

Release Notes, QuickBooks SDK 3.0

11/7/03

Welcome to the QuickBooks Software Development Kit (SDK) Version 3.0. This software is designed for use with QuickBooks 2004; selected features of this software can be used with QuickBooks 2002 and 2003 as well.

The system requirements for the QuickBooks SDK are the same as those for the version of QuickBooks that is running. QuickBooks 2004 will run on Microsoft Windows 98, 2000, Me, or XP operating systems. QuickBooks 2002 and 2003 will run on Windows NT, 98, 2000, Me, or XP operating systems. (The SDK does not run in unattended mode—that is, with automatic login—on Windows 98, First Edition; it does, however, run in unattended mode on Windows 98, Second Edition.)

Table of Contents

- New Request Processor: QBXMLRP2 (p. 2)
- Support for QuickBooks Products (p. 2)
- Installation Instructions (p. 3)
- New Files of Interest (p. 4)
- Documentation (p. 5)
- Tools (p. 6)
- Samples (p. 6)
- Tips for Using Version 3.0 (p. 8)
- Known Bugs and Workarounds (p. 9)
- Bugs Fixed in This Release (p. 9)
- Changes to the qbXML Specification (p. 10)
- If You Have Questions . . . (p. 13)
- Summary of New Features (p. 14)
- Details of Changes in SDK 3.0 (p. 17)
 - Table 1: New Requests/Responses
 - Table 2: Elements Added to Requests
 - Table 3: Elements Added to Responses and Response Aggregates
 - Table 4: Enumeration Changes
 - Table 5: Changes to Reports

New Request Processor: QBXMLRP2

This release introduces a new version of the Request Processor, **QBXMLRP2**. This new Request Processor is included in QuickBooks 2004 (and supports QuickBooks 2003 and 2002 as well). If you are using qbXML, you must use QBXMLRP2 instead of QBXMLRP for SDK 3.0 requests. ProcessRequest will fail in QBXMLRP if it is asked to process 3.0 requests.

We strongly recommend that you upgrade existing applications to use QBXMLRP2 because it resolves problems that can occur with QBXMLRP when multiple versions of QuickBooks are installed. The SDK 3.0 Release includes a redistributable installer and merge module for QBXMLRP2, which you can include in the installation for your application.

For information on new methods contained in the new Request Processor, see the *Developer's Guide for QuickBooks* and the *Developer's Guide for the QBFC Library*.

Support for QuickBooks Products

The following table lists the different versions of QuickBooks and the corresponding versions of the qbXML specifications that support them. (*Note: The SDK does not support QuickBooks Basic.*)

Version of QuickBooks	qbXML Specification Version
QuickBooks 2004 (and Enterprise 4.0) (using QBXMLRP2)	3.0, 2.1, 2.0, 1.1, 1.0
QuickBooks 2004 (and Enterprise 4.0) (using QBXMLRP)	2.1, 2.0, 1.1, 1.0
QuickBooks 2003 (R7 and above) (and Enterprise 3.0)*	2.1, 2.0, 1.1, 1.0
QuickBooks 2003 (and Enterprise 2.0)*	2.0, 1.1, 1.0
QuickBooks 2002 (R2 and above) (and Enterprise)*	1.1, 1.0
QuickBooks 2002 (R1)*	1.0
QuickBooks Online Edition (v. 10)	3.0, 2.1, 2.0
QuickBooks 2004, Canadian Edition (using QBXMLRP2)	CA3.0, CA2.0
QuickBooks 2004, Canadian Edition (using QBXMLRP)	CA2.0
QuickBooks 2003, Canadian Edition*	CA2.0
<small>*Earlier releases of QuickBooks will work with either version of the Request Processor (QBXMLRP2 or QBXMLRP). We recommend you upgrade existing applications to use QBXMLRP2. Features new in SDK 3.0 are available only with QBXMLRP2; in addition, this Request Processor solves some problems found in the older Request Processor.</small>	

The following table lists the different versions of the QuickBooks Foundation Class (QBFC) Library and the versions of the qbXML specification that support them.

<i>QBFC Library</i>	<i>qbXML Specification Version</i>
QBFC3 (including support for QuickBooks Online Edition and Canadian Editions of QuickBooks)	3.0, 2.1, 2.0, 1.1, 1.0, CA3.0, CA2.0
QBFC2_1 (including support for QuickBooks Online Edition)	2.1, 2.0, 1.1, 1.0
QBFC2	2.0, 1.1, 1.0
QBFC2CA	CA2.0
QBFC1	1.1, 1.0

Installation Instructions

To install the SDK using the installer, execute *QBSDK30.exe*. The installer also adds a menu (*Start>Programs>QuickBooks SDK 3.0>*) that provides convenient access to SDK tools, samples, and documentation (including the *Onscreen Reference*). You can also access the qbXML Validator and qbXML Test applications through the "Send To" context menu for files.

If you have previously installed a Beta release of the SDK, be sure to uninstall the Beta version of the software before you install this final release. Failure to do so may cause improper copying of the most recent SDK documentation.

Registering an Application for QuickBooks Online Edition

Before an application can connect with QuickBooks Online Edition using the SDK, it must be registered with the Intuit Developer Network. To register your application and obtain an application ID, please follow the instructions in Chapter 3 of the *Developer's Guide for QuickBooks Online Edition* and visit <http://appreg.quickbooks.com> to register your application.

Importing the Correct Library

For Visual C++ applications (if you are not using the QBFC Library), you need to import the library named *qbxmlrp2.dll*, which is installed in *C:\Program Files\Common Files\Intuit\QuickBooks*. For Visual Basic applications, select

Project > References from within your VB project. Find "QBXMLRP2 1.0 Type Library" in the list and check the box next to it, then click OK. If you don't find the entry in the list, you can browse to it.

Migrating QBFC Applications to Version 3.0

If you want to use SDK 3.0 requests, you will need to change QBFC code to refer to the new QBFC Library. For example, in Visual Basic, replace "QBFC2_1Lib" with "QBFC3Lib."

In SDK 3.0, the CreateMsgSetRequest method now takes a new first parameter in addition to the major version and minor version to specify the country. The value should be a string "US" or "CA" for US and Canada respectively. When using QBFC with QuickBooks Online Edition, this value is ignored (since QuickBooks Online Edition is US only), but we recommend that you supply the "US" value anyway for QuickBooks Online Edition.

You must also make a change to the DataExtAdd request when you are migrating an application from QBFC2 or QBFC2_1 to QBFC3. The ORListTxn must be changed to ORListTxnWithMacro.

See the *Developer's Guide for the QBFC Library* for additional tips on migrating QBFC applications to a new SDK version.

The QBFC3 library requires the Microsoft system DLL ShFolder.dll to be installed on the system. Newer Microsoft operating systems (Windows ME, Windows XP, etc.) have this DLL pre-installed. This DLL is installed if needed by QuickBooks and by the QBFC3 stand-alone installer using the SHFolder.exe redistributable installer from Microsoft. If you plan to use the QBFC3 merge module in your installer *and* you want your application to work with RDS (where QuickBooks may not be installed on the machine where your application is installed) then your installer must include and execute the SHFolder.exe installer from Microsoft. Technical issues prevent us from including this action in the QBFC3 merge module itself.

New Files of Interest

This release includes documentation, tools, and samples. The following files and directories may be of particular interest to you:

- This release includes .NET Primary Interop Assemblies (PIAs) for

QBXMLRP2, QBFC3, and QBFC2.1. These assemblies are included in the redistributable installers and merge modules for each library. If you have previously built a local interop assembly for use with your .NET integrations, please be sure to update them to use the primary interop assemblies included with this release instead.

- *doc\qbxml\ops30.xml*
The sample file containing all qbXML requests and responses.
- *OSR\OnscreenRef*
The Onscreen Reference for SDK 3.0.
- *samples\qbd\vb\qbxml\DataEvents*
A simple test application that shows how to subscribe to data events and implement a callback for those events.
- *samples\qbd\vb\qbfc\Events*
Illustrates event handling using the QBFC Library.
- *samples\qbd\cpp\qbxml\UIandEventTest*
Shows how to subscribe to events (data, UI and UI extension events) and handle notifications.
- *samples\qbd\vb\qbxml\sdktest*
A simple test application that shows how to integrate with QuickBooks SDK and allows the user to select a qbXML request file and submit it to QuickBooks for processing. This sample has been updated to use QBXMLRP2 so that it can handle files containing all QBXML requests (including QBXML 3.0 requests).
- *samples\qbd\vb\qbfc\UI-and-eBay*
A sample application showing a simple bridge between eBay sales data and QuickBooks, this sample also highlights several aspects of integrating with the QuickBooks UI, including adding and removing menu items and requesting that the QuickBooks UI open and display a specific transaction form.
- *samples\xmlfiles\CustomFieldAddValue.xml*: adds a value to a custom field for an existing customer
- *samples\xmlfiles\CustomFieldDelValue.xml*: clears a value in a custom field for an existing customer
- *samples\xmlfiles\CustomFieldModValue.xml*: modifies a value in a custom field for an existing customer
- *samples\xmlfiles\DisplayTxn.xml*: posts the QuickBooks transaction edit window for an existing transaction (purchase order, as shown in the sample)
- *samples\xmlfiles\TemplateQuery.xml*: displays the templates that are defined for the current company file.

- *samples\xmlfiles\DisplayList.xml*: displays the Other Name edit form for adding a new Other Name to QuickBooks.

All new samples require QuickBooks 2004, the QBXMLRP2 Request Processor, and the QuickBooks 2004 sample product-based company file. (The QBXMLRP2 Request Processor is installed automatically by QuickBooks 2004.)

Files in New Locations

The SDK configuration file (*qbsdk.ini*) and the SDK log file (*qbsdklog.txt*) have been moved to the following directory: *C:\Program Files\Common Files\Intuit\QuickBooks*.

Standalone (.exe) installers for redistributable components of the SDK have been moved to the SDK's *tools\installers* directory. This includes the QBFC redistributables, which used to be in the SDK's *tools\QBFC* directory. This move is being made because we now provide QBXMLRP2 redistributable components, which are independent of QBFC.

Documentation

This release includes new SDK manuals in both PDF and HTML. Along with the other major features in the SDK, the manuals and the *Onscreen Reference* can be conveniently accessed through the Start menu.

For best results, use Internet Explorer version 6 or higher to view the *Onscreen Reference*. If you use Netscape Navigator, you may not be able to view all the information.

Tools

The **qbXML Validator** tool contains new functionality that allows you to verify qbXML requests that correspond to these combinations:

- 3.0, 2.1, 2.0, 1.1, 1.0 for U.S. editions of QuickBooks
- 3.0, 2.1, and 2.0 for QuickBooks Online Edition
- CA3.0 and CA2.0 for Canadian editions of QuickBooks

The Validator is installed automatically as part of the SDK installation.

The **SDKTest** application is located in the *\tools\SDKTest* folder. This useful development tool reads a qbXML request file, sends it to QuickBooks, and returns a response. The command line version of SDKTest is *samples\qbd\cpp\qbxml\sdktest\sdktest.exe*.

The file *tools\SDKTest\SDKTestPlus3.exe* is an enhanced version of SDKTest

that uses QBXMLRP2 and allows the user to control every aspect of the connection to QuickBooks. It supports sending subscription request XML as well as data integration XML. The source code for this tool is in *samples\qbdt\vb\qbxml\SDKTextPlus3*.

The file *tools\SubscriptionManager\SubView.exe* is a tool for viewing and removing application subscriptions from QuickBooks.

Samples

The Samples directory contains a *readme.html* (which can be accessed from *Start > Programs > QuickBooks SDK 3.0 > Samples*). This file lists each sample and provides links to open the project directory, the project itself, and the executable form of the application.

The SDK 3.0 installer verifies that the MSXML4.0SP2 API from Microsoft is installed on your system. If MSXML is not present, the SDK 3.0 installer will install it for you. This API is used by the SDK's Validator tool and several samples.

Downloading MSXML 4.0

If you want to run the sample applications on a system that does not have MSXML installed (for example, if you want to use Remote Data Sharing from a system that does not have the SDK installed), you will need to install MSXML. In addition, if your application is dependent on MSXML Core Services, you will need to redistribute that software in your installer. The MSXML Core Services software is available from <http://msdn.microsoft.com>

Building the C++ Samples

The `#import "qbxmlrp2.dll"` statement in the C++ samples allows these applications to use the COM interface supported by the QuickBooks SDK. This statement will cause an error when you are building the samples in Microsoft Visual Studio if the Include path is not set to include the location of this DLL. The DLL is installed with QuickBooks and resides in the QuickBooks common directory, currently defined as *C:\Program Files\Common Files\Intuit\QuickBooks*. Add this directory to the

Include path by choosing the Tools menu, Options command, and the Directories tab in Visual Studio.

UIandEventTest Samples

In order to build the EventHandler C++ sample, you will need to specify the QuickBooks executable path for the sample project. The method for doing this is different for Visual Studio 6.0 and Visual Studio .NET.

For Visual Studio 6.0, select "Options" from the "Tools" menu. Select the "Directories" tab and then select "Executable Files". Add the QuickBooks executable path to the list of directories here.

For Visual Studio .NET, select "Properties" from the "Project" menu. Under the "C/C++" options, select "General" and add the QuickBooks executable path to the "Additional Include Directories". Under the "MIDL" options, also select "General" and add the QB path to "Additional Include Directories".

All of these C++ samples also use *QBXMLRP2.DLL* in the QuickBooks "common" directory. The project files currently specify this path as *C:\Program Files\Common Files\Intuit\QuickBooks*. If this path is different on your machine, you will need to change it in the project files.

Known Problems in VB Sample Applications

The Visual Basic sample applications allow you to view a response from QuickBooks. If you try to view a response that is greater than 32K, you will receive an "Out of memory" error. This happens because the VB sample applications use a TextBox control, which has a 32K limit, rather than a RichTextBox control. In a Windows 98 or Windows Me environment, this can corrupt memory, and the application might not terminate properly.

There is a known issue (not related to qbXML) with the AddReceivePayment sample application. The MS Visual Studio application must be installed before you can run AddReceivePayment, or you will receive the following error message when you run the application: "Class not registered. You need the following file to be installed on your machine. MSSTDFMT.DLL."

Tips for Using Version 3.0

Using Payroll Reports

If your application is accessing a company file that is currently signed up for a subscription to a payroll service, you can generate payroll reports and query PayrollNonWageItems.

To test this feature, you can use one of Intuit's sample company files (even though it will not be signed up for a subscription to a payroll service). Please be aware that QuickBooks will record an artificial creation date to any test data that you add to a sample company file.

Using Remote Data Sharing (RDS)

This release includes a redistributable version of Remote Data Sharing 3.0. RDS 3.0 works only for applications using QBXMLRP2. Please note that RDS 3.0 does not support event notification or user interface extensions.

Beginning with SDK 3.0, the Remote Data Sharing (RDS) server supports automatic login mode, also called *batch mode*. In this access mode, QuickBooks is installed but not running on the machine with the RDS server. Instead, the QuickBooks user authorizes the RDS server to log in automatically to the desired set of company files. Subsequently, remote applications can access any of these company files.

Applications using the older Request Processor (QBXMLRP) should use RDS 2.1.

In Chapter 9 (Remote Data Sharing and Your Application) of the Concepts Manual, there is a discussion of merge modules and installers. Missing from this description is the following requirement:

Make sure you have the Microsoft VC (VC_CRT.msm), VC++ (VC_STL.msm), and VC I/O (VC_CRT_IO.msm) runtime library merge modules, which are required because the SDK merge modules install components that depend on the Visual C and C++ version 7 runtime libraries. These Microsoft merge modules are included with most MSI-based install builders, or you can get them directly from Microsoft. When you add the VC_CRT.msm, VC_STL.msm, and VC_CRT_IO.msm modules to the installer you are responsible for configuring them to set their target directory to the Windows system directory.

Known Bugs and Workarounds

- Even though TrackLostEvents is optional and should default to None according to the spec, leaving it out in DataEventSubscriptionAddRq may actually sometimes default it to All. So it's recommended that applications explicitly specify a TrackLostEvents value in their data event subscription request to ensure that they obtain the desired behavior.

Bugs Fixed in This Release

- The RowData element is removed from Time reports in this release. In previous releases, the RowData elements returned in Time reports may contain incorrect values.
- The string length for RefNumber in ReceivePaymentAdd has been changed from 11 to 20.
- In SDK 3.0 and above, the correct AverageCost is now returned for ItemInventoryAdd. In earlier versions of the SDK, under certain circumstances, a value of 0.00 was returned.
- The EmployeeAdd status message now always includes the name that was in conflict. Previously, under certain circumstances, the name was not included.
- In SDK 3.0 and above, DateMacros correctly operate on the fiscal year or calendar year as specified. (Note that the erroneous behavior in calls to pre-3.0 versions of the SDK is preserved.)
- The IsActive element in OtherNameModRq is now working.
- The status messages in List Mod responses now use the ListID instead of the Name field to identify the missing element. Previously, because the Name field is not guaranteed to exist, some responses were missing the necessary element information.
- An "unsupported field" warning is now returned when the Rate field is included in a TimeTrackingAdd request.
- In SDK 3.0, you can filter an Audit Trail report using the ReportModifiedDateRangeFilter or the ReportModifiedDateRangeMacro. These filters use the time stamp for the transaction.
- An error is now returned if a negative sales price is specified for ItemDiscountAddRq. This value must always be a positive value.
- An error is now returned if the application attempts to specify a Subtotal item line. The Subtotal item value is always calculated and set by QuickBooks. If the application attempts to set this value, it receives status code 3040 and the error message, "You cannot set the amount for items of this type."

Changes to the qbXML Specification

- In OtherName -Add, -Mod, and -Ret: Suffix, Mobile, and Pager are now marked as Deprecated because QuickBooks does not support these fields.
- The HostInfo aggregate was added to all three event types.
- In the EmployeePayrollInfo aggregate, an enumerated element has been

added: **UseTimeDataToCreatePaychecks**. Possible values for this element are NotSet, UseTimeData, and DoNotUseTimeData. This element replaces the existing Boolean value IsUsingTimeDataToCreatePaychecks. Please modify your code to use the new enum value. The older Boolean value (IsUsingTimeDataToCreatePaychecks) should no longer be used and will be deprecated in a future release.

- Maximum length of oldMessageSetID and NewMessageSetID: In this release, the maximum length is 24 characters. However, in Releases 1.0 and 1.1, the maximum length was 23 characters. To ensure that your code works with all releases, it is recommended that you consider the maximum length to be 23 characters.
- For ItemInventoryMod, if you attempt to clear COGSAccountRef or AssetAccountRef, you'll receive statusCode 3170 (OBJECT_CANNOT_BE_MODIFIED) if the spec version is 2.1 or earlier, or statusCode 3190 (CANNOT_CLEAR_REQUIRED_ELEMENT) if the spec version is 3.0 or greater. Note that this is an example of statusCode versioning, where we preserve an existing statusCode for 2.1 and earlier requests, while introducing an improved statusCode starting with 3.0.
- An attempt to clear the Name of an Item, SalesRep, Customer, Vendor, or OtherName now returns a 3190 error if the qbXML specification version is 3.0 or greater. Otherwise, it will continue returning the error that was returned before (3090 for Items, 3170 for all others). We don't return 3190 for Employee even with the 3.0 spec version because the Name field was never required for EmployeeAdd. This is another example of statusCode versioning.
- In previous versions of QuickBooks, if a signed application had an invalid or revoked certificate, QBXMLRP would return 0x80040420 "The QuickBooks user has denied access." This behavior is fixed in this release. For the invalid certificate case, both QBXMLRP and QBXMLRP2 will return 0x80040419 "This application's certificate is invalid. An application must have a valid certificate to access QuickBooks company data files." For the revoked certificate case, QBXMLRP will return 0x80040419 (QBXMLRP and older versions of QuickBooks did not distinguish between invalid and revoked certificates, so both are treated as invalid), and QBXMLRP2 will return the new code 0x8004042c "This application's certificate has been revoked by its certification authority. An application must have a valid certificate to access QuickBooks company data files."
- The GroupWithUndepositedFunds BOOLTYPE element was removed from ItemPaymentMod. As a result, the way you would "set" the "Group with undeposited funds" radio button is to either attempt to clear the DepositToAccountRef, or to specify the account whose SpecialAccountType is UndepositedFunds for the DepositToAccountRef. Similarly, for ItemPaymentAdd, you can set the "Group with Undeposited

Funds" radio button either by not specifying DepositToAccountRef at all, or by specifying the special UndepositedFunds account for the DepositToAccountRef.

- The behavior of transaction add requests when `<RefNumber>` is blank (i.e. `<RefNumber/>` or `<RefNumber></RefNumber>`) has changed. It used to select the next sequential reference number since the last one used by QuickBooks (as though no `<RefNumber>` had been provided). It now LEAVES IT BLANK. (For example, with CheckAdd requests, this would be important since you don't know the number until the check is printed.) This behavior applies to all transactions.
- The `<DateMacro>` element has a large number of possible values that were added in 3.0. See the *Onscreen Reference* for information on which values are valid for specific versions of the qbXML specification.
- Added IsActive to EmployeeMod.
- Added BudgetSummaryReportQuery request and response.
- Added enumerated values to IncludeColumn for reports: PaidAmount, ProgressAmount, ProgressPercent, UserEdit, TaxTableVersion.
- Added BillableStatus element to JournalDebitLine and JournalCreditLine.
- Added Country element to HostRet and HostInfo (in Events) aggregates. Possible values are "US", "CA", "UK", and "AU". This value reflects which country the product is built for, so in the U.S. edition of QuickBooks 2004 release, only "US" is returned.
- The order of some enum values has changed. This will affect you only if you are using QBFC and you use numbers to set enum values; for example, if you use function calls such as SetValue(1) instead of SetValue(atAccountsPayable). We recommend against using numbers to set enum values. If your programming language makes it hard for you to use enum symbols such as atAccountsPayable, use the SetAsString function instead of SetValue—for example, SetAsString("AccountsPayable"). That way, your code will not be affected when the order of the enum values changes.
- If any of these columns
 - Employee
 - PayrollItemDetail
 - PayrollYtdDetail

is used in the General Summary Report Query in the "Summarize Columns By" element and the application is not granted access to personal data, status code 3261 is returned, with the status message, "The integrated application has no permission to access personal data. The QuickBooks administrator can grant permission to access personal data through the

Integrated Application preferences.

- Where applicable, status code 3000 (“Invalid object ID”) has been replaced by status code 3150 (“Missing required element”).
- In the Employee Add, Modify, and Query requests, the military state element is now supported. Values are AA, AE, and AP.
- The ServiceDate field in Invoice and Purchase Order requests can now be cleared as well as modified.

If You Have Questions . . .

For IDN support, please visit <http://developer.intuit.com/support>.

For community support, visit <http://idnforums.intuit.com>.

Summary of New Features

This summary is divided into two parts. The first part provides an overview of the changes to QuickBooks SDK in Release 3.0. The second part provides a detailed listing of changes at the message and element level.

Part One: Overview of Changes in SDK 3.0

Several highlights of this release have already been described in these Release Notes. These highlights include

- A new Request Processor, QBXMLRP2, which can be redistributed along with your application
- A unified QuickBooks Foundation Class (QBFC) Library that supports both US and Canadian editions of QuickBooks as well as QuickBooks Online Edition
- An enhanced Remote Data Sharing (RDS) Library that supports automatic login.

The following paragraphs describe additional highlights of this release.

Event Notification

Your application can now subscribe to QuickBooks data, UI, and UI extension events so that it can be notified when they occur within QuickBooks. Data events occur when the QuickBooks user (or an integrated application) adds, modifies, or deletes data. UI events are triggered when the company file is opened or closed. UI extension events are triggered when the user selects a menu item added by your application. (For more information, see Chapter 5, "Event Notification," in the *Concepts Manual*.)

UI Integration with QuickBooks

This SDK release allows your application to control the QuickBooks user interface in a number of ways. First, your application can now add a menu item (with accompanying submenu items) to the QuickBooks UI. Second, your application can now display reports inside QuickBooks. Third, your application can also (1) display a form to create a new list element or transaction or (2) display a form for editing an existing list element or transaction. (For more information, see Chapter 6, "Integrating with the QuickBooks UI," in the *Concepts Manual*.)

Custom Fields and Data Extensions

You can now add, modify, and delete custom fields using the SDK. These fields are visible both from your application and from within QuickBooks. (Custom fields are implemented using the existing DataExt- requests.)

The DataExtAddRq has been modified to use macros so that you can add data extensions to new transactions in the original request message set--your application doesn't need to wait for the Add response before adding data extension values.

Reports

This release adds eight payroll reports (available if your application is accessing a company file that is currently signed up for a subscription to a payroll service). It also adds three new filters (Detail Level, Posting Status, and Modify Date) as well as two sales order reports (OpenSalesOrderByCustomer and OpenSalesOrderByItem, both in the General Detail report category) and five budget summary reports (BalanceSheetBudgetOverview, BalanceSheetBudgetVsActual, ProfitAndLossBudgetOverview, ProfitAndLossBudgetPerformance, ProfitAndLossBudgetVsActual).

Expanded Data Access

As promised, we've expanded the ability to modify transactions. With SDK 3.0, you can modify bills, statement charges, credit memos, estimates, and sales orders, in addition to the invoice and purchase order support added in SDK 2.1. Plus, your application can now add, modify, and query customer credit card information (provided the QuickBooks administrative user has granted the necessary permission to access personal data).

This release adds a new list item, ItemFixedAsset (Add, Query, Modify, and Delete). It also adds Modify support for ItemDiscount, ItemGroup, ItemInventoryAssembly, ItemOtherCharge, ItemPayment, ItemSalesTax, ItemSalesTaxGroup, ItemSubtotal, and SalesRep list items.

Your application can now query for templates and use them when adding or modifying transactions in QuickBooks. (This feature is available for invoices, estimates, sales orders, sales receipts, credit memos, and purchase orders.)

This release supports the ability to modify most entity elements (applies to customers, employees, vendors, and other names; especially see VendorMod and CustomerMod). It also adds the PayrollItemNonWage query (requires subscription to a payroll service).

Part Two: Details of Changes in SDK 3.0

The following sections contain details on new request and response messages as well as on new and changed elements within response aggregates. See the updated *Onscreen Reference* for complete listings of all messages. Also see the example file *qbxmlops30.xml*.

Table 1: New Requests/Responses in SDK 3.0

Request	Category
ItemDiscountMod	List
ItemFixedAssetAdd	List
ItemFixedAssetMod	List
ItemFixedAssetQuery	List
ItemGroupMod	List
ItemInventoryAssemblyMod	List
ItemOtherChargeMod	List
ItemPaymentMod	List
ItemSalesTaxGroupMod	List
ItemSalesTaxMod	List
ItemSubtotalMod	List
PayrollItemNonWageQuery	List
SalesRepMod	List
TemplateQuery	List
BillMod	Transaction
ChargeMod	Transaction
CreditMemoMod	Transaction
EstimateMod	Transaction
ItemReceiptQuery	Transaction
SalesOrderMod	Transaction
ListDisplayAdd	UI Integration
ListDisplayMod	UI Integration
TxnDisplayAdd	UI Integration
TxnDisplayMod	UI Integration
+ new message set for QBXMLSubscription (see <i>Onscreen Reference</i> for details)	Subscription

+ DataEventRecoveryInfoDel and DataEventRecoveryInfoQuery (added to QBXML message set)	Events
--	--------

Table 2: Elements Added to Requests in SDK 3.0

Request	Element Added	Category
CustomerAdd	PreferredPaymentMethodRef CreditCardInfo Notes	List
CustomerMod	IsActive ParentRef CustomerTypeRef TermsRef SalesRepRef SalesTaxCodeRef ItemSalesTaxRef ResaleNumber CreditLimit PreferredPaymentMethodRef CreditCardInfo (protected – that is, requires access to personal data granted by the QuickBooks administrative user) JobStatus JobStartDate JobProjectedEndDate JobEndDate JobDesc JobTypeRef Notes IsStatementWithParent (QuickBooks Online Edition only) DeliveryMethod (QuickBooks Online Edition only)	List

Table 2 (cont'd): Elements Added to Requests in SDK 3.0		
EmployeeAdd	Notes UseTimeDataToCreate- Paychecks (inside EmployeePayrollInfo)	List
EmployeeMod	IsActive Notes UseTimeDataToCreate- Paychecks (inside EmployeePayrollInfoMod; EmployeePayrollInfoMod changed to protected)	List
OtherNameAdd	Notes	List
OtherNameMod	Notes	List
VendorAdd	Notes	List
VendorMod	IsActive Notes VendorTypeRef TermsRef CreditLimit VendorTaxIdent IsVendorEligibleFor1099	List
BillPaymentCheckAdd	Memo	Transaction
BillPaymentCreditCardAdd	Memo	Transaction
ChargeAdd	OverrideItemAccountRef	Transaction
ChargeQuery	IncludeLinkedTxns	Transaction
CreditMemoAdd	TemplateRef	Transaction
EstimateAdd	IsActive	Transaction
EstimateAdd	TemplateRef	Transaction
EstimateQuery	IncludeLinkedTxns	Transaction
InvoiceAdd InvoiceMod	TemplateRef	Transaction
JournalEntryAdd	IsAdjustment BillableStatus (inside JournalCreditLine and JournalDebitLine)	Transaction

Table 2 (cont'd): Elements Added to Requests in SDK 3.0		
PurchaseOrderAdd PurchaseOrderMod	TemplateRef	Transaction
PurchaseOrderQuery	IncludeLinkedTxns	Transaction
SalesOrderAdd	TemplateRef	Transaction
SalesReceiptAdd	TemplateRef	Transaction
DataExtAdd DataExtDel DataExtMod	TxnLineID	Data Extensions

Table 3: Elements Added to Responses and Response Aggregates in SDK 3.0

Response/Resp. Aggregate	Element Added	Category
CustomerRet	PreferredPaymentMethodRef CreditCardInfo (protected)	List
EmployeePayrollInfo	UseTimeDataToCreatePaychecks (inside EmployeePayrollInfo or EmployeePayrollInfoMod)	List
ItemQueryRs	ItemFixedAssetRet	List
OtherNameRet	Notes	List
ChargeRet	IsPaid LinkedTxn	Transaction
CreditMemoRet	TemplateRef TxnLineID (inside CreditMemoLineGroupRet)	Transaction
EstimateRet	IsActive LinkedTxn TemplateRef TxnLineID (inside EstimateLineGroupRet)	Transaction
InvoiceRet	TemplateRef	Transaction
ItemGroupLineRet	TxnLineID	Transaction

JournalEntryRet	IsAdjustment BillableStatus (inside JournalCreditLine and JournalDebitLine)	Transaction
LinkedTxn	LinkType	Transaction
PurchaseOrderRet	TemplateRef LinkedTxn	Transaction
SalesOrderRet	TemplateRef TxnLineID (inside SalesOrderLineGroupRet)	Transaction
SalesReceiptRet	TemplateRef TxnLineID (inside SalesReceiptLineGroupRet)	Transaction
HostRet	Country IsAutomaticLogin QBFileMod	Other
CompanyActivityRet	LastCondenseTime	Other
CompanyRet	IsSampleCompany	Other

Table 4: Enumeration Changes in SDK 3.0

Enum Type	Enum Value Added	Category
ListDelType	ItemFixedAsset	List
AssignToObject	ItemReceipt	Data Extensions
RemoveFromObject	ItemReceipt	Data Extensions
TxnDataExtType	ItemReceipt	Data Extensions
TxnType	ItemReceipt	Transaction
TxnDelType	ItemReceipt SalesTaxPaymentCheck	Transaction
TxnVoid	ItemReceipt	Transaction

Table 4 cont'd: Enumeration Changes in SDK 3.0

DateMacro	All, Today, ThisWeek, ThisMonth, ThisCalendarQuarter, ThisCalendarQuarterToDate, ThisFiscalQuarter, ThisCalendarYear, ThisCalendarYearToDate, ThisFiscalYear, Yesterday, LastWeekToDate, LastMonthToDate, LastCalendarQuarter, LastCalendarQuarterToDate, LastFiscalQuarterToDate, LastCalendarYear, LastCalendarYearToDate, LastFiscalYearToDate, NextWeek, NextFourWeeks, NextMonth, NextCalendarQuarter, NextCalendarYear, NextFiscalQuarter, NextFiscalYear	
-----------	---	--

Table 5: Summary of Changes to Reports in SDK 3.0

Element Added	Report or Filter Type
BudgetSummaryReport	new report type; includes 5 reports: BalanceSheetBudgetOverview BalanceSheetBudgetVsActual ProfitAndLossBudgetOverview ProfitAndLossBudgetPerformance ProfitAndLossBudgetVsActual
PayrollDetailReport	new report type; includes 5 reports: Payroll Item Detail Payroll Transaction Detail Payroll Detail Review Payroll Transactions by Payee Employee State Taxes Detail
PayrollSummaryReport	new report type; includes 3 reports: Payroll Summary Employee Earnings Summary Payroll Liability Balances
OpenSalesOrderByCustomer OpenSalesOrderByItem	new enum value for GeneralDetailReportType enumeration type
DisplayReport element	all reports
ReportDetailLevelFilter ReportModifiedDateRangeFilter <i>or</i> ReportModifiedDateRangeMacro ReportPostingStatusFilter	filters added to all report queries (except TimeReportQuery and BudgetSummaryReport)
FixedAsset (this is an enumeration)	ItemTypeFilter (in ReportItemFilter)

New enumerated values for ColType	TaxTableVersion UserEdit IsAdjustment Invoiced
New enumerated values for IncludeColumn	Invoiced PaidAmount ProgressAmount ProgressPercent UserEdit TaxTableVersion