate the SPLUSRDTA string (10-character limit) for the print spool job, so it can be displayed by native operating system tools, such as DSPSPLF and WRKSPLF.

Adapter Enhancements

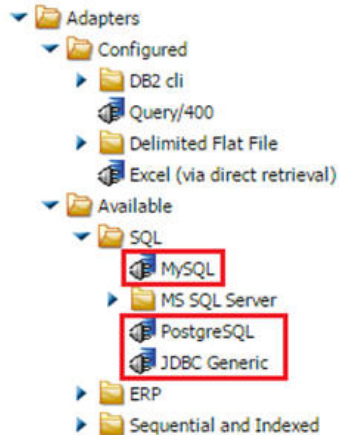
In this section:

New Adapters
SQL Adapters
Adapter for DB2
Adapter for Microsoft SQL Server
Sequential and Indexed Adapters

The following new features apply to adapters.

New Adapters

New adapters for MySQL, PostgreSQL, and JDBC Generic are available with the Standard edition. These adapters are JDBC-based adapters and are available from the SQL Adapter folder, as shown in the following image.



SQL Adapters

In this section:

- Improved Handling of a Star Schema With Fan Trap
- Optimization of Simplified Character Functions
- Optimization of Simplified Date and Date-Time Functions
- Optimization of Legacy Character and Numeric Functions
- Optimization of Date and Date-Time Functions
- Optimization of Simplified Character Functions

The following new features apply to all SQL adapters.

Improved Handling of a Star Schema With Fan Trap

When a star schema contains a segment with aggregated facts and a lower-level segment with the related detail-level facts, a request that performs aggregation on both levels and returns them sorted by the higher level can experience the multiplicative effect. This means that the fact values that are already aggregated may be re-aggregated and, therefore, return multiplied values.

When the adapter detects the multiplicative effect, it turns optimization off in order to handle the request processing and circumvent the multiplicative effect. However, performance is degraded when a request is not optimized.

A new context analysis process has been introduced in this release that detects the multiplicative effect and generates SQL script commands that retrieve the correct values for each segment context. These scripts are then passed to the RDBMS as subqueries in an optimized SQL statement.

To activate the context analysis feature, click *Change Common Adapter Settings* on the Adapters page of the Web Console. Then select Yes for the *FCA* parameter in the *Miscellaneous Settings* section and click Save, as shown in the following image.

Change Settings for Common Adapter

Save settings in Select

Customize data type mappings

DECOMPOSE-DATE Decompose Date fields into components

DATEFMT Format for Date fields

Miscellaneous settings

FCA Activate Foctransform Context Analysis

BLEND-MODE Mode for processing blended structures

LITE-MODE Simplified mode driven by configuration parameters

ETL-TRG-DBMS Default ETL Target DBMS

Optimization of Simplified Character Functions

Simplified character functions have streamlined parameter lists, similar to those used by SQL functions. They are optimized against a wide variety of Relational data sources. The simplified character functions introduced in this release are:

- SUBSTRING - Extracts a substring from a source character string.
- TOKEN - Extracts a token from a string.
- LTRIM - Removes all blanks from the left end of a character string.
- RTRIM - Removes all blanks from the right end of a character string.
- TRIM_ - Removes leading characters, trailing characters, or both from a character string.
- CHAR_LENGTH - Returns the character length of a source string.
- LOWER - Returns the character string with all letters lowercase.
- UPPER - Returns the character string with all letters uppercase.
- POSITION - Returns the position of the first occurrence of a specified sample string in a source string.

The SQL Optimization Report provides optimization information for each function by adapter.

Optimization of Simplified Date and Date-Time Functions

Simplified date and date-time functions have streamlined parameter lists, similar to those used by SQL functions. They are optimized against a wide variety of Relational data sources. The simplified date and date-time functions introduced in this release are:

- ❑ DTRUNC - given a date or timestamp and a component, DTRUNC returns the first date within the period specified by that component.
- ❑ DTADD - given a date in standard date or date-time format, DTADD returns a new date after adding the specified number of a supported component. The returned date format is the same as the input date format.
- ❑ DTPART - given a date in standard date or date-time format and a component, DTPART returns the component value in integer format.
- ❑ DTDIFF - given two dates in standard date or date-time formats, DTDIFF returns the number of given component boundaries between the two dates. The returned value has integer format for calendar components or double precision floating point format for time components.

The SQL Optimization Report provides optimization information for each function by adapter.

Optimization of Legacy Character and Numeric Functions

The following functions have been optimized for most SQL adapters, as reflected in the SQL Optimization Report.

- ❑ TRIMV with the options L or B (left or both).
- ❑ TRIM.
- ❑ POWER (*fld ** exponent*).
- ❑ ARGLEN.

Optimization of Date and Date-Time Functions

The following functions are optimized as reflected in the SQL Optimization Report.

- ❑ **DATEADD.** With YEAR, MONTH, and DAY parameters.
- ❑ **DATECVT.** Optimized for DB2 with 6-digit legacy date formats I6YMD, I6MDY, I6DMY, P6YMD, P6MDY, P6DMY, A6YMD, A6MDY, and A6DMY.
- ❑ **DATEDIFF.** With YEAR, MONTH, and DAY parameters.
- ❑ **DATEMOV.** With BOW, EOW, BOM, EOM, BOQ, EOQ, BOY, and EOY parameters.

- ❑ **DPART.** With YEAR, MONTH, DAY, QUARTER, and WEEKDAY parameters.
- ❑ **HADD.** With YEAR, MONTH, DAY, HOUR, MINUTE, and SECOND parameters.
- ❑ **HDATE.**
- ❑ **HDIFF.** With YEAR, MONTH, DAY, HOUR, MINUTE, and SECOND parameters.
- ❑ **HDTM.** With the following parameter combinations: "8,'HYMDS'", "8,'HYMDs'", and "10,'HYMDm'".
- ❑ **HGETC.**
- ❑ **HPART.** With YEAR, MONTH, DAY, QUARTER, WEEKDAY, DAY-OF-YEAR, WEEK, HOUR, MINUTE, SECOND, MILLISECOND, and MICROSECOND parameters. The optimized HPART function with the WEEK parameter is compatible with FOCUS only in the presence of the WEEKFIRST=ISO2 setting.
- ❑ **TODAY.**

Optimization of Simplified Character Functions

The following simplified functions are optimized to SQL:

- ❑ SUBSTRING
- ❑ LTRIM
- ❑ RTRIM
- ❑ TRIM_
- ❑ CHAR_LENGTH
- ❑ LOWER
- ❑ UPPER

The SQL Optimization Report provides optimization information for each function by adapter.

Note: Simplified functions are easier to use and are likely to be optimized in a wider range of engines than legacy functions.

Adapter for DB2

In this section:

Case-Insensitive Filtering
Metadata

The following new features apply to the Adapter for DB2.

Case-Insensitive Filtering

Case-insensitive filtering is now supported for local IBM i DB2 connections. The setting, SET COLLATION = SRV_CI, is dynamically respected at any time during procedure execution. You can add this setting as a custom setting using the Administration Console or as an entry in the Reporting Server configuration file, edasprof.prf.

Metadata

How to:

Update a Synonym

The Refresh Synonym option performed on a one-part synonym will yield a new FOC1784 message to use the Update Synonym option. The reason for the change in behavior is that the operating system search path for the user and the DB2 search path may not be in sync, resulting in misleading results. To prevent this, it is necessary to use the Update Synonym option.

Procedure: How to Update a Synonym

1. Right-click the Repository folder that contains your synonym and select *Metadata New*.
2. Right-click the *DB2 cli *LOCAL* connection and select *Create or Update Synonym*.
3. Enter the Library name for the synonym and then click *Next*.
4. From the Create Metadata drop-down menu, select the *Update* option.
5. If your synonym contained a prefix or suffix, type it using the prefix and suffix fields.
6. Select the Table name and then click *Next*.

The Update Synonyms for DB2 dialog box opens, as shown in the following image:

Update Synonyms for DB2 (*LOCAL)

One synonym candidate was selected

Check box to allow attributes from the DBMS catalog to override attributes from the existing synonym

Category/Property	Description
<input type="checkbox"/> FILE	
<input type="checkbox"/> REMARKS	Provides descriptive information about the data source.
<input type="checkbox"/> SEGMENT	
<input type="checkbox"/> SEGMENT	The name used to identify this segment.
<input type="checkbox"/> CARDINALITY	Defines how many members of a dimension can be retrieved for a report.
<input type="checkbox"/> TABLENAME	Identifies the table or view.
<input type="checkbox"/> CONNECTION	Indicates a previously declared connection.
<input checked="" type="checkbox"/> KEY	Lists columns that constitute the primary key for the table.
<input checked="" type="checkbox"/> FOREIGN_KEY	Foreign Key Name.
<input checked="" type="checkbox"/> INDEX	Index Name.
<input type="checkbox"/> FIELD	

17 / 17

Back Next

- Choose the synonym properties you wish to refresh and then click *Next* to complete the Update Synonym process.

Adapter for Microsoft SQL Server

In this section:

Capture REMARKS/DESCRIPTION/TITLE in Synonym
Support for HINT Clause

The following new features apply to the Adapter for Microsoft SQL Server.

Capture REMARKS/DESCRIPTION/TITLE in Synonym

The Adapter for Microsoft SQL Server can retrieve and store view properties (if present) in a Master File as the following attributes:

- MS_Description on View** is mapped to the Master File REMARKS attribute.
- MS_Description on View Column** is mapped to the Master File DESCRIPTION attribute.
- MS_Title on View Column** is mapped to the Master File TITLE attribute.

Support for HINT Clause

The Adapter for Microsoft SQL Server can speed up the DML SELECT by specifying a HINT clause through the following SQL SET command:

```
SQL SQLengine SET HINT hint_syntax
```

where:

```
hint_syntax
```

Is the DBMS proprietary HINT statement, supplied with comment marks. For example:

```
SQL SQLMSS SET HINT /* +USE_HASH */ sets USE_HASH hint in Microsoft SQL Server;  
SQL SQLORA SET HINT removes the hint that was set before.
```

The TABLE command places the hint at the end of the generated query for Microsoft SQL Server. The hint (or hint combination) will only be set when the adapter constructs a single SELECT statement. It does not occur in the case of a FOCUS-managed join.

The end user is responsible for the hint or hint combination syntax.

Sequential and Indexed Adapters

In this section:

Adapter for Excel (via Direct Retrieval)

This section provides new features for the Sequential and Indexed adapters.

Adapter for Excel (via Direct Retrieval)

In this section:

File Listener for Excel Worksheet

This section provides detailed descriptions of new features for the Adapter for Excel (via direct retrieval).

File Listener for Excel Worksheet

The File Listener component of DataMigrator can process Microsoft Excel workbooks that are delivered to a specified directory.

In addition to the synonym for the workbook as Excel (via direct read), this requires a synonym for a flat file with the normal file listener parameters specified. An additional segment contains a copy of the synonym for the Excel workbook.

Known Issues

In this section:

Business Intelligence Portal
 InfoAssist
 Developer Workbench
 JD Edwards Adapters
 Report Broker
 Spreadsheet Client
 National Language Support
 DataMigrator/Data Management Console
 Metadata
 Documentation

The following are known issues and will be addressed in a future version of DB2 Web Query.

Business Intelligence Portal

This section addresses the known issues for BI Portal.

Upload Data and Upload Wizard

- ❑ NLS characters are not supported in the following places when using the Upload Data and Upload Wizard:
 - ❑ Excel Worksheet name.
 - ❑ Folder name from where the Upload Wizard is being launched.

Workaround: Use invariant (A-Z and 0-9) characters. Support for NLS characters will be added in a future Hotfix.

- ❑ The Upload Wizard fails to upload data. This happens when DB2 Web Query DataMigrator is installed and the user running the Upload Wizard is not licensed to use DataMigrator, that is, a user who is not a member of the DevWorkBench group. This requirement will be removed in a future Hotfix.

Workaround: If a license is available, add the user to the DevWorkBench group using the Security Center. Otherwise, launch the Upload Wizard from a user who is licensed to use DataMigrator, that is, a user who is a member of the DevWorkBench group.

- ❑ Run-Time Enablement (RTE) may cause the Upload Wizard to fail. This will happen if the user's current library (CURLIB) is not defined in the user's active RTE environment. The Upload Wizard writes out a temporary file during the upload process.

Workaround: It is recommended to add both QGPL and the user's CURLIB (if different from the default, QGPL) to the user's active RTE environment using the WRKWQRTE command.

Metadata Wizard

- ❑ The Adapters for Query/400, DB Heritage, and JD Edwards should not be used during the Import Metadata step as these are multi-step Create Synonym processes, which are not yet supported by the Metadata Wizard.

Workaround: Use the Metadata New option to create synonyms for these Adapter types.

InfoAssist

This section addresses the known issues for InfoAssist.

- ❑ Some navigation and arrow keys (for example, Delete, Home, and End) do not work in the following areas of the application:
 - ❑ Text input box for Prompts (Delete key).
 - ❑ Join Description of the Edit Join dialog box (arrow keys).

Workaround: Use the mouse and Backspace key.

Note: This applies to Firefox versions 27.0.01 and higher.

Developer Workbench

This section addresses the known issues for Developer Workbench.

- ❑ The Upload Data option is not available through the Developer Workbench product.

Workaround: Access the Upload Data option using the Business Intelligence Portal.
- ❑ Developer Workbench online help is hosted on a remote server and requires access to the Internet.

Workaround: A PDF version of the help will be available on the DB2 Web Query Wiki.

JD Edwards Adapters

This section addresses the known issues for the JD Edwards Adapters.

To use the Alternate Language option in the Adapter for JD Edwards EnterpriseOne and the Adapter for JD Edwards World:

1. Configure the adapter with *UDC Direct File Access* unchecked.
2. Run the *Refresh Metadata* with *Alternate Language File* unchecked.
This step will create the `udcdicdb` table.
3. Run the *Refresh Metadata* with *Alternate Language File* checked and enter the default language code.
This step will create the `altdicdb` table.
4. Create all the synonyms needed.

Note: Steps 2 and 3 need to be run, as needed, whenever UDC descriptions are updated.

Report Broker

This section addresses the known issues for Report Broker.

- ❑ Migration of Public Distribution Lists will generate the following error:

```
ERROR IBFSservice - setShares -
opShareBasic/opShareAdvanced to this resource denied - user:qwqadmin
res:/WFC/Repository/untitled/ReportBroker/~ownerID/distribution_list.adr
[2014-03-11 00:00:00,605] ERROR IBFSAddrBookConverter - Failed
to share
'/WFC/Repository/untitled/ReportBroker/~ownerID/distribution_list.adr'
with IBFS:/SSYS/GROUPS/EVERYONE
```

The distribution list will be migrated as a private object. To make this published:

1. Move the distribution list to a published folder.
 2. Right-click on the distribution list and select the *Publish* option.
- ❑ Schedules may not run when using custom date intervals.
Workaround: Use a different recurring interval, such as monthly or weekly. This will be fixed in a future Hotfix.

Spreadsheet Client

This section addresses the known issues for Spreadsheet Client.

DB2 Web Query InfoAssist, when opened from Spreadsheet Client, does not support the following:

- ❑ Excel Pivot output type
- ❑ Table of Contents format option

Both of these options will be disabled from Spreadsheet Client InfoAssist in a future release of DB2 Web Query.

National Language Support

This section addresses the known issues for National Language Support (NLS). This section only applies to Turkish and can be ignored otherwise.

In Turkish, running an active Flash or active PDF report in InfoAssist or from BI Portal generates a FOC3357 error.

Workaround: Edit the /QIBM/UserData/qwebqry/base80/client/wfc/etc/nlscfg.err file and add the following line to the end of the file:

```
ENCODING = Cp1026
```

Also, edit the /QIBM/UserData/qwebqry/ibi/srv77/wfs/etc/odin.cfg file by modifying the JSCOM3 Listener block to include an additional argument on the JVM_OPTIONS parameter, as follows:

```
;JSCOM3 Listener
NODE = JSS
BEGIN
  PROTOCOL = TCP
  CLASS = JAVASERVER
  PORT = 12335
  AWT_HEADLESS = Y
  JVM_OPTIONS = -Djava.version=1.6|-Duser.language=en
  JVM_MAX_HEAP = 512
  IBI_CLASSPATH = /home/sqljdbc40/sqljdbc.jar:/home/sqljdbc20/sqljdbc.jar
END
```

DataMigrator/Data Management Console

Important: The renaming of Application Directories in the Data Management Console is not recommended as it may result in execution failures with the contents of the Application Directory.

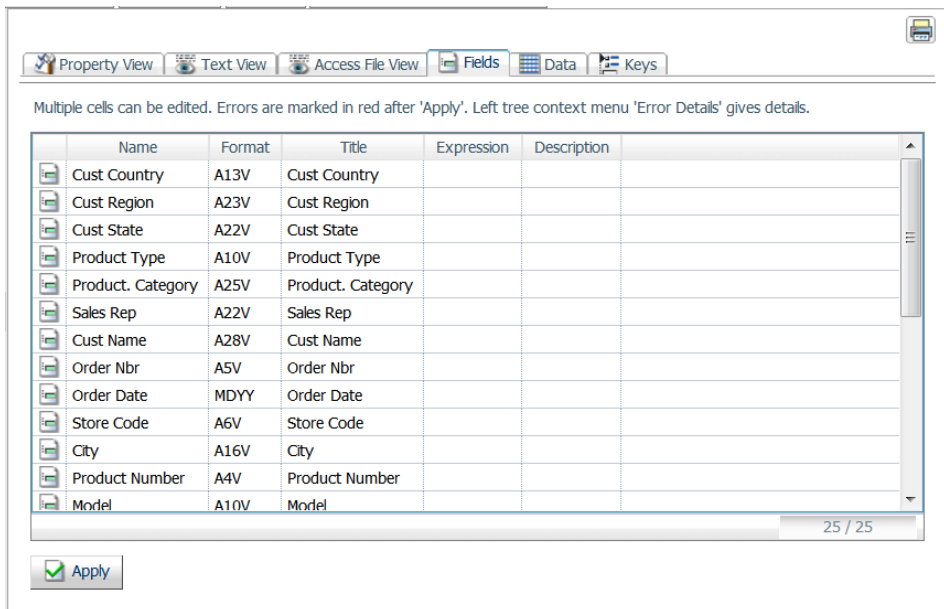
Metadata

This section addresses the known issues for metadata.

- ❑ Creating a synonym for a Query/400 file from the right-click folder Metadata Edit option does not generate the associated Web Query procedure in the repository.

Workaround: Right-click a folder and select the *Metadata New* option to create synonyms for Query/400.

- ❑ Editing a synonym using the Fields tab of the Synonym Editor generates a "Failed to load, server may be down, status =404" error upon applying edits. The Synonym Editor tabs are shown in the following image.



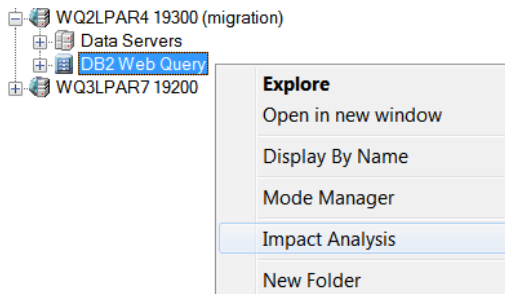
Workaround: Click the *Property View* tab to make any edits to the synonym.

Documentation

This section addresses updates to the documentation.

- ❑ In the Developer Workbench *Analyzing Metadata and Procedures with the Impact Analysis Tool* topic, the documentation indicates that the Impact Analysis tool can be launched from the Synonym Editor or from the user interface. Regarding the latter, launching the Impact Analysis tool from the Data Server node on the Explorer tree will limit the search path to the Reporting Server application directories. The only procedures that can reside in these directories are DataMigrator for i data flows.

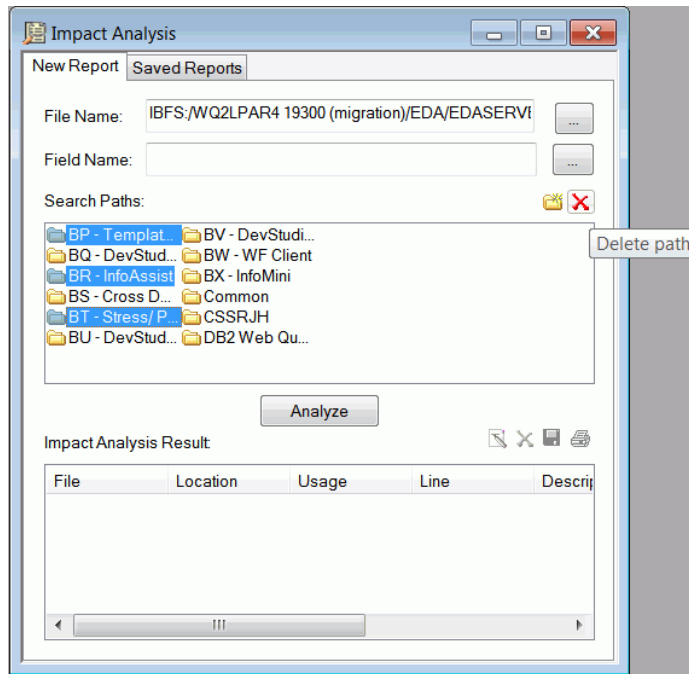
In order for the Impact Analysis tool to be able to search the DB2 Web Query repository, it is necessary to launch the tool from the DB2 Web Query node, and not the Data Servers node, as shown in the following image.



The search path is determined by the location where you launch the Impact Analysis tool. Launching the tool from the DB2 Web Query node will search the entire repository for procedures that are impacted. There are two ways to limit the scope of the search:

- 1.** Launch the Impact Analysis tool from a specific top-level folder in the repository.






- While in the Impact Analysis tool, use the Search Paths filter to delete any top level folders that are not required, as shown in the following image.















































Note: Using the Impact Analysis tool from the DB2 Web Query repository node requires a user ID that belongs to the WebQueryAdministrator group.











Web Browser Support

The following table provides browser support information for Web Query product components for Web Query Version 2.2.0.

 Supported
  Supported with minor issues
  Under evaluation
 Not evaluated
  Not supported

Web Query Component	Internet Explorer v11 (32-bit)	Internet Explorer v10 (32-bit)	Internet Explorer v9 (32-bit)	Firefox v38	Safari v8.0.6	Chrome v44
Reporting						
Active Reports						
HTML Composer pages	Standards mode 	Standards mode 	Standards mode 			
InfoAssist						
HTML Reporting Features						
HTML format (No JavaScript)						
JavaScript components	Standards Mode 	Standards Mode 				
<input type="checkbox"/> Accordion <input type="checkbox"/> HFREEZE <input type="checkbox"/> On-demand Paging	Compatibility View 	Compatibility View 				
<input type="checkbox"/> Multi-drill						
<input type="checkbox"/> Table of Contents (BYTOC)						

Web Query Component	Internet Explorer v11 (32-bit)	Internet Explorer v10 (32-bit)	Internet Explorer v9 (32-bit)	Firefox v38	Safari v8.0.6	Chrome v44
Graph Requests (See Graph request notes below for additional information)						
Server-generated graphs						
Browser-generated (HTML5)						
Browser-generated (AHTML)						
Browser-generated (AFLEX, APDF)						
Web Interfaces						
Amper Auto-prompting						
BI Portal						
OLAP						
Report Broker						
Desktop Tools						
Developer Workbench (Requires Internet Explorer)						

Web Query Component	Internet Explorer v11 (32-bit)	Internet Explorer v10 (32-bit)	Internet Explorer v9 (32-bit)	Firefox v38	Safari v8.0.6	Chrome v44
Administration Tools						
Security Center					—	
Web Query Administration Console					—	

Note:

- ❑ Simple HTML Web Query reports can be viewed on any browser.
- ❑ Support for presenting images and graphs in HTML, DHTML, and DHTML compound reports is provided using an image embedding facility based on the client browser. Output generated by Internet Explorer browsers or in scenarios where the browser is unknown (such as distributed by Report Broker) supports image inclusion through the creation of a web archive file (.mht). For all other browsers, images are 64-bit encoded within the generated .htm file.
- ❑ Graph request notes:
 - ❑ Server-generated graphs refer to traditional graph requests that are generated on the Reporting Server, and then embedded as a bitmap or vector image in a document or webpage. This includes the following output formats:
 - ❑ Bitmap: PNG, JPG
 - ❑ Vector: PDF (but not active PDF), SVG
 - ❑ Browser-generated graphs refer to graphs that are rendered inside the browser. This is done using JavaScript in HTML5 compatible browsers and by Flash in older versions of Internet Explorer that are either not HTML5 compatible or are being run in a mode that is not HTML5 compatible. Browser-generated graphs are utilized in both standard HTML5 output (FORMAT JSCHART) and in Active Technologies (FORMAT AHTML and FORMAT AFLEX, APDF).

Note: Applet-based charts (Java Graph engine running client-side) have been deprecated in Web Query Version 2.1 and are no longer tracked in this matrix.






- ❑ Mac Users: Firefox browser is supported on the Macintosh operating system. Firefox browser functionality is consistent with the Safari web browser.
- ❑ Adobe Reader support:
 - ❑ Adobe XI is certified.
 - ❑ Adobe X is certified.
 - ❑ Adobe Acrobat Reader Version 9 is supported.
- ❑ Drill-down links do not work when using an embedded PDF viewer available in some browser versions. Refer to the configuration information for the specific browser on how to change the Application Options settings for the relevant content types so that the browser will automatically use the Adobe Reader.





















Mobile Browser Support













































Web Query was tested on the following devices:

- ❑ iOS 8.4 tablet
- ❑ iOS 8.02 tablet
- ❑ iOS 8.04 phone
- ❑ iOS 6.1.3 touch
- ❑ Android 4.4.4 tablet and phone
- ❑ Mobile Faves version 3.2.0.2

The following table provides mobile browser support information for Web Query product components for Web Query Version 2.2.0.

-  Supported
  Supported with minor issues
  Under evaluation
-  Not evaluated
  Not supported

Web Query Component	iOS Safari	iOS MobileFaves	Android Chrome	Android MobileFaves
Reporting				
Active Reports				
HTML Canvas pages				
PDF			 (Requires third-party apps)	
Excel			 (Requires third-party apps)	
PowerPoint (PPTX)	 (Requires third-party apps)		 (Requires third-party apps)	

Web Query Component	iOS Safari	iOS MobileFaves	Android Chrome	Android MobileFaves
HTML Reporting Features				
HTML format (<i>No JavaScript</i>)				
JavaScript components				
<input type="checkbox"/> Accordion <input type="checkbox"/> HFREEZE <input type="checkbox"/> On-demand Paging				
<input type="checkbox"/> Multi-drill				
<input type="checkbox"/> Table of Contents (BYTOC)				
Graph Requests				
Server-generated graphs				
Browser-generated (HTML5)				
Browser-generated (AHTML)				
Interfaces				
Amper Auto-prompting				
Report Broker Explorer	 Tablets		 Tablets	
Report Broker Console	 Tablets		 Tablets	
Report Broker Scheduling Tool	 Tablets		 Tablets	

Note: Web Query developer tools (InfoAssist) are not supported for mobile.

