



intelligent accounting online

accountsIQ Integration Layer - Overview

accountsIQ (AIQ) provides a full **Application Programming Interface (API)** to enable the secure update of data from and to external 3rd party applications.

The API provides the best possible toolset to enable seamless integration with a variety of other applications such as webshops, CRM systems, EPOS, payroll systems, stock control and so on.

Using the AIQ API eliminates the need for time consuming re-keying of data across multiple systems.

If you're a developer, the AIQ API enables you to link your application seamlessly with AIQ thus widening the potential user base for your product. You can quickly register with us to become an Integration Partner. Partners can send or read data to/from accountsIQ using a predefined set of methods. It is possible for partners to send data to multiple customer databases on the accountsiq platform.

This document provides the following

- [Getting started with the API](#)
- [Explanation of the key API functions available and how to use them](#)
- [Case studies describing how the API can be used in practice](#)
- [How to Guide –Detailed explanation of an approach to integrating an EPOS system with AIQ](#)

Getting started with the API

Firstly you need to register with us (send an email to development@accountsIQ.com) and we can provide you with access to a test account to try out the API.

A full set of documentation for the API is provided [here](#). This document describes the various API calls that are available together with usage examples in C# and VB, the underlying data structures and terms and conditions governing the usage of the API.

All testing is carried out on our staging environment providing you with a safe area to carry out integration testing with your chosen application.

Authentication

Once you have registered with us and have been provided with a test account, we provide an encrypted partner key. A customer key identifying the specific database(s) can be created from within the application.

This key can be obtained from the Integration tab on the Company Information screen. Each transfer of data requires these keys to be submitted as part of the request.

Company Information

Information Integration

accountsIQ provides a two-way integration with several third party systems. When the integration link is set up for your user account, you will need to request a user key (below) and provide it to the third party software.

For more information, please contact support@accountsIQ.com or your third party vendor.

Please type in your password:

You submit a HTTPS request using the required method, supplying the relevant parameters. The API validates this request and responds with either a success or a failure notification. If successful, the data is immediately imported into the tables within the relevant customer database. A failure message indicating the cause is supplied if the validation fails.

In addition, you can use methods within the API to extract data from accountsIQ for display or reporting within your own application. This is useful if you need to display stock or customer balance data for example within another application.

API Methods

Most of the core datasets within accountsIQ are exposed through the API. The choice of which methods to use is largely down to the type of integration you are attempting to carry out.

Please use the API documentation to learn about how each integration method works. As an introduction, the following table summarises some of the most important methods and how they can be used.

API Method	
<p>Create/Update/Get Customers</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'UpdateCustomer' ▪ 'GetCustomer' ▪ 'GetNewCustomerFromDefaults' 	<p>Create or update customer records in the system directly from another application (e.g. CRM system, webshop).</p> <p>These are stored within the main accounts table in AIQ and accessible from the customer maintenance functions.</p> <p>Use the 'UpdateCustomer' function for this purpose. This method has a parameter to enable you to either create a new customer or update an existing customer record depending on the requirement.</p> <p>The 'GetCustomer' record enables you to retrieve an existing customer record by submitting the customer code as a parameter.</p> <p>The GetNewCustomerFromDefaults returns a new customer pre-populated with the customer defaults values. The system maintains a set of default settings for each new customer created in the system. This is maintainable from the front-end.</p>
<p>Create/Update Sales Orders</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'GetNewSalesOrder' ▪ 'GetOrder' ▪ 'GetOrdersByCustomerCode' 	<p>Create or update sales order records that originate from another application such as a sales order processing system or website.</p> <p>The 'GetNewSalesOrder' function creates a new order for a given customer account. It fills the order defaulting the information with the customer information.</p> <p>Use 'GetOrdersByCustomerCode' to retrieve a list of orders by customer code.</p>

API Method	
<p>Create/Update Sales Deliveries</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'DeliverOrderLines' ▪ 'GetNewDeliveryLine' 	<p>Register the delivery of a selected order with full update to the stock module within accountsIQ.</p> <p>You can choose to dispatch stock from multiple locations and multiple racks (bins) within these locations.</p> <p>The 'DeliverOrderLines' method delivers partially/fully some or all of the lines of a given order.</p> <p>An order can be delivered in multiple parts, each one allowing the user to create a related invoice.</p> <p>Several deliveries which have not been invoiced yet can be merged together to create a single invoice.</p> <p>The 'GetNewDeliveryLine' function creates a delivery line based on an existing order line. This line can then be used during the partial/full delivery of an order.</p>
<p>Create/Update Product (Stock) Items</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'SaveStockItem' ▪ 'GetStockItem' ▪ 'GetStockItemList' 	<p>Add product items directly into the product tables within accountsIQ. Useful if your product catalogue is maintained within another system and you wish to keep it synchronised with accountsIQ.</p> <p>Use the 'SaveStockItem' function to update or create a new stock item in an entity.</p> <p>Use 'GetStockItem' to retrieve the details of a specific stock item by stock item code or 'GetStockItemList' to retrieve a list of all active and inactive stock items</p>
<p>Create/Update/Post Sales Invoices</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'GetNewSalesInvoice' ▪ 'SaveInvoice' ▪ 'PostInvoice' ▪ 'CreateBatchSalesInvoice' ▪ 'GetNewBatchSalesInvoice' 	<p>Create sales invoice records in the system directly from another application. This will post directly into the sales ledger updating the relevant accounts.</p> <p>There are two types of invoices in the system:</p> <ol style="list-style-type: none"> 1. Product Invoices: These are invoices with specific line items (products). Product invoices are initially saved with a 'Processed' or draft status. Once they have been confirmed they can be posted. Only posted invoices are updated to the sales ledger 2. Batch Invoices: These are invoices with no line items and can be used to invoice services etc. Batch invoices are

API Method	
	<p>posted automatically when they are saved</p> <p>The GetNewSalesInvoice creates a new product invoice for a given customer account. It fills the invoice defaulting the information with the related customer's information.</p> <p>The 'SaveInvoice' function saves a modified existing product invoice record back to the system.</p> <p>The 'PostInvoice' function posts an existing product invoice to the transaction table.</p> <p>'CreateBatchSalesInvoice' posts a batch invoice directly into the system.</p> <p>The postings are as follow:</p> <ul style="list-style-type: none"> • One Net line for each batch sales invoice line, hitting the specified GL account of each line • One VAT line for each batch sales invoice line, hitting the default Sales VAT control account (as specified in the Required account list) • One balancing control line per sales batch invoice hitting the debtors control specified on the customer account. <p>'GetNewBatchSalesInvoice' returns a new batch sales invoice record pre-filled with defaults for a given customer.</p>
<p>Create/Allocate Sales Receipts</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'Save Sales Receipt' ▪ 'Allocate Transactions' 	<p>Create sales receipts allocated to specific customer invoices. This will create the relevant postings to the customer account and company bank account</p> <p>The 'SaveSalesReceipt' function creates a receipt for the indicated customer.</p> <p>Use the 'AllocateTransactions' function to allocate two outstanding transactions together.</p>

API Method	
<p>Create/Update/Post Purchase Invoices</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'GetNewPurchaseInvoice' ▪ 'SaveInvoice' ▪ 'PostInvoice' ▪ 'CreateBatchPurchaseInvoice' ▪ 'GetNewBatchPurchaseInvoice' 	<p>Create purchase invoice records in the system directly from another application. This will post directly into the sales ledger updating the relevant supplier accounts.</p> <p>There are two types of invoices in the system:</p> <ol style="list-style-type: none"> 1. Product Invoices: These are invoices with specific line items (products). Product invoices are initially saved with a 'Processed' or draft status. Once they have been confirmed they can be posted. Only posted invoices are updated to the sales ledger 2. Batch Invoices: These are invoices with no line items and can be used to invoice services etc. Batch invoices are posted automatically when they are saved <p>The GetNewPurchaseInvoice creates a new product invoice for a given supplier account. It fills the invoice defaulting the information with the related supplier's information.</p> <p>The 'SaveInvoice' function saves a modified existing product invoice record back to the system.</p> <p>The 'PostInvoice' function posts an existing product invoice to the transaction table.</p> <p>'CreateBatchPurchaseInvoice' posts a batch invoice directly into the system.</p> <p>'GetNewBatchSalesInvoice' returns a new batch sales invoice record pre-filled with defaults for a given customer.</p>
<p>Create/Update a GL Journal</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'CreateGeneralJournal' 	<p>Create or update a General Ledger Journal transaction that originates from another system. This would create a movement between two or more general ledger accounts, e.g. the posting of a salary journal from a payroll system.</p> <p>Use the 'CreateGeneralJournal' for this purpose.</p>
<p>Extract a list of GL Accounts</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'GetGLAccountList' 	<p>Use the 'GetGLAccountList' to obtain a listing of General Ledger Accounts directly from the system. Useful for posting a journal into accountsIQ from within another application</p>

API Method	
<p>Extract a list of Departments</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'GetDepartmentList' 	<p>Use the 'GetDepartmentList' to obtain a listing of departments/tracking codes directly from the system. Useful for posting a journal into accountsIQ from within another application</p>
<p>Extract a list of Customer Statement Details</p> <p>Related Methods:</p> <ul style="list-style-type: none"> ▪ 'GetCustomersStatement' 	<p>Use the 'GetCustomersStatement' function to extract a full list of customer statement data for a selected set of customers. Useful for enabling the display of customer statement data within another application</p>

Integration Case Studies

The following provides a number of real-life examples of integrations carried out using the accountsIQ API:

Case Studies	
Field Sales PDA Access	<p>Our API enables the field sales operatives of a finished goods merchandiser to create sales orders, make deliveries, invoice customers, and process customer payments using standard handheld PDAs. All data is seamlessly updated into accountsIQ from this mobile application in real-time.</p> <p>Operatives can make stock enquiries, look up customer account details, and sell more products on the road making them much more effective and reducing administrative overhead that would otherwise be required.</p>
Electronic products website pushing product sales data into accountsIQ	<p>The API enables this e-commerce web-shop to be fully integrated with AIQ. Buying customers register on the website and their details are sent directly into AIQ. When they make a product purchase, a sales invoice is created in AIQ updating the sales ledger in the process.</p> <p>Once the online payment is processed, a sales receipt for the value of the credit card purchase is then created and allocated against the relevant customer invoice, updating the sales ledger and company bank account straightaway</p>
Charity Website taking donations	<p>The API enables donations from a charity website to be automatically updated into AIQ. People making donations are recorded as new customers and their credit card donations are recorded as sales invoices and sales receipts in the system</p>
Catering company updating invoices from their Sales & Purchase Order Processing System	<p>This catering company processes thousands of invoices per week using their in-house Sales Order & Purchase Processing system.</p> <p>They use the API to send across these records directly into accountsIQ thereby eliminating the need for any re-keying of data</p>
Time Recording & Billing with an Accounting Practice	<p>We integrated AIQ with the chosen time recording and billing system used within an accounting practice.</p> <p>Fee notes/invoices for client jobs created within the time recording system are updated to accountsIQ together with client and job information</p>
Payroll Integration	<p>We've collaborated with an online payroll system supplier to enable integration with accountsIQ. Payroll information generated in the payroll system can be</p>

Case Studies

imported directly into accountsIQ by creating a general ledger journal that posts to the relevant accounts. This is achieved by means of mapping the pay codes in the payroll system to the GL codes in AIQ.

This helps ensure that the two systems remain synchronised and eliminates any errors that could arise from the re-keying of data

Next Steps

To get started with integration contact our Support and Implementation team (support@accountsIQ.com).

We will provide you with a full set of documentation on the functions available and help with any questions.

We can then provide access to a staging environment where test integrations can be tried out.

How to Guide: Integration between AccountsIQ and an EPOS system

The following table sets out the potential integration points between an EPOS system and AccountsIQ.

The exact level of integration will depend on the functionality within the EPOS, where relevant functions like Stock Control and Purchase Ordering will take place and what level of reporting and analysis is required within accountsIQ.

The table below gives a guide to the potential integration between EPOS and accountsIQ and which Webservice component in accountsIQ should be used.

Transaction	Description
Sales – <i>without</i> product detail	<p>A summary of the sales values by department (analysis codes) from the EPOS can be updated to accountsIQ via the 'GetNewBatchInvoice' function.</p> <p>These invoices are posted to an EPOS account setup in the accountsIQ Debtors Ledger. One overall EPOS account can be set up or alternatively you can set up one for each till or even each employee (assuming the details can be extracted from the EPOS at that level).</p> <p>The invoice can be a daily or weekly summary depending on the level of detail you would like to be able to drill down to within accountsIQ.</p> <p>We recommend a daily summary invoice. This will include a breakdown of the net sales by department (setup as analysis codes) and by VAT code.</p> <p>The till or employee code linked in the EPOS to each sale can be used to create a separate summary invoice for each till/employee if desired which in turn is linked to a specific Debtors account code for that till/employee.</p>
Sales – <i>with</i> product detail	<p>If the ability to produce sales analysis by product/product group is required within accountsIQ or if stock control is to be maintained in accountsIQ, then 'GetNewSalesInvoice' method can be used.</p> <p>In this case daily/weekly summary invoices as above can be posted but with a breakdown per product of the sales for each product for the day/week and till/employee for each invoice posted.</p> <p>Stock control should be maintained either in the EPOS system or in accountsIQ but not in both. This will largely depend on whether the EPOS has stock control capability and whether most stock movement is via EPOS – i.e.: if sales are all through EPOS then the EPOS is the best place to control stock.</p> <p>If there are multiple other sources of movement, e.g.: Sales via website, then stock is better controlled within accountsIQ. If controlled within the EPOS, maintenance of stock balances should be turned off in accountsIQ and product breakdown in accountsIQ is then only used for sales analysis purposes.</p>

Transaction	Description
Cash Summary	<p>To balance off the relevant EPOS Debtors accounts, a cash summary posting needs to be made of the cash that went through the till. This will usually be one summary per day/week by type (e.g.: cash, cheques, credit cards, vouchers etc). The 'SaveSalesReceipt' function can be used to post these to the relevant EPOS Debtors accounts.</p> <p>The bank account is automatically updated at the same time so it is important that these can be reconciled to lodgements by linking to a lodgement reference.</p> <p>The accountsIQ auto bank reconciliation function will pick up easily if it can match the reference and amount.</p> <p>Multiple debtors' receipts can be linked to one lodgement. If there is a facility in the EPOS to create a lodgement then it might be preferable to post the summary cash over based on each lodgement. Alternatively the next lodgement reference could be used and reset once a lodgement has been done.</p> <p>Vouchers can be difficult to deal with from an accounting point of view. Strictly speaking you want to be able to deal with the sale as a normal sale (although it may be linked to a promotion to determine success of the promotion) and the voucher is handled as a receipt. A voucher control account can be setup in accountsIQ to track all vouchers and this can be "washed out" or journalled as a promotional expense from there.</p>
Adjustments	<p>Differences inevitably arise between the Sales that went through the EPOS and the cash that has been reconciled from the till and lodged to the bank. The discrepancies can be handled via journal adjustments in the EPOS account in the accountsIQ Sales Ledger.</p> <p>However, some EPOS systems allow you to record adjustments to account for such differences. These adjustments can be posted to the relevant EPOS Debtors accounts using the Debtors Journal Webservice. Adjustments can arise for many different reasons and often a reason code is used in the EPOS which can be linked to a GL code used in accountsIQ. For example:</p> <ul style="list-style-type: none"> ➤ Robbery ➤ Money used as petty cash ➤ Money used to pay sundry expenses ➤ Discrepancies that arose on end of day till reconciliation (sometimes linked to specific tills or employees) <p>These will effectively be posted as journal adjustments to the EPOS account in the accountsIQ sales ledger and should help to balance the Sales postings with the cash postings to allow them to be fully allocated.</p>
Synchronise Product files	<p>It is important that the product and pricing details are controlled from either the EPOS or accountsIQ but again not both. If maintained within the EPOS then the product details in accountsIQ need to be synchronised so that the product detail Sales invoices can be updated to accountsIQ as above.</p> <p>In this case the 'SaveStockItem' can be used to send product/price updates to the accountsIQ system.</p>

Transaction	Description
	<p>If the product file is to be maintained in accountsIQ (usually if other sources of sales other than EPOS or if multiple outlets/franchisees to be controlled from a centrally maintained product file). In this case the xxxx Webservice can be used to request an update from accountsIQ</p>
<p>Statistics</p>	<p>For the design of good Key Performance Indicators (KPIs) in accountsIQ, user defined statistics can be maintained by analysis code (e.g.: department).</p> <p>Often the EPOS system stores and can produce these stats by day or week and these can be used to update the relevant user defined statistics in accountsIQ using the Statistics Update Webservice. These would include stats like;</p> <ul style="list-style-type: none"> ➤ Number of customers ➤ Number of items sold ➤ Number of staff hours or full-time Equivalents (FTEs) ➤ Number of complaints or refunds
<p>Suggested Purchase Order</p>	<p>Some EPOS systems, particularly where stock is maintained within the EPOS allow the user to raise a Purchase Order or to suggest a Purchase Order for product that is hitting set minimum levels.</p> <p>These POs or PO requests can be posted to the accountsIQ system for processing using the GetNewPurchaseOrder function. This is particularly applicable where multiple sites are involved, particularly in a franchise environment where the franchisor supplies the product.</p> <p>The PO is posted to the franchise copy of accountsIQ and is automatically updated to the franchisor books as a Sales Order to be fulfilled.</p>
<p>Goods Received</p>	<p>Once goods have been received into the EPOS (i.e.: where stock control is done from the EPOS) this information can be passed over to accountsIQ for matching against the invoice from the supplier.</p> <p>The goods received are stored against the related Purchase Order in accountsIQ which is then matched against the invoice when received from the Supplier or Franchisor. The 'DeliverOrderLines' function can be used for this purpose.</p>