QAD .NET User Interface Release Notes

March 2013

These release notes include information about QAD .NET UI for QAD Enterprise Applications 2013 – Enterprise Edition.

Review this document *before* proceeding with any phase of a QAD .NET UI implementation.

These release notes include the following sections:

Installation and Configuration Information

The following summarizes installation and configuration changes for this version of the QAD .NET UI.

For information on the QAD .NET UI release history, see the *Platform and Product Availability Guide*, available from the General Reference section of the QAD support site (http://support.qad.com).

The QAD Document Library (http://www.qad.com/documentlibrary/) offers a complete set of all QAD user guides, training guides, and other materials.

Note If upgrading from a previous version, be sure to review the release notes for the versions between your current version and this version.

Cumulative Patch Information

Important Before installing the QAD .NET UI 2013, be sure to go to the QAD Store (http://store.qad.com) to check for the latest cumulative patch to QAD .NET UI 2013. You must install the patch after installing the QAD .NET UI.

Release Summary

QAD .NET UI Version: 2013 EE

Product Versions: QAD Enterprise Applications 2013 – Enterprise Edition

Microsoft .NET Framework Version: 4

Microsoft .NET Framework 4 must be installed on client machines. You can download and install it from the Microsoft Download Center (http://www.microsoft.com/en-us/download/details.aspx?id=17851).

Tomcat Versions: 5.5.*x*, 6.0.*x*, and 7.0.*x*.

Operating System: The QAD .NET UI client runs on Windows XP SP3, Windows Vista, Windows 7, and Windows 8 (Desktop Mode). The QAD .NET UI can run on 64-bit Windows, but only in 32-bit mode.

Microsoft Internet Explorer Version: 9 or higher. If running Windows XP SP3, you can use Internet Explorer 8, but see “HTML5 Process Maps and Internet Explorer” on page 5.

The Connection Manager SSH implementation uses the jSch component by JCraft, Inc., distributed in accordance with the license agreement (http://www.jcraft.com/jsch/LICENSE.txt).

Supported Languages

The user interface supports the following languages in this release:

Chinese (Simplified)	English (US)	Italian	Portuguese (Brazilian)
Chinese (Traditional)	French	Japanese	Spanish (Castilian)
Dutch	German	Polish	Spanish (Latin American)

The following languages have some support, but new terms added in this release may appear in English:

Bulgarian	Greek	Norwegian	Slovenia
Czech	Hungarian	Romanian	Swedish
Danish	Korean	Russian	Turkish
Finnish	Lithuanian	Slovak	Ukrainian

Georgia SoftWorks Windows SSH or Telnet Server Installation

The QAD .NET UI uses an SSH or telnet server for two purposes:

- On the database server, it is used to run a pool of sessions that support maintenance programs, reports, and inquiries.
- It enables the client terminal interface for programs that only run in terminal mode (see “Programs in Terminal Mode Only” on page 18) as well as any custom programs that do not conform to QAD programming standards.

QAD .NET UI 2013 now supports secure shell (SSH) as well as telnet. You can use either SSH or telnet; the default is now to use SSH (see “Connection Manager: Secure Shell (SSH) Default” on page 13). Typically, the SSH or telnet server runs on a UNIX (or Linux) machine. If you plan to use a UNIX machine for the SSH server, QAD recommends using the SSH daemon, which comes standard on all UNIX distributions. If you plan to use a UNIX machine for the telnet server, you can use the default telnet service provided with the operating system.

If you want to run the SSH or telnet server on a Windows machine rather than a UNIX (or Linux) machine, use the Georgia SoftWorks (GSW) SSH Server or Telnet Server. This software is not included on the QAD .NET UI installation media: you must download the latest version of the software to obtain the most recent patches and functionality from Georgia SoftWorks:

http://www.georgiasoftworks.com/

For the Georgia SoftWorks SSH Server, see:
http://www.georgiasoftworks.com/products/ssh2/ssh2_server.php.

For the Georgia SoftWorks Telnet Server, see:
http://www.georgiasoftworks.com/products/uts/overview.php.

Refer to the Georgia SoftWorks documentation for installation information as well as software and system sizing requirements.

The Georgia Softworks Power Features pack provides a session monitor, which is helpful for troubleshooting connection issues.

Registering the Georgia SoftWorks Software

To register the software, you provide a product ID to Georgia SoftWorks so that a serial number can be generated for your product. The serial number identifies server hardware and software components. If these components change or are upgraded, contact Georgia SoftWorks about generating a new product ID and serial number.

Important If you need to reinstall or are planning to move your installation to a different platform, or if you are a sales agent or a distributor, include that information on the registration.

1. Select the Registration icon from the program group in the Start menu.
2. In the Georgia SoftWorks Product Registration window, enter your customer information. The information that displays in the Product Information section is system-generated.
3. Set Sessions Requested to 100. This is the number QAD automatically supplies with your registration.
4. Choose Save to File to save this information, or choose Print. Then, follow the appropriate step to supply the product ID to Georgia SoftWorks:
   a. E-mail the saved registration form file to Georgia SoftWorks at:

      registration@georgiasoftworks.com

When your form is received, a serial number is generated for your product and is returned to you by e-mail.

b FAX the printed registration form to Georgia SoftWorks at 706-265-1020. When your form is received, a serial number is generated for your product and is returned to you by FAX.

When you receive your serial number, return to the Georgia SoftWorks Product Registration window and enter it in the appropriate field in the registration form. Click Register.

Tomcat Configuration

As part of the software prerequisites for installing QAD Enterprise Edition, you must install the Tomcat web server in the `$CATALINA_HOME` directory as described in the *QAD Enterprise Applications – Enterprise Edition Installation Guide*.

Note When you install Tomcat, be sure to update the `tomcat-users.xml` file in the `/conf` directory to include user, password, and role settings for the admin and pronav users. For example:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <user name="tomcat" password="tomcat" roles="tomcat" />
  <user name="role1"  password="tomcat" roles="role1"  />
  <user name="both"   password="tomcat" roles="tomcat,role1" />
  <user name="admin"  password="mfgpro" roles="qadadmin,manager,manager-gui,admin,pronav" />
  <user name="pronav" password="editor" roles="pronav" />
</tomcat-users>
```

Process Maps Installation

The process maps are delivered separately from the QAD .NET UI. The process maps for QAD Enterprise Applications 2013 – Enterprise Edition are included with the QAD 2013 EE release media and are available on the QAD Store (http://store.qad.com). The QAD Deployment Toolkit (QDT) installs the process map components (including the viewer and editor) along with the process map content as part of the overall installation process for the product. However, you should then get the latest process map content for QAD 2013 EE from the QAD Store.

Process Map Configuration Settings

QDT now installs the process map viewer, editor, and related components on the home server as a stand-alone web application named pronav (`tomcat/webapps/pronav`). The following settings in the client session configuration file (`client-session.xml`) specify the default configuration:

```
<!-- Process map settings -->
<ProcessMapBaseUrl>${DesktopProtocol}://${DesktopHost}:${DesktopPort}/pronav</ProcessMapBaseUrl>
  <qad.url.process.editor>${ProcessMapBaseUrl}/ProcessEditor.jsp</qad.url.process.editor>
  <qad.url.process.viewer>${ProcessMapBaseUrl}/ProcessViewer.jsp</qad.url.process.viewer>
```

Previously, the process viewer and editor were in `/tomcat/webapps/<environment>` and the process map content was in `/tomcat/webapps/<environment>/WEB-INF/pronav`.

Now process maps are included in an environment named pronav (`/tomcat/webapps/pronav`) by default. A benefit of this approach is that you can now define a single process map installation that can be shared across multiple environments.

Note The configuration setting for the process map images (QAD_IMG), set in Administration > Process Admin, now requires a fully qualified domain name. The settings now include a QAD_PMAP_ROOT setting to specify the URL to the process map installation (`http://server.domain.com:port/pronav/`) and then the QAD_IMG setting is `{QAD_PMAP_ROOT}images/`.

HTML5 Process Maps and Internet Explorer

The Process Viewer and Process Editor have been updated to support HTML5.

Computers running the QAD .NET UI 2013 EE client should be upgraded to use the most recent version of Internet Explorer available for the version of Windows.

For instance, if running Windows 7 or Windows Vista, you should have Internet Explorer 9, which supports HTML5.

If running Windows XP SP3, be sure to upgrade to Internet Explorer 8. (Internet Explorer 9 is not supported on Windows XP.)

Note With Internet Explorer 8, the Process Editor will not work properly when opened inside the QAD .NET UI because Internet Explorer 8 does not support HTML5.

If you want to edit process maps and are running Windows XP SP3 with Internet Explorer 8, a workaround is to install a different browser that supports HTML5 such as Chrome or Firefox, and then access the Process Editor as a stand-alone web application based on where it is installed.

You can identify the URL for the Process Editor by choosing Help | View Configuration and searching for "process" to find the URLs for both the Process Editor and Process Viewer. The Process Editor URL will typically have the format `http://server.domain.com:port/pronav/ProcessEditor.jsp`. Enter this URL in the browser that supports HTML5.

Easy On Boarding and Integrated Customization Toolkit Process Maps

Process maps now include additional maps for Easy On Boarding (EOB) and the Integrated Customization Toolkit (ICT).

Easy On Boarding, available as a QAD Services engagement, simplifies the implementation process by pre-populating most standard data and pre-configuring standard processes. Companies can adjust both data and processes later, but the process streamlines the implementation tasks. For more information about Easy On Boarding, please contact QAD Services.

The Easy On Boarding process maps are organized into industry verticals including Automotive, Life Sciences, Industrial, Electrical, Consumer Products, and Food and Beverage. Nodes on the maps include links to Easy On Boarding training, documentation, and other attachments that are provided during a QAD Services engagement.

You can only access the Easy On Boarding attachments if you have a QAD Services engagement for QAD Easy On Boarding.

Note The location of the Easy On Boarding attachments is specified by the Attachments setting on the Administration | Process Admin — Process Properties screen. If you do not have a QAD Services engagement and you try to access the attachments from a process map node, you will get a "file not found" error message. However, if you set Attachments to `ProcessViewer.jsp?ProcessName=eob_attachments&f=`, a process map displays instead that lets you know you can only access the attachments if you have a QAD Services engagement.

Internet Explorer and QAD .NET UI Client Installation

Warning The 64-bit version of Internet Explorer does not install the QAD .NET UI client, even if you are running Internet Explorer as an administrator. You must use the 32-bit version of Internet Explorer (typically located in C:\Program Files (x86)\Internet Explorer\iexplore.exe) to install the QAD .NET UI client. A situation in which this is likely to occur is when a user inadvertently creates a shortcut to the 64-bit version of Internet Explorer rather than the 32-bit version, and then uses that shortcut to launch Internet Explorer.

Reporting Framework Sample Reports

When upgrading from a version of the QAD .NET UI earlier than 2.9.4 where the Reporting Framework included six sample reports, note that the six reports have been removed in the newer versions. If you keep the previous menu system data, and the AppServer has the upgraded version without the .p programs (proxies) for the sample reports, you get an error when you launch the reports from the menu. If you no longer need these reports, you can delete them from the menu system using Menu System Maintenance. If you would like to continue to use these sample reports, you can copy the six proxy programs from your previous system to the new system and compile them. Alternatively, delete them from the menu system, and then install the six sample reports included on the Reporting Framework Source CD, following the instructions included with the CD, which is available for download from the QAD Store (see “Reporting Framework Source and Samples on QAD Store” on page 17).

Application Changes

If upgrading from a previous version of the QAD .NET UI, be sure to review the release notes for the versions between your current version and this version.

InBox Messages to Inactive Users

Previously, you could send InBox messages to inactive or nonexistent users without getting an error message. This issue has been resolved so that you now get an error message (“The following addresses were not recognized”) when you send or save the message.

Browse Maintenance Print

You can now print the visual relationships between tables shown in Browse Maintenance by choosing File > Print.

Refresh Button on Browse Toolbar

The browse toolbar now includes a refresh button next to the paging buttons so you can easily refresh the display of browse data. The refresh button is included on both drill-down and lookup browses.

Generalized Code Scrolling Improvement

Fields with generalized codes allow you to scroll through a pull-down display of field values using the up and down arrow keys. Previously, you could mistakenly change field values while using the arrow keys. Now, when the up or down arrow key is used, it first opens the pull-down instead of immediately changing the field value.

Browse Option for Switching of CSV Delimiter

In many non-US English environments, the delimiter for CSV export/import is a semicolon rather than a comma. You can now specify the delimiter in the client session configuration file (client-session.xml) browse settings (`<Browse>...<csvdelimiter>,</csvdelimiter>...</Browse>`), where the default is a comma.

Browse Timeout Configuration

Previous releases have introduced browse performance controls that can be configured in the `client-session.xml` file's `<Browse>...<Notification>` settings. Additionally, the QAD .NET UI includes a mechanism to set the timeout on a browse "get all records" request to a single value with a `<TreatGetAllAsOneRequest>` setting in the `client-session.xml` configuration file. The "get all records" request is actually a series of requests to the server. Setting the parameter to false will treat each request separately for timeout. Setting the parameter to true will use one timer for the set of requests. Suppose the timeout value is 5 minutes and 3 calls are made, each taking 4 minutes. A false setting would not time out, as no single request exceeds 5 minutes. A true setting will time out on the second request, as the 5 minutes is used up.

Configurable Screens Added Frame Correction

Previously, an added frame was appearing only on the first line of a multi-line frame. This issue has been resolved.

Messages Display To Column

The Messages display includes a To column along with the columns for From, Date, and Subject.

Browse Control Default Changes

Previously, the browse control defaults were to provide no warnings and then cancel all long-running browses after five minutes. Now, the defaults are to provide warnings after two minutes and then cancel all long-running browses after three minutes. These browse control settings are specified in client-session.xml. The new setting is: `<timeout browseId="" warnAt="2" cancelAfter="3"/>`.

Metric Pie Chart Placement

Previously, the bottom metric pie chart was placed too far to the right if there was a horizontal scroll bar to the right when the group box was expanded. This issue has been resolved.

Operational Metrics Screen Focus Correction

Previously, a focus issue was causing the screen to scroll to the top or side when any chart was hovered over by the cursor when scroll bars were present. Now the focus goes to the chart that was hovered over, and the screen jumps to that chart instead of the top of the page.

Operational Metrics Print Format Correction

Previously, when you printed a metric with charts expanded that were high enough to span multiple pages, the charts could sometimes get cut across two pages instead of being kept on the same page as one chart image. This issue has been resolved.

Operational Metrics Print Memory Correction

Previously, in Operational Metrics, using Print or Print Preview from a metric that has never had its history charts expanded could result in an out-of-memory error and in some cases cause a red X to be displayed on the screen. This issue has been resolved.

Operational Metrics Target Range Correction

Previously, the metric editor target-range dropdown control did not change the number field when the dropdown was set to 1. This issue has been resolved.

Operational Metrics Autosizing Correction

Previously, a metric group display was not resized after the creation of a new metric in a group. This issue has been resolved.

URN Support for Browse Collections

The system's URN format can now specify browse collections, using the format `urn:collections:`*`collection_ID`*.

Browse Cancel Button

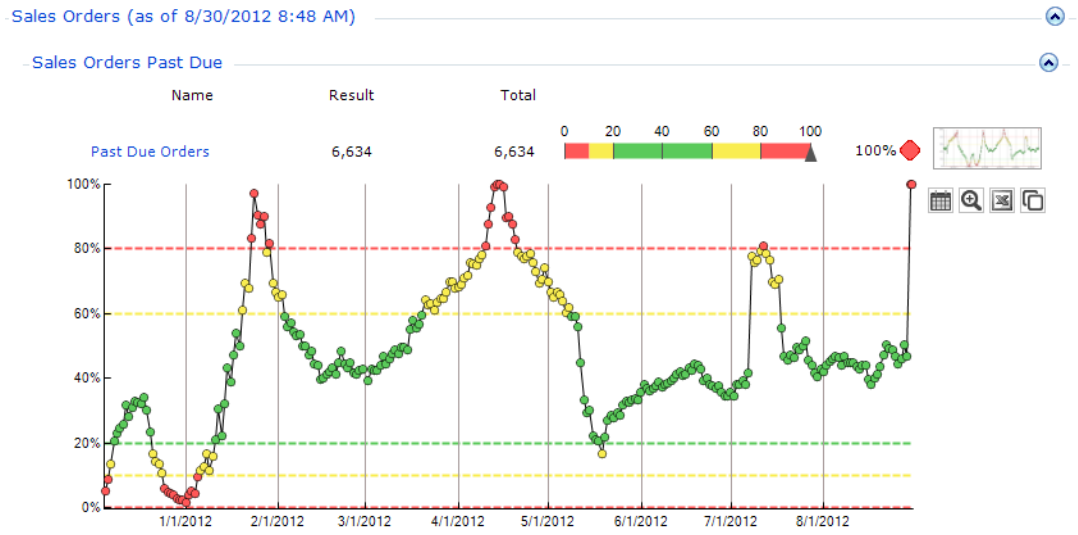
The toolbar for browses includes a Cancel button so that users can stop long-running browses.

AIA Timeout Setting

The browse performance controls apply in AIA environments, but for other types of requests in AIA environments, you can configure the AIA timeout setting in the `client-session.xml` files's `<AppServer>...<Timeout>` setting (in seconds). The default setting of 0 directs the system to use the Progress default, which is 100 seconds.

Operational Metrics History

With the QAD Operational Metrics History, you can view changes in operational metrics over time. The system stores the history of operational metric activity and then generates charts of the data for you.



You can then explore the chart data further, change the time range, scroll right and left or zoom in and out of particular areas of interest, and export the chart data to Excel for further analysis.

When you open an operational metric page, the system uses the most recent history data to display the initial view of the metric page if the data is less than 24 hours old. This allows the page to be displayed more quickly than if the underlying metric browses were queried to retrieve the data.

The system saves history data whenever the browse queries for the metrics are run. In addition to the history data, a pie chart that summarizes the metric results is also saved so that the Operational Metrics View process map can show the most recently generated results.

The system queries the metric browses (and saves history data) in the following situations:

- When you click an operational metric's Refresh button.
- When you open an operational metric and the history data is more than 24 hours old. (Note that 24 hours is the default; the interval is configurable and applies to all metrics in the system.)
- When the QAD_OpMetricsAutoRun report is run. Typically, this report is run when metrics are scheduled for periodic running, but if a report administrator runs this report from the report designer, it also runs the metric and its processes.

The history data is never deleted from the system.

A new "as of" label next to the metric name indicates the date and time of the metric data being displayed. If the browse queries are currently running, "loading..." is displayed next to the metric name; when they finish running, the "as of" time shows the current time.

Metric history data will contain at most one history record per day for a given metric. If the metric's browses run more than once in a given day, the history reflects the most recent run.

Configuring Operational Metrics History

Configuring operational metrics history includes the following:

- Operational Metrics History Update Interface
- Operational Metrics History and the QAD Reporting Framework

Operational Metrics History Update Interval

By default, if operational metric history data is more than 24 hours old, the system updates the data (by running the metric browses) when you launch a metric collection. Otherwise, the most recent history data is used to display the metric more quickly.

You can change the time interval by adding (and modifying) the following to the `client-session.xml` configuration file:

```
<Metrics>

...

      <StaleDataAllowedHours>24</StaleDataAllowedHours>

...

</Metrics>
```

The time interval applies to all metrics in the system.

Operational Metrics History and the QAD Reporting Framework

Operational Metrics History uses the QAD Reporting Framework report server's scheduled batch mode to auto-run a special report that runs the desired metric and generates and stores metric history data. The new QAD-supplied report, QAD_OpMetricsAutoRun, is used for auto-running the metrics in scheduled batch mode.

Your system must be configured to run scheduled reports in scheduled batch mode (see the Scheduled Batch Mode section in the *Reporting Framework User Guide*'s Administering Reports chapter).

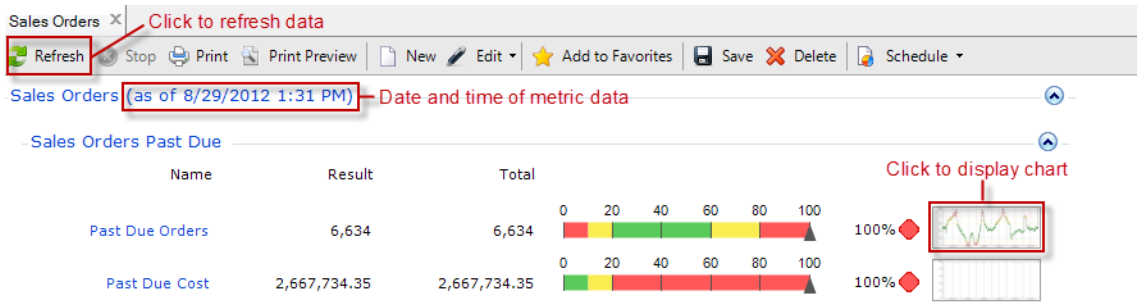
Additionally, the Set Up a Scheduled Batch section in the *Reporting Framework User Guide*'s Administering Reports chapter describes how to create a parameter file to contain command line parameters with fixed values, using a params.pf file as an example. In that file, you must add the following line in order for the metric report to run properly:

```
-enable:qad.plugin.opmetrics
```

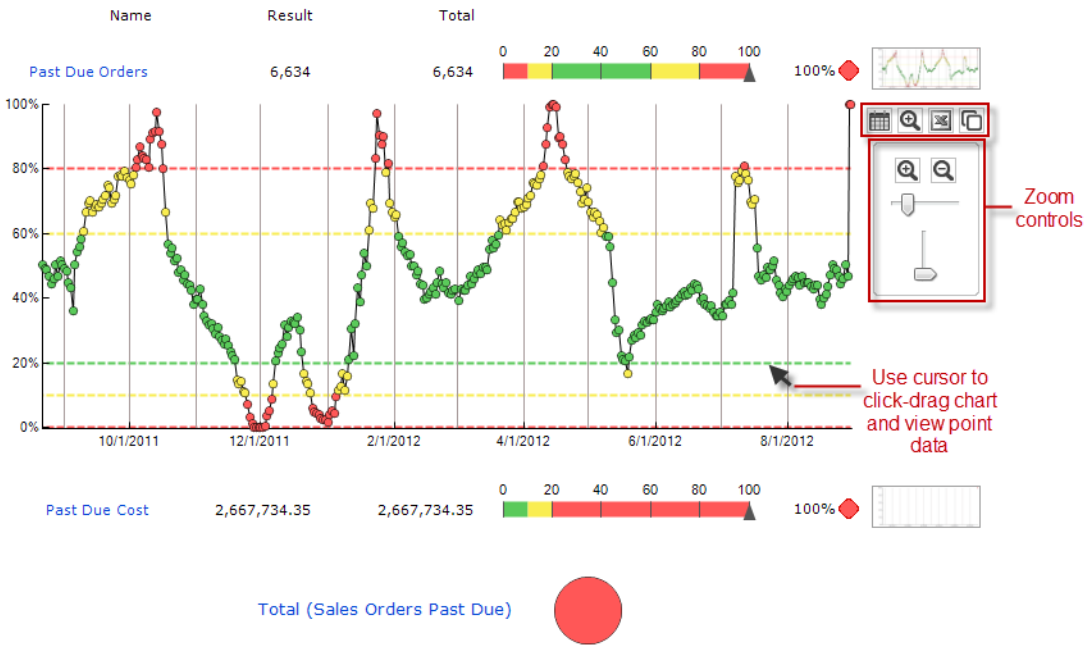
Using Operational Metrics History

With the QAD Operational Metrics History, you can view changes in operational metrics over time. The system stores the history of operational metric activity and then generates graphs of the data for you.

For example, in the Sales Orders metric collection, open the Sales Orders Past Due metric. Notice the thumbnail images to the right of the displays for Past Due Orders and Past Due Cost. Click on the thumbnail image for Past Due Orders:



When you click on the chart thumbnail, a metric history chart displays:



Navigating the History Chart

Under the thumbnail image of the chart, click the icons that allow you to:

- Change the time range of the display (1 week, 1 month, 3 months, 6 months, 1 year, 2 years, 3 years).
- Toggle the Zoom control to:
  - Zoom in
  - Zoom out
  - Zoom horizontally
  - Zoom vertically
- Export the history data to Excel for further analysis.
- Detach the chart to a separate window to enlarge the view.

You can also click-drag to navigate the metric history chart.

If you mouse over a data point, it shows you its value. The color of the dot on a given day corresponds to the metric result for that day. Boundaries for the result ranges (where red indicates an error, yellow indicates a warning, and green indicates good) are also displayed on the chart as colored dotted lines.

Clicking the chart thumbnail again hides the metric history chart.

Scheduling Batch Processes for Operational Metrics History

Although metric history gets generated whenever a user manually refreshes a metric (or opens a metric that does not have recent history data), you can schedule metrics to be run at regular intervals to guarantee the regular creation of history data. For example, you might want to schedule a certain metric to be auto-run daily and a different metric to be auto-run weekly.

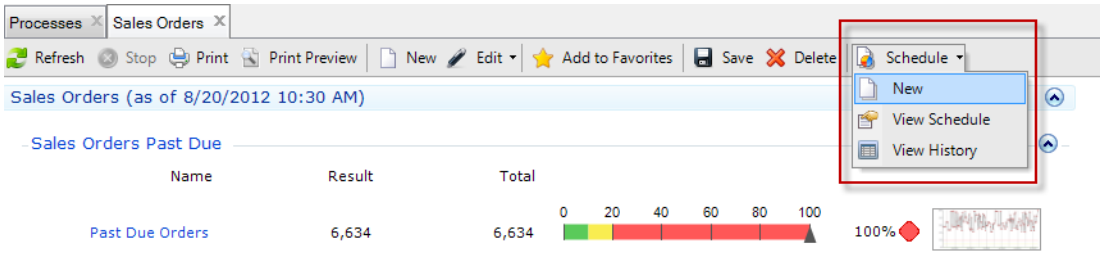
Operational Metrics History uses the QAD Reporting Framework report server's scheduled batch mode to auto-run a special report that runs the desired metric and generates and stores metric history data. This

must be configured as described in “Operational Metrics History and the QAD Reporting Framework” on page 10.

Scheduling and Running Batch Processes

You can schedule and edit these batch processes directly from a metric display.

From the toolbar, choose Schedule to schedule batch processes:



The Schedule pull-down options include:

- New — schedule a new batch process. Enter a valid batch ID.

  Note Batch IDs must first be defined by an administrator using Batch ID Maintenance. It can be useful to name the batches according to the time interval at which the report server is configured to run that batch. For example, you might define a batch called “daily” that is configured to run every day and another batch called “weekly” that is configured to run once a week. When a batch ID is specified by the user, the metric auto-running only occurs if a report server is configured to process that batch ID.

- View Schedule — view currently scheduled batch processes in a browse. You can view further details and modify the batch process by right-clicking on the ID and choosing Scheduled Report History, Parameters, and Scheduled Report Maintenance. Use Scheduled Report Maintenance to modify batch details. (See the *Reporting Framework User Guide*’s Maintaining Scheduled Reports section for further information.)

- View History — view previously run batch processes in a browse that includes their status, such as New, Waiting, Running, Complete, or Error. (See the *Reporting Framework User Guide*’s Viewing Report History section for further information.)

Saving Operational Metrics as Favorites

To save a metric as a favorite, you can either:

- Drag the menu item for the metric from the Applications pane to the Favorites pane.
- When the metric is open, click the Add to Favorites button in the toolbar.

Important These two ways do not create the new favorite in the same way:

When you drag the metric from the Applications pane to the Favorites pane, the new item in the Favorites pane points to exactly the same metric with the same history data. Clicking on the favorite opens the same metric as clicking on the metric in the Applications pane.

When you click Add to Favorites, however, a new metric is created and the new item in the Favorites pane points to the new metric. The reason it creates a new metric is that you are free to make custom changes to it before saving it as a favorite. As with all favorites, the new favorite metric is only visible to the user that saved it. Although the new favorite can have the same name (by default) as the one on the Applications pane, clicking on the favorite opens the new metric. Although the new metric is based on the same browses, the history data saved for the new metric is different. Additionally, the new metric saved as a favorite by using Add to Favorites does not have the scheduling functionality.

Note If you want the metric saved as a favorite to be the same metric (with the same history data) as the one on the Applications pane, be sure to drag the menu item from the Applications pane to the Favorites pane.

Attaching Metrics History Manually

Metrics history is attached to specific metrics collections, which reside as XML files on the QAD home server. If these XML files are replaced by new or modified files (during an upgrade, for example), the system reconnects the existing history to the new XML files. However, in some cases, due to certain changes in the new (or old) XML files (changes in metric names, for example), the history might not get reconnected. To address this, a history connection screen can be enabled, allowing an administrator to manually connect the history to metrics. To enable this feature:

1. Edit the client-session.xml file, and find the `<Metrics>` element.
2. Inside the `<Metrics>` element, add the following:

   ```
   <ManualAttachHistory>true</ManualAttachHistory>
   ```

3. Launch the QAD .NET UI client.

Now, when an administrator (someone with access to create metric collections) runs a metric and right-clicks on the metric name, there will be a History menu item with the following options:

- Merge — this brings up a window in which the current history chart will be displayed along with a chart in which can be displayed one or more sets of history that are available to merge with the current history.
- Replace — this brings up the same window as above, only the selected history set completely replaces the current history, instead of merging with it.
- Unlink — this disconnects the metric from any history, allowing history to start fresh.

The changes done in all three cases only take effect if the collection is saved using the Replace option.

Connection Manager: Secure Shell (SSH) Default

The Connection Manager now specifies secure shell (SSH) rather than telnet by default. To view and change the settings, in Administration | Connection Manager, under Functions, click Update configuration settings:

Host. The machine name or IP address of the SSH or telnet server.

Port. The port number for SSH or telnet. The default is `22` (SSH). For telnet, the port number is `23`. (Previously, the default was `23` for telnet).

Protocol. Specifies the connection protocol as `ssh` (the default) or `telnet`. (Previously, the default was `telnet`.)

Startup Script. The server log-in prompts and the responses to these prompts, separated with the pipe symbol (|). The standard order is:
`loginPrompt|userid|passwordPrompt|$PASSWD|osPrompt|cd UIConfigDir|osPrompt|startScript`
For example:
`login:|mfg|Password:|$PASSWD|$|cd /user/mfg/work|$|exec /user/mfg/work/scripts/connmgr.wrap`

Note The same startup script can be used for both SSH and telnet. If using SSH, the login credentials in the script are ignored if they are specified in the *Server Startup User* and *Server Startup Password* settings. For SSH, if you have defined *Server Startup User* and *Server Startup Password*, you can remove the login credentials from the script but the first four token delimiters must still be included in the script. For example, for telnet, the script might be:

```
login:|mfg|Password:|$PASSWD|$|cd /user/mfg/work|$|exec
/user/mfg/work/scripts/connmgr.wrap
```

For SSH, however, with the login credentials defined in *Server Startup User* and *Server Startup Password*, the script can be:

```
||||$|cd /user/mfg/work|$|exec /user/mfg/work/scripts/connmgr.wrap
```

Server Startup User. Specifies the user ID of the server startup user, if not specified in the startup script. (This new setting is only used for SSH.)

Server Startup Password. The password for the session startup script, if not specified in the startup script. It is encrypted on entry.

Reporting Framework Changes

The following summarize Reporting Framework changes for this release.

QAD_Translate VBscript Object

Report script objects provide dynamic functionality that can be called from VBscript code within report layout designs. A new script object (QAD_Translate) has been added in this release to provide access to translated labels from VBscript code, adding more possibilities than previously existed in the use of label report fields. This script object is especially useful in situations where combinations of labels and other report data need to appear in a single label field; previously report developers would use techniques such as hidden label fields to implement such features.

The QAD_Translate script object provides a number of methods to retrieve translated labels from the QAD label master repository. Each label is uniquely identified by a label term key which points to a number of translated strings in different languages, possibly containing several variations of different lengths (small, medium, and large). Some labels also contain a stacked label variation, which uses the ! symbol within the translated text to denote the positions where the text should be continued on the next line of display.

Note Fields using stacked labels in the report design should have sufficient height to accommodate the stacked label. For example, if the stacked label indicates two lines of text (has one ! character), then the fields should be twice the normal height for that font.

QAD_Translate.GetBestFitString(labelTerm, field)

Gets the translated string indicated by the input label term key for the user's language. This method will automatically select the longest of the translated labels (small, medium, or large) that will fit in the specified report field, taking into account the field's width and font properties.

Table 1
QAD_Translate.GetBestFitString(labelTerm, field)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
field	*Input*	Object	The report field whose width and font are to be inspected for choosing the best fit label. The field can be specified using a statement such as `report.fields ("Field4").`
	Return Value		The largest label for the specified term key that will fit into the specified field.

QAD_Translate.GetBestFitString(labelTerm, field,erpLanguage)

Gets the translated string indicated by the input label term key for the specified language. This method will automatically select the longest of the translated labels (small, medium, or large) that will fit in the specified report field, taking into account the field's width and font properties.

Table 2
QAD_Translate.GetBestFitString(labelTerm, field, erpLanguage)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
field	*Input*	Object	The report field whose width and font are to be inspected for choosing the best fit label. The field can be specified using a statement such as `report.fields("Field4").`
erpLanguage	*Input*	String	The QAD Label Master language identifier.
	Return Value		The largest label for the specified term key that will fit into the specified field.

QAD_Translate.GetLongLabel(labelTerm)

Gets the translated string for the long label indicated by the input label term key for the user's language.

Table 3
QAD_Translate.GetLongLabel(labelTerm)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
	Return Value		The long label for the specified term key.

QAD_Translate.GetLongLabel(labelTerm, erpLanguage)

Gets the translated string for the long label indicated by the input label term key for the specified language.

Table 4
QAD_Translate.GetLongLabel(labelTerm,erpLanguage)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
erpLanguage	*Input*	String	The QAD Label Master language identifier.
	Return Value		The long label for the specified term key.

QAD_Translate.GetMediumLabel(labelTerm)

Gets the translated string for the medium label indicated by the input label term key for the user's language.

Table 5
QAD_Translate.GetMediumLabel(labelTerm)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
	Return Value		The medium label for the specified term key.

QAD_Translate.GetMediumLabel(labelTerm, erpLanguage)

Gets the translated string for the medium label indicated by the input label term key for the specified language.

Table 6
QAD_Translate.GetMediumLabel(labelTerm,erpLanguage)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
erpLanguage	*Input*	String	The QAD Label Master language identifier.
	Return Value		The medium label for the specified term key.

QAD_Translate.GetShortLabel(labelTerm)

Gets the translated string for the short label indicated by the input label term key for the user's language.

Table 7
QAD_Translate.GetShortLabel(labelTerm)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
	Return Value		The short label for the specified term key.

QAD_Translate.GetShortLabel(labelTerm, erpLanguage)

Gets the translated string for the short label indicated by the input label term key for the specified language.

Table 8
QAD_Translate.GetShortLabel(labelTerm,erpLanguage)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
erpLanguage	*Input*	String	The QAD Label Master language identifier.
	Return Value		The short label for the specified term key.

QAD_Translate.GetStackedLabel(labelTerm)

Gets the translated string for the stacked label indicated by the input label term key for the user's language.

Table 9
QAD_Translate.GetStackedLabel(labelTerm)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
	Return Value		The stacked label for the specified term key.

QAD_Translate.GetStackedLabel(labelTerm, erpLanguage)

Gets the translated string for the stacked label indicated by the input label term key for the specified language.

Table 10
QAD_Translate.GetStackedLabel(labelTerm,erpLanguage)

Name	Input/Output	Data Type	Description
labelTerm	*Input*	String	The label term key.
erpLanguage	*Input*	String	The QAD Label Master language identifier.
	Return Value		The stacked label for the specified term key.

Temporary Memory Usage

Issues with temporary memory usage have now been fixed, improving performance and stability for multiple runs of reports with large data sets.

Reporting Framework INT64 Support

The Reporting Framework now supports the INT64 Progress data type in data sources.

Warning When a data source has a Progress INT64 field that is being displayed in a report layout, it is represented in the report as a long integer. This maintains the precision of data from the data source and displays properly. However, if calculated fields are created in the report layout (using Report Resource Designer) which do calculations on the INT64 field, then these calculated fields are represented as decimal fields (type “double”). Double fields can represent large numbers, but they do so at the expense of precision. As a result, large INT64 values when used in calculated fields may lose precision, although this only starts to occur in the quadrillions. A solution to this for Progress Data Source programs is to do the calculation in the data source program itself and pass it back as an INT64 field instead of doing the calculation in the layout definition using VBScript.

Data Source Dynamic Settings

It is now possible to write code in Progress report data source programs that allows the data source to dynamically determine report settings such as language, date and number formats, and search criteria location in the report output. For more details, see the *Reporting Framework User Guide*.

Reporting Framework Source and Samples on QAD Store

The Reporting Framework Source and Samples download (Reporting Framework Source CD) is available from the QAD Store (http://store.qad.com). The download contains sample reports, templates, and portions of the Reporting Framework’s Progress code that are relevant for Progress data source developer usage. This does not include the source code of the data source programs used by the hundreds of new reports that have been developed recently using the framework; it only includes the generic pieces that are part of the framework itself. The download also includes documentation, coding tools, and examples for using the Scheduled Report and Run Report APIs, which allow developers to write applications to schedule or run reports.

Scheduled Report API No Longer Limited To rptAdmin Role

In previous releases, programs that schedule reports using the Scheduled Report API had to pass input authentication credentials corresponding to a user in the rptAdmin role, which authorized the scheduling of any report in the system. The authorization mechanism has been changed to allow for finer-grained

authorization at the report level, and allows any QAD user credentials to be used. The new authorization mechanism leverages QAD menu security in a fashion similar to how browses are authorized. The following rules now apply:

- If the report is not on the menu system at all, full access to run the report is given.
- If the report is on the menu system, the user's access to that report in the given domain is checked; if the user has access then the report can be scheduled.
- If the user does not have access, the API does not schedule the report and returns an error (error code 107: user role permissions error).

For more information about the Scheduled Report API, see the Reporting Framework Source and Samples download, available on the QAD Store (http://store.qad.com).

Language Options

The report viewer's Settings options now include a new Language setting in the General tab. Use the Language pull-down to choose the language in which to display the report. When a user changes the report language setting, it only affects the report currently being run, and remain in effect until that report program tab is closed. Other program tabs are not affected by the report language change.

Like all report settings, the choice of language is saved if the user clicks the Save As button to save the report settings (including search conditions) for quick recall in future runs.

Number To Words Conversion for 50 in French

The QAD_NumberUtil.ToWords .NET script object converts the input number into word strings in the specified language. Previously, in the French language, the script object was incorrectly converting the number 50 to "cinquente" instead of the correct "cinquante." This issue has been resolved.

Programs in Terminal Mode Only

Some programs are only available in Terminal mode, which emulates the Character UI within the QAD .NET UI. You navigate the program in the same way as in the Character UI. The following programs are only available in Terminal mode:

- Accounts Not To Convert Maint
- AP Integrity Report
- Archive File Reload
- Call Queue Manager
- Change Deferred/Accrued Accounts
- CIM Data Load Process Monitor
- Combined Integrity Checks
- Compile Programs
- Convert Ship Qty in Ship UM
- Count Program
- Create Records for Printer Output
- Database Connect
- Database Disconnect
- Database Table Size Inquiry

- Debug CIM Document
- Dump Export/Import Doc for Edit
- End User Time Zone Change Util
- Escalation Monitor
- Exit to Operating System
- Export/Import Document Query
- Field Eligibility Maintenance
- Fixed Asset Maintenance
- Fixed Assets Integrity Report
- GL Integrity Report
- GLRW Mismatch A/C Code
- Initial Euro Exchange Rate Copy
- Inventory Integrity Report
- License Registration
- Multiple Time Zones Startup Util
- PO Integrity Report
- Process Import Documents
- Program Level
- Program/Text File Display
- Receive Import Documents
- Reload Edited Export/Import Doc
- Required Ship Schedule Update
- Send Export Documents
- Sequence Maintenance
- Server Time Zone Change Util
- Set Multiple BOL Print Utility
- Ship-From to AR
- Trading Partner Library Load
- Trading Partner Library Unload
- WIP Integrity Report