WebFOCUS

Server New Features

WebFOCUS Reporting Server Release 8203

DataMigrator Server Release 7708

June 05, 2018
Contents

1. Server Enhancements .......................................................... 7
   New Dependencies Analysis Column ........................................... 8
   Specifying Connection for Formatted File Targets ......................... 8
   Redesign of Connect to Data Page ............................................. 9
   Configuring Geographic Information .......................................... 20
   Linking to Your WebFOCUS Client Repository ............................. 39
   Upload Support for JSON ...................................................... 42
   Automatic Scrolling Of the Business View Pane When Dragging and Dropping ................. 42
   Adding Captions for Check List Values ...................................... 42
   Using a LIKE Relation With a Text Box Filter .............................. 43
   Comparing Values to a Lookup File ........................................... 46
   Converting a Filter Value List to a Variable Expression ................... 48
   Storing Server Log Files in a User-Specified Location .................... 49
   Downloading a Deferred Jobs Report ........................................... 49
   Applications Page Redesign ................................................... 49
   Bubble Help for Guiding the User to the Next Step ......................... 50
   Modern Design of Server Web Console ....................................... 50
   Left-Click Menu on the Web Console ......................................... 51
   edastart Command Option for Updating Connection Attributes ........... 52
   Turning a Configured Server Into a Docker Container ..................... 52
   Setting the Maximum Width of a Trace Line ................................ 54
   Reporting Server on z/OS 64-Bit Version ................................... 54
   Debugging a Dialogue Manager Procedure .................................. 54
   Kerberos Support for JDBC-Based Data Sources ............................ 70
   Git Integration With the WebFOCUS Reporting Server ..................... 70

2. Adapter Enhancements ....................................................... 71
   SQL Adapters .................................................................... 71
   Diagnostics: Improved Outer Join Rejection Messages for SQL Adapters .... 71
   Extended Bulk Load for SQL Adapters: Auxiliary Connection Updates .... 71
   Adapter for Amazon Athena .................................................. 72
   Adapter for Amazon Redshift: Readonly Property for IDENTITY Columns .... 72
   Adapter for Apache Drill: TIME and TIMESTAMP Fields .................. 72

Server New Features 3
Adapter for WebFOCUS .......................................................... 79
Sequential and Indexed Adapters ...................................................... 79
Storing Images on Filesystem From a Database Table ......................... 79
Test for Write Access to SFTP Server .............................................. 79
Adapter for Hyperledger Fabric ..................................................... 80
Statistics Adapters .................................................................. 80
Adapter for Rserve Integration ..................................................... 80
XML Adapters ....................................................................... 89
Pivot Support for JSON Documents ................................................ 89
Adapter for JSON: Reflect the WebFOCUS Query in HOLD FORMAT JSON ................ 89

3. DataMigrator Enhancements .................................................. 91
Adapters ........................................................................... 91
Adapter for Salesforce.com: Extended Bulk Load Now Available ............... 91
Test for Write Access to SFTP Server .............................................. 91
Extended Bulk Load for SQL Adapters: Auxiliary Connection Updates ........... 92
Creating Data Fields as TIMESTAMP for Jethro and Impala ....................... 92
Pivot Support for JSON Documents ................................................ 92
Data Management Console ........................................................... 93
Modern Design of the Data Management Console .................................. 93
Storing Images on Filesystem From a Database Table .................................. 93
Cluster Manager Available for DataMigrator Scheduler ............................... 93
Data and Process Flows ................................................................ 93
Specifying Connection for Formatted File Targets ..................................... 93
Data Profiling ....................................................................... 94
Data Profiling Enhancement - Forecast and Distribution Charts ................... 94
Reporting ........................................................................... 94
New Dependencies Analysis Column .................................................. 94

4. Resource Analyzer and Resource Governor Enhancements ...................... 95
Monitoring at the Data Service Level ................................................... 95

5. Reporting Language Enhancements ........................................... 97
New Punctuation Option for Numbers Using SET CDN .................................. 98
Two-Part Names Support in ? DEFINE ............................................. 99
Contents

Stacking Duplicate Columns in Multi-Verb Requests Based on AS Names .......................... 99
Enhancements to Error Messages .................................................................................. 102
DTIME: Extracting Time Components From a Date-Time Value ..................................... 102
EDAPRINT: Inserting a Custom Message in the EDAPRINT Log File ............................. 104
Using FORECAST in a COMPUTE Command ................................................................ 105
  Calculating Trends and Predicting Values With FORECAST ....................................... 106
    FORECAST Processing ............................................................................................... 107
    Using a Simple Moving Average ............................................................................. 110
    Using Single Exponential Smoothing ..................................................................... 114
    Using Double Exponential Smoothing .................................................................. 117
    Using Triple Exponential Smoothing ..................................................................... 118
    Using a Linear Regression Equation ...................................................................... 121
    Distinguishing Data Rows From Predicted Rows .................................................... 123
PATTERNS: Returning a Pattern That Represents the Structure of the Input String ........... 125
Simplified Statistical Functions .................................................................................... 126
  CORRELATION: Calculating the Degree of Correlation Between Two Sets of Data ........ 127
  KMEANS_CLUSTER: Partitioning Observations Into Clusters Based on the Nearest Mean Value ........................................................................................................... 128
  STDDEV: Calculating the Standard Deviation for a Set of Data Values ....................... 131
  MULTIREGRESS: Creating a Multivariate Linear Regression Column ......................... 132
RSERVE: Running an R Script ....................................................................................... 134
Displaying a Caret Symbol (<) in Heading Objects ....................................................... 139
Displaying Syntax Components in Heading Objects ...................................................... 140
Representing a Null String ........................................................................................... 142
Enhancements to the BYDISPLAY Parameter .............................................................. 143
Enhancement to SET COUNTWIDTH .......................................................................... 146
Change to Default Value for SET MISSINGTEST ......................................................... 146
Setting MISSING ON Behavior for DEFINE and COMPUTE .......................................... 149
Enhancement to the SUMPREFIX Parameter .................................................................. 151
Enhancements to Sort Performance .............................................................................. 152
STRING Data Type ....................................................................................................... 154
Accordion By Row Enhanced Interface ........................................................................ 154
OpenType Fonts Embedded in PDF Output Files ........................................................... 154
Server Enhancements

The server provides a wide range of capabilities and tools for adapter configuration, metadata creation, application and path management, security control, communications configuration, and for monitoring, tuning, and troubleshooting server performance. Authorized users can perform most server administration tasks from a graphical Web Console. The server supports WebFOCUS reporting functions, extraction, load and transformation functions, and analysis and data access control functions.

In this chapter:

- New Dependencies Analysis Column
- Specifying Connection for Formatted File Targets
- Redesign of Connect to Data Page
- Configuring Geographic Information
- Linking to Your WebFOCUS Client Repository
- Upload Support for JSON
- Automatic Scrolling Of the Business View Pane When Dragging and Dropping
- Adding Captions for Check List Values
- Using a LIKE Relation With a Text Box Filter
- Comparing Values to a Lookup File
- Converting a Filter Value List to a Variable Expression
- Storing Server Log Files in a User-Specified Location
- Downloading a Deferred Jobs Report
- Applications Page Redesign
- Bubble Help for Guiding the User to the Next Step
- Modern Design of Server Web Console
- Left-Click Menu on the Web Console
- edastart Command Option for Updating Connection Attributes
- Turning a Configured Server Into a Docker Container
- Setting the Maximum Width of a Trace Line
- Reporting Server on z/OS 64-Bit Version
- Debugging a Dialogue Manager Procedure
- Kerberos Support for JDBC-Based Data Sources
- Git Integration With the WebFOCUS Reporting Server
New Dependencies Analysis Column

The Dependencies Analysis report for a DataMigrator flow or synonym shows all of the synonyms and flows that it references.

A new column in the report shows the last modified date for each dependent file.

Specifying Connection for Formatted File Targets

In previous releases, using DataMigrator with a Formatted File target adapter would fail if the Flat File adapter had a connection for an [S]FTP server. In addition, by default, an application directory name was specified for a synonym, which was not needed.

Now, when a data flow uses a new target with a Formatted File adapter, a new Connection box appears. This option enables you to specify a configured connection. The default value is <local>. 
Redesign of Connect to Data Page

The Connect to Data (Adapters) page has been redesigned to have separate lists for configured adapters and available adapters, as shown in the following image.

The Configured Adapters list shows all adapters and connections that have been configured.

The Available list shows all available adapters. If a connection has already been configured for an adapter, a check mark is displayed next to the adapter name. A drop-down list enables you to select an adapter category to view on the list.

Many of the options for adapters and connections, such as configuring a new connection or creating a synonym, open in a new panel on the Connect to Data page.
Reference: Configured Adapters List

The Configured Adapters panel lists all adapters and connections that have been configured. You can create synonyms and cluster business views from this panel, add new or duplicate connections, remove connections, and view and edit properties.

Adapter Menu

By right-clicking an adapter name or clicking the down arrow on an adapter line on the list, you open a context menu. The items on the menu may differ for different categories of adapters. For an SQL Adapter, the following options are available.

- **Add Connection.** Opens the Add Connection page to configure a new connection for the adapter.
- **Change Settings.** Opens the Change Settings page for this adapter so you can edit the settings specific to this adapter.
  
  **Note:** Once saved in the server profile edasprof, the settings will have global effects.
- **Remove.** Removes the configuration for this adapter.
- **Help.** Opens the help file for this adapter.

Connection Menu

By right-clicking a connection name or clicking the down arrow on a connection line on the list, you open a context menu with all or some of the following options, depending on the adapter.

- **Show DBMS objects.** Opens a panel for creating synonyms using this connection.
  
  **Note:** For delimited files and Excel files, the corresponding option is to show local files. When you click this option, a file picker dialog box opens. When you select a file, the Create Synonym page opens for the selected file.
- **Properties.** Opens the Change Connect Parameters page for the connection.
- **Test.** Tests access to the DBMS. If successful, a report is displayed showing connection information for the DBMS.
- **Duplicate Connection.** Opens a Duplicate Connection panel with the same connection name, suffixed with _dup, and the same server as the original connection. You can use this to connect to a different database than the one in the original connection, or use a different type of security model.
- **Configure Bulk Load.** For some adapters, such as the Adapter for Redshift ODBC, opens a bulk load configuration page.
- **Test Bulk.** Tests whether extended bulk load is available and displays a message.
- **Test DBMS Case Sensitivity.** Displays a panel with information about case sensitivity for this DBMS.
- **Impact Analysis.** Displays a list of files that reference this connection.
- **Delete.** Deletes the connection.

**Reference:** **Available Adapters List**

The Available list shows all available adapters, whether already configured or not. If the adapter already has a configured connection, a check mark displays next to its name. If the adapter has multiple versions, when you right-click it or click the down arrow, you select which version to configure, as shown in the following image.
You can select a category of adapter to display on the list, as shown in the following image.
Instead of displaying the available adapters as a list, you can display them as icons by clicking View as Icon on the Available pane menu, as shown in the following image.

Reference: Creating Synonyms

To create synonyms, right-click a connection and click Show local files (for delimited files) or Show DBMS objects from the context menu.

For delimited or Excel files, a file picker dialog box opens. Select a file and click OK. The Select Synonym Candidates page opens, as shown in the following image.
For an SQL adapter, the Available Objects page opens, as shown in the following image.

![Available Objects for MS SQL Server ODBC (CON01)](image)

Creating Synonyms for an SQL Adapter

The Show DBMS Objects page has been designed so that you can enter all the parameters needed to create or update base synonyms or create a cluster synonym on a single page.

Select the object type and enter the owner/schema and database from which to show objects.
You can search for objects by entering a term in the search text box and clicking Search, as shown in the following image.

From the Action drop-down list, you can select Create Base Synonyms (for individual tables or objects), Create Cluster Synonym with BV (for creating a synonym that references multiple base synonyms), or Update Base Synonyms, as shown in the following image.

Creating Base Synonyms

To create base synonyms, click the check box next to one or more tables for which you want to create a synonym. You can edit the Default Synonym Name. When you are finished, click Create Base Synonyms on the ribbon. The synonyms are created and added to the Application specified.

Updating Base Synonyms

1. Click the check box next to one or more tables for which you want to update the synonym.
   You can edit the Default Synonym Name.

2. Click Update Base Synonyms on the ribbon.
A page opens that enables you to select attributes from the DBMS catalog that will override attributes from the existing synonym, as shown in the following image.

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

<table>
<thead>
<tr>
<th>Category/Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remarks</td>
<td>Provides description information about the data source.</td>
</tr>
<tr>
<td>Segment</td>
<td>The name used to identify this segment.</td>
</tr>
<tr>
<td>Cardinality</td>
<td>Defines how many members of a dimension can be retrieved for a report.</td>
</tr>
<tr>
<td>Tablename</td>
<td>Identifies the table or view.</td>
</tr>
<tr>
<td>Connection</td>
<td>Indicates a previously declared connection.</td>
</tr>
<tr>
<td>Key</td>
<td>Lists columns that constitute the primary key for the table.</td>
</tr>
<tr>
<td>Foreign Key Name</td>
<td>For use with foreign keys.</td>
</tr>
<tr>
<td>Index</td>
<td>Index Name.</td>
</tr>
<tr>
<td>Field</td>
<td>Field Name. The name used to reference this element of data in a request.</td>
</tr>
<tr>
<td>Title</td>
<td>An alternative report column title for the field. You can split the text across as many as five separate title lines by separating the lines with a comma (,).</td>
</tr>
<tr>
<td>Usage Format</td>
<td>Describes how to format a field when displaying it in a report or using it in a calculation.</td>
</tr>
<tr>
<td>Actual</td>
<td>Describes the type and length of data as it is actually stored in the data source.</td>
</tr>
<tr>
<td>Missing</td>
<td>Enables null values to be entered into and read from a field in data sources that support null data, such as RODIS data sources and most relational data sources.</td>
</tr>
<tr>
<td>Description</td>
<td>An optional attribute that enables you to provide comments and other documentation for a field within the Master File. You can include any comment up to 32 (2048) characters in length.</td>
</tr>
</tbody>
</table>

3. Click **Next**.

A status page opens showing messages about the success of the updates. The synonyms are added to the Application specified.

4. Click **Close**.
Creating a Cluster Synonym with a Business View (BV).

When you select *Create Cluster Synonym with BV* on the Action list, the DBMS objects list changes so that you can both select objects to add to the cluster and assign them as facts or dimensions, as shown in the following image.

![Available Objects for MS SQL Server ODBC (CON01)](image-url)
You can select each fact and each dimension manually, or you can right-click a fact table name or click the down arrow next to a fact table name and select one of the following:

- **Show Related Dimensions.** Displays a report of related dimensions, as shown in the following image.
Add Related Dimensions. Selects all related dimensions automatically, as shown in the following image.

When you have selected the fact and dimension tables, click Create Cluster Synonym in the Next group on the ribbon.

The Save As dialog box opens with a default name for the cluster in the application directory selected on the Show DBMS Objects page. You can change directories and edit the name.

Click OK to create the cluster.
Reference: Ribbon Options

Tuning and Troubleshooting Group

- **Help Menu Options**
  - **Complete Adapter List.** Displays a report listing all adapters by category and their properties, such as support for Unicode, bulk load, and change data capture.
  - **SQL APT Status.** Displays a report listing all SQL functions and keywords. The report indicates, for each SQL adapter, whether the function or keyword is passed to the SQL engine as is, converted to the DBMS-specific version of SQL, or not converted or passed so that it will be processed by WebFOCUS.
  - **SQL Optimization Report.** Displays a report identifying whether WebFOCUS functions are optimized for each SQL adapter. You can select an adapter subcategory and a type of function.
  - **Data Types.** Displays a data type report for all SQL adapters, a subcategory of SQL adapters, or a specific adapter. You can also select a specific server data type for the report. The report shows how the DBMS data type is mapped to server USAGE and ACTUAL formats.
  - **Adapters License Count.** Displays a report showing your license code, the number of adapters you are licensed to configure, and the number of configured adapters.

- **Change Common Adapter Settings.** Opens a page for setting options that apply to all adapters.

View Group

- **Reset.** Resets the page to its initial view.
- **Show Less.** Hides the Tuning and Troubleshooting group. The View menu then displays a Show More option for unhiding the hidden group.

Configuring Geographic Information

Geographic roles have been unified to access 58 world administrative boundaries down to the postal code level. The server has a geographic configuration editor for customizing the list of geographic roles, basemaps, reference layers, and demographic layers. A customized geographic role can reference either an uploaded shapefile or an existing Esri Feature Layer.
Procedure: How to Edit Geographic Configuration Settings

By default, the server is configured for unified geographic roles. This configuration is controlled by the following settings.

1. Go to Workspace and click Edit settings on the Geo Services button on the ribbon.

The Change Settings for GEO services page opens, as shown in the following image.

2. The following settings control unified geographic roles.

   - **GEO_UNIFIED_ROLE.** This compatibility setting must be turned on (the default) to activate unified geographic roles. Turn it off to use geographic roles from prior releases. On indicates that the new shorter set of geographic roles that combines subsets of previously used roles will be used. Off will use geographic roles from previously shipped releases.
**GEO_MAP_PROVIDER.** Assigns names of the providers of geographic maps. The list of names should be separated with slashes to be used by transformation code for mapping. Default is ALL. The currently supported set of providers includes WFRS (geographic boundaries distributed in the Reporting Server) and ESRI.

**GEO_ALIASING.** On enables aliasing to map administrative entities. Aliasing is a mechanism to support alternative names/spellings to the administrative names used as keys to find the corresponding geometry. Caution, incorrect results will be reported on a map in the case where the column data contains variations of names/spellings for the same administrative entity (resolved to the same key). Alias names are stored in .csv files. Shipped aliasing files are located in the geomaps sub-directory of the etc folder under EDAHOME. The naming convention is geo_srv_dbl_geo_role. Each data file has an associated synonym with the same name. Aliasing is currently supported for the four geo roles COUNTRY, STATE, COUNTY, and CITY. The STATE aliasing data file includes the valid country name for each state. COUNTY and CITY files include valid country and state names. *Valid* means the actual key value used to fetch a geometry. The default value is off.

**GEO_ALIASING_APP.** Sets an application name for user-added .csv files with aliases. DEFAULT means no user files. User-added alias data files are supported for the four geo roles COUNTRY, STATE, COUNTY, and CITY (see the description for the GEO_ALIASING setting). The default value is DEFAULT, which uses the server aliasing files.

The best practice is to copy the desired .csv file from the _edahome/etc/geomaps folder to the application folder named in this setting, and edit it, changing aliases or adding records with new ones. There are four focexecs named geo_srv_mapkey_<geo_role> that have mandatory parameter ISO2 country name. The following is a request example that reports city names for South Africa sorted by state/province name.

```ex
geo_srv_mapkey_city ISO2='ZA'
```

Following is an example of user-created records (based on the obtained report):

"South Africa", "Gauteng", "Johannesburg", "City of Johannesburg"
"South Africa", "Western Cape", "Paarl", "Drakenstein"

**GEO_ALIASING_FIPS.** On enables FIPS aliasing to US map administrative entities. This setting requires GEO_ALIASING to be on. The data file with the United State FIPS aliases is processed as an extension to the main aliases data file. The default value is off.
- **GEO_COUNTRY.** Assigns a default value for an automatically added DEFINE field with GEOGRAPHIC_ROLE ‘Country’. This mechanism is in effect during the create metadata process when the setting GEOROLES-AUTO is ON and a column with geo role COUNTRY is not detected. The generated DEFINE field will be used to create geo hierarchies required in mapping. The default value is 'United States'.

- **GEOROLE_2_MBRLEVEL.** CUSTOM enables customized per country mapping of the geo roles (STATE, COUNTY and CITY) to the MBR administrative levels. STANDARD sets uniform mapping of geo roles (STATE, COUNTY and CITY) for all countries to the MBR administrative levels (1, 3 and 5 respectively). There is a geo_role2mbrlev focexec to list countries with customized admin levels. The report does not depend on this setting.

3. Click Save.

**Procedure: How to Edit the Geographic Configuration**

The GEO configuration editor provides a tool for editing or adding properties for geographic roles, basemaps, reference layers, and demographic layers. In addition, it enables you to add maps and shapefiles to the configuration.

1. Go to Workspace and click **Edit Configuration** on the Geo Services button on the ribbon.

   The **GEO configuration editor** opens displaying the configured geographic roles, as shown in the following image.

   ![GEO configuration editor](image)

   You can select the following objects from the Object drop-down list.

   - Role.
   - Basemap.
   - ContextLayer.
You can edit the properties for a basemap or context layer (reference layer or demographic layer), or add a new one. You cannot edit the properties of a standard geographic role. To add a customized geographic role, you first add an Esri map or a shapefile (WFRS map) and assign the geographic role to the map or shapefile.

The following standard unified geographic roles are configured by default and cannot be changed. These geographic roles create a hierarchy that can be used to drill down or up between levels of administration in maps, reports, or charts.

- CONTINENT.
- COUNTRY.
- STATE.
- COUNTY.
- CITY.
- POSTAL CODE.

The following describes columns for geographic roles in the configuration editor.

**name**

Is the unique name of the geographic role. It cannot have spaces, but it can have underscores (_).

Next to the name is an indicator of whether the role is a standard role or a customized role.

**title**

Is the description of the geographic role that is displayed in reports and in drop-down lists in the WebFOCUS tools.

**returned_geometry**

Is the type of geographic data returned from the map service for rendering on the map.

Valid values include:

- GEOMETRY_AREA
- GEOMETRY_POINT
- GEOMETRY_LINE

2. To add a new basemap or customize an existing basemap, select Basemap from the Object drop-down list.
The following image shows the GEO configuration editor with the Basemap object selected.

**GEO configuration editor**

<table>
<thead>
<tr>
<th>name</th>
<th>icon</th>
<th>title</th>
</tr>
</thead>
<tbody>
<tr>
<td>streets</td>
<td>qb/streets_map_108x72.png</td>
<td>World Street Map</td>
</tr>
<tr>
<td>satellite</td>
<td>qb/imagery_map_108x72.png</td>
<td>World Imagery</td>
</tr>
<tr>
<td>terrain</td>
<td>qb/terrain_map_108x72.png</td>
<td>Terrain with Labels</td>
</tr>
<tr>
<td>gray</td>
<td>qb/gray_map_108x72.png</td>
<td>Light Gray Canvas Map</td>
</tr>
<tr>
<td>dark-gray</td>
<td>qb/darkgray_map_108x72.png</td>
<td>Dark Gray Canvas Map</td>
</tr>
<tr>
<td>oceans</td>
<td>qb/oceans_map_108x72.png</td>
<td>Oceans Map</td>
</tr>
<tr>
<td>national-geographic</td>
<td>qb/national_geographic_map_108x72.png</td>
<td>National Geographic World Map</td>
</tr>
<tr>
<td>osm</td>
<td>qb/open_street_map_108x72.png</td>
<td>Open Street Map</td>
</tr>
<tr>
<td>hybrid</td>
<td>qb/hybrid_map_108x72.png</td>
<td>World Imagery with Labels</td>
</tr>
<tr>
<td>topo</td>
<td>qb/topo_map_108x72.png</td>
<td>World Topographic Map</td>
</tr>
<tr>
<td>None</td>
<td>qb/none_map_108x72.png</td>
<td>BasemapNone</td>
</tr>
</tbody>
</table>

Following is a description of the properties used for basemap configuration.

**name**

Is the name of the basemap.

Next to the name is an indicator of whether the basemap is a standard basemap or a customized basemap.

**icon**

Is the name of the thumbnail for the basemap (for a standard basemap) or the URL to the thumbnail (for a customized basemap) that will appear on the Basemap drop-down list in the WebFOCUS tools or the Change Basemap map widget.

**title**

Is a title to display on the Basemap drop-down list in the WebFOCUS tools or the Change Basemap map widget.

**url**

Is the URL to the map service that provides the basemap, for a customized basemap. For a standard basemap, the URL is already stored in the server geographic configuration file and is not displayed.

**type**

Valid values are tiled and vector.
**addon.json**

Specifies additional JSON properties for rendering the map.

- To customize the properties of an existing basemap, click the down arrow next to a basemap name or right-click the basemap line and click **Customize BASEMAP**.

The Customize Basemap dialog box opens, as shown in the following image.

```
Customize BASEMAP: streets

- name: streets
- icon: disk/streets_map_100x72.png
- title: World Street Map
- type: tiled
- url:

addon.json:
```

Edit the properties you want to change. If you change the URL, you can click **Verify** to make sure the map service is valid and accessible.

When you are finished, click **OK**, then click **Save** on the GEO configuration editor Basemaps page.

- To add a new basemap to the configuration, click **Add**.
The *Create a BASEMAP* dialog box opens, as shown in the following image.

Enter a name for the basemap, a URL to the thumbnail, a title to display, and the URL to the map service that provides the basemap, and click *Verify*.

When you have configured the properties, click *OK*, then click *Save* on the GEO configuration editor Basemap page.

3. To add a new context layer or customize an existing context layer, select *ContextLayer* from the Object drop-down list.
The following image shows the GEO configuration editor with the ContextLayer object selected.

Following is a description of the properties used for context layer configuration.

**name**

Is the name of the context layer.

Next to the name is an indicator of whether the context layer is a standard context layer or a customized context layer.

**authorization**

Is the type of authentication needed to access this context layer. Valid values are:

- **silent.** Credentials for your ArcGIS application are provided in the connection string of the Adapter for Esri ArcGIS.

  **Note:** For instructions for configuring the Adapter for Esri ArcGIS, see the Adapter Administration manual.

- **none.** No authorization is needed.

- **named.** User credentials are provided in the connection string of the Adapter for Esri.

- **on premises.** User credentials for a locally hosted ArcGIS server are provided in the connection string of the Adapter for Esri.

**layer type**

Is the type of context layer. For a cached layer, the layer type is tile. For a layer that is rendered dynamically, the layer type is featurelayer.

**title**

Is a title to display on the demographic layer drop-down list in the WebFOCUS tools.
uri
Is the URL to the map service that provides the context layer.

addon_json
Specifies additional JSON properties needed for rendering the context layer. For example, smartMapping properties define the border styles within the context layer.

To customize the properties of an existing context layer, click the down arrow next to a context layer name or right-click the context layer line and click Customize context layer. The Customize CONTEXTLAYER dialog box opens, as shown in the following image.

Edit the properties you want to change. If you change the URI, you can click Verify to make sure the map service is valid and accessible.

When you are finished, click OK, then click Save on the GEO configuration editor Context Layers page.

To add a new context layer to the configuration, click Add.
The Create a Customized CONTEXTLAYER dialog box opens, as shown in the following image.

Enter a name for the context layer, the authorization type, a layer type, a title to display, any additional JSON needed for rendering the context layer, and the URI to the map service that provides the context layer, and click Verify.

When you have configured the properties, click OK, then click Save on the GEO configuration editor Context Layer page.

**Reference:** Editing the List of Geographic Roles

To add a geographic role to the configuration, you can add a new Esri map or a shapefile hosted by the Server and associate a geographic role with the new map. You can also implement NUTS geographic roles support.

**Adding a New Role for an Esri Map**

To add an Esri geographic role, click Add ESRI map on the Geographic Configuration Editor.
The *Create a new ESRI map role* dialog box opens, as shown in the following image.

Configure the following map service properties.

**name**
Is a name for the geographic role.

**title**
Is a title to display in the WebFOCUS tools.

**returned_geography**
Select the type of geometry that is returned from the map service for this role. Valid values are:

- **GEOMETRY_AREA.** Returns JSON polygon definitions.
- **GEOMETRY_LINE.** Returns JSON line definitions.
- **GEOMETRY_POINT.** Returns a JSON point.

**url**
Is the URL to the map service that provides the geographic data.
Click **Verify** after entering the URL to verify that the map service is available by going to the specified URL.

**Service Parameters**
Add as parameters any additional geographic roles needed to identify the exact location of the new role. For example, a city name needs state and country parameters.

The following properties add the WebFOCUS Regions role to the configuration.

![Edit geography role REGION](image)

**Note:** The *parameter name* corresponds to the field name in the FeatureLayer referenced in the following URL:

http://services7.arcgis.com/L95Wwv9oJtQ0tjAs/ArcGIS/rest/services/wfretail_sub_regions/FeatureServer/0

Click **OK** when you have finished configuring the properties.

The new role is added to the configuration as a customized role, as shown in the following image.

![GEO configuration editor](image)

Click **Save** to save this role to the configuration.
The following request uses the WebFOCUS Regions geographic role in a map request.

```
DEFINE FILE WF_RETAIL_LITE
REGION/A50 (GEOGRAPHIC_ROLE=REGION) = BUSINESS_SUB_REGION;
END

GRAPH FILE WF_RETAIL_LITE
SUM COGS_US
BY REGION
WHERE COUNTRY_NAME EQ 'United States'
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH CHOROPLETH
ON GRAPH SET STYLE *
TYPE=REPORT, CHART-LOOK=com.esri.map, $
TYPE=DATA, COLUMN=N2, BUCKET=color, $
*GRAPH_JS_FINAL
"extensions": {
"com.esri.map": {
"overlayLayers": [
{
"ibiDataLayer": {
"map-metadata": {
"map_by_field": "REGION"
}
}
],
"baseMapInfo": {
"customBaseMaps": [
{
"ibiBaseLayer": "gray"
}
]
}
}
*END
ENDSTYLE
END
```
Adding a New Role for a Server-Hosted Map

A server-hosted map is based on a shapefile. You must upload the shapefile (.dbf) to an application folder accessible to the server. The server will transform it to ibijson format.

An ESRI shape file is actually a collection of at least four files:

- **.dbf file.** The .dbf file is a standard database file used to store attribute data and object IDs. A .dbf file is mandatory for shape files.

- **.shp file.** The .shp file is a mandatory Esri file that gives features their geometry. Every shapefile has its own .shp file that represents spatial vector data.

- **.shx file.** The .shx file is a mandatory Esri shape index position file. This type of file is used to search forward and backwards.

- **.prj file.** The .prj file is an optional file that contains the metadata associated with the shapefiles coordinate and projection system.

All files must have exactly the same name and to be located in the same directory. If they are not, the shapefile conversion will fail.

When there are several possible keys associated with a geometry, a drop down list of detected key names will be displayed. Select any one of these fields. No selection required when there is a single geometry key.

The shapefile should only be in the GCS_WGS_1984 - World Geodetic System 1984 (decimal degrees) coordinate system.
To add a geographic role for a Server-hosted map, click *Add WFRS map* on the Geographic Configuration Editor.

The *Add WFRS hosted map* dialog box opens, as shown in the following image.

Configure the following properties.

**role name**
Is a name for the geographic role.

**Geometry type**
Select either POLYGON or POINT from the drop-down list.

**Esri shape**
Enter the name of the application directory where the shapefile resides, or click the ellipsis (...) to navigate to the application directory. Then select the .dbf file for the role.

**Load to app**
Enter the name of the application directory where you want to place the ibjson file, or click the ellipsis (...) to navigate to the application directory.

**Quantization type**
Quantization is the process of transforming a large set of input values to a smaller set of values. When transforming the shapefile, the server will quantize points that are too close together in order to optimize map rendering performance. Two methods are available for quantization, LINEAR or GRID. The default is LINEAR.

**Quantization_X**
Is the threshold value for the x-axis.

**Quantization_Y**
Is the threshold value for the y-axis.
If the map has multiple keys, a drop-down list displays so that you can select one, as shown in the following image.

Click OK when you have finished configuring the properties.

The new role is added to the configuration as a customized role, as shown in the following image.

Click Save to save this role to the configuration.
You can test the role by right-clicking the role in the configuration editor and clicking Test. A sample map will be generated, as shown in the following image.

Adding NUTS Support

Nomenclature of territorial units for statistics (NUTS) are geographic roles specific to the European Union.

To add NUTS geographic roles to the configuration, click Add NUTS support on the Geographic Configuration Editor.
The NUTS geographic roles are added, as shown in the following image.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Title</th>
<th>Return Geometry</th>
<th>Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUTS0</td>
<td>customized</td>
<td>Country (NUTS level 0)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS0_CC</td>
<td>customized</td>
<td>Country (NUTS level 0)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS1</td>
<td>customized</td>
<td>Region (NUTS level 1)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS1_CC</td>
<td>customized</td>
<td>Region (NUTS level 1)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS2</td>
<td>customized</td>
<td>Province (NUTS level 2)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS2_CC</td>
<td>customized</td>
<td>Province (NUTS level 2)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS3</td>
<td>customized</td>
<td>District (NUTS level 3)</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>NUTS3_CC</td>
<td>customized</td>
<td>District (NUTS level 3)</td>
<td>GEOMETRYAREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>CONTINENT</td>
<td>standard</td>
<td>Continent</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>COUNTRY</td>
<td>standard</td>
<td>Country</td>
<td>GEOMETRY_AREA</td>
<td>ESRI</td>
</tr>
<tr>
<td>STATE</td>
<td>standard</td>
<td>State</td>
<td>GEOMETRY_AREA</td>
<td>WFRS/ESRI</td>
</tr>
<tr>
<td>COUNTY</td>
<td>standard</td>
<td>County</td>
<td>GEOMETRY_AREA</td>
<td>WFRS/ESRI</td>
</tr>
<tr>
<td>CITY</td>
<td>standard</td>
<td>City</td>
<td>GEOMETRY_POINT</td>
<td>WFRS/ESRI</td>
</tr>
<tr>
<td>POSTAL-CODE</td>
<td>standard</td>
<td>Postal code</td>
<td>GEOMETRY_AREA</td>
<td>WFRS/ESRI</td>
</tr>
</tbody>
</table>

Click Save to save these roles to the configuration.

**Adding Support for Extended Postal Codes**

Click *Add extended postal codes* to add support for Level 1 and Level 2 postal codes used in certain countries.

**Customizing Vocabulary Rules**

For each geographic role, a set of vocabulary rules define how to recognize when a field name should automatically be assigned to that role. If you right-click a role, you can click *Customize vocabulary* from the context menu.

Elements in a rule are connected by the Boolean logic operation OR (only one needs to be satisfied). Each vocabulary element contains words enclosed with special characters. Words in the rule element are connected by the Boolean logic operation AND (all need to be satisfied).

A word may be prefixed and/or suffixed with the percent character (%), which is a placeholder for any sequence of characters. If an element contains more than one word, each word has to be prefixed by the character plus (+) or minus (-). Plus indicates that the word must be found in the column name. Minus indicates that word must not be found in the column name.
For example, the following are the vocabulary rules for the role COUNTRY.

To add another rule, click *Add optional*. When you are finished, click *OK*. Click *Save* to save these rules to the configuration.

**Linking to Your WebFOCUS Client Repository**

You can map an application directory to access your WebFOCUS Client Repository.

In order to map an application to your WebFOCUS Client Repository, you must have the WebFOCUS Client REST Adapter configured, as described in the *Adapter Administration* manual.

**Procedure:** How to Link to an Existing WebFOCUS Client Repository

1. From the Applications page, click *Manage* on the ribbon, then *External Repository*, then *WebFOCUS Client Repository*, then *Link to Existing*.

   You can also right-click the Application Directories tree and click *Manage*, then *External Repository*, then *WebFOCUS Client Repository*, then *Link to Existing*. 

Server New Features

39
Linking to Your WebFOCUS Client Repository

The WebFOCUS Client Repository connection page opens, as shown in the following image.

2. Enter a prefix or accept the default value. The prefix can be any string but, by default, it is set to the machine name running the WebFOCUS Client. This name is derived from the WebFOCUS Client REST Adapter connection.

3. Click OK.

The New Application pane opens with the Application Type set to map to your WebFOCUS Client REST connection, as shown in the following image.

4. Enter or select values for the following parameters.

**Application Type**
Select *Application mapping to hostname - WebFOCUS Client REST (CON01)*

**Application Name**
Enter an application name for the mapped repository application, or accept the default name.

**Physical location**
The default WebFOCUS Repository location will be automatically entered based on your entries on the first connection screen. However, you can click the ellipsis to browse for a location.
Map to
Select one of the following mapping selections.

- **Existing application.** The Repository directories will be added to an existing application directory.
- **New application (directory will be created).** A new directory for the WebFOCUS Client Repository files will be created.
- **Existing application, recreated (all files will be deleted).** An existing application directory will be deleted and recreated with the WebFOCUS Repository files.

Add directory to APPPATH
This check box is selected, by default, so the application will show on the Application Directories tree.

Position in APPPATH
By default, the position is Last. You can select First.

Profile
Select a profile from the drop-down list. The default is edasprof.

5. Click **OK**.

The application is added to the Application Directories tree. You can manage WebFOCUS Client Repository files using this application directory.

**Procedure:** How to Delete a Link to a WebFOCUS Client Repository

1. From the Applications page, click **Manage** on the ribbon, then **External Repository**, then **WebFOCUS Client Repository**, then **List and Delete**.

You can also right-click the Application Directories tree and click **Manage**, then **External Repository**, then **WebFOCUS Client Repository**, then **List and Delete**.

A pane opens listing the WebFOCUS Client Repositories that are linked as applications.

2. Right-click a host name in the Prefix column, or click the down arrow in the Prefix column, and click **Delete reference to remote repository**.

A dialog box opens asking you to confirm that you want to delete the reference.

3. Click **OK**.

The application directory will not be deleted, but the reference to the WebFOCUS Repository will no longer be available to the server.
Upload Support for JSON

The Upload Wizard now supports upload of .json files.

Automatic Scrolling Of the Business View Pane When Dragging and Dropping

When a synonym is open in DataAssist, if you need to drag a field to a location below the visible area of the screen, the pane will automatically scroll as you drag, so you can drop the field where you need to place it.

Adding Captions for Check List Values

If you use a check list filter for a field that contains numbers or codes, you can configure the filter to have an additional column into which you can drag a field that will display a description of each value. For example, if you list product codes as a check list, you can also list the associated product names.

The check list is the default type of filter for integer and alphanumeric fields. For other numeric fields, the default filter type is the slider. However, you can use the Display option on the filter pane drop-down menu to change the type of filter.

Once you have created a check list for a numeric field, you can click Two columns (drop a field) from the Display option on the filter pane drop-down menu, as shown in the following image.
Next, you can drag a field that will describe these values to the second column, as shown in the following image.

**Note:** The additional description column does not become part of the filter. To clear the additional column, click *Plain* instead of *Two columns (drop a field)* from the Display option on the filter pane drop-down menu.

**Using a LIKE Relation With a Text Box Filter**

When you create text box filter, the Relation menu on the filter pane drop-down menu now has Like as an option, as shown in the following image.
To use LIKE, you enter a pattern for data values to match.

You can click a row in the filter pane and add a LIKE pattern in the text box.

In the LIKE pattern:

- If you enter an underscore character (\_) in any position of the pattern, any single character in that position will match the pattern.

- If you enter a percent symbol (%), any number of characters in that position will match the pattern. The % symbol is not needed at the end of a pattern, it is added automatically, as shown in the following image, where the entered pattern was P.

![Model, LIKE](image)

You can add additional rows in order to add additional patterns by clicking the plus sign (+) at the top of the filter pane.

For the pattern entered above, any model starting with the letter P will match the pattern.
In order to apply the filter, click the Apply filter button (the check mark) at the top of the panel.

Right-click the filter in the Business View pane and click Sample Data to open an output window for the filter. Click Sample Data in this window to display values that match the pattern, as shown in the following image.
Comparing Values to a Lookup File

You can create a filter that compares the values in a source file to the values in a lookup file. The filter has the value 1 if the value from the source file is found in the lookup file, and the value 0 (zero) if the value from the source file is not found in the lookup file.

From the filter pane drop-down menu, point to Lookup other file and click Check: has value.

The Select Lookup Synonym dialog box opens, as shown in the following image.

![Select Lookup Synonym dialog box](image)

The top of the dialog box has check boxes for conditions for automatically matching to the lookup field. By default, foreign-to-primary key match and matching names are selected. You can change the selections.
Once you have made your selections, select an application directory. The synonyms with fields that match your selections display in the right pane, as shown in the following image.

Select the synonym you want to use. The DB_INFILE dialog box opens. If it doesn't automatically find a match, you can select the matching fields. However, if it could find a match using the methods checked at the top of the Select Lookup Synonym dialog box, the source field and lookup field are listed, as shown in the following image.

When you are finished, click OK.
To see the filter that was created, right-click the filter name in the Tables and Columns pane, and click Properties. The DB_INFILE calculation is displayed in the Expression text box, as shown in the following image.

If you want to change the lookup properties, you can edit the expression, and click Apply.

**Converting a Filter Value List to a Variable Expression**

When you insert a check box filter using the EQ (equal) or NE (not equal) relation for a field that has a small list of possible values that do not change often, you can convert the filter to a variable expression that improves performance.

The option to convert the filter to a variable expression is on the Values option of the filter pane drop-down menu, as shown in the following image.
The variable expression is added to the synonym under Variables. Right-click the variable expression and click **Properties** to see the expression, as shown in the following image.

### Storing Server Log Files in a User-Specified Location

By default, the server log files edaprint.log, hliprint.log, rmldata.log, and edadata.log are stored in the edatemp directory. This directory is created under the EDATEMP location which, by default, is set to EDACONF. To override this location, set the EDALOG environment variable to a physical directory prior to starting the server. The edatemp directory will still be created under EDATEMP.

### Downloading a Deferred Jobs Report

You can download a deferred jobs report to your personal computer. On the Workspace page, open the Special Services and Listeners folder. Click the arrow next to Scheduler or right-click **Scheduler**, and click **Deferred List** on the context menu. On the Deferred List page, click the arrow or right-click a deferred job and click **Download**, as shown in the following image.

The default location for the downloaded report is your Downloads directory.

### Applications Page Redesign

The Applications page on the Web Console has been redesigned to include:

- Multi-selection of files and folders for copy, cut, paste, and delete.
- Column-based filters.
- An upgraded integrated search.
- A customized layout with drag and drop of all frames on all Web Console pages and the ability to view items as lists, or tabs.
Bubble Help for Guiding the User to the Next Step

Context sensitive bubble help pops up to guide the user to the next step when performing Web Console procedures. For example, the following image shows bubble help that pops up when creating a cluster synonym with a Business View:

![Available Objects for MS SQL Server ODBC (CON01)]

<table>
<thead>
<tr>
<th>Available Objects for MS SQL Server ODBC (CON01)</th>
</tr>
</thead>
<tbody>
<tr>
<td>? Object Type</td>
</tr>
<tr>
<td>? Owner/Schema</td>
</tr>
<tr>
<td>? Database</td>
</tr>
<tr>
<td>Customize data type mappings</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>Action</td>
</tr>
<tr>
<td>Header</td>
</tr>
<tr>
<td>Fact</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
<tr>
<td>select</td>
</tr>
</tbody>
</table>

Modern Design of Server Web Console

The server Web Console has been modernized with the use of Google material design icons and the Awesome font.
Left-Click Menu on the Web Console

Web Console items that have a popup context menu now have a down arrow that you can click to open the menu, as shown in the following image.
edastart Command Option for Updating Connection Attributes

This option is controlled by the Left Click on tree to Show Menu option on the Preferences page of the User menu found on the top right of the Web Console page, as shown in the following image.

![Preferences Page](image)

edastart Command Option for Updating Connection Attributes

The edastart command now has options to update adapter connection attributes. For example, you can use the following command to update the password.

```
UPDATE_PROFILE prof SUFFIX suffix CONNECTION conname NEW_PASS passwd
```

where:

- **prof**
  - Is the name of the profile in which to change the attributes.

- **suffix**
  - Is the suffix for the adapter whose connection attributes are being updated.

- **conname**
  - Is the name of the connection for which the attributes are being updated.

- **passwd**
  - Is the new password for the connection.

Turning a Configured Server Into a Docker Container

A Docker container is a standalone executable package containing everything needed to run a piece of software while isolating it from the environment.
The server now includes a UNIX/Linux shell script that you can use as a template for turning a configured server into a Docker container. Turning a configuration into a Docker container requires a working Docker environment where the server administrator is allowed to create and manage Docker containers. The shell script is in the EDAHOME/bin directory and is named ibidockerize.sh.

**Procedure: How to Turn a Server Configuration Into a Docker Container**

Prior to editing the script, visit the Docker web site (https://www.docker.com/) for information on Docker features to accomplish your task.

The template is written for UNIX and Linux, however, the same concepts/steps apply to Windows (it is up to the customer to rewrite it for windows, if needed).

1. Copy the ibidockerize.sh file from the EDAHOME/bin directory to a local directory and open it for editing.
   
   **Note:** The script is highly commented and easy to follow (the actual program code is less than 10 lines).

2. Edit the installation paths and HTTP ports to the ones used in your server configuration.

3. Add paths for any additional software needed to support the configuration, such as DBMS software and JVM software.
   
   You can edit the script to conform to the needs of your site.

4. Save and run the script to create the container.
   
   You can then redistribute the container according to your needs.
Setting the Maximum Width of a Trace Line

By default, the maximum width of a trace line has been increased to 230 characters to increase the readability of traces by reducing wrapped lines. You can change this default value on the Workspace page. Open the Logs and Traces folder, click the arrow or right-click Traces, and click Configure. The Traces Configuration page opens, as shown in the following image.

![Traces Configuration Page](image)

Enter a new value in the Set column width of trace text box, and click Save and Restart Server.

Reporting Server on z/OS 64-Bit Version

The Reporting Server on z/OS is now a 64-bit version. This implementation allows a larger number of simultaneous agents and connections in a single server instance, provides performance improvements, and enables the server to access additional memory so it can take advantage of modern core language features that require a larger memory footprint.

Debugging a Dialogue Manager Procedure

The Web Console now has a Dialogue Manager Debugger tool. Using this tool, you can execute a procedure that contains Dialogue Manager commands while watching and changing the values of the Dialogue Manager variables in the procedure. You can set and remove breakpoints in the procedure, step into each line of code, step over execution of external procedures, continue to a breakpoint, execute an ad-hoc procedure, and open HOLD files created in the procedure. You can save the session information of one debug session.

While the procedure is executing, the Dialogue Manager Debugger shows the values of all local variables, global variables, and, if requested, system variables. If a TABLE request is executed, the debugger shows the FOCUS stack with the commands being executed. If a variable that has not been defined is used in the procedure, the Dialogue Manager Debugger prompts you for a value.
Note that when the Dialogue Manager Debugger is executing a procedure, it is not running that procedure in an interactive WebFOCUS environment, so interactive Dialogue Manager commands (such as -PROMPT) that require a response from the user are not executed. Also, you cannot edit the procedure in the debugger environment. If you want to edit the procedure, you must do it from the Applications page or in a text editor.

**Procedure:** How to Use the Dialogue Manager Debugger

The Dialogue Manager Debugger is for use with WebFOCUS procedures that contain Dialogue Manager commands.

1. To open a procedure in the Dialogue Manager Debugger, right-click the procedure or click the down arrow on the procedure row, point to Run Advanced, and click Debug.
The procedure opens in the Dialogue Manager Debugger, as shown in the following image.

The left panel displays the procedure with line numbers. The Debug panel displays the variables and stacks used in the procedure. Once you start execution, an output panel opens on the right showing all output.

The Debug panel displays the following items.

- **Watch Local Amper Variables (&).** Displays values of the local variables you add to the list. Adding variables to this list enables you to change their values at breakpoints.

- **All Local Amper Variables (&).** Displays the values of all local variables in the procedure. You cannot change variable values from this list.
Note: If you want to include system variables, once the procedure is running, right-click All Local Amper Variables (&), and click Show system ampers, as shown in the following image.

- **Watch Global Amper Variables (&&)**. Displays values of the global variables you add to the list. Adding variables to this list enables you to change their values at breakpoints.

- **All Global Amper Variables (&&)**. Displays the values of all global variables in the procedure. You cannot change variable values from this list.

- **Breakpoints**. Lists the breakpoints you set.

- **Focus stack**. Lists the WebFOCUS commands being executed, for example, a TABLE request.

- **Execution stack**. Lists the procedures being executed.

2. To set a breakpoint, right-click a Dialogue Manager command line or click the down arrow on the Dialogue Manager command line where you want the breakpoint to occur, and click Set Breakpoint.

A breakpoint is a stopping point for execution. You may want to set one or more breakpoints in order to examine or change variable values at those points, run an ad-hoc request, or open a HOLD file created in the procedure.
Once you set a breakpoint, a solid circle appears in the Brk column on that line of the procedure, and the breakpoint is added to the list of breakpoints in the Debug panel, as shown in the following image.

You can disable or clear a breakpoint by right-clicking the line or clicking the down arrow on the Dialogue Manager command line where the breakpoint is set and clicking Disable Breakpoint or Clear Breakpoint. A disabled breakpoint appears as a white circle with only an outline.

You can also right-click Breakpoints in the Debug panel and click one of the following options from the context menu.

- Add new breakpoint.
Enable all breakpoints.

Disable all breakpoints.

Remove all breakpoints.

If you click Add new breakpoint, a dialog box opens for you to enter a procedure name and line number for the breakpoint, as shown in the following image.

If the breakpoint is for the current procedure, leave the Focexec text box blank.

Click Submit to set the breakpoint.

3. To add variables to a watch list, right-click Watch Local Amper Variables (&) or Watch Global Amper Variables (&&) or click the down arrow next to one of these lists, and click either Add new amper or Add all ampers.

By default, as variables are set or changed, they are listed under All Local Amper Variables (&) or All Global Amper Variables (&&). You can add specific or all variables to the Watch Local Amper Variables (&) list or the Watch Global Amper Variables (&&) list if you want to change their values at a breakpoint.

If you click Add new amper, a dialog box opens in which you enter the name of the variable to watch (without the preceding ampersands).
The variable is added to the watch list, as shown in the following image.

4. To start the debugger, click **Start** on the ribbon.

   The commands for executing the procedure and running an ad-hoc request become active.

   If the procedure references a variable that has not been given a value, a dialog box opens to prompt you for a value when the variable is encountered, as shown in the following image.

The following options are available from the ribbon.

**Execution Options**

- **Step Into.** Steps into commands in an external procedure (invoked by the EX or INCLUDE commands). Otherwise, Step Into executes to the next Dialogue Manager command in the procedure.
- **Step Over.** Executes to the next Dialogue Manager command in the procedure, bypassing commands in an external procedure (invoked by the EX or -INCLUDE command).

- **Continue.** Executes all commands to the next breakpoint.

- **Step Out.** Executes the remainder of the current procedure without stopping at breakpoints.

- **Stop Debugging.** Ends the debugging session and stops the execution of all procedures in the execution stack. You can click Start to start a new debugging session.

**Ad Hoc Options**

- **Run Ad Hoc Focexec.** Opens a window for entering and running a FOCEXEC, as shown in the following image.

![Image of ad hoc Focexec window]

**File Options**

- **Open Focexec.** Opens a file picker dialog box that enables you to open a FOCEXEC file (.fex).

- **Open Master.** Opens a file picker dialog box that enables you to open a Master File (.mas).
Open Hold File. Opens dialog box in which you can enter the name of a HOLD file created in the debugging session as long as the procedure is still running.

Save Debug Session. Saves the current debugging session. Only one session can be saved. A new session will overwrite the existing saved session.

Restore Debug Session. Restores the saved debugging session.

Clear Debug Session. Clears the saved debugging session.

Switch Viewing File. If you have opened or executed multiple files, lets you select among them.

Close. Closes a file you opened from the File options button, if it is the file currently being viewed. The root FOCEXEC cannot be closed. To close the debugger, along with the file used to open the debugger, click the X at the top right of the debugger page.

Enter a value and click Submit.

Example: Using the Dialogue Manager Debugger

The following procedure is named dmdbugger3.fex. It includes a variable named &CTR whose value is set in the procedure, and a variable named &REGION whose value is not set in the procedure. The procedure contains a TABLE request and two external requests. The first external request displays output and the second creates a HOLD file named cathold.

```
-SET &CTR = 1;
-TYPE REQUEST # &CTR;
TABLE FILE WF_RETAIL_LITE
SUM DAYSDELAYED AS DAYS
BY TIME_MTH
WHERE BUSINESS_REGION EQ '&REGION'
WHERE_GROUPED DAYSDELAYED GT 200
ON TABLE SET PAGE NOPAGE
END
-RUN
-TYPE END OF REQUEST &CTR;
-SET &CTR = &CTR+1;
-TYPE REQUEST # &CTR;
EX doc77x/catfex2
-RUN
-TYPE END OF REQUEST &CTR;
-SET &CTR = &CTR+1;
-TYPE REQUEST # &CTR;
EX doc77x/catfex3
-RUN
-TYPE END OF REQUEST &CTR;
-TYPE END OF PROCEDURE
```
The following image shows this procedure open in the debugger, with breakpoints set on Dialogue Manager commands before and after each request. The breakpoints are listed on the Debug panel.

```
1. Server Enhancements

2. Server New Features  63
```
Clicking Start, then Continue stops at the first breakpoint, where the value of the &CTR local variable is shown as 1. The value of the &GEO_UNIFIED_ROLE global variable is shown as ON. The execution stack shows the name and line number of the procedure being executed, as shown in the following image.
Clicking Step Into opens the following dialog box that asks you to enter a value for the &REGION variable.

![Enter value dialog box]

Entering EMEA and clicking Submit proceeds to the next Dialogue Manager command. REGION is added to the list of local variables, and the first -TYPE message displays in the output window, as shown in the following image.

![Output window with local variables and -TYPE message]
Before the request is actually run, the TABLE commands to be executed are listed on the Focus Stack, as shown in the following image.

After the request is run, the output displays in the output window. The next request in the procedure is issued using a -INCLUDE command. The external procedure is opened in the debugger if either the last debugger command was Step Into, or the last debugger command was Continue and the external procedure contains a breakpoint on a line that will be executed. The following image shows the external procedure open in the debugger.
The last request in the procedure, run using an EX command, creates a HOLD file named cathold. After the request is run, but while the procedure is still running, you can open the HOLD file by clicking Open Hold File from the File menu, as shown in the following image.
A dialog box opens where you can enter the name of the hold file, as shown in the following image.
After you enter the name and click *Submit*, a report showing the HOLD file data displays in the output window, as shown in the following image.

After the procedure finishes executing, the debugger stops. You can click *Start* to start a new execution.
**Kerberos Support for JDBC-Based Data Sources**

Kerberos, an authentication protocol that provides a strong authentication for computer networks, has been enhanced for this release. Single sign-on is now supported from the browser, to the WebFOCUS Reporting Server, to JDBC-based data sources.

With server security set to OPSYS and Kerberos authentication enabled, a Windows user can connect to the server from a supported browser (such as, Internet Explorer 11, Chrome, or Firefox) to the Web Console without having to re-enter their credentials.

The simplified process results in the browser passing a stored Kerberos ticket to the WebFOCUS Reporting Server, which then passes it on to a JDBC data source that uses Kerberos and is configured with Security as Trusted.

**Git Integration With the WebFOCUS Reporting Server**

Git, a Distributed Version Control System (DVCS), is now supported by the DataMigrator Server. Git can be used on both the Reporting Server Web Console and the DataMigrator Data Management Console.

The distributed nature of Git allows a vast range of workflow possibilities for projects and teams. The two most commonly used paradigms are Centralized Workflow and Integration-manager Workflow. This Git integration with the DataMigrator Server works with both paradigms seamlessly.
This section describes new adapter features. All adapters can be used for WebFOCUS Reporting and SQL requests, and as sources for DataMigrator flows.

The server supports adapters designed to access a wide variety of data sources. Using the graphical Web Console, you can configure these adapters and create the metadata you need to seamlessly access the corresponding type of data.

On the Web Console, adapters are grouped as SQL, XML-based, ERP, OLAP, Procedures, Sequential and Indexed, DBMS, and Social Media.

In this chapter:

- SQL Adapters
- OLAP Adapters
- ERP Adapters
- Procedures Adapters
- Sequential and Indexed Adapters
- Statistics Adapters
- XML Adapters

### SQL Adapters

This section provides descriptions of new features for SQL adapters.

### Diagnostics: Improved Outer Join Rejection Messages for SQL Adapters

More explanation has been added to messages that result from a rejected outer join request.

### Extended Bulk Load for SQL Adapters: Auxiliary Connection Updates

Using Extended Bulk Load with selected adapters requires that intermediate data files are transferred using [S]FTP to a host server. These adapters include:

- Apache Hive
- Cloudera Impala
In previous releases, this was done by adding a connection to the Flat File adapter and was limited to a single connection per server.

As of this release, a new Configure Bulk Load option is available, and allows you to configure an additional connection for the combination of server and adapter with a unique name. This option can be accessed by right-clicking a connection for a configured data adapter.

**Adapter for Amazon Athena**

The Adapter for Amazon Athena is new in this release. The Amazon Athena query service allows you to access data stored in Amazon Simple Storage Service (Amazon S3) by using standard SQL.

**Adapter for Amazon Redshift: Readonly Property for IDENTITY Columns**

The Adapter for Amazon Redshift supports IDENTITY data types as read-only with the property FIELDTYPE=R set in the Master File.

**Adapter for Apache Drill: TIME and TIMESTAMP Fields**

The Adapter for Apache Hive can now be used to read columns described with the TIME and TIMESTAMP data types, provided the storage plug-in used supports it.

**Adapter for Apache Spark**

The Adapter for Apache Spark is new in this release. It provides access to data using the Spark Thrift Server and JDBC driver. This adapter can be used when:

- Spark is installed with a Hadoop distribution, such as Hortonworks or MapR, and data is stored in HDFS.
- Spark is installed independently, without Hadoop, and data is stored in NFS.

**Support for DSN-Less Connections to Db2 on UNIX, Linux, and Windows**

On UNIX, Linux, and Windows you can specify the host name, port and database name instead of a DSN when configuring a connection to Db2 in the Web Console.
Setting the Db2 Effective Trusted Context User From a Procedure

A Db2 trusted context enables the connected WebFOCUS user ID to be used for database interactions instead of the authenticated user ID for the connection. Use the following command to set the trusted context user ID in a WebFOCUS procedure.

```
ENGINE DB2 SET TRUSTED_USERID conname \tcuser,tcpass
```

where:

- **conname**
  
  Is the connection name for the trusted context. This connection must be the active connection for the server when the SET TRUSTED_USERID command is issued.

- **tcuser**
  
  Is the trusted context user ID.

- **tcpass**
  
  Is the trusted context password.

**Adapter for Db2: Limited Support for Tables With CLOB Columns on i5**

While CLOB and BLOB columns cannot be used for Change Data Capture, synonyms can now be created for Table Log Records that contain them so that the other columns can be used. Attempts to retrieve CLOB and BLOB columns from Table Log Records return a MISSING value.

**Adapter for Db2: CDC for i5 Unicode Supports Multiple CCSIDs**

The Adapter for DB2 on IBM i with a Unicode server now supports a Change Data Capture (CDC/IUD) load flow when a target has columns with different Coded Character Set Identifiers (CCSIDs).

**Adapter for Cache: JDBC Adapter**

A JDBC adapter for access to Cache data sources is available in this release.

**Adapter for EXASol: Extended Bulk Load Support From Windows**

Extended Bulk Load to the EXASol system is now supported from Windows by using an auxiliary connection to a non-Windows "buffer host", which is also known as a FTP_node.
Creating Data Fields as TIMESTAMP for Jethro and Impala

In previous releases, attempting to create a table in Jethro or Impala from a synonym using the Recreate DBMS Table (or CREATE FILE) option in the DMC with fields described as YYMD or Date would fail with a syntax error. This was due to the fact that a data type would not be specified.

As of this release, such fields are created as TIMESTAMP and an error does not occur.

**Note:** Neither Jethro or Impala have a DATE datatype.

Adapter for Oracle: Extended Bulk Load Support for BLOB and CLOB Columns

The Adapter for Oracle now supports CLOB and BLOB Extended Bulk loading that provides improved performance using the Direct Path API.

Adapter for Oracle: Support for System-Versioned Temporal Tables

Temporal queries against Oracle System-Versioned tables have been implemented by introducing fields with TEMPORALPROPERTY SYSTEMTIME. Such a field is added by editing the synonym for the table. For example:

```
DEFINE FIELD1/HYYMDm WITH REALFLD TEMPORALPROPERTY SYSTEMTIME = ;$
```

The following are requirements for the field with TEMPORALPROPERTY SYSTEMTIME.

- The field can have any valid field name.
- The datatype must be a date-time datatype with sufficient precision.
- The WITH clause is required and must reference a real table column.

Fields with TEMPORALPROPERTY=SYSTEMTIME can only be used in a single WHERE test of one of the following two types.

- **WHERE field EQ expression**

  This type of WHERE test translates to an ORACLE AS OF TIMESTAMP clause.

- **WHERE field FROM expression1 TO expression2**

  This type of WHERE test translates to a VERSIONS BETWEEN TIMESTAMP clause.
where:

```
expression, expression1, expression2
```

Must be date-time expressions or constants. When used in a FROM-TO expression, the constants `DT('0001-01-01')` and `DT('9999-12-31')` represent the oldest and most recent applicable timestamps. They translate to the ORACLE keywords MINVALUE and MAXVALUE.

**Adapter for Microsoft SQL Server ODBC: Version 2017 Support**

SQL Server 2017 is supported with the MSODBC version of the adapter.

**Adapter for Microsoft SQL Server ODBC: SQL Server 2017/2016 Always Encrypted Support**

The Windows ODBC version of the Adapter for SQL Server supports the SQL Server 2017/2016 Always Encrypted native feature under the following conditions:

- The columns have to be encrypted using SQL Server Management Studio prior to creating any synonyms.
- The adapter connection is configured with the connection string keyword `ColumnEncryption=Enabled`.

When a synonym is created, the Access File will contain the attribute `ENCRIPT_TYPE={DETERMINISTIC|RANDOMIZED}` for the encrypted columns.

Certain operations on encrypted columns cannot be performed in SQL. The adapter logic will account for that, and those operations will be performed by the WebFOCUS Reporting Server.

**Adapter for Microsoft SQL Server: Extended Bulk Load Support for BLOB and CLOB Columns**

The Adapter for Microsoft SQL Server now supports CLOB and BLOB Extended Bulk loading that provides improved performance. To take advantage of this feature the version of Microsoft BCP utility should be 11.0 or higher.

**Adapter for Microsoft SQL Server OLE DB Support**

Microsoft reversed its previous decision, and un-deprecated OLE DB technology on 3/31/2018, with the release of MSOLEDBSQL Driver version 18. Microsoft calls this release the third generation of OLE DB technology.

Accordingly, Information Builders removed *obsolescent* status from its OLE DB adapter, and introduced support of new MSOLEDBSQL Driver 18.

Customers may now choose between MS SQL Server OLE DB or ODBC adapters, depending on required features.
Note that first (SQLOLEDB) and second (SQLNCLI/SNAC) generations of Microsoft OLE DB Providers remain deprecated.

**Adapter for Microsoft SQL Server Azure Data Warehouse**

The Adapter for Microsoft SQL Server Azure Data Warehouse is new in this release. It enables you to access data stored in the Microsoft parallel processing data warehouse architecture.

**Adapter for SQL Server: Azure SQL Database Support**

The Adapters for SQL Server (suffixes SQLMSS and MSODBC) now support read/write access to the Azure Database. The MSODBC version of the Adapter is recommended on platforms where the MS ODBC driver is available.

**Adapter for Teradata CLI: Extended Bulk Load Support for BLOB, CLOB, and String Columns**

The CLI Adapter for Teradata now supports loading CLOB and BLOB data into fields of an existing target using Extended Bulk (MERGE INTO command transpires into LOB_DEFERRED_BY_NAME method).

**Note:**

- in previous server releases, CLOB/BLOB write functionality was available only with the ODBC version of the adapter and not using Extended Bulk Load.
- The Teradata TPT API, version 15.10.01.08 or higher is recommended in order to take advantage of this feature.

**Adapter for Teradata CLI: Support for Stored Procedures with Dynamic Result Set**

The CLI Adapter for Teradata now supports reporting from Teradata stored procedures that return a dynamic result set.

**Adapter for Teradata: FASTLOAD**

As of this release, when the adapter for Teradata has the BULKLOAD setting set to ON, the Teradata load procedures FASTLOAD or MULTILOAD are used for best performance.

Previously, when creating a new table in Teradata from the Web Console using Custom Copy or the HOLD...FORMAT SQLDBC (FOCUS) command, a parameterized INSERT statement was generated, which was not the fastest option.

**OLAP Adapters**

This section contains descriptions of features for OLAP adapters.
Adapter for Microsoft SQL Server Analysis Services: Support for Microsoft Azure

The Adapter for Microsoft SQL Server Analysis Services supports connection to and querying from the Analysis Services Server hosted by Microsoft Azure.

ERP Adapters

This section describes new features for ERP adapters.

Adapter for Salesforce.com: Extended Bulk Load Now Available

Extended Bulk Load is now available on the adapter for Salesforce.com, making it the very first non-SQL adapter with this capability. The Salesforce.com adapter can be found in the ERP adapter folder list.

Support for SAP S/4 HANA, the Next Generation of ERP Business Suite From SAP Replaces SAP ECC/ERP

Support for SAP S/4 HANA, the next generation of ERP business suite from SAP replaces SAP ECC/ERP. The Adapter for SAP now supports SAP S/4 HANA. Customers who rely on the current Adapter for SAP can now take advantage of all the benefits of SAP S/4 HANA, while reusing business logic and data in new application initiatives.

Adapter for SAP/R3: Support for Strict Mode in Release 7.40 and Up

Starting with Release 7.40, SAP implemented a stricter check for and enforcement of Open SQL syntax rules:

- List elements in Open SQL statements are separated by a comma.
- Host variables in Open SQL statements are escaped by an at symbol (@).

To enable a user to control use of strict mode, the following command has been introduced.

```
ENGINE SQLSAP SET SQLSTRICTMODE {ON|OFF|AUTO}
```

where:

ON

Enforces use of strict mode, regardless of the target SAP system capabilities.

OFF

Turns off strict mode.

AUTO

Sets strict mode automatically depending on the target SAP system kernel release (Kernel 740 SP02 and higher support ON). This is the default value.
To revert to the automatic setting, issue the following command.

```
ENGINE SQLSAP SET SQLSTRICTMODE RESET
```

Procedures Adapters

This section provides descriptions of new features for procedures adapters.

Adapter for REST: RESTTSCQ Support for Custom Headers

The Adapter for REST now supports custom headers in the FILEDEF RESTTSCQ REST request trace.

Adapter for REST: Create Synonym Support for Custom Header Information

The Create Synonym page for the Adapter for REST now supplies a text box in which you can enter custom header name=value pairs that are passed in the HTTP header of the REST web service call. Each name=value pair must be separated from the previous pair with a semicolon (;). For example:

```
Content-Type="CDF";PARAM1="ABC"
```

The custom header information is only passed if the web service call requires the custom header parameters. You can change the parameter values at runtime using a WHERE or IF command.

Adapter for REST: Support for URL in the Synonym

You can edit a synonym for the Adapter for REST to contain the endpoint URL for unsecured services instead of placing it in the connection string.

Delete or null out the CONNECTION parameter in the Access File and add the SERVICEURL attribute to point to the endpoint URL. If the OBJECT attribute exists in the Access File, the full URL is a concatenation of the SERVICEURL and OBJECT values. For example, consider an Access File that contains the following attributes:

```
CONNECTION=,
SERVICEURL=http://api.geonames.org,
OBJECT=ABC,
```

The endpoint URL will be:

```
http://api.geonames.org/ABC
```
**Adapter for WebFOCUS**

The Adapter for WebFOCUS integrates with the WebFOCUS Client through WebFOCUS RESTful Web Services. Explicit and Password Passthru authentication to WebFOCUS is handled by configuring the connection to the adapter. A successful authentication returns a Cross-Site Request Forgery (CSRF) token that is automatically passed on each WebFOCUS RESTful Web Service call.

When you right-click the connection and click *Show DBMS objects*, you can then click *Create Synonym(s) and Examples*, which loads a set of examples that include Master Files, Access Files, and Procedures. Each of these examples performs a specific WebFOCUS function, such as security maintenance, ReportCaster scheduling, and folder maintenance.

To configure the adapter, right-click the adapter on the Available list and click *Configure*. The Add Connection page opens. Configure the following parameters.

- **Connection Name**
  Is a name for the connection.

- **WebFOCUS Base Url**
  Is the URL used to access WebFOCUS, in the form http://computername:port/ibi_apps.

- **Security**
  Select Explicit or Password Passthru. For Explicit authentication, enter the WebFOCUS user ID and password.

**Sequential and Indexed Adapters**

This section provides new feature descriptions for sequential and indexed adapters.

**Storing Images on Filesystem From a Database Table**

In previous releases, when using DataMigrator to copy data from a database table with a field described as BLOB for a column that contains a picture to a Delimited Flat File target, the name of a temporary file was written to that file.

Now, in the location specified for the target file, a like-named sub-directory is created that contains a file with each image from the source table with the appropriate extension (file type). This is currently available for the following file formats: .gif, .jpeg, .png, or .svg.

**Test for Write Access to SFTP Server**

As of Release 7.7.08, when testing a connection using [S]FTP, both reading and writing are tested and the message displayed indicates the success or failure of each.

In previous releases, when configuring a connection to an [S]FTP server for fixed or delimited flat files, the Test option would only test reading from the remote server.
However, these adapters could be used to write to the remote server, either independently or as an auxiliary connection for an adapter, such as Impala or Drill, that required a file to be transferred to the server where they were running.

In these cases, even though the test successfully passed, the use of the adapter could still fail. An example of this is if the user did not have write access to the specified directory.

**Adapter for Hyperledger Fabric**

The Adapter for Hyperledger Fabric is new in this release. It allows you to seamlessly communicate with IBM Hyperledger Fabric, which provides support for permissioned blockchain applications. Blockchain technology is a distributed record of transactions validated and stored at multiple peer locations in a network. The records stored in the blockchain are immutable, providing an extremely high level of security and integrity.

**Statistics Adapters**

This section contains descriptions of features for statistics adapters.

**Adapter for Rserve Integration**

The Adapter for Rserve enables you to define a connection to Rserve for remotely executing R scripts to be used with WebFOCUS reports and charts as a summary (Compute) or virtual field.

Rserve is a TCP/IP server that allows users to run R scripts directly from WebFOCUS or the Reporting Server without the need to initialize R or link to an R library. The original Rserve paper is available at [http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings/Urbanek.pdf](http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings/Urbanek.pdf).

For more information on Rserve, see [https://rforge.net/Rserve/index.html](https://rforge.net/Rserve/index.html).
**Reference:** Configuring the Adapter for Rserve

On the Connect to Data page of the Reporting Server Web Console, right-click *Rserve* on the Available list, and click *Configure*, as shown in the following image.

The Add Rserve to Configuration page opens, as shown in the following image.

Enter the following connection parameters.

**Connection Name**

Is a name for this connection.

**Server**

Is the hostname of the Rserve host.

Rserve does not have to be installed on the Reporting Server host machine. However, it must be accessible to the Reporting Server.
Port
Is the port on which the Rserve host listens.

Security
There are three methods by which a user can be authenticated when connecting to Rserve:

- **Explicit.** The user ID and password are explicitly specified for each connection and passed to Rserve, at connection time, for authentication.

  If you select Explicit authentication, enter the Rserve user ID and password.

- **Password Passthru.** The user ID and password received from the client application are passed to Rserve, at connection time, for authentication.

- **Trusted.** The adapter connects to Rserve using the rules for an impersonated process that are relevant to the current operating system.

**IBI_CLASSPATH**
Are additional Java Class directories or full path jar names to be available for Java Services, each on a separate line.

The files REngine.jar and RServeEngine.jar are required. They are available on the Rserve web site, [https://www.rforge.net/Rserve/files/](https://www.rforge.net/Rserve/files/). You must add the path to the location of these jar files to IBI_CLASSPATH.

Click **Configure**.

If the configuration is successful, the following message displays.

**Rserve successfully added to configuration**

In addition, you can click **Test** on the context menu for the connection.

**Reference:** Creating a Synonym for an R Script

Each R script used with the Adapter for Rserve must have a synonym that describes the script’s independent variables and dependent variable. The Master File will contain the list of input (independent) variables and the output (dependent) variable. The Access File will contain information about the script and data files.

The synonym will be created using a sample file that contains only the fields that are input parameters for the script. A few rows of sample data are sufficient for the Adapter for Rserve to determine the appropriate data types and lengths of the parameters. The sample file must be a .csv file.
To create a synonym for an R script, right-click a connection for the Adapter for Rserve and click Create metadata objects. The Create Synonym for Rserve page opens, as shown in the following image, where the connection name is MyRserve.

Select or enter values for the following parameters.

**R Script location on R server**
Is the remote location, including script name and extension, for the R script file (.R). Leave blank if the R script is in an application folder accessible to the Reporting Server.

**Select file with sample input data for the R Script**
Open the file picker (…) to select the application directory and file that contains the sample data for creating the synonym. Click OK.

**Application**
Open the file picker (…) to select the application that contains the R script. Select the R script file from the file picker and click OK.

**Synonym Name**
Enter a name for the resulting synonym, or accept the default name.

When you have finished entering the synonym creation parameters, click Create Synonym on the ribbon, as shown in the following image.
Example: Creating a Synonym for an R Script

The following R script named wine_run_model.R predicts Bordeaux wine prices based on the average growing season temperature, the amount of rain during the harvest season, the amount of rain during the winter, and the age of the wine.

```r
# filename: wine_run_model.r

args <- commandArgs(trailingOnly=TRUE)
input_file <- file.path(args[1])
output_file <- file.path(args[2])

wine_test <- read.csv(input_file)

wine_model <- readRDS("/prediction/wine_model.rds")

results <- predict(wine_model, newdata = wine_test)

colnames(results) <- c('Price')
write.csv(results, file=output_file, row.names=FALSE)
```

The following sample data file named wine_input_sample.csv contains the names and sample values for the independent variables used in this model.

"AGST", "HarvestRain", "WinterRain", "Age"
16.1667, 122, 717, 4
16, 74, 578, 3

The synonym creation page for this script is shown in the following image.
Clicking *Create Synonym* on the ribbon generates the wine_run_model synonym. The Master File, wine_run_model.mas, describes the independent (input) variables and the dependent (output) variable, as shown below:

```plaintext
FILENAME=WINE_RUN_MODEL, SUFFIX=RSERVE , $
SEGMENT=INPUT_DATA, SEGTYPE=S0, $
   FIELDNAME=AGST, ALIAS=AGST, USAGE=D9.4, ACTUAL=STRING,
   MISSING=ON,
   TITLE='AGST', $
   FIELDNAME=HARVESTRAIN, ALIAS=HarvestRain, USAGE=I11, ACTUAL=STRING,
   MISSING=ON,
   TITLE='HarvestRain', $
   FIELDNAME=WINTERRAIN, ALIAS=WinterRain, USAGE=I11, ACTUAL=STRING,
   MISSING=ON,
   TITLE='WinterRain', $
   FIELDNAME=AGE, ALIAS=Age, USAGE=I11, ACTUAL=STRING,
   MISSING=ON,
   TITLE='Age', $
SEGMENT=OUTPUT_DATA, SEGTYPE=U, PARENT=INPUT_DATA, $
   FIELDNAME=PRICE, ALIAS=Price, USAGE=D18.14, ACTUAL=STRING,
   MISSING=ON,
   TITLE='Price', $
```

The Access File, wine_run_model.acx, describes the names and locations of the R script and the sample data file, as shown below.

```plaintext
SEGNAME=INPUT_DATA,  
   CONNECTION=MyRserve, 
   R_SCRIPT=/prediction/wine_run_model.r, 
   R_SCRIPT_LOCATION=WFRS, 
   R_INPUT_SAMPLE_DAT=prediction/wine_input_sample.csv, $
```

---

2. Adapter Enhancements
Now that the synonym has been created for the model, the model will be used to run against the following data file named wine_forecast.csv.

```
Year,Price,WinterRain,AGST,HarvestRain,Age,FrancePop
1952,7.495,600,17.1167,160,31,43183.569
1953,8.0393,690,16.7333,80,30,43495.03
1955,7.6858,502,17.15,130,28,44217.857
1957,6.9845,420,16.1333,110,26,45152.252
1958,6.7772,582,16.1667,187,25,45653.805
1959,8.0757,485,17.4833,187,24,46128.638
1960,6.5188,763,16.4167,290,23,46583.995
1961,8.4937,830,17.3333,38,22,47128.005
1962,7.388,697,16.3,52,21,48088.673
1963,6.7127,608,15.7167,155,20,48798.99
1964,7.3094,402,17.2667,96,19,49356.943
1965,6.2518,602,15.3667,290,23,49801.821
1966,7.7443,819,16.5333,86,17,50254.966
1967,6.8398,714,16.2333,118,16,50650.406
1968,6.2435,610,16.2,292,15,51034.413
1969,6.3459,575,16.55,244,14,51470.276
1970,7.5883,622,16.6667,89,13,51918.389
1971,7.1934,551,16.7667,112,12,52431.647
1972,6.2049,536,14.9833,158,11,52894.183
1973,6.6367,376,17.0667,247,7,53332.805
1974,6.2941,574,16.3,184,9,53689.61
1975,7.292,572,16.95,171,8,53955.042
1976,7.1211,418,17.65,247,7,54159.049
1977,6.2587,821,15.5833,87,6,54378.362
1978,7.186,763,15.8167,51,5,54602.193
```

The data file can be any type of file that R can read. In this case it is another .csv file. This file needs a synonym in order to be used in a report request.

The following image shows the synonym creation page for wine_forecast.csv using the Adapter for Delimited Files.
Following is the generated Master File, wine_forecast.mas.

FILENAME=WINE_FORECAST, SUFFIX=DFIX, CODEPAGE=1252, DATASET=prediction/wine_forecast.csv, $
SEGMENT=WINE_FORECAST, SEGTYPE=S0, $
FIELDNAME=YEAR1, ALIAS=Year, USAGE=I6, ACTUAL=A5V, MISSING=ON, TITLE='Year', $
FIELDNAME=PRICE, ALIAS=Price, USAGE=D8.4, ACTUAL=A7V, MISSING=ON, TITLE='Price', $
FIELDNAME=WINTERRAIN, ALIAS=WinterRain, USAGE=I5, ACTUAL=A3V, MISSING=ON, TITLE='WinterRain', $
FIELDNAME=AGST, ALIAS=AGST, USAGE=D9.4, ACTUAL=A8V, MISSING=ON, TITLE='AGST', $
FIELDNAME=HARVESTRAIN, ALIAS=HarvestRain, USAGE=I5, ACTUAL=A3V, MISSING=ON, TITLE='HarvestRain', $
FIELDNAME=AGE, ALIAS=Age, USAGE=I4, ACTUAL=A2V, MISSING=ON, TITLE='Age', $
FIELDNAME=FRANCEPOP, ALIAS=FrancePop, USAGE=D11.3, ACTUAL=A11V, MISSING=ON, TITLE='FrancePop', $

Following is the generated Access File, wine_forecast.acx.

SEGNAME=WINE_FORECAST, DELIMITER=',', ENCLOSURE='''', HEADER=YES, CDN=COMMAS_DOT, CONNECTION=<local>, $

The following request, wine_forecast_price_report.fex, uses the RSERVE built-in function to run the script and return a report.

-*wine_forecast_price_report.fex
TABLE FILE PREDICTION/WINE_FORECAST
PRINT
  YEAR
  WINTERRAIN
  AGST
  HARVESTRAIN
  AGE

  COMPUTE PREDICTED_PRICE/D18.2 MISSING ON ALL=
    RSERVE(prediction/wine_run_model, AGST, HARVESTRAIN, WINTERRAIN, AGE, Price); AS 'Predicted,Price'

ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END

Server New Features
The output is shown in the following image.

<table>
<thead>
<tr>
<th>Year</th>
<th>WinterRain</th>
<th>AGST</th>
<th>HarvestRain</th>
<th>Age</th>
<th>Predicted Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1952</td>
<td>600</td>
<td>17.1167</td>
<td>160</td>
<td>31</td>
<td>7.72</td>
</tr>
<tr>
<td>1953</td>
<td>690</td>
<td>16.7333</td>
<td>80</td>
<td>30</td>
<td>7.87</td>
</tr>
<tr>
<td>1955</td>
<td>502</td>
<td>17.1500</td>
<td>130</td>
<td>28</td>
<td>7.68</td>
</tr>
<tr>
<td>1957</td>
<td>420</td>
<td>16.1333</td>
<td>110</td>
<td>26</td>
<td>7.00</td>
</tr>
<tr>
<td>1958</td>
<td>582</td>
<td>16.4167</td>
<td>187</td>
<td>25</td>
<td>7.02</td>
</tr>
<tr>
<td>1959</td>
<td>485</td>
<td>17.4833</td>
<td>187</td>
<td>24</td>
<td>7.54</td>
</tr>
<tr>
<td>1960</td>
<td>763</td>
<td>16.4167</td>
<td>290</td>
<td>23</td>
<td>6.76</td>
</tr>
<tr>
<td>1961</td>
<td>830</td>
<td>17.3333</td>
<td>38</td>
<td>22</td>
<td>8.36</td>
</tr>
<tr>
<td>1962</td>
<td>697</td>
<td>16.3000</td>
<td>52</td>
<td>21</td>
<td>7.51</td>
</tr>
<tr>
<td>1963</td>
<td>608</td>
<td>15.7167</td>
<td>155</td>
<td>20</td>
<td>6.63</td>
</tr>
<tr>
<td>1964</td>
<td>402</td>
<td>17.2667</td>
<td>96</td>
<td>19</td>
<td>7.56</td>
</tr>
<tr>
<td>1965</td>
<td>602</td>
<td>15.3667</td>
<td>267</td>
<td>18</td>
<td>5.92</td>
</tr>
<tr>
<td>1966</td>
<td>819</td>
<td>16.5333</td>
<td>86</td>
<td>17</td>
<td>7.56</td>
</tr>
<tr>
<td>1967</td>
<td>714</td>
<td>16.2333</td>
<td>118</td>
<td>16</td>
<td>7.11</td>
</tr>
<tr>
<td>1969</td>
<td>575</td>
<td>16.5500</td>
<td>244</td>
<td>14</td>
<td>6.60</td>
</tr>
<tr>
<td>1970</td>
<td>622</td>
<td>16.6667</td>
<td>89</td>
<td>13</td>
<td>7.32</td>
</tr>
<tr>
<td>1971</td>
<td>551</td>
<td>16.7667</td>
<td>112</td>
<td>12</td>
<td>7.19</td>
</tr>
<tr>
<td>1972</td>
<td>536</td>
<td>14.9833</td>
<td>158</td>
<td>11</td>
<td>5.88</td>
</tr>
<tr>
<td>1973</td>
<td>376</td>
<td>17.0667</td>
<td>123</td>
<td>10</td>
<td>7.09</td>
</tr>
<tr>
<td>1974</td>
<td>574</td>
<td>16.3000</td>
<td>184</td>
<td>9</td>
<td>6.57</td>
</tr>
<tr>
<td>1975</td>
<td>572</td>
<td>16.9500</td>
<td>171</td>
<td>8</td>
<td>6.99</td>
</tr>
<tr>
<td>1976</td>
<td>418</td>
<td>17.6500</td>
<td>247</td>
<td>7</td>
<td>6.92</td>
</tr>
<tr>
<td>1977</td>
<td>821</td>
<td>15.5833</td>
<td>87</td>
<td>6</td>
<td>6.71</td>
</tr>
<tr>
<td>1978</td>
<td>763</td>
<td>15.8167</td>
<td>51</td>
<td>5</td>
<td>6.91</td>
</tr>
</tbody>
</table>
XML Adapters

This section provides new feature descriptions for XML-based adapters.

Pivot Support for JSON Documents

Both the Data Management Console and Web Console provide a capability to Pivot a database column or field containing multiple values, such as a delimited list or a JSON document. This creates a new segment in the synonym that describes the internal structure of the field.

This option is now available for JSON documents. In previous releases, if the source synonym was for a JSON document itself, this option was not available.

Adapter for JSON: Reflect the WebFOCUS Query in HOLD FORMAT JSON

The following HOLD syntax for creating a JSON HOLD file uses the BY fields in the report request to create a hierarchy of the JSON output.

```sql
ON TABLE HOLD NOKEYS AS app/filename FORMAT JSON STRUCTURE HIERARCHY
```

Example: Retaining the Query Structure Using HOLD FORMAT JSON

The following request retains the BY field hierarchy in the JSON HOLD file.

```sql
TABLE FILE WF_RETAIL_LITE
SUM COGS_US REVENUE_US
BY BUSINESS_REGION
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
ON TABLE HOLD NOKEYS AS STRUCTJSON FORMAT JSON STRUCTURE HIERARCHY
END
```
The partial JSON output resulting from this request is shown in the following image.

```json
{
    "sort_key": [
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Accessories", "PRODUCTSUBCATEGORY": "Chargers", "verbs": [{"COSGS_US": 3735.00, "REVENUE_US": 7493.22}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Accessories", "PRODUCTSUBCATEGORY": "Headphones", "verbs": [{"COSGS_US": 82280.00, "REVENUE_US": 124148.55}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Accessories", "PRODUCTSUBCATEGORY": "Universal Remote Controls", "verbs": [{"COSGS_US": 79392.00, "REVENUE_US": 79392.00}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Camcorders", "PRODUCTSUBCATEGORY": "Handheld", "verbs": [{"COSGS_US": 38740.04, "REVENUE_US": 78621.88}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Camcorders", "PRODUCTSUBCATEGORY": "Professional", "verbs": [{"COSGS_US": 63944.00, "REVENUE_US": 81272.20}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Camcorders", "PRODUCTSUBCATEGORY": "Standard", "verbs": [{"COSGS_US": 84767.00, "REVENUE_US": 119886.10}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Computers", "PRODUCTSUBCATEGORY": "Smartphone", "verbs": [{"COSGS_US": 47610.00, "REVENUE_US": 72849.83}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Media Player", "PRODUCTSUBCATEGORY": "Blu Ray", "verbs": [{"COSGS_US": 319254.00, "REVENUE_US": 410102.65}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Media Player", "PRODUCTSUBCATEGORY": "DVD Players", "verbs": [{"COSGS_US": 51699.00, "REVENUE_US": 8605.19}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Media Player", "PRODUCTSUBCATEGORY": "Streaming", "verbs": [{"COSGS_US": 3978.00, "REVENUE_US": 6793.09}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Stereo Systems", "PRODUCTSUBCATEGORY": "Home Theater Systems", "verbs": [{"COSGS_US": 104442.80, "REVENUE_US": 157117.68}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Stereo Systems", "PRODUCTSUBCATEGORY": "Receivers", "verbs": [{"COSGS_US": 69100.00, "REVENUE_US": 97280.26}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Stereo Systems", "PRODUCTSUBCATEGORY": "Speaker Kits", "verbs": [{"COSGS_US": 114900.00, "REVENUE_US": 190667.94}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Stereo Systems", "PRODUCTSUBCATEGORY": "iPod Docking Station", "verbs": [{"COSGS_US": 43842.08, "REVENUE_US": 69594.47}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Televisions", "PRODUCTSUBCATEGORY": "Flat Panel TV", "verbs": [{"COSGS_US": 100430.00, "REVENUE_US": 123923.37}]
        },
        {
            "BUSINESS_REGION": "EMEA", "PRODUCTCATEGORY": "Video Production", "PRODUCTSUBCATEGORY": "Video Editing", "verbs": [{"COSGS_US": 78722.00, "REVENUE_US": 112414.72}]
        },
        {
            "BUSINESS_REGION": "North America", "PRODUCTCATEGORY": "Accessories", "PRODUCTSUBCATEGORY": "Chargers", "verbs": [{"COSGS_US": 4290.09, "REVENUE_US": 8496.82}]
        },
        {
            "BUSINESS_REGION": "North America", "PRODUCTCATEGORY": "Accessories", "PRODUCTSUBCATEGORY": "Headphones", "verbs": [{"COSGS_US": 102014.00, "REVENUE_US": 181484.64}]
        }
    ]
}
```
Chapter 3

DataMigrator Enhancements

This section describes the new features for DataMigrator.

DataMigrator represents a broad category of tools designed to facilitate and automate the extraction and integration of data. From source extraction through target load, data is transformed through the application of business rules. Once the transformation is complete, the data is loaded into table structures that have been optimized for a particular application.

For more information on any of these new features, see the DataMigrator User’s Guide.

In this chapter:

- Adapters
- Data Management Console
- Data and Process Flows
- Data Profiling
- Reporting

Adapters

The following section provides descriptions of new features for adapters.

Adapter for Salesforce.com: Extended Bulk Load Now Available

Extended Bulk Load is now available on the adapter for Salesforce.com, making it the very first non-SQL adapter with this capability. The Salesforce.com adapter can be found in the ERP adapter folder list.

Test for Write Access to SFTP Server

As of Release 7.7.08, when testing a connection using [S]FTP, both reading and writing are tested and the message displayed indicates the success or failure of each.

In previous releases, when configuring a connection to an [S]FTP server for fixed or delimited flat files, the Test option would only test reading from the remote server.
However, these adapters could be used to write to the remote server, either independently or as an auxiliary connection for an adapter, such as Impala or Drill, that required a file to be transferred to the server where they were running.

In these cases, even though the test successfully passed, the use of the adapter could still fail. An example of this is if the user did not have write access to the specified directory.

**Extended Bulk Load for SQL Adapters: Auxiliary Connection Updates**

Using Extended Bulk Load with selected adapters requires that intermediate data files are transferred using [S]FTP to a host server. These adapters include:

- Apache Hive
- Cloudera Impala
- Jethro
- EXASol

In previous releases, this was done by adding a connection to the Flat File adapter and was limited to a single connection per server.

As of this release, a new Configure Bulk Load option is available, and allows you to configure an additional connection for the combination of server and adapter with a unique name. This option can be accessed by right-clicking a connection for a configured data adapter.

**Creating Data Fields as TIMESTAMP for Jethro and Impala**

In previous releases, attempting to create a table in Jethro or Impala from a synonym using the Recreate DBMS Table (or CREATE FILE) option in the DMC with fields described as YYMD or Date would fail with a syntax error. This was due to the fact that a data type would not be specified.

As of this release, such fields are created as TIMESTAMP and an error does not occur.

**Note:** Neither Jethro or Impala have a DATE datatype.

**Pivot Support for JSON Documents**

Both the Data Management Console and Web Console provide a capability to Pivot a database column or field containing multiple values, such as a delimited list or a JSON document. This creates a new segment in the synonym that describes the internal structure of the field.

This option is now available for JSON documents. In previous releases, if the source synonym was for a JSON document itself, this option was not available.
Data Management Console

The following section provides descriptions of new features for the Data Management Console (DMC).

Modern Design of the Data Management Console

The Data Management Console has been modernized with the use of Google material design icons and the Awesome font.

Storing Images on Filesystem From a Database Table

In previous releases, when using DataMigrator to copy data from a database table with a field described as BLOB for a column that contains a picture to a Delimited Flat File target, the name of a temporary file was written to that file.

Now, in the location specified for the target file, a like-named sub-directory is created that contains a file with each image from the source table with the appropriate extension (file type). This is currently available for the following file formats: .gif, .jpeg, .png, or .svg.

Cluster Manager Available for DataMigrator Scheduler

DataMigrator has been updated to include the Cluster Manager (CLM), a scalable workload manager which enables distribution of the requests across servers. The addition of CLM to DataMigrator, which automatically switches to other application servers in the event that the main server is unavailable, provides a sense of security to businesses by ensuring that essential functions can continue regardless of a service disruption.

Data and Process Flows

The following section provides descriptions of new features for data and process flows.

Specifying Connection for Formatted File Targets

In previous releases, using DataMigrator with a Formatted File target adapter would fail if the Flat File adapter had a connection for an [S]FTP server. In addition, by default, an application directory name was specified for a synonym, which was not needed.

Now, when a data flow uses a new target with a Formatted File adapter, a new Connection box appears. This option enables you to specify a configured connection. The default value is <local>. 
Data Profiling

The following section provides descriptions of new features for data profiling.

Data Profiling Enhancement - Forecast and Distribution Charts

Data Profiling provides data characteristics for the columns in a synonym. This release introduces two new options: Forecast Chart and Distribution Chart.

The Forecast Chart allows data profiling on numeric fields, and displays the values used for both the actual data values and forecast values in a bar chart. For the selected numeric field, you can specify the field to sort by, the number of predictions you wish to see, and the forecast type. The Forecast Chart is available for numeric fields only.

The Distribution Chart feature displays the most frequent values, along with a report that shows the basic data profiling statistics for the field.

Reporting

The following section provides descriptions of new features for reporting.

New Dependencies Analysis Column

The Dependencies Analysis report for a DataMigrator flow or synonym shows all of the synonyms and flows that it references.

A new column in the report shows the last modified date for each dependent file.
This section describes the new features for Resource Analyzer.

Resource Analyzer provides Information Systems (IS) organizations with the ability to manage the growing volume and unpredictable nature of ad hoc data access.

Resource Governor controls monitoring, system configuration parameters, and governing rules. It provides preemptive governing for requests issued to both relational and non-relational data sources.

Together, Resource Analyzer and its partner product, Resource Governor, are designed specifically to help IS organizations analyze and control end user data access.

In this chapter:

- Monitoring at the Data Service Level

Monitoring at the Data Service Level

Resource Management now provides the administrator the ability to enable and disable monitoring at the Data Service level.

Every Data Service defined in Workspace will be displayed in the Resource Management tree with a single right-click menu item that allows you to view properties. The Properties pane gives you the option to set Monitor status for the Data Service to either On or Off. By default, the setting will be On so that all Data Services are monitored.
WebFOCUS is a complete information control system with comprehensive features for retrieving and analyzing data. It enables you to create reports quickly and easily. It also provides facilities for creating highly complex reports, but its strength lies in the simplicity of the request language. You can begin with simple queries and progress to complex reports as you learn about additional facilities.

**In this chapter:**
- New Punctuation Option for Numbers Using SET CDN
- Two-Part Names Support in ? DEFINE
- Stacking Duplicate Columns in Multi-Verb Requests Based on AS Names
- Enhancements to Error Messages
- DTIME: Extracting Time Components From a Date-Time Value
- EDAPRINT: Inserting a Custom Message in the EDAPRINT Log File
- Using FORECAST in a COMPUTE Command
- PATTERNS: Returning a Pattern That Represents the Structure of the Input String
- Simplified Statistical Functions
- Displaying a Caret Symbol (<) in Heading Objects
- Displaying Syntax Components in Heading Objects
- Representing a Null String
- Enhancements to the BYDISPLAY Parameter
- Enhancement to SET COUNTWIDTH
- Change to Default Value for SET MISSINGTEST
- Setting MISSING ON Behavior for DEFINE and COMPUTE
- Enhancement to the SUMPREFIX Parameter
- Enhancements to Sort Performance
- STRING Data Type
- Accordion By Row Enhanced Interface
- OpenType Fonts Embedded in PDF Output Files
New Punctuation Option for Numbers Using SET CDN

The CDN parameter has a new option, SPACES_DOT or SPACEP, that displays a space as the thousands separator and a period as the decimal separator.

Syntax: How to Specify Punctuation for Numbers

```
SET CDN = option
```

where:

```
option
```

Is one of the following.

- **DOTS_COMMA** or **ON** sets the decimal separator as a comma and the thousands separator as a period.
- **COMMAS_DOT** or **OFF** sets the decimal separator as a period and the thousands separator as a comma.
- **SPACES_COMMA** or **SPACE** sets the decimal separator as a comma, and the thousands separator as a space.
- **SPACES_DOT** or **SPACEP** sets the decimal separator as a period, and the thousands separator as a space.
- **QUOTES_COMMA** or **QUOTE** sets the decimal point as a comma and the thousands separator as an apostrophe.
- **QUOTES_DOT** or **QUOTEP** sets the decimal point as a period and the thousands separator as an apostrophe.

Example: Displaying a Space as the Thousands Separator and a Period as the Decimal Separator

The following request uses the SPACEP option for the CDN parameter to display a space as the thousands separator and a period as the decimal separator.

```
TABLE FILE WF_RETAIL_LITE
SUM COGS_US GROSS_PROFIT_US
BY PRODUCT_CATEGORY
ON TABLE SET CDN SPACEP
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF, $
ENDSTYLE
END
```
The output is shown in the following image.

<table>
<thead>
<tr>
<th>Product</th>
<th>Cost of Goods</th>
<th>Gross Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessories</td>
<td>$342,877.00</td>
<td>$156,674.40</td>
</tr>
<tr>
<td>Camcorder</td>
<td>$453,205.00</td>
<td>$214,049.48</td>
</tr>
<tr>
<td>Computers</td>
<td>$109,281.00</td>
<td>$70,480.46</td>
</tr>
<tr>
<td>Media Player</td>
<td>$779,593.00</td>
<td>$230,197.56</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>$857,042.00</td>
<td>$359,351.89</td>
</tr>
<tr>
<td>Televisions</td>
<td>$227,820.00</td>
<td>$58,340.68</td>
</tr>
<tr>
<td>Video Production</td>
<td>$180,540.00</td>
<td>$78,639.84</td>
</tr>
</tbody>
</table>

**Two-Part Names Support in ? DEFINE**

If a DEFINE command is issued using a two-part name (appname/filename), the output from the ? DEFINE query command will show the two-part name.

**Example:** Returning a Two-Part Name From the ? DEFINE Command

The following DEFINE command is issued using a two-part name.

```
DEFINE FILE ibisamp/GGSALES
DIFF = DOLLARS-BUDDOLLARS;
END
```

Issuing the ? DEFINE ibisamp/GGSALES command returns the following output.

<table>
<thead>
<tr>
<th>FILE NAME TYPE</th>
<th>FIELD NAME</th>
<th>FORMAT</th>
<th>SEGMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBISAMP/GGSALES</td>
<td>DIFF</td>
<td>D12.2</td>
<td>1</td>
</tr>
</tbody>
</table>

**Stacking Duplicate Columns in Multi-Verb Requests Based on AS Names**

You can use the SET DUPLICATECOL command to reformat report requests that use multiple display commands, placing aggregated fields in the same column above the displayed field.

By default, each new display command in a request generates additional sort field and display field columns. With DUPLICATECOL set to OFF, each field occupies only one column in the request, with the values from each display command stacked under the values for the previous display command.
In prior releases, the duplicate columns were matched based on field names. Now, fields can also be matched based on AS names. An AS name will not be matched to a field name. When a field has an AS name, it will only be matched to other fields that have the same AS name.

**Example:** Stacking Duplicate Columns in Multi-Verb Requests Based on AS Names

The following request has three display commands. The first sums the CURR_SAL field. The second sums the SALARY field by department. The third prints the GROSS field by department and last name. Each field is assigned the same AS name, even the CURR_SAL field.

```
TABLE FILE EMPLOYEE
SUM CURR_SAL AS CURR_SAL ED_HRS
SUM SALARY AS CURR_SAL ED_HRS BY DEPARTMENT AS 'DEPT'
PRINT FIRST_NAME GROSS AS CURR_SAL ED_HRS BY DEPARTMENT BY LAST_NAME
ON TABLE SET DUPLICATECOL OFF
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF, SIZE=10, $
VERBSET=1, COLOR=RED,$
VERBSET=2, COLOR=BLUE,$
VERBSET=3,COLOR=BLACK,$
ENDSTYLE
END
```
The partial output is shown in the following image.

<table>
<thead>
<tr>
<th>DEPT</th>
<th>LAST_NAME</th>
<th>FIRST_NAME</th>
<th>CURR_SAL</th>
<th>ED_HRS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIS</td>
<td>BLACKWOOD</td>
<td>ROSEMARIE</td>
<td>$1,815.00</td>
<td>75.00</td>
</tr>
<tr>
<td>MIS</td>
<td>ROSEMARIE</td>
<td>ROSEMARIE</td>
<td>$1,815.00</td>
<td>75.00</td>
</tr>
<tr>
<td>MIS</td>
<td>ROSEMARIE</td>
<td>ROSEMARIE</td>
<td>$1,815.00</td>
<td>75.00</td>
</tr>
<tr>
<td>MIS</td>
<td>ROSEMARIE</td>
<td>ROSEMARIE</td>
<td>$1,815.00</td>
<td>75.00</td>
</tr>
<tr>
<td>MIS</td>
<td>ROSEMARIE</td>
<td>ROSEMARIE</td>
<td>$1,815.00</td>
<td>75.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,255.00</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,255.00</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,255.00</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,255.00</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,147.75</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,147.75</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,147.75</td>
<td>45.00</td>
</tr>
<tr>
<td>CROSS</td>
<td>BARBARA</td>
<td>BARBARA</td>
<td>$2,147.75</td>
<td>45.00</td>
</tr>
<tr>
<td>GREENSPAN</td>
<td>MARY</td>
<td>MARY</td>
<td>$750.00</td>
<td>25.00</td>
</tr>
<tr>
<td>GREENSPAN</td>
<td>MARY</td>
<td>MARY</td>
<td>$750.00</td>
<td>25.00</td>
</tr>
<tr>
<td>GREENSPAN</td>
<td>MARY</td>
<td>MARY</td>
<td>$750.00</td>
<td>25.00</td>
</tr>
<tr>
<td>GREENSPAN</td>
<td>MARY</td>
<td>MARY</td>
<td>$720.84</td>
<td>25.00</td>
</tr>
<tr>
<td>JONES</td>
<td>DIANE</td>
<td>DIANE</td>
<td>$1,540.00</td>
<td>50.00</td>
</tr>
<tr>
<td>JONES</td>
<td>DIANE</td>
<td>DIANE</td>
<td>$1,540.00</td>
<td>50.00</td>
</tr>
<tr>
<td>JONES</td>
<td>DIANE</td>
<td>DIANE</td>
<td>$1,540.00</td>
<td>50.00</td>
</tr>
<tr>
<td>JONES</td>
<td>DIANE</td>
<td>DIANE</td>
<td>$1,479.50</td>
<td>50.00</td>
</tr>
<tr>
<td>MCCOY</td>
<td>JOHN</td>
<td>JOHN</td>
<td>$1,540.00</td>
<td>.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
<tr>
<td>SMITH</td>
<td>MARY</td>
<td>MARY</td>
<td>$1,100.00</td>
<td>36.00</td>
</tr>
</tbody>
</table>
Enhancements to Error Messages

Many error messages have been simplified and unnecessary messages have been eliminated.

Two lines have been added to error messages related to parsing a Master File. The line that triggered the message is displayed below a line with a pointer to the position of the error. The pointer consists of a series of dashes and a down arrow (-----v) that identifies the location of the error in the line.

For example, in the WF_RETAIL_LITE Master File, the DEFINE command for SHIPMENT_UNITY has been edited to have an extra equal sign (=).

FILENAME=WF_RETAIL_LITE, TRANS_FILE=_EDAHOME/NLS/dt,
REMARKS='Lite Cluster Join of Fact Tables Sales and Shipments for Demo Database', BV_NAMESPACE=OFF,$
SEGMENT=WF_RETAIL_SALES, PARENT=., CRFILE=retail8203/facts/
wf_retail_sales, CRINCLUDE=ALL,
DESCRIPTION='Sales Fact','$
DEFINE SALE_UNITY/I9C WITH ID_SALES=1;
TITLE='Sale Unit(s)', DESCRIPTION='Unity Value (1) for each Sales Fact','$
SEGMENT=WF_RETAIL_SHIPMENTS, PARENT=., CRFILE=retail8203/facts/
wf_retail_shipments, CRINCLUDE=ALL,
DESCRIPTION='Shipments Fact',$
DEFINE = SHIPMENT_UNITY/I9C WITH ID_SHIPFACT=1;
TITLE='Shipments Unit(s)', DESCRIPTION='Unity Value (1) for each Shipment Fact',$

Running a request against this Master File produces the following messages.

(FOC03601) ERROR AT OR NEAR LINE 10 IN APP01/WF_RETAIL_LITE
(FOC1822) Invalid symbol in MFD parser: >=<

--------v

DEFINE = SHIPMENT_UNITY/I9C WITH ID_SHIPFACT=1;

DTIME: Extracting Time Components From a Date-Time Value

Given a date-time value and time component keyword as input, DTIME returns the value of all of the time components up to and including the requested component. The remaining time components in the value are set to zero. The field to which the time component is returned must have a time format that supports the component being returned.

Syntax: How to Extract a Time Component From a Date-Time Value

DTIME(datetime, component)
where:

datetime
Date-time

Is the date-time value from which to extract the time component. It can be a field name or a date-time literal.

component
Keyword

Valid values are:

- TIME. The complete time portion is returned. Its smallest component depends on the input date-time format. Nanoseconds are not supported or returned.
- HOUR. The time component up to and including the hour component is extracted.
- MINUTE. The time component up to and including the minute component is extracted.
- SECOND. The time component up to and including the second component is extracted.
- MILLISECOND. The time component up to and including the millisecond component is extracted.
- MICROSECOND. The time component up to and including the microsecond component is extracted.

**Example:** Extracting Time Components

The following request defines two date-time fields:

- TRANSTIME contains the extracted time components from TRANSDATE down to the minute.
- TRANSTIME2 extracts all of the time components from the literal date-time value 2018/01/17 05:45:22.777888.
DEFINE FILE VIDEOTR2
TRANSTIME/HHISsm = DTIME(TRANSDATE, MINUTE);
TRANSTIME2/HHISsm = DTIME(DT(2018/01/17 05:45:22.777888), TIME);
END
TABLE FILE VIDEOTR2
SUM TRANSTIME TRANSTIME2
BY MOVIECODE
BY TRANSDATE
WHERE MOVIECODE CONTAINS 'MGM'
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END

The output is shown in the following image.

<table>
<thead>
<tr>
<th>MOVIECODE</th>
<th>TRANSDATE</th>
<th>TRANSTIME</th>
<th>TRANSTIME2</th>
</tr>
</thead>
<tbody>
<tr>
<td>145MGM</td>
<td>1999/11/06</td>
<td>02:12:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td>243MGM</td>
<td>1991/06/19</td>
<td>04:11:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td>259MGM</td>
<td>1991/06/19</td>
<td>07:18:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td>284MGM</td>
<td>1999/06/18</td>
<td>03:30:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td>505MGM</td>
<td>1996/06/21</td>
<td>01:16:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td>518MGM</td>
<td>1991/06/24</td>
<td>04:43:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td></td>
<td>1998/10/03</td>
<td>02:41:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td></td>
<td>1999/11/18</td>
<td>10:27:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td>688MGM</td>
<td>1998/03/19</td>
<td>07:23:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td></td>
<td>1999/04/22</td>
<td>06:19:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td></td>
<td>1999/10/22</td>
<td>06:25:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td></td>
<td>1999/10/30</td>
<td>06:29:00.000000</td>
<td>05:45:22.777888</td>
</tr>
<tr>
<td></td>
<td>1999/11/19</td>
<td>10:26:00.000000</td>
<td>05:45:22.777888</td>
</tr>
</tbody>
</table>

EDAPRINT: Inserting a Custom Message in the EDAPRINT Log File

The EDAPRINT function enables you to add a text message into the EDAPRINT log file and assign it a message type. The returned value of the function is zero (0).

Syntax: How to Insert a Message in the EDAPRINT Log File

EDAPRINT(message_type, 'message')
where:

**message_type**

Keyword

Can be one of the following message types.

- **I.** Informational message.
- **W.** Warning message.
- **E.** Error message.

'**message**'

Is the message to insert, enclosed in single quotation marks.

**Example:** Inserting a Custom Message in the EDAPRINT Log File

The following procedure inserts three messages in the EDAPRINT log file.

- `SET &I = EDAPRINT(I, 'This is a test informational message');`
- `SET &W = EDAPRINT(W, 'This is a test warning message');`
- `SET &E = EDAPRINT(E, 'This is a test error message');`

The output is shown in the following image.

**Using FORECAST in a COMPUTE Command**

The FORECAST feature in prior releases was implemented for use in a RECAP command. However, the use of RECAP imposed limitations on placement of the FORECAST field in the output and use of sort fields.

Implementing FORECAST in a COMPUTE command eliminates these limitations and enables you to place the FORECAST calculation in a Master File. For the COMPUTE version of FORECAST, each type of calculation has its own version of the FORECAST function.
Calculating Trends and Predicting Values With FORECAST

You can calculate trends in numeric data and predict values beyond the range of those stored in the data source by using the FORECAST feature. FORECAST can be used in a report or graph request.

The calculations you can make to identify trends and forecast values are:

- **Simple moving average (FORECAST_MOVAVE).** Calculates a series of arithmetic means using a specified number of values from a field. For details, see Using a Simple Moving Average on page 110.

- **Exponential moving average.** Calculates a weighted average between the previously calculated value of the average and the next data point. There are three methods for using an exponential moving average:
  - **Single exponential smoothing (FORECAST_EXPAVE).** Calculates an average that allows you to choose weights to apply to newer and older values. For details, see Using Single Exponential Smoothing on page 114.
  - **Double exponential smoothing (FORECAST_DOUBLEXP).** Accounts for the tendency of data to either increase or decrease over time without repeating. For details, see Using Double Exponential Smoothing on page 117.
  - **Triple exponential smoothing (FORECAST_SEASONAL).** Accounts for the tendency of data to repeat itself in intervals over time. For details, see Using Triple Exponential Smoothing on page 118.

- **Linear regression analysis (FORECAST_LINEAR).** Derives the coefficients of a straight line that best fits the data points and uses this linear equation to estimate values. For details, see Using a Linear Regression Equation on page 121.

When predicting values in addition to calculating trends, FORECAST continues the same calculations beyond the data points by using the generated trend values as new data points. For the linear regression technique, the calculated regression equation is used to derive trend and predicted values.

FORECAST performs the calculations based on the data provided, but decisions about their use and reliability are the responsibility of the user. Therefore, the user is responsible for determining the reliability of the FORECAST predictions, based on the many factors that determine how accurate a prediction will be.
FORECAST Processing

You invoke FORECAST processing by including one of the FORECAST functions in a COMPUTE command. FORECAST performs the specified calculation for all the existing data points and then continues them to generate the number of predicted values that you request. The parameters needed for FORECAST include the field to use in the calculations, the number of predictions to generate, and whether to display the input field values or the calculated values on the report output for the rows that represent existing data points.

FORECAST operates on the lowest sort field in the request. This is either the last ACROSS field in the request or, if the request does not contain an ACROSS field, it is the last BY field. The FORECAST calculations start over when the highest-level sort field changes its value. In a request with multiple display commands, FORECAST operates on the last ACROSS field (or if there are no ACROSS fields, the last BY field) of the last display command. When using an ACROSS field with FORECAST, the display command must be SUM or COUNT.

**Syntax:**

**How to Calculate Trends and Predict Values**

FORECAST_MOVAVE calculation

```plaintext
COMPUTE result_field[/fmt] = FORECAST_MOVAVE(display, infield, interval, npredict, npoint1);
```

FORECAST_EXPAVE calculation

```plaintext
COMPUTE result_field[/fmt] = FORECAST_EXPAVE(display, infield, interval, npredict, npoint1);
```

FORECAST_DOUBLEXP calculation

```plaintext
COMPUTE result_field[/fmt] = FORECAST_DOUBLEXP(display, infield, interval, npredict, npoint1, npoint2);
```

FORECAST_SEASONAL calculation

```plaintext
COMPUTE result_field[/fmt] = FORECAST_SEASONAL(display, infield, interval, npredict, nperiod, npoint1, npoint2, npoint3);
```
FORECAST_LINEAR calculation

```
COMPUTE result_field[/fmt] = FORECAST_LINEAR(display, infield, interval, npredict);
```

where:

- `result_field` is the field containing the result of FORECAST. It can be a new field, or the same as `infield`. This must be a numeric field; either a real field, a virtual field, or a calculated field.

- `fmt` is the display format for `result_field`. The default format is D12.2. If `result_field` was previously reformatted using a DEFINE or COMPUTE command, the format specified in the COMPUTE command is respected.

- `display` keyword

  Specifies which values to display for rows of output that represent existing data. Valid values are:

  - **INPUT_FIELD.** This displays the original field values for rows that represent existing data.
  
  - **MODEL_DATA.** This displays the calculated values for rows that represent existing data.

  **Note:** You can show both types of output for any field by creating two independent COMPUTE commands in the same request, each with a different display option.

- `infield` is any numeric field. It can be the same field as `result_field`, or a different field. It cannot be a date-time field or a numeric field with date display options.

- `interval` is the increment to add to each sort field value (after the last data point) to create the next value. This must be a positive integer. To sort in descending order, use the BY HIGHEST phrase. The result of adding this number to the sort field values is converted to the same format as the sort field.

  For date fields, the minimal component in the format determines how the number is interpreted. For example, if the format is YMD, MDY, or DMY, an interval value of 2 is interpreted as meaning two days; if the format is YM, the 2 is interpreted as meaning two months.

- `npredict` is the number of predictions for FORECAST to calculate. It must be an integer greater than or equal to zero. Zero indicates that you do not want predictions, and is only
supported with a non-recursive FORECAST. For the SEASONAL method, npredict is the number of *periods* to calculate. The number of *points* generated is:

\[ n_{\text{period}} \times n_{\text{predict}} \]

\[ n_{\text{period}} \]
For the SEASONAL method, is a positive whole number that specifies the number of data points in a period.

\[ n_{\text{point1}} \]
Is the number of values to average for the MOVAVE method. For EXPAVE, DOUBLEEXP, and SEASONAL, this number is used to calculate the weights for each component in the average. This value must be a positive whole number. The weight, \( k \), is calculated by the following formula:

\[ k = \frac{2}{1 + n_{\text{point1}}} \]

\[ n_{\text{point2}} \]
For DOUBLEEXP and SEASONAL, this positive whole number is used to calculate the weights for each term in the trend. The weight, \( g \), is calculated by the following formula:

\[ g = \frac{2}{1 + n_{\text{point2}}} \]

\[ n_{\text{point3}} \]
For SEASONAL, this positive whole number is used to calculate the weights for each term in the seasonal adjustment. The weight, \( p \), is calculated by the following formula:

\[ p = \frac{2}{1 + n_{\text{point3}}} \]

**Reference:** Usage Notes for FORECAST

- The sort field used for FORECAST must be in a numeric or date format.

- When using simple moving average and exponential moving average methods, data values should be spaced evenly in order to get meaningful results.

- The use of column notation is not supported in the FORECAST expression. Column notation continues to be supported as before outside of this expression. The process of generating the FORECAST values creates extra columns that are not printed in the report output. The number and placement of these additional columns varies depending on the specific request.

- Missing values may lead to unexpected or unusable results and are not recommended for use with FORECAST_LINEAR.
If you use the ESTRECORDS parameter to enable the external sort to better estimate the amount of sort work space needed, you must take into account that FORECAST with predictions creates additional records in the output.

In a styled report, you can assign specific attributes to values predicted by FORECAST with the StyleSheet attribute WHEN=FORECAST. For example, to make the predicted values display with the color red, use the following syntax in the TABLE request:

```plaintext
ON TABLE SET STYLE *  
  TYPE=DATA, COLUMN=MYFORECASTSORTFIELD, WHEN=FORECAST, COLOR=RED, $  
ENDSTYLE
```

**Reference:** FORECAST Limits

The following are not supported with a COMPUTE command that uses FORECAST:

- BY TOTAL command.
- MORE, MATCH, FOR, and OVER phrases.

**Using a Simple Moving Average**

A simple moving average is a series of arithmetic means calculated with a specified number of values from a field. Each new mean in the series is calculated by dropping the first value used in the prior calculation, and adding the next data value to the calculation.

Simple moving averages are sometimes used to analyze trends in stock prices over time. In this scenario, the average is calculated using a specified number of periods of stock prices. A disadvantage to this indicator is that because it drops the oldest values from the calculation as it moves on, it loses its memory over time. Also, mean values are distorted by extreme highs and lows, since this method gives equal weight to each point.

Predicted values beyond the range of the data values are calculated using a moving average that treats the calculated trend values as new data points.

The first complete moving average occurs at the \(n^{th}\) data point because the calculation requires \(n\) values. This is called the lag. The moving average values for the lag rows are calculated as follows: the first value in the moving average column is equal to the first data value, the second value in the moving average column is the average of the first two data values, and so on until the \(n^{th}\) row, at which point there are enough values to calculate the moving average with the number of values specified.
**Example: Calculating a New Simple Moving Average Column**

This request defines an integer value named PERIOD to use as the independent variable for the moving average. It predicts three periods of values beyond the range of the retrieved data. The MOVAVE column on the report output shows the calculated moving average numbers for existing data points.

```
DEFINE FILE GGSALES
SDATE/YYM = DATE;
SYEAR/Y = SDATE;
SMONTH/M = SDATE;
PERIOD/I2 = SMONTH;
END
TABLE FILE GGSALES
SUM UNITS DOLLARS
COMPUTE MOVAVE/D10.1= FORECAST_MOVAVE(MODEL_DATA, DOLLARS,1,3,3);
BY CATEGORY BY PERIOD
WHERE SYEAR EQ 97 AND CATEGORY NE 'Gifts'
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```
In the report, the number of values to use in the average is 3 and there are no UNITS or DOLLARS values for the generated PERIOD values.

Each average (MOVAVE value) is computed using DOLLARS values where they exist. The calculation of the moving average begins in the following way:

- The first MOVAVE value (801,123.0) is equal to the first DOLLARS value.
- The second MOVAVE value (741,731.5) is the mean of DOLLARS values one and two: \((801,123 + 682,340) / 2\).
The third MOVAVE value (749,513.7) is the mean of DOLLARS values one through three:
\[(801,123 + 682,340 + 765,078) / 3.\]

The fourth MOVAVE value (712,897.3) is the mean of DOLLARS values two through four:
\[(682,340 + 765,078 + 691,274) / 3.\]

For predicted values beyond the supplied values, the calculated MOVAVE values are used as
new data points to continue the moving average. The predicted MOVAVE values (starting with
694,975.6 for PERIOD 13) are calculated using the previous MOVAVE values as new data
points. For example, the first predicted value (694,975.6) is the average of the data points
from periods 11 and 12 (620,264 and 762,328) and the moving average for period 12
(702,334.7). The calculation is: 694,975 = (620,264 + 762,328 + 702,334.7)/3.

Example:  
Displaying Original Field Values in a Simple Moving Average Column

This request defines an integer value named PERIOD to use as the independent variable for
the moving average. It predicts three periods of values beyond the range of the retrieved data.
It uses the keyword INPUT_FIELD as the first argument in the FORECAST parameter list. The
trend values do not display in the report. The actual data values for DOLLARS are followed by
the predicted values in the report column.

```
DEFINE FILE GGSALES
SDATE/YYM = DATE;
SYEAR/Y = SDATE;
SMONTH/M = SDATE;
PERIOD/I2 = SMONTH;
END
TABLE FILE GGSALES
SUM UNITS DOLLARS
COMPUTE MOVAVE/D10.1 = FORECAST_MOVAVE(INPUT_FIELD,DOLLARS,1,3,3);
BY CATEGORY BY PERIOD
WHERE SYEAR EQ 97 AND CATEGORY NE 'Gifts'
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```
Using Single Exponential Smoothing

The single exponential smoothing method calculates an average that allows you to choose weights to apply to newer and older values.

The following formula determines the weight given to the newest value.

\[ k = \frac{2}{1+n} \]

where:

\( k \)  
Is the newest value.

<table>
<thead>
<tr>
<th>Category</th>
<th>PERIOD</th>
<th>Unit Sales</th>
<th>Dollar Sales</th>
<th>MOVAVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td>1</td>
<td>61666</td>
<td>801123</td>
<td>801,123.0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>54870</td>
<td>682340</td>
<td>682,340.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>61608</td>
<td>765078</td>
<td>765,078.0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>57050</td>
<td>691274</td>
<td>691,274.0</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>59229</td>
<td>720444</td>
<td>720,444.0</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>58466</td>
<td>742457</td>
<td>742,457.0</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>60771</td>
<td>747253</td>
<td>747,253.0</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>54633</td>
<td>655896</td>
<td>655,896.0</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>57829</td>
<td>730317</td>
<td>730,317.0</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>57012</td>
<td>724412</td>
<td>724,412.0</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>51110</td>
<td>620264</td>
<td>620,264.0</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>58981</td>
<td>762328</td>
<td>762,328.0</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Food</td>
<td>1</td>
<td>54394</td>
<td>672727</td>
<td>672,727.0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>54894</td>
<td>699073</td>
<td>699,073.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>52713</td>
<td>642802</td>
<td>642,802.0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>58026</td>
<td>718514</td>
<td>718,514.0</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>53289</td>
<td>660740</td>
<td>660,740.0</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>58742</td>
<td>734705</td>
<td>734,705.0</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>60127</td>
<td>760586</td>
<td>760,586.0</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>55622</td>
<td>695235</td>
<td>695,235.0</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>55787</td>
<td>683340</td>
<td>683,140.0</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>57340</td>
<td>713768</td>
<td>713,768.0</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>57439</td>
<td>710138</td>
<td>710,138.0</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>57290</td>
<td>705315</td>
<td>705,315.0</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Is an integer greater than one. Increasing $n$ increases the weight assigned to the earlier observations (or data instances), as compared to the later ones.

The next calculation of the exponential moving average (EMA) value is derived by the following formula:

$$EMA = (EMA \times (1-k)) + (\text{datavalue} \times k)$$

This means that the newest value from the data source is multiplied by the factor $k$ and the current moving average is multiplied by the factor $(1-k)$. These quantities are then summed to generate the new EMA.

**Note:** When the data values are exhausted, the last data value in the sort group is used as the next data value.

**Example:**  **Calculating a Single Exponential Smoothing Column**

The following defines an integer value named PERIOD to use as the independent variable for the moving average. It predicts three periods of values beyond the range of retrieved data.

```
DEFINE FILE GGSALES
SDATE/YYM = DATE;
SYEAR/Y = SDATE;
SMONTH/M = SDATE;
PERIOD/I2 = SMONTH;
END
TABLE FILE GGSALES
SUM UNITS DOLLARS
COMPUTE EXPAVE/D10.1= FORECAST_EXPAVE(MODEL_DATA,DOLLARS,1,3,3);
BY CATEGORY BY PERIOD
WHERE SYEAR EQ 97 AND CATEGORY NE 'Gifts'
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
```
The output is shown in the following image:

<table>
<thead>
<tr>
<th>Category</th>
<th>PERIOD</th>
<th>Unit Sales</th>
<th>Dollar Sales</th>
<th>EXPAVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td>1</td>
<td>61666</td>
<td>801123</td>
<td>801,123.0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>54870</td>
<td>682340</td>
<td>741,731.5</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>61608</td>
<td>765078</td>
<td>753,404.8</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>57050</td>
<td>691274</td>
<td>722,339.4</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>59229</td>
<td>720444</td>
<td>721,391.7</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>58466</td>
<td>742457</td>
<td>731,924.3</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>60771</td>
<td>747253</td>
<td>739,588.7</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>54633</td>
<td>655896</td>
<td>697,742.3</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>57829</td>
<td>730317</td>
<td>714,029.7</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>57012</td>
<td>724412</td>
<td>719,220.8</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>51110</td>
<td>620264</td>
<td>669,742.4</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>58981</td>
<td>762328</td>
<td>716,035.2</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>739,181.6</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>750,754.8</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>756,541.4</td>
</tr>
<tr>
<td>Food</td>
<td>1</td>
<td>54394</td>
<td>672727</td>
<td>672,727.0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>54894</td>
<td>699073</td>
<td>685,900.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>52713</td>
<td>642802</td>
<td>664,351.0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>58026</td>
<td>718514</td>
<td>691,432.5</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>53289</td>
<td>660740</td>
<td>676,086.3</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>58742</td>
<td>734705</td>
<td>705,395.6</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>60127</td>
<td>760586</td>
<td>732,990.8</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>55622</td>
<td>695235</td>
<td>714,112.9</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>55787</td>
<td>683140</td>
<td>698,626.5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>57340</td>
<td>713768</td>
<td>706,197.2</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>57459</td>
<td>710138</td>
<td>708,167.6</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>57290</td>
<td>705315</td>
<td>706,741.3</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>706,028.2</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>705,671.6</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>705,493.3</td>
</tr>
</tbody>
</table>

In the report, three predicted values of EXPAVE are calculated within each value of CATEGORY. For values outside the range of the data, new PERIOD values are generated by adding the interval value (1) to the prior PERIOD value.

Each average (EXPAVE value) is computed using DOLLARS values where they exist. The calculation of the moving average begins in the following way:

- The first EXPAVE value (801,123.0) is the same as the first DOLLARS value.
- The second EXPAVE value (741,731.5) is calculated as follows. Note that because of rounding and the number of decimal places used, the value derived in this sample calculation varies slightly from the one displayed in the report output:

\[
n = 3 \quad \text{(number used to calculate weights)}
\]

\[
k = \frac{2}{1+n} = \frac{2}{4} = 0.5
\]

\[
\text{EXPAVE} = \text{EXPAVE} \times (1-k) + \text{new-DOLLARS} \times k = (801123 \times 0.5) + (682340 \times 0.50) = 400561.5 + 341170 = 741731.5
\]
The third EXPAVE value (753,404.8) is calculated as follows:

\[ \text{EXPAVE} = (\text{EXPAVE} \times (1-k)) + (\text{new-DOLLARS} \times k) = (741731.5 \times 0.5) + (765078 \times 0.50) = 370865.75 + 382539 = 753404.75 \]

Using Double Exponential Smoothing

Double exponential smoothing produces an exponential moving average that takes into account the tendency of data to either increase or decrease over time without repeating. This is accomplished by using two equations with two constants.

The first equation accounts for the current time period and is a weighted average of the current data value and the prior average, with an added component (b) that represents the trend for the previous period. The weight constant is k:

\[ \text{DOUBLEXP}(t) = k \times \text{datavalue}(t) + (1-k) \times ((\text{DOUBLEXP}(t-1) + b(t-1)) \]

The second equation is the calculated trend value, and is a weighted average of the difference between the current and previous average and the trend for the previous time period. b(t) represents the average trend. The weight constant is g:

\[ b(t) = g \times (\text{DOUBLEXP}(t)-\text{DOUBLEXP}(t-1)) + (1-g) \times (b(t-1)) \]

These two equations are solved to derive the smoothed average. The first smoothed average is set to the first data value. The first trend component is set to zero. For choosing the two constants, the best results are usually obtained by minimizing the mean-squared error (MSE) between the data values and the calculated averages. You may need to use nonlinear optimization techniques to find the optimal constants.

The equation used for forecasting beyond the data points with double exponential smoothing is

\[ \text{forecast}(t+m) = \text{DOUBLEXP}(t) + m \times b(t) \]

where:

\[ m \]

Is the number of time periods ahead for the forecast.
Example: Calculating a Double Exponential Smoothing Column

The following sums the ACTUAL_YTD field of the CENTSTMT data source by period, and calculates a single exponential and double exponential moving average. The report columns show the calculated values for existing data points.

TABLE FILE CENTSTMT
SUM ACTUAL_YTD
COMPUTE EXP/D15.1 = FORECAST_EXPAVE(MODEL_DATA,ACTUAL_YTD,1,0,3);
DOUBLEXP/D15.1 = FORECAST_DOUBLEXP(MODEL_DATA,ACTUAL_YTD,1,0,3,3);
BY PERIOD
WHERE GL_ACCOUNT LIKE '3%%'
ON TABLE SET STYLE * GRID=OFF,$
END

The output is shown in the following image:

<table>
<thead>
<tr>
<th>PERIOD</th>
<th>Actual</th>
<th>EXP</th>
<th>DOUBLEXP</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002/01</td>
<td>12,957,681</td>
<td>12,957,681.0</td>
<td>12,957,681.0</td>
</tr>
<tr>
<td>2002/02</td>
<td>25,441,971</td>
<td>19,199,826.0</td>
<td>22,439,246.3</td>
</tr>
<tr>
<td>2002/03</td>
<td>39,164,321</td>
<td>29,182,073.5</td>
<td>34,791,885.1</td>
</tr>
<tr>
<td>2002/04</td>
<td>52,733,326</td>
<td>40,957,699.8</td>
<td>48,845,816.0</td>
</tr>
<tr>
<td>2002/05</td>
<td>66,765,920</td>
<td>53,861,809.9</td>
<td>63,860,955.9</td>
</tr>
<tr>
<td>2002/06</td>
<td>80,952,492</td>
<td>67,407,150.9</td>
<td>79,188,052.9</td>
</tr>
</tbody>
</table>

Using Triple Exponential Smoothing

Triple exponential smoothing produces an exponential moving average that takes into account the tendency of data to repeat itself in intervals over time. For example, sales data that is growing and in which 25% of sales always occur during December contains both trend and seasonality. Triple exponential smoothing takes both the trend and seasonality into account by using three equations with three constants.

For triple exponential smoothing you need to know the number of data points in each time period (designated as L in the following equations). To account for the seasonality, a seasonal index is calculated. The data is divided by the prior season index and then used in calculating the smoothed average.

- The first equation accounts for the current time period, and is a weighted average of the current data value divided by the seasonal factor and the prior average adjusted for the trend for the previous period. The weight constant is k:
SEASONAL(t) = k \cdot (\text{data value}(t)/\text{I}(t-L)) + (1-k) \cdot (\text{SEASONAL}(t-1) + b(t-1))

The second equation is the calculated trend value, and is a weighted average of the difference between the current and previous average and the trend for the previous time period. \(b(t)\) represents the average trend. The weight constant is \(g\):

\[ b(t) = g \cdot (\text{SEASONAL}(t)-\text{SEASONAL}(t-1)) + (1-g) \cdot (b(t-1)) \]

The third equation is the calculated seasonal index, and is a weighted average of the current data value divided by the current average and the seasonal index for the previous season. \(I(t)\) represents the average seasonal coefficient. The weight constant is \(p\):

\[ I(t) = p \cdot (\text{data value}(t)/\text{SEASONAL}(t)) + (1 - p) \cdot I(t-L) \]

These equations are solved to derive the triple smoothed average. The first smoothed average is set to the first data value. Initial values for the seasonality factors are calculated based on the maximum number of full periods of data in the data source, while the initial trend is calculated based on two periods of data. These values are calculated with the following steps:

1. The initial trend factor is calculated by the following formula:
   \[ b(0) = (1/L) \left( \frac{(y(L+1)-y(1))}{L} + \frac{(y(L+2)-y(2))}{L} + \ldots + \frac{(y(2L)-y(L))}{L} \right) \]

2. The calculation of the initial seasonality factor is based on the average of the data values within each period, \(A(j)\) (\(1<=j<=N\)):
   \[ A(j) = \frac{\left( y((j-1)L+1) + y((j-1)L+2) + \ldots + y(jL) \right)}{L} \]

3. Then, the initial periodicity factor is given by the following formula, where \(N\) is the number of full periods available in the data, \(L\) is the number of points per period and \(n\) is a point within the period (\(1<=n<=L\)):
   \[ I(n) = \left( \frac{y(n)/A(1) + y(L+n)/A(2) + \ldots + y((N-1)L+n)/A(N)}{N} \right) \]

The three constants must be chosen carefully. The best results are usually obtained by choosing the constants to minimize the mean-squared error (MSE) between the data values and the calculated averages. Varying the values of npoint1 and npoint2 affect the results, and some values may produce a better approximation. To search for a better approximation, you may want to find values that minimize the MSE.
The equation used to forecast beyond the last data point with triple exponential smoothing is:

\[
\text{forecast}(t+m) = (\text{SEASONAL}(t) + m \times b(t)) / I(t-L+\text{MOD}(m/L))
\]

where:

\( m \)

Is the number of periods ahead for the forecast.

**Example:** Calculating a Triple Exponential Smoothing Column

In the following, the data has seasonality but no trend. Therefore, \( npoint2 \) is set high (1000) to make the trend factor negligible in the calculation:

\[
\text{TABLE FILE VIDEOTRK}
\]
\[
\text{SUM TRANSTOT}
\]
\[
\text{COMPUTE SEASONAL/D10.1 = FORECAST_SEASONAL(MODEL_DATA,TRANSTOT, 1,3,3,3,1000,1));}
\]
\[
\text{BY TRANSDATE}
\]
\[
\text{WHERE TRANSDATE NE '19910617'}
\]
\[
\text{ON TABLE SET STYLE *}
\]
\[
\text{GRID=OFF,}\$
\]
\[
\text{ENDSTYLE}
\]
\[
\text{END}
\]

In the output, \( npredict \) is 3. Therefore, three periods (nine points, \( nperiod \times npredict \)) are generated.

<table>
<thead>
<tr>
<th>TRANSDATE</th>
<th>TRANSTOT</th>
<th>SEASONAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>91/06/18</td>
<td>21.25</td>
<td>21.3</td>
</tr>
<tr>
<td>91/06/19</td>
<td>38.17</td>
<td>31.0</td>
</tr>
<tr>
<td>91/06/20</td>
<td>14.23</td>
<td>34.6</td>
</tr>
<tr>
<td>91/06/21</td>
<td>44.72</td>
<td>53.2</td>
</tr>
<tr>
<td>91/06/24</td>
<td>126.28</td>
<td>75.3</td>
</tr>
<tr>
<td>91/06/25</td>
<td>47.74</td>
<td>82.7</td>
</tr>
<tr>
<td>91/06/26</td>
<td>40.97</td>
<td>73.7</td>
</tr>
<tr>
<td>91/06/27</td>
<td>60.24</td>
<td>62.9</td>
</tr>
<tr>
<td>91/06/28</td>
<td>31.00</td>
<td>66.3</td>
</tr>
<tr>
<td>91/06/29</td>
<td></td>
<td>45.7</td>
</tr>
<tr>
<td>91/06/30</td>
<td></td>
<td>94.1</td>
</tr>
<tr>
<td>91/07/01</td>
<td></td>
<td>53.4</td>
</tr>
<tr>
<td>91/07/02</td>
<td></td>
<td>72.3</td>
</tr>
<tr>
<td>91/07/03</td>
<td></td>
<td>140.0</td>
</tr>
<tr>
<td>91/07/04</td>
<td></td>
<td>75.8</td>
</tr>
<tr>
<td>91/07/05</td>
<td></td>
<td>98.9</td>
</tr>
<tr>
<td>91/07/06</td>
<td></td>
<td>185.8</td>
</tr>
<tr>
<td>91/07/07</td>
<td></td>
<td>98.2</td>
</tr>
</tbody>
</table>
Using a Linear Regression Equation

The linear regression equation estimates values by assuming that the dependent variable (the new calculated values) and the independent variable (the sort field values) are related by a function that represents a straight line:

\[ y = mx + b \]

where:

- \( y \) is the dependent variable.
- \( x \) is the independent variable.
- \( m \) is the slope of the line.
- \( b \) is the y-intercept.

FORECAST_LINEAR uses a technique called Ordinary Least Squares to calculate values for \( m \) and \( b \) that minimize the sum of the squared differences between the data and the resulting line.

The following formulas show how \( m \) and \( b \) are calculated.

\[
m = \frac{(\sum xy - (\sum x \cdot \sum y)/n)}{(\sum x^2 - (\sum x)^2/n)}
\]

\[
b = (\sum y)/n - (m \cdot (\sum x)/n)
\]

where:

- \( n \) is the number of data points.
- \( y \) is the data values (dependent variables).
- \( x \) is the sort field values (independent variables).

Trend values, as well as predicted values, are calculated using the regression line equation.
**Example:** Calculating a New Linear Regression Field

The following request calculates a regression line using the VIDEOTRK data source of QUANTITY by TRANSDATE. The interval is one day, and three predicted values are calculated.

```
TABLE FILE VIDEOTRK
SUM QUANTITY
COMPUTE FORTOT=FORECAST_LINEAR(MODEL_DATA, QUANTITY, 1, 3);
BY TRANSDATE
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

The output is shown in the following image:

<table>
<thead>
<tr>
<th>TRANSDATE</th>
<th>QUANTITY</th>
<th>FORTOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/17/91</td>
<td>12</td>
<td>6.63</td>
</tr>
<tr>
<td>06/18/91</td>
<td>2</td>
<td>6.57</td>
</tr>
<tr>
<td>06/19/91</td>
<td>5</td>
<td>6.51</td>
</tr>
<tr>
<td>06/20/91</td>
<td>3</td>
<td>6.45</td>
</tr>
<tr>
<td>06/21/91</td>
<td>7</td>
<td>6.39</td>
</tr>
<tr>
<td>06/24/91</td>
<td>12</td>
<td>6.21</td>
</tr>
<tr>
<td>06/25/91</td>
<td>8</td>
<td>6.15</td>
</tr>
<tr>
<td>06/26/91</td>
<td>2</td>
<td>6.09</td>
</tr>
<tr>
<td>06/27/91</td>
<td>9</td>
<td>6.03</td>
</tr>
<tr>
<td>06/28/91</td>
<td>3</td>
<td>5.97</td>
</tr>
<tr>
<td>06/29/91</td>
<td></td>
<td>5.91</td>
</tr>
<tr>
<td>06/30/91</td>
<td></td>
<td>5.85</td>
</tr>
<tr>
<td>07/01/91</td>
<td></td>
<td>5.79</td>
</tr>
</tbody>
</table>

**Note:**

- Three predicted values of FORTOT are calculated. For values outside the range of the data, new TRANSDATE values are generated by adding the interval value (1) to the prior TRANSDATE value.
- There are no QUANTITY values for the generated FORTOT values.
- Each FORTOT value is computed using a regression line, calculated using all of the actual data values for QUANTITY.
TRANSDATE is the independent variable (x) and QUANTITY is the dependent variable (y). The equation is used to calculate QUANTITY FORECAST trend and predicted values.

The following version of the request charts the data values and the regression line.

```
GRAPH FILE VIDEOTRK
SUM QUANTITY
COMPUTE FORTOT=FORECAST_LINEAR(MODEL_DATA,QUANTITY,1,3);
BY TRANSDATE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET LOOKGRAPH VLINE
END
```

The output is shown in the following image.

![Output](image)

**Distinguishing Data Rows From Predicted Rows**

To make the report output easier to interpret, you can create a field that indicates whether the FORECAST value in each row is a predicted value. To do this, define a virtual field whose value is a constant other than zero. Rows in the report output that represent actual records in the data source will appear with a value that is not zero. Rows that represent predicted values will display zero. You can also propagate this field to a HOLD file.
Example: Distinguishing Data Rows From Predicted Rows

In the following example, the DATA_ROW virtual field has the value 1 for each row in the data source. It has the value zero for the predicted rows. The PREDICT field is calculated as YES for predicted rows, and NO for rows containing data. In addition, the StyleSheet attribute WHEN=FORECAST is used to display the predicted values for the FORTOT field in red.

```plaintext
DEFINE FILE VIDEOTRK
DATA_ROW/11 = 1;
END
TABLE FILE VIDEOTRK
SUM TRANSTOT DATA_ROW
COMPUTE
PREDICT/A3 = IF DATA_ROW NE 0 THEN 'NO' ELSE 'YES' ;
FORTOT/D12.2=FORECAST_LINEAR(MODEL_DATA,TRANSTOT,1,3);
BY TRANSDATE
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF, $
TYPE=DATA, COLUMN=FORTOT, WHEN=FORECAST, COLOR=RED, $
ENDSTYLE
END
```

The output is shown in the following image:

<table>
<thead>
<tr>
<th>TRANSDATE</th>
<th>TRANSTOT</th>
<th>DATA_ROW</th>
<th>PREDICT</th>
<th>FORTOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/17/91</td>
<td>57.03</td>
<td>10</td>
<td>NO</td>
<td>38.58</td>
</tr>
<tr>
<td>06/18/91</td>
<td>21.25</td>
<td>2</td>
<td>NO</td>
<td>40.32</td>
</tr>
<tr>
<td>06/19/91</td>
<td>38.17</td>
<td>4</td>
<td>NO</td>
<td>42.07</td>
</tr>
<tr>
<td>06/20/91</td>
<td>14.23</td>
<td>3</td>
<td>NO</td>
<td>43.81</td>
</tr>
<tr>
<td>06/21/91</td>
<td>44.72</td>
<td>6</td>
<td>NO</td>
<td>45.55</td>
</tr>
<tr>
<td>06/24/91</td>
<td>126.28</td>
<td>9</td>
<td>NO</td>
<td>50.78</td>
</tr>
<tr>
<td>06/25/91</td>
<td>47.74</td>
<td>7</td>
<td>NO</td>
<td>52.52</td>
</tr>
<tr>
<td>06/26/91</td>
<td>40.97</td>
<td>2</td>
<td>NO</td>
<td>54.26</td>
</tr>
<tr>
<td>06/27/91</td>
<td>60.24</td>
<td>7</td>
<td>NO</td>
<td>56.00</td>
</tr>
<tr>
<td>06/28/91</td>
<td>31.00</td>
<td>3</td>
<td>NO</td>
<td>57.74</td>
</tr>
<tr>
<td>06/29/91</td>
<td></td>
<td>0</td>
<td>YES</td>
<td>59.48</td>
</tr>
<tr>
<td>06/30/91</td>
<td></td>
<td>0</td>
<td>YES</td>
<td>61.23</td>
</tr>
<tr>
<td>07/01/91</td>
<td></td>
<td>0</td>
<td>YES</td>
<td>62.97</td>
</tr>
</tbody>
</table>
PATTERNS: Returning a Pattern That Represents the Structure of the Input String

PATTERNS returns a string that represents the structure of the input argument. The returned pattern includes the following characters:

- A is returned for any position in the input string that has an uppercase letter.
- a is returned for any position in the input string that has a lowercase letter.
- 9 is returned for any position in the input string that has a digit.

Note that special characters (for example, +/-=%) are returned exactly as they were in the input string.

The output is returned as variable length alphanumeric.

Syntax: How to Return a String That Represents the Pattern Profile of the Input Argument

\[
PATTERNS(string)
\]

where:

- string
  - Alphanumeric

  Is a string whose pattern will be returned.

Example: Returning a Pattern Representing an Input String

The following request returns patterns that represent customer addresses.

```plaintext
DEFINE FILE WF_RETAIL_LITE
Address_Pattern/A40V = PATTERNS(ADDRESS_LINE_1);
END

TABLE FILE WF_RETAIL_LITE
PRINT FST.ADDRESS_LINE_1 OVER
Address_Pattern
BY ADDRESS_LINE_1 NOPRINT SKIP-LINE
WHERE COUNTRY_NAME EQ 'United States'
WHERE CITY_NAME EQ 'Houston' OR 'Indianapolis' OR 'Chapel Hill' OR 'Bronx'
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```
Simplified Statistical Functions

The following simplified statistical functions have been added.

- CORRELATION, which returns the degree of correlation between two independent sets of data.
- KMEANS_CLUSTER, which partitions n observations into k clusters in which each observation belongs to the cluster with the nearest mean.
- STDDEV, which quantifies the amount of variation or dispersion of a set of data values.
MULTIREGRESS, which derives a linear equation that best fits the data values of one or more numeric fields, and uses this equation to create a new column in the report output.

RSERVE, which runs an R script that returns vector output.

These functions can be called in a COMPUTE command and operate on the internal matrix that is generated during TABLE request processing. The STDDEV and CORRELATION functions can also be called as a verb object in a display command.

Prior to calling a statistical function, you need to establish the size of the partition on which these functions will operate, if the request contains sort fields.

Note: It is recommended that all numbers and fields used as parameters to these functions be double-precision.

Syntax: How to Specify the Partition Size for Simplified Statistical Functions

SET PARTITION_ON = {FIRST|PENULTIMATE|TABLE}

where:

FIRST
  Uses the first (also called the major) sort field in the request to partition the values.

PENULTIMATE
  Uses the next to last sort field where the COMPUTE is evaluated to partition the values. This is the default value.

TABLE
  Uses the entire internal matrix to calculate the statistical function.

CORRELATION: Calculating the Degree of Correlation Between Two Sets of Data

The CORRELATION function calculates the correlation coefficient between two numeric fields. The function returns a numeric value between zero (0.0) and 1.0.

Reference: Calculate the Correlation Coefficient Between Two Fields

CORRELATION(field1, field2)

where:

field1
  Numeric
  Is the first set of data for the correlation.
field2
Numeric

Is the second set of data for the correlation.

**Example: Calculating a Correlation**

The following request calculates the correlation between the DOLLARS and BUDDOLLARS fields converted to double precision.

```
DEFINE FILE ibisamp/ggsales
DOLLARS/D12.2 = DOLLARS;
BUDDOLLARS/D12.2 = BUDDOLLARS;
END
TABLE FILE ibisamp/ggsales
SUM DOLLARS BUDDOLLARS
CORRELATION(DOLLARS, BUDDOLLARS)
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

The output is shown in the following image.

<table>
<thead>
<tr>
<th>DOLLARS</th>
<th>BUDDOLLARS</th>
<th>CORRELATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>46,156,290.00</td>
<td>46,220,778.00</td>
<td>.895691073</td>
</tr>
</tbody>
</table>

**KMEANS_CLUSTER: Partitioning Observations Into Clusters Based on the Nearest Mean Value**

The KMEANS_CLUSTER function partitions observations into a specified number of clusters based on the nearest mean value. The function returns the cluster number assigned to the field value passed as a parameter.

**Note:** If there are not enough points to create the number of clusters requested, the value -10 is returned for any cluster that cannot be created.

**Syntax:**

```
KMEANS_CLUSTER(number, percent, iterations, tolerance, [prefix1.]field1[, [prefix1.]field2 ...])
```
where:

**number**
- Integer
- Is number of clusters to extract.

**percent**
- Numeric
- Is the percent of training set size (the percent of the total data to use in the calculations).
  - The default value is AUTO, which uses the internal default percent.

**iterations**
- Integer
- Is the maximum number of times to recalculate using the means previously generated.
  - The default value is AUTO, which uses the internal default number of iterations.

**tolerance**
- Numeric
- Is a weight value between zero (0) and 1.0. The value AUTO uses the internal default tolerance.

**prefix1, prefix2**
- Defines an optional aggregation operator to apply to the field before using it in the calculation. Valid operators are:
  - **SUM.** which calculates the sum of the field values. SUM is the default value.
  - **CNT.** which calculates a count of the field values.
  - **AVE.** which calculates the average of the field values.
  - **MIN.** which calculates the minimum of the field values.
  - **MAX.** which calculates the maximum of the field values.
  - **FST.** which retrieves the first value of the field.
  - **LST.** which retrieves the last value of the field.

**Note:** The operators PCT., RPCT., TOT., MDN., MDE., RNK., and DST. are not supported.

**field1**
- Numeric
- Is the set of data to be analyzed.
**field2**

Numeric

Is an optional set of data to be analyzed.

**Example:** **Partitioning Data Values Into Clusters**

The following request partitions the DOLLARS field values into four clusters and displays the result as a scatter chart in which the color represents the cluster. The request uses the default values for the percent, iterations, and tolerance parameters by passing them as the value 0 (zero).

```
SET PARTITION_ON = PENULTIMATE
GRAPH FILE GGSALES
PRINT UNITS DOLLARS
COMPUTE KMEAN1/D20.2 TITLE 'K-MEANS' = KMEANS_CLUSTER(4, AUTO, AUTO, AUTO, DOLLARS);
ON GRAPH SET LOOKGRAPH SCATTER
ON GRAPH PCHOLD FORMAT JSCART
ON GRAPH SET STYLE *
INCLUD=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
   type = data, column = N2, bucket = y-axis,$
   type = data, column = N1, bucket = x-axis,$
   type = data, column = N3, bucket = color,$
   GRID = OFF,$
*GRAPH_JS_FINAL
   colorScale: {
      colorMode: 'discrete',
      colorBands: [{start: 1, stop: 1.99, color: 'red'}, {start: 2, stop: 2.99, color: 'green'},
         {start: 3, stop: 3.99, color: 'yellow'}, {start: 3.99, stop: 4, color: 'blue'}]
   }
*END
ENDSTYLE
END
```
The output is shown in the following image.

**STDDEV: Calculating the Standard Deviation for a Set of Data Values**

The STDDEV function returns a numeric value that represents the amount of dispersion in the data. The set of data can be specified as the entire population or a sample. The standard deviation is the square root of the variance, which is a measure of how observations deviate from their expected value (mean). If specified as a population, the divisor in the standard deviation calculation (also called degrees of freedom) will be the total number of data points, N. If specified as a sample, the divisor will be N-1.

If \( x_i \) is an observation, N is the number of observations, and \( \mu \) is the mean of all of the observations, the formula for calculating the standard deviation for a population is:

\[
\sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \mu)^2}
\]

To calculate the standard deviation for a sample, the mean is calculated using the sample observations, and the divisor is N-1 instead of N.

**Reference:**  *Calculate the Standard Deviation in a Set of Data*

\[
\text{STDDEV(field, sampling)}
\]
where:

- **field**: Numeric
  - Is the set of observations for the standard deviation calculation.

- **sampling**: Keyword
  - Indicates the origin of the data set. Can be one of the following values:
    - **P** Entire population.
    - **S** Sample of population.

**Example: Calculating a Standard Deviation**

The following request calculates the standard deviation of the DOLLARS field converted to double precision.

```plaintext
DEFINE FILE ibisamp/ggsales
DOLLARS/D12.2 = DOLLARS;
END
TABLE FILE ibisamp/ggsales
SUM DOLLARS STDDEV(DOLLARS,S)
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

The output is shown in the following image.

<table>
<thead>
<tr>
<th>DOLLARS</th>
<th>DOLLARS</th>
</tr>
</thead>
<tbody>
<tr>
<td>46,156,290.00</td>
<td>6,157.711080272</td>
</tr>
</tbody>
</table>

**MULTIREGRESS: Creating a Multivariate Linear Regression Column**

MULTIREGRESS derives a linear equation that best fits a set of numeric data points, and uses this equation to create a new column in the report output. The equation can be based on one or more independent variables.

The equation generated is of the following form, where $y$ is the dependent variable and $x_1$, $x_2$, and $x_3$ are the independent variables.

$$ y = a_1 x_1 + a_2 x_2 + a_3 x_3 + b $$
When there is one independent variable, the equation represents a straight line. When there are two independent variables, the equation represents a plane, and with three independent variables, it represents a hyperplane. You should use this technique when you have reason to believe that the dependent variable can be approximated by a linear combination of the independent variables.

**Syntax:** How to Create a Multivariate Linear Regression Column

```
MULTIREGRESS(input_field1, [input_field2, ...])
```

where:

```
input_field1, input_field2 ...
```

Are any number of field names to be used as the independent variables. They should be independent of each other. If an input field is non-numeric, it will be categorized to transform it to numeric values that can be used in the linear regression calculation.

**Example:** Creating a Multivariate Linear Regression Column

The following request uses the DOLLARS and BUDDOLLARS fields to generate a regression column named Estimated_Dollars.

```
GRAPH FILE GGSALES
SUM BUDUNITS UNITS BUDDOLLARS DOLLARS
COMPUTE Estimated_Dollars/F8 = MULTIREGRESS(DOLLARS, BUDDOLLARS);
BY DATE
ON GRAPH SET LOOKGRAPH LINE
ON GRAPH PCHOLD FORMAT JSCHART
ON GRAPH SET STYLE *
INCLUDE=IBFS:/FILE/IBI_HTML_DIR/ibi_themes/Warm.sty,$
type=data, column = n1, bucket = x-axis,$
type=data, column= dollars, bucket=y-axis,$
type=data, column= buddollars, bucket=y-axis,$
type=data, column= Estimated_Dollars, bucket=y-axis,$
*GRAPH_JS
"series":["series":2, "color":"orange"]
*END
ENDSTYLE
END
```
The output is shown in the following image. The orange line represents the regression equation.

**RSERVE: Running an R Script**

You can use the RSERVE function in a COMPUTE command to run an R script that returns vector output. This requires that you have a configured Adapter for Rserve.

**Syntax:**

```r
RSERVE(rserve_mf, input_field1, ...input_fieldn, output)
```

where:

- `rserve_mf` is the synonym for the R script.
- `input_field1, ...input_fieldn` are the independent variables used by the R script.
- `output` is the dependent variable returned by the R script. It must be a single column (vector) of output.
Example: Using RESERVE to Run an R Script

The R script named wine_run_model.R predicts Bordeaux wine prices based on the average growing season temperature, the amount of rain during the harvest season, the amount of rain during the winter, and the age of the wine.

Using a configured connection (named MyRserve) for the Adapter for Rserve, and a sample data file named wine_input_sample.csv, you create the following synonym for the R script, as described in Adapter for Rserve Integration on page 80.

Master File

FILENAME=WINE_RUN_MODEL, SUFFIX=RSERVE , $  
SEGMENT=INPUT_DATA, SECTYPE=S0, $  
  FIELDNAME=AGST, ALIAS=AGST, USAGE=D9.4, ACTUAL=STRING,  
  MISSING=ON,  
  TITLE='AGST', $  
  FIELDNAME=HARVESTRAIN, ALIAS=HarvestRain, USAGE=I11, ACTUAL=STRING,  
  MISSING=ON,  
  TITLE='HarvestRain', $  
  FIELDNAME=WINTERRAIN, ALIAS=WinterRain, USAGE=I11, ACTUAL=STRING,  
  MISSING=ON,  
  TITLE='WinterRain', $  
  FIELDNAME=AGE, ALIAS=Age, USAGE=I11, ACTUAL=STRING,  
  MISSING=ON,  
  TITLE='Age', $  
SEGMENT=OUTPUT_DATA, SECTYPE=U, PARENT=INPUT_DATA, $  
  FIELDNAME=PRICE, ALIAS=Price, USAGE=D18.14, ACTUAL=STRING,  
  MISSING=ON,  
  TITLE='Price', $

Access File

SENGNAME=INPUT_DATA,  
  CONNECTION=MyRserve,  
  R_SCRIPT=/prediction/wine_run_model.r,  
  R_SCRIPT_LOCATION=WFRS,  
  R_INPUT_SAMPLE_DAT=prediction/wine_input_sample.csv, $
Now that the synonym has been created for the model, the model will be used to run against the following data file named wine_forecast.csv.

<table>
<thead>
<tr>
<th>Year</th>
<th>Price</th>
<th>WinterRain</th>
<th>AGST</th>
<th>HarvestRain</th>
<th>Age</th>
<th>FrancePop</th>
</tr>
</thead>
<tbody>
<tr>
<td>1952</td>
<td>7.495</td>
<td>600</td>
<td>17.1167</td>
<td>160</td>
<td>31</td>
<td>43183.569</td>
</tr>
<tr>
<td>1953</td>
<td>8.0393</td>
<td>690</td>
<td>16.7333</td>
<td>80</td>
<td>30</td>
<td>43495.03</td>
</tr>
<tr>
<td>1955</td>
<td>7.6858</td>
<td>502</td>
<td>17.15</td>
<td>130</td>
<td>28</td>
<td>44217.857</td>
</tr>
<tr>
<td>1957</td>
<td>6.9845</td>
<td>420</td>
<td>16.1333</td>
<td>110</td>
<td>26</td>
<td>45152.252</td>
</tr>
<tr>
<td>1959</td>
<td>8.0757</td>
<td>485</td>
<td>17.4833</td>
<td>187</td>
<td>24</td>
<td>46128.638</td>
</tr>
<tr>
<td>1960</td>
<td>6.5188</td>
<td>763</td>
<td>16.4167</td>
<td>290</td>
<td>23</td>
<td>46853.995</td>
</tr>
<tr>
<td>1961</td>
<td>8.4937</td>
<td>830</td>
<td>17.3333</td>
<td>38</td>
<td>22</td>
<td>47128.005</td>
</tr>
<tr>
<td>1962</td>
<td>7.388</td>
<td>697</td>
<td>16.3</td>
<td>52</td>
<td>21</td>
<td>48088.673</td>
</tr>
<tr>
<td>1963</td>
<td>6.7127</td>
<td>608</td>
<td>15.7167</td>
<td>155</td>
<td>20</td>
<td>48798.99</td>
</tr>
<tr>
<td>1964</td>
<td>7.3094</td>
<td>402</td>
<td>17.2667</td>
<td>96</td>
<td>19</td>
<td>49356.943</td>
</tr>
<tr>
<td>1965</td>
<td>6.2518</td>
<td>602</td>
<td>15.3667</td>
<td>267</td>
<td>18</td>
<td>49801.821</td>
</tr>
<tr>
<td>1966</td>
<td>7.7443</td>
<td>819</td>
<td>16.5333</td>
<td>86</td>
<td>17</td>
<td>50254.966</td>
</tr>
<tr>
<td>1967</td>
<td>6.8398</td>
<td>714</td>
<td>16.2333</td>
<td>118</td>
<td>16</td>
<td>50650.406</td>
</tr>
<tr>
<td>1968</td>
<td>6.2435</td>
<td>610</td>
<td>16.2</td>
<td>292</td>
<td>15</td>
<td>51034.413</td>
</tr>
<tr>
<td>1969</td>
<td>6.3459</td>
<td>575</td>
<td>16.55</td>
<td>244</td>
<td>14</td>
<td>51470.276</td>
</tr>
<tr>
<td>1970</td>
<td>7.5883</td>
<td>622</td>
<td>16.6667</td>
<td>89</td>
<td>13</td>
<td>51918.389</td>
</tr>
<tr>
<td>1971</td>
<td>7.1934</td>
<td>551</td>
<td>16.7667</td>
<td>112</td>
<td>12</td>
<td>52431.647</td>
</tr>
<tr>
<td>1972</td>
<td>6.2049</td>
<td>536</td>
<td>14.9833</td>
<td>158</td>
<td>11</td>
<td>52894.183</td>
</tr>
<tr>
<td>1973</td>
<td>6.6367</td>
<td>376</td>
<td>17.0667</td>
<td>123</td>
<td>10</td>
<td>53332.805</td>
</tr>
<tr>
<td>1974</td>
<td>6.2941</td>
<td>574</td>
<td>16.3</td>
<td>184</td>
<td>9</td>
<td>53689.61</td>
</tr>
<tr>
<td>1975</td>
<td>7.292</td>
<td>572</td>
<td>16.95</td>
<td>171</td>
<td>8</td>
<td>53955.042</td>
</tr>
<tr>
<td>1976</td>
<td>7.1211</td>
<td>418</td>
<td>17.65</td>
<td>247</td>
<td>7</td>
<td>54159.049</td>
</tr>
<tr>
<td>1977</td>
<td>6.2587</td>
<td>821</td>
<td>15.5833</td>
<td>87</td>
<td>6</td>
<td>54378.362</td>
</tr>
<tr>
<td>1978</td>
<td>7.186</td>
<td>763</td>
<td>15.8167</td>
<td>51</td>
<td>5</td>
<td>54602.193</td>
</tr>
</tbody>
</table>

The data file can be any type of file that R can read. In this case it is another .csv file. This file needs a synonym in order to be used in a report request. You create the synonym for this file using the Adapter for Delimited Files.

Following is the generated Master File, wine_forecast.mas.

```plaintext
FILENAME=WINE_FORECAST, SUFFIX=DFIX, CODEPAGE=1252,
DATASET=prediction/wine_forecast.csv,
SEGMENT=WINE_FORECAST, SEGTYPE=S0,
FIELDNAME=YEAR1, ALIAS=Year, USAGE=I6, ACTUAL=A5V,
MISSING=ON, TITLE='Year',
FIELDNAME=PRICE, ALIAS=Price, USAGE=D8.4, ACTUAL=A7V,
MISSING=ON, TITLE='Price',
FIELDNAME=WINTERRAIN, ALIAS=WinterRain, USAGE=I5, ACTUAL=A3V,
MISSING=ON, TITLE='WinterRain',
FIELDNAME=AGST, ALIAS=AGST, USAGE=D9.4, ACTUAL=A8V,
MISSING=ON, TITLE='AGST',
FIELDNAME=HARVESTRAIN, ALIAS=HarvestRain, USAGE=I5, ACTUAL=A3V,
MISSING=ON, TITLE='HarvestRain',
FIELDNAME=AGE, ALIAS=Age, USAGE=I4, ACTUAL=A2V,
MISSING=ON, TITLE='Age',
FIELDNAME=FRANCEPOP, ALIAS=FrancePop, USAGE=D11.3, ACTUAL=A11V,
MISSING=ON, TITLE='FrancePop',
```
Following is the generated Access File, wine_forecast.acx.

```
SEGNAME=WINE_FORECAST, DELIMITER=',', ENCLOSURE='', HEADER=YES,
CDN=COMMAS_DOT, CONNECTION=<local>, $
```

The following request, wine_forecast_price_report.fex, uses the RSERVE built-in function to run the script and return a report.

```
-*wine_forecast_price_report.fex
TABLE FILE PREDICTION/WINE_FORECAST
PRINT
  YEAR
  WINTERRAIN
  AGST
  HARVESTRAIN
  AGE

  COMPUTE PREDICTED_PRICE/D18.2 MISSING ON ALL=
  RSERVE('prediction/wine_run_model, AGST, HARVESTRAIN, WINTERRAIN, AGE, Price'); AS 'Predicted,Price'

ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF, $
ENDSTYLE
END
```
The output is shown in the following image.

<table>
<thead>
<tr>
<th>Year</th>
<th>WinterRain</th>
<th>AGST</th>
<th>HarvestRain</th>
<th>Age</th>
<th>Predicted Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1952</td>
<td>600</td>
<td>17.1167</td>
<td>160</td>
<td>31</td>
<td>7.72</td>
</tr>
<tr>
<td>1953</td>
<td>690</td>
<td>16.7333</td>
<td>80</td>
<td>30</td>
<td>7.87</td>
</tr>
<tr>
<td>1955</td>
<td>502</td>
<td>17.1500</td>
<td>130</td>
<td>28</td>
<td>7.68</td>
</tr>
<tr>
<td>1957</td>
<td>420</td>
<td>16.1333</td>
<td>110</td>
<td>26</td>
<td>7.00</td>
</tr>
<tr>
<td>1958</td>
<td>582</td>
<td>16.4167</td>
<td>187</td>
<td>25</td>
<td>7.02</td>
</tr>
<tr>
<td>1959</td>
<td>485</td>
<td>17.4833</td>
<td>187</td>
<td>24</td>
<td>7.54</td>
</tr>
<tr>
<td>1960</td>
<td>763</td>
<td>16.4167</td>
<td>290</td>
<td>23</td>
<td>6.76</td>
</tr>
<tr>
<td>1961</td>
<td>830</td>
<td>17.3333</td>
<td>38</td>
<td>22</td>
<td>8.36</td>
</tr>
<tr>
<td>1962</td>
<td>697</td>
<td>16.3000</td>
<td>52</td>
<td>21</td>
<td>7.51</td>
</tr>
<tr>
<td>1963</td>
<td>608</td>
<td>15.7167</td>
<td>155</td>
<td>20</td>
<td>6.63</td>
</tr>
<tr>
<td>1964</td>
<td>402</td>
<td>17.2667</td>
<td>96</td>
<td>19</td>
<td>7.56</td>
</tr>
<tr>
<td>1965</td>
<td>602</td>
<td>15.3667</td>
<td>267</td>
<td>18</td>
<td>5.92</td>
</tr>
<tr>
<td>1966</td>
<td>819</td>
<td>16.5333</td>
<td>86</td>
<td>17</td>
<td>7.56</td>
</tr>
<tr>
<td>1967</td>
<td>714</td>
<td>16.2333</td>
<td>118</td>
<td>16</td>
<td>7.11</td>
</tr>
<tr>
<td>1969</td>
<td>575</td>
<td>16.5500</td>
<td>244</td>
<td>14</td>
<td>6.60</td>
</tr>
<tr>
<td>1970</td>
<td>622</td>
<td>16.6667</td>
<td>89</td>
<td>13</td>
<td>7.32</td>
</tr>
<tr>
<td>1971</td>
<td>551</td>
<td>16.7667</td>
<td>112</td>
<td>12</td>
<td>7.19</td>
</tr>
<tr>
<td>1972</td>
<td>536</td>
<td>14.9833</td>
<td>158</td>
<td>11</td>
<td>5.88</td>
</tr>
<tr>
<td>1973</td>
<td>376</td>
<td>17.0667</td>
<td>123</td>
<td>10</td>
<td>7.09</td>
</tr>
<tr>
<td>1974</td>
<td>574</td>
<td>16.3000</td>
<td>184</td>
<td>9</td>
<td>6.57</td>
</tr>
<tr>
<td>1975</td>
<td>572</td>
<td>16.9500</td>
<td>171</td>
<td>8</td>
<td>6.99</td>
</tr>
<tr>
<td>1976</td>
<td>418</td>
<td>17.6500</td>
<td>247</td>
<td>7</td>
<td>6.92</td>
</tr>
<tr>
<td>1977</td>
<td>821</td>
<td>15.5833</td>
<td>87</td>
<td>6</td>
<td>6.71</td>
</tr>
<tr>
<td>1978</td>
<td>763</td>
<td>15.8167</td>
<td>51</td>
<td>5</td>
<td>6.91</td>
</tr>
</tbody>
</table>
Displaying a Caret Symbol (<) in Heading Objects

The caret or less than symbol (<) signals that the next item in a heading object is a field name. Using it in a heading object without a field name produces an error message similar to the following, and the request is not run.

(FOC804) REFERENCED OBJECT <item> IS NOT FOUND IN MFD <master_file>

In order to display the caret symbol in a heading object, use two consecutive caret symbols (<<).

Example: Displaying a Less Than Symbol in a Heading

The following request computes the difference between REVENUE_US and COGS_US and displays those rows in which the difference is less than 100,000.

```
TABLE FILE WF_RETAIL_LITE
HEADING CENTER
" Difference <<  100,000"
" 
SUM COGS_US REVENUE_US
COMPUTE Difference/D20.2 = REVENUE_US - COGS_US;
BY PRODUCT_CATEGORY
WHERE TOTAL Difference LT 100000
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

The heading displays the text Difference < 100,000, as shown in the following image.
Displaying Syntax Components in Heading Objects

You can automatically display syntax components from your report or chart request in heading objects by adding one or more of the following attributes:

- `<REQUEST.FILTERS>`. Lists the WHERE and IF conditions in the request.
- `<REQUEST.VERB_OBJECTS>`. Lists the display fields referenced in the request.
- `<REQUEST.SORT_KEYS>`. Lists all sort fields in the request.
- `<REQUEST.BYKEYS>`. Lists all BY sort fields in the request.
- `<REQUEST.ACROSSKEYS>`. Lists all ACROSS sort fields in the request.
- `<REQUEST.VERB_OBJECTSCONTEXT>`. Lists the display command syntax used for each display field.
- `<REQUEST.SORT_KEYSCONTEXT>`. Lists all sort phrases in the request.

Note: The syntax component breaks onto multiple lines if the heading line length extends beyond the width of the report or chart container.
Example: Displaying Report Syntax Components

The following request displays all available syntax components from the report request in the heading. The spot markers (<+0>) are used to separate heading items so they can be styled separately in the StyleSheet.

```
TABLE FILE WF_RETAIL_LITE
HEADING
"Display Objects: <+0> <REQUEST.VERB_OBJECTS"
"Sort Fields: <+0> <REQUEST.SORT_KEYs"
"BY Fields: <+0> <REQUEST.BYKEYS"
"ACROSS Fields: <+0> <REQUEST.ACROSSKEYS"
"Filters: <+0> <REQUEST.FILTERS"

"Display Commands: <+0> <REQUEST.VERB_OBJECTS_CONTEXT"
"Sort Phrases: <+0> <REQUEST.SORT_KEYs_CONTEXT"

SUM COGS_US REVENUE_US
COMPUTE RATIO/D12.2=REVENUE_US/COGS_US;
BY PRODUCT_CATEGORY SUBTOTAL COGS_US
ACROSS BUSINESS_REGION ACROSS-TOTAL
WHERE BUSINESS_REGION NE 'Oceania'
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
TYPE=REPORT, COLOR = BLUE, SIZE=10, GRID=OFF,$
TYPE=HEADING, ITEM=1, OBJECT=TEXT, FONT=Courier, COLOR=BLUE, STYLE=BOLD,$
TYPE=HEADING, ITEM=2, FONT=Courier, COLOR=TEAL, STYLE=ITALIC,$
TYPE=TITLE, FONT=ARIAL, STYLE=BOLD, COLOR=NAVY,$
TYPE=ACROSSTITLE, FONT=ARIAL, STYLE=BOLD, COLOR=NAVY,$
TYPE=ACROSSVALUE, FONT=ARIAL, STYLE=ITALIC, COLOR=NAVY, SIZE=10,$
ENDSTYLE
END
```
Representing a Null String

In prior releases, two consecutive single quotation marks represented a string with one blank character. For example:

FIELD1/A1 = "";

Starting in this release, two consecutive single quotation marks will represent a null value with format A1V and an actual length of 0 (zero), when the field has MISSING ON. For example:

FIELD1/A1 MISSING ON = "";

If you want a blank character, add a blank between the single quotation marks. For example:

FIELD1/A1 = " ";
Enhancements to the BYDISPLAY Parameter

By default, a sort field value displays only on the first row or column of the set of detail rows or columns generated for that sort field value. You can control this behavior using the BYDISPLAY parameter. In prior releases, BYDISPLAY was only supported for styled output formats and only for BY sort fields. However, it now applies to all output formats and can control display of ACROSS values as well as BY values.

**Syntax:**

**How to Control Display of Sort Field Values on Report Output**

```
SET BYDISPLAY = {OFF|ON|BY|ACROSS|ALL}
```

```
ON TABLE SET BYDISPLAY {OFF|ON|BY|ACROSS|ALL}
```

where:

**OFF**

Displays a sort field value only on the first line or column of the report output for the sort group and on the first line or column of a page. OFF is the default value.

**ON or BY**

Displays the relevant BY field value on every line of report output produced. BY is a synonym for ON.

**ACROSS**

Displays the relevant ACROSS field value on every column of report output produced.

**ALL**

Displays the relevant BY field value on every line of report output and the relevant ACROSS field value on every column of report output.
**Example:** Controlling Display of Sort Field Values on Report Output

The following request generates a report on which sort field values only display when they change (BYDISPLAY OFF).

```
-SET &BYDISP = OFF;
SET BYDISPLAY = &BYDISP
TABLE FILE WF_RETAIL_LITE
HEADING CENTER
" BYDISPLAY = &BYDISP"
""
SUM QUANTITY_SOLD DAYSDELAYED
BY PRODUCT_CATEGORY
BY PRODUCT_SUBCATEG
ACROSS BUSINESS_REGION
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

The output is shown in the following image.
Changing **BYDISPLAY** to **ON** or **BY** displays **BY** field values on every row, as shown in the following image.

### BYDISPLAY = BY

<table>
<thead>
<tr>
<th>Product Category</th>
<th>Product Subcategory</th>
<th>EMEA Quantity Sold</th>
<th>Days Delayed</th>
<th>North America Quantity Sold</th>
<th>Days Delayed</th>
<th>Oceania Quantity Sold</th>
<th>Days Delayed</th>
<th>South America Quantity Sold</th>
<th>Days Delayed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessories</td>
<td>Charger</td>
<td>187</td>
<td>55</td>
<td>220</td>
<td>84</td>
<td>.</td>
<td>.</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>Accessories</td>
<td>Headphones</td>
<td>403</td>
<td>137</td>
<td>470</td>
<td>180</td>
<td>2</td>
<td>0</td>
<td>56</td>
<td>18</td>
</tr>
<tr>
<td>Accessories</td>
<td>Universal Remote Controls</td>
<td>287</td>
<td>102</td>
<td>325</td>
<td>132</td>
<td>7</td>
<td>7</td>
<td>51</td>
<td>23</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Handheld</td>
<td>466</td>
<td>200</td>
<td>527</td>
<td>209</td>
<td>4</td>
<td>1</td>
<td>94</td>
<td>20</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Professional</td>
<td>23</td>
<td>4</td>
<td>35</td>
<td>17</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Standard</td>
<td>344</td>
<td>157</td>
<td>404</td>
<td>130</td>
<td>1</td>
<td>0</td>
<td>58</td>
<td>37</td>
</tr>
<tr>
<td>Computers</td>
<td>Smartphone</td>
<td>384</td>
<td>162</td>
<td>411</td>
<td>151</td>
<td>5</td>
<td>2</td>
<td>65</td>
<td>27</td>
</tr>
<tr>
<td>Media Player</td>
<td>Blu Ray</td>
<td>1,206</td>
<td>495</td>
<td>1,395</td>
<td>518</td>
<td>15</td>
<td>11</td>
<td>203</td>
<td>78</td>
</tr>
<tr>
<td>Media Player</td>
<td>DVD Players</td>
<td>32</td>
<td>14</td>
<td>48</td>
<td>17</td>
<td>.</td>
<td>.</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>Media Player</td>
<td>Streaming</td>
<td>88</td>
<td>30</td>
<td>124</td>
<td>53</td>
<td>3</td>
<td>0</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Home Theater Systems</td>
<td>741</td>
<td>230</td>
<td>847</td>
<td>340</td>
<td>4</td>
<td>5</td>
<td>138</td>
<td>59</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Receivers</td>
<td>266</td>
<td>90</td>
<td>282</td>
<td>109</td>
<td>.</td>
<td>.</td>
<td>68</td>
<td>21</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Speaker Kits</td>
<td>451</td>
<td>172</td>
<td>476</td>
<td>190</td>
<td>.</td>
<td>.</td>
<td>95</td>
<td>33</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>iPod Docking Station</td>
<td>529</td>
<td>186</td>
<td>673</td>
<td>272</td>
<td>7</td>
<td>0</td>
<td>102</td>
<td>62</td>
</tr>
<tr>
<td>Televisions</td>
<td>Flat Panel TV</td>
<td>159</td>
<td>67</td>
<td>166</td>
<td>62</td>
<td>1</td>
<td>0</td>
<td>36</td>
<td>15</td>
</tr>
<tr>
<td>Video Production</td>
<td>Video Editing</td>
<td>381</td>
<td>139</td>
<td>445</td>
<td>150</td>
<td>3</td>
<td>1</td>
<td>58</td>
<td>27</td>
</tr>
</tbody>
</table>

Changing **BYDISPLAY** to **ACROSS** displays **ACROSS** field values over every column, as shown in the following image.

### BYDISPLAY = ACROSS

<table>
<thead>
<tr>
<th>Product Category</th>
<th>Product Subcategory</th>
<th>EMEA Quantity Sold</th>
<th>Days Delayed</th>
<th>North America Quantity Sold</th>
<th>Days Delayed</th>
<th>Oceania Quantity Sold</th>
<th>Days Delayed</th>
<th>South America Quantity Sold</th>
<th>Days Delayed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessories</td>
<td>Charger</td>
<td>187</td>
<td>55</td>
<td>220</td>
<td>84</td>
<td>.</td>
<td>.</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>Accessories</td>
<td>Headphones</td>
<td>403</td>
<td>137</td>
<td>470</td>
<td>180</td>
<td>2</td>
<td>0</td>
<td>56</td>
<td>18</td>
</tr>
<tr>
<td>Accessories</td>
<td>Universal Remote Controls</td>
<td>287</td>
<td>102</td>
<td>325</td>
<td>132</td>
<td>7</td>
<td>7</td>
<td>51</td>
<td>23</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Handheld</td>
<td>466</td>
<td>200</td>
<td>527</td>
<td>209</td>
<td>4</td>
<td>1</td>
<td>94</td>
<td>20</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Professional</td>
<td>23</td>
<td>4</td>
<td>35</td>
<td>17</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Standard</td>
<td>344</td>
<td>157</td>
<td>404</td>
<td>130</td>
<td>1</td>
<td>0</td>
<td>58</td>
<td>37</td>
</tr>
<tr>
<td>Computers</td>
<td>Smartphone</td>
<td>384</td>
<td>162</td>
<td>411</td>
<td>151</td>
<td>5</td>
<td>2</td>
<td>65</td>
<td>27</td>
</tr>
<tr>
<td>Media Player</td>
<td>Blu Ray</td>
<td>1,206</td>
<td>495</td>
<td>1,395</td>
<td>518</td>
<td>15</td>
<td>11</td>
<td>203</td>
<td>78</td>
</tr>
<tr>
<td>Media Player</td>
<td>DVD Players</td>
<td>32</td>
<td>14</td>
<td>48</td>
<td>17</td>
<td>.</td>
<td>.</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>Media Player</td>
<td>Streaming</td>
<td>88</td>
<td>30</td>
<td>124</td>
<td>53</td>
<td>3</td>
<td>0</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Home Theater Systems</td>
<td>741</td>
<td>230</td>
<td>847</td>
<td>340</td>
<td>4</td>
<td>5</td>
<td>138</td>
<td>59</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Receivers</td>
<td>266</td>
<td>90</td>
<td>282</td>
<td>109</td>
<td>.</td>
<td>.</td>
<td>68</td>
<td>21</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Speaker Kits</td>
<td>451</td>
<td>172</td>
<td>476</td>
<td>190</td>
<td>.</td>
<td>.</td>
<td>95</td>
<td>33</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>iPod Docking Station</td>
<td>529</td>
<td>186</td>
<td>673</td>
<td>272</td>
<td>7</td>
<td>0</td>
<td>102</td>
<td>62</td>
</tr>
<tr>
<td>Televisions</td>
<td>Flat Panel TV</td>
<td>159</td>
<td>67</td>
<td>166</td>
<td>62</td>
<td>1</td>
<td>0</td>
<td>36</td>
<td>15</td>
</tr>
<tr>
<td>Video Production</td>
<td>Video Editing</td>
<td>381</td>
<td>139</td>
<td>445</td>
<td>150</td>
<td>3</td>
<td>1</td>
<td>58</td>
<td>27</td>
</tr>
</tbody>
</table>
Changing BYDISPLAY to ALL displays BY field values on every row and ACROSS field values over every column, as shown in the following image.

<table>
<thead>
<tr>
<th>Product Category</th>
<th>Product Subcategory</th>
<th>EMEA Quantity Sold</th>
<th>EMEA Days Delayed</th>
<th>North America Quantity Sold</th>
<th>North America Days Delayed</th>
<th>Oceania Quantity Sold</th>
<th>Oceania Days Delayed</th>
<th>South America Quantity Sold</th>
<th>South America Days Delayed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessories</td>
<td>Charger</td>
<td>187</td>
<td>55</td>
<td>220</td>
<td>84</td>
<td>5</td>
<td>2</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>Accessories</td>
<td>Headphones</td>
<td>403</td>
<td>137</td>
<td>470</td>
<td>280</td>
<td>2</td>
<td>0</td>
<td>56</td>
<td>18</td>
</tr>
<tr>
<td>Accessories</td>
<td>Universal Remote Controls</td>
<td>287</td>
<td>102</td>
<td>325</td>
<td>182</td>
<td>7</td>
<td>7</td>
<td>51</td>
<td>23</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Handheld</td>
<td>466</td>
<td>200</td>
<td>527</td>
<td>209</td>
<td>4</td>
<td>1</td>
<td>94</td>
<td>20</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Professional</td>
<td>23</td>
<td>4</td>
<td>35</td>
<td>17</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>Camcorder</td>
<td>Standard</td>
<td>344</td>
<td>157</td>
<td>404</td>
<td>130</td>
<td>1</td>
<td>0</td>
<td>58</td>
<td>37</td>
</tr>
<tr>
<td>Computers</td>
<td>Smartphone</td>
<td>384</td>
<td>182</td>
<td>411</td>
<td>151</td>
<td>2</td>
<td>2</td>
<td>65</td>
<td>27</td>
</tr>
<tr>
<td>Media Player</td>
<td>Blu Ray</td>
<td>1,206</td>
<td>495</td>
<td>1,395</td>
<td>518</td>
<td>15</td>
<td>11</td>
<td>203</td>
<td>78</td>
</tr>
<tr>
<td>Media Player</td>
<td>DVD Players</td>
<td>32</td>
<td>14</td>
<td>48</td>
<td>17</td>
<td>.</td>
<td>.</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>Media Player</td>
<td>Streaming</td>
<td>88</td>
<td>30</td>
<td>124</td>
<td>53</td>
<td>3</td>
<td>0</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Home Theater Systems</td>
<td>741</td>
<td>239</td>
<td>847</td>
<td>349</td>
<td>4</td>
<td>5</td>
<td>138</td>
<td>59</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Receivers</td>
<td>266</td>
<td>90</td>
<td>282</td>
<td>109</td>
<td>.</td>
<td>.</td>
<td>68</td>
<td>21</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>Speaker Kits</td>
<td>451</td>
<td>172</td>
<td>476</td>
<td>199</td>
<td>.</td>
<td>.</td>
<td>95</td>
<td>33</td>
</tr>
<tr>
<td>Stereo Systems</td>
<td>iPod Docking Station</td>
<td>529</td>
<td>186</td>
<td>673</td>
<td>272</td>
<td>7</td>
<td>0</td>
<td>102</td>
<td>62</td>
</tr>
<tr>
<td>Televisions</td>
<td>Flat Panel TV</td>
<td>159</td>
<td>67</td>
<td>166</td>
<td>62</td>
<td>1</td>
<td>0</td>
<td>36</td>
<td>15</td>
</tr>
<tr>
<td>Video Production</td>
<td>Video Editing</td>
<td>381</td>
<td>139</td>
<td>445</td>
<td>150</td>
<td>3</td>
<td>1</td>
<td>58</td>
<td>27</td>
</tr>
</tbody>
</table>

**Enhancement to SET COUNTWIDTH**

You can now specify an integer value for the width of a COUNT column on report output. The syntax is:

```
SET COUNTWIDTH = {off|on|n}
```

where:

- **OFF**
  Makes the format of the COUNT field I5. This is the default value.

- **ON**
  Makes the format of the COUNT field I9.

- **n**
  Enables you to specify a width for the COUNT field up to the maximum integer format supported in your operating environment.

**Change to Default Value for SET MISSINGTEST**

The MISSINGTEST parameter now has three values, OLD, SPECIAL, and NEW. The value NEW has been added and is the default value.
In prior releases, by default, when an IF-THEN-ELSE expression was used to calculate a result and the IF expression evaluated to zero (for numeric expressions) or blank (for alphanumeric expressions), the left hand side was checked to see if it had MISSING ON. If it did, and only some values were needed (NEEDS SOME), the result of the IF expression was MISSING, not true or false. The outcome returned was also MISSING, not the result of evaluating the THEN or ELSE expression. The SET MISSINGTEST = NEW command eliminates the missing test for the IF expression so that either the THEN expression or the ELSE expression will be evaluated and returned as the result. This is the new default behavior.

**Syntax:**

How to Control Evaluation of MISSING in IF-THEN-ELSE Expressions

```plaintext
SET MISSINGTEST = {NEW|OLD|SPECIAL}
```

where:

**NEW**

Excludes the IF expression from the missing values evaluation so that the IF expression results in either true or false, not MISSING. If it evaluates to true, the THEN expression is used to calculate the result. If it evaluates to false, the ELSE expression is used to calculate the result. This is the default value.

**OLD**

Includes the IF expression in the missing values evaluation. If the IF expression evaluates to MISSING and the missing field only needs some missing values, the result is also MISSING.

**SPECIAL**

Is required for passing parameters to RStat.
**Example: Using SET MISSINGTEST With IF-THEN-ELSE Expressions**

The following request defines a field named MISS_FIELD that contains a missing value for the country name Austria. In the TABLE request there are two calculated values, CALC1 and CALC2 that test this field in IF-THEN-ELSE expressions. Both of these fields have MISSING ON and need only some missing values to be missing:

```
SET MISSINGTEST = OLD
DEFINE FILE wf_retail_lite
MISS_FIELD/A10 MISSING ON = IF COUNTRY_NAME NE 'Austria' THEN 'DATAEXISTS' ELSE MISSING;
END

TABLE FILE wf_retail_lite
SUM COGS_US MISS_FIELD
COMPUTE CALC1/A7 MISSING ON = IF ((MISS_FIELD EQ '') OR (MISS_FIELD EQ MISSING)) THEN 'THEN' ELSE 'ELSE';
COMPUTE CALC2/A7 MISSING ON = IF ((MISS_FIELD EQ MISSING) OR (MISS_FIELD EQ '')) THEN 'THEN' ELSE 'ELSE';
BY COUNTRY_NAME
WHERE BUSINESS_REGION EQ 'EMEA'
WHERE COUNTRY_NAME LT 'E'
ON TABLE SET NODATA 'MISSING'
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

Running the request with MISSINGTEST=OLD produces the output shown in the following image:

<table>
<thead>
<tr>
<th>Customer</th>
<th>Cost of Goods</th>
<th>MISS_FIELD</th>
<th>CALC1</th>
<th>CALC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>$426,570.00</td>
<td>MISSING</td>
<td>MISSING</td>
<td>THEN</td>
</tr>
<tr>
<td>Belgium</td>
<td>$3,800.00</td>
<td>DATAEXISTS</td>
<td>ELSE</td>
<td>ELSE</td>
</tr>
<tr>
<td>China</td>
<td>$563.00</td>
<td>DATAEXISTS</td>
<td>ELSE</td>
<td>ELSE</td>
</tr>
<tr>
<td>Denmark</td>
<td>$1,346.00</td>
<td>DATAEXISTS</td>
<td>ELSE</td>
<td>ELSE</td>
</tr>
</tbody>
</table>

Note that for Austria, MISS_FIELD is MISSING.

- In CALC1, the expression MISS_FIELD EQ ' ' is evaluated to true. MISS_FIELD IS MISSING is not evaluated at all because evaluation stops when it can be determined that the result of the expression is true. However, since the expression compared the field to blank, it is checked to see if the result field supports missing values. Since it does, the final result is MISSING.
In CALC2, the expression MISS_FIELD EQ MISSING is true. MISS_FIELD EQ ' ' is not evaluated at all because evaluation stops when it can be determined that the result of the expression is true. No missing check is needed, so the result of the IF expression is TRUE, and the THEN expression is evaluated and returned as the result.

Changing the SET command to SET MISSINGTEST=NEW and rerunning the request produces the output shown in the following image. The IF expressions in CALC1 and CALC2 both evaluate to true because neither expression is checked to see if the result field supports missing, so the THEN expression is evaluated and returned as the result in both cases.

<table>
<thead>
<tr>
<th>Customer</th>
<th>Cost of Goods</th>
<th>MISS_FIELD</th>
<th>CALC1</th>
<th>CALC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>$426,570.00</td>
<td>MISSING</td>
<td>THEN</td>
<td>THEN</td>
</tr>
<tr>
<td>Belgium</td>
<td>$3,800.00</td>
<td>DATAEXISTS</td>
<td>ELSE</td>
<td>ELSE</td>
</tr>
<tr>
<td>China</td>
<td>$563.00</td>
<td>DATAEXISTS</td>
<td>ELSE</td>
<td>ELSE</td>
</tr>
<tr>
<td>Denmark</td>
<td>$1,346.00</td>
<td>DATAEXISTS</td>
<td>ELSE</td>
<td>ELSE</td>
</tr>
</tbody>
</table>

Setting MISSING ON Behavior for DEFINE and COMPUTE

When a virtual field or calculated value can have missing values, you can specify whether all or some of the field values used in the expression that creates the DEFINE or COMPUTE field must be missing to make the result field missing. If you do not specify ALL or SOME for a DEFINE or COMPUTE with MISSING ON, the default value is SOME.

The SET parameter MISS_ON enables you to specify whether SOME or ALL should be used for MISSING ON in a DEFINE or COMPUTE that does not specify which to use.

**Syntax:**

**How to Set a Default Value for MISSING ON in DEFINE and COMPUTE**

```plaintext
SET MISS_ON = {SOME|ALL}
```

where:

- **SOME**
  
  Indicates that if at least one field in the expression has a value, the temporary field has a value (the missing values of the field are evaluated as 0 or blank in the calculation). If all of the fields in the expression are missing values, the temporary field has a missing value. SOME is the default value.
ALL

Indicates that if all the fields in the expression have values, the temporary field has a value. If at least one field in the expression has a missing value, the temporary field has a missing value.

Example: Setting a Default Value for MISSING ON in DEFINE and COMPUTE

The following request creates three virtual fields that all have MISSING ON. Field AAA has all missing values. Field BBB is missing only when the category is Gifts and has the value 100 otherwise. Field CCC is the sum of AAA and BBB.

```
SET MISS_ON = SOME
DEFINE FILE GGSALES
AAA/D20 MISSING ON = MISSING;
BBB/D20 MISSING ON = IF CATEGORY EQ 'Gifts' THEN MISSING ELSE 100;
CCC/D20 MISSING ON = AAA + BBB;
END
TABLE FILE GGSALES
SUM
AAA
BBB
CCC
BY CATEGORY
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
END
```

Running the request with SET MISS_ON=SOME (the default) shows that CCC has a value unless both AAA and BBB are missing.

<table>
<thead>
<tr>
<th>Category</th>
<th>AAA</th>
<th>BBB</th>
<th>CCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td>.</td>
<td>144,000</td>
<td>144,000</td>
</tr>
<tr>
<td>Food</td>
<td>.</td>
<td>144,000</td>
<td>144,000</td>
</tr>
<tr>
<td>Gifts</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>
Changing SET MISS_ON to ALL, produces the following output. CCC is assigned a missing value because one of the fields used to calculate it is always missing.

<table>
<thead>
<tr>
<th>Category</th>
<th>AAA</th>
<th>BBB</th>
<th>CCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td>.</td>
<td>144,000</td>
<td>.</td>
</tr>
<tr>
<td>Food</td>
<td>.</td>
<td>144,000</td>
<td>.</td>
</tr>
<tr>
<td>Gifts</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>

**Enhancement to the SUMPREFIX Parameter**

The SUMPREFIX parameter allows you to specify which value will be displayed when aggregating an alphanumeric or smart date field in the absence of any prefix operator. The default value is LST, which will return the physical last value within the sort group. FST has been available, which will return the first physical value in the sort group. The setting has been enhanced to include MIN and MAX to return either the minimum value or maximum value within the sort group.

MIN and MAX correspond to the SQL MIN and MAX functions and can be easily converted to SQL when optimizing a request against a relational database.

**Syntax:** How to Control Display of Aggregated Alphanumeric or Smart Date Fields

```
SET SUMPREFIX = {FST|LST|MIN|MAX}
```

where:

- **FST**
  Displays the first value when alphanumeric or smart date data types are aggregated.

- **LST**
  Displays the last value when alphanumeric or smart date data types are aggregated. LST is the default value.

- **MIN**
  Displays the minimum value in the sort order set by your server code page and configuration when alphanumeric or smart date data types are aggregated.

- **MAX**
  Displays the maximum value in the sort order set by your server code page and configuration when alphanumeric or smart date data types are aggregated.
Example: Displaying the Minimum Value for an Aggregated Alphanumeric Field

The following request sets SUMPREFIX to MIN and displays the aggregated PRODUCT_CATEGORY and DAYSDELAYED values as well as the minimum, maximum, first, and last PRODUCT_CATEGORY values. In each row, the aggregated PRODUCT_CATEGORY value matches the MIN.PRODUCT_CATEGORY value. The DAYSDELAYED numeric field is not affected by the SUMPREFIX value and is aggregated.

```
SET SUMPREFIX = MIN
TABLE FILE wf_retail_lite
SUM PRODUCT_CATEGORY DAYSDELAYED MIN.PRODUCT_CATEGORY MAX.PRODUCT_CATEGORY
  FST.PRODUCT_CATEGORY LST.PRODUCT_CATEGORY
BY BRAND
WHERE BRAND GT 'K' AND BRAND LT 'U'
ON TABLE SET PAGE NOLEAD
ON TABLE SET STYLE *
GRID=OFF,$
ENDSTYLE
END
```

The output is shown in the following image.

<table>
<thead>
<tr>
<th>Brand</th>
<th>Product Category</th>
<th>Days Delayed</th>
<th>MIN Product Category</th>
<th>MAX Product Category</th>
<th>FST Product Category</th>
<th>LST Product Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>LG</td>
<td>Media Player</td>
<td>339</td>
<td>Media Player</td>
<td>Televisions</td>
<td>Media Player</td>
<td>Televisions</td>
</tr>
<tr>
<td>Logitech</td>
<td>Accessories</td>
<td>114</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Accessories</td>
</tr>
<tr>
<td>Niles Audio</td>
<td>Accessories</td>
<td>150</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Accessories</td>
</tr>
<tr>
<td>Onkyo</td>
<td>Stereo Systems</td>
<td>140</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Panasonic</td>
<td>Camcorder</td>
<td>422</td>
<td>Camcorder</td>
<td>Televisions</td>
<td>Camcorder</td>
<td>Televisions</td>
</tr>
<tr>
<td>Philips</td>
<td>Stereo Systems</td>
<td>229</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Pioneer</td>
<td>Accessories</td>
<td>339</td>
<td>Stereo Systems</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Polk Audio</td>
<td>Stereo Systems</td>
<td>93</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Roku</td>
<td>Media Player</td>
<td>85</td>
<td>Media Player</td>
<td>Media Player</td>
<td>Media Player</td>
<td>Media Player</td>
</tr>
<tr>
<td>Samsung</td>
<td>Accessories</td>
<td>525</td>
<td>Stereo Systems</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Sonyo</td>
<td>Camcorder</td>
<td>298</td>
<td>Stereo Systems</td>
<td>Camcorder</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Sennheiser</td>
<td>Accessories</td>
<td>128</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Accessories</td>
<td>Accessories</td>
</tr>
<tr>
<td>Sharp</td>
<td>Media Player</td>
<td>252</td>
<td>Stereo Systems</td>
<td>Media Player</td>
<td>Stereo Systems</td>
<td>Stereo Systems</td>
</tr>
<tr>
<td>Sony</td>
<td>Accessories</td>
<td>1,100</td>
<td>Stereosystems</td>
<td>Televisions</td>
<td>Accessories</td>
<td>Televisions</td>
</tr>
<tr>
<td>Thomson Grass</td>
<td>Video</td>
<td>166</td>
<td>Video</td>
<td>Video</td>
<td>Video</td>
<td>Video</td>
</tr>
<tr>
<td>Valley</td>
<td>Production</td>
<td></td>
<td>Production</td>
<td>Production</td>
<td>Production</td>
<td>Production</td>
</tr>
<tr>
<td>Toshiba</td>
<td>Media Player</td>
<td>7</td>
<td>Media Player</td>
<td>Media Player</td>
<td>Media Player</td>
<td>Media Player</td>
</tr>
</tbody>
</table>

Enhancements to Sort Performance

The sorting procedure has been enhanced to analyze the request being processed and the amount of sort memory available in order reduce the amount of disk I/O. The sort strategy is controlled by the specifics of the request and the values of the SORTMATRIX and SORTMEMORY parameters.
**Reference:** Sort Parameters

**SORTMATRIX**

The SORTMATRIX parameter controls whether to employ in-memory sorting with decreased use of external memory. The syntax is

```sql
SET SORTMATRIX = {SMALL|LARGE}
```

where:

**SMALL**

Creates a single sort matrix of up to 2048 rows, and uses a binary search based insertion sort with aggregation during retrieval. The maximum number of rows in this matrix has been determined to provide the best performance for this type of sort. If the sort matrix becomes full, it is written to a file called FOCSORT on disk, the in-memory matrix is emptied, and retrieval continues, writing to FOCSORT as many times as necessary. When the end of data is detected, the remaining rows are written to FOCSORT and the merge routine merges all of the sort strings in FOCSORT (which, in extreme cases, may require multiple merge phases), while also completing the aggregation.

**LARGE**

Creates a large matrix or multiple small matrices in memory, when adequate memory is available as determined by the SORTMEMORY parameter. LARGE is the default value. The goal of this strategy is to do as much sorting as possible in internal memory before writing any records to disk. Whether disk I/O is necessary at all in the sorting process depends on the amount of memory allocated for sorting and the size of the request output. If the amount of SORTMEMORY is not large enough to meaningfully make use of the LARGE strategy, the sort will default to the SMALL strategy. The LARGE strategy greatly reduces the need for disk I/O and, if disk I/O is required after all (for very large output), it virtually eliminates the need for multiple merge phases.

**SORTMEMORY**

The SORTMEMORY parameter controls the amount of internal memory available for sorting. The syntax is

```sql
SET SORTMEMORY = {n|512}
```

where:

**n**

Is the positive number of megabytes of memory available for sorting. The default value is 512.
STRING Data Type

Certain relational data sources support a data type called STRING to store alphanumeric data that has an unlimited length. This type of data can be mapped to the TX data type. However, text fields have limitations on their use in WebFOCUS sort and selection phrases.

The format specification for a STRING field has no length specification. The length is determined on retrieval. For example:

FIELD1/STRING

The STRING data type has all of the functionality of alphanumeric data types in WebFOCUS. The limit to a STRING field value length is 2 GB. It can be propagated to relational data sources that have the STRING data type and to delimited HOLD files, where both the USAGE and ACTUAL formats are generated as STRING.

Accordion By Row Enhanced Interface

Accordion By Row reports are HTML reports that offer an interactive interface to data aggregated at multiple levels, by presenting the sort fields within an expandable tree. By default, the report will present the highest dimension or sort field (BY value) and the aggregated measures associated with each value. The tree control can be used to open or close each dimension and view the associated aggregated values. Clicking the plus or minus sign or the arrow next to a sort field value opens new rows that display the next lower level sort field values and subtotals. The lowest level sort field, when expanded, displays the aggregated data values.

Using the Accordion By Row enhanced interface, navigation is easier when working with wide and large reports in a portal page, the data automatically resizes to fit the size of the container, and the column widths automatically adjust based on the largest data value or column title, whichever is larger. The SET EXPANDBYROWTREE=ON command in a procedure enables the enhanced Accordion By Row feature.

For more information on Accordion By Row reports, see the Creating Reports With WebFOCUS Language technical content.

OpenType Fonts Embedded in PDF Output Files

In addition to Type 1 fonts, you can add and configure OpenType fonts to be embedded in PDF output files. OpenType fonts enable more flexibility in customizing PDF files, such as support for expanded character sets and layout features, and cross-platform compatibility.

The font extension for OpenType fonts is .OTF. You can only use OTF fonts with content in Compact Font Format (CFF).
Learn More

For more information on WebFOCUS font support, see the *Creating Reports With WebFOCUS Language* technical content.
Feedback

Customer success is our top priority. Connect with us today!

Information Builders Technical Content Management team is comprised of many talented individuals who work together to design and deliver quality technical documentation products. Your feedback supports our ongoing efforts!

You can also preview new innovations to get an early look at new content products and services. Your participation helps us create great experiences for every customer.

To send us feedback or make a connection, contact Sarah Buccellato, Technical Editor, Technical Content Management at Sarah_Buccellato@ibi.com.

To request permission to repurpose copyrighted material, please contact Frances Gambino, Vice President, Technical Content Management at Frances_Gambino@ibi.com.
WebFOCUS

Server New Features
WebFOCUS Reporting Server Release 8203
DataMigrator Server Release 7708