

Microsoft Dynamics® C5

Factsheet on OIOUBL

Setup, user and technical guide

TABLE OF CONTENTS

Introduction.....	3
Setting up OIOUBL	4
XML schema.....	4
Setting up XML conversion.....	8
Setting up Actions	9
Setting up action lists for customers and vendors	10
Specifying EAN numbers for customers and vendors	11
Specifying VAT number for customers and vendors	11
Specifying which customers and vendors can accept incoming OIOUBL documents.....	11
User guide.....	12
Outgoing XML files	12
Incoming XML files.....	12
XML Inbox.....	12
XML conversion (automatic update).....	16
Further processing of sales/purchase orders created by OIOUBL import	17
Requirements for extended OIOUBL functionality	19
OIOUBL – known restrictions.....	20
Online validation of OIOUBL documents	21
Technical guide	22
known restrictions.....	35

INTRODUCTION

In connection with the new requirements laid down in the "Statute on information in the OIOUBL Electronic Invoice for use with invoicing of public sector organisations" a solution that makes it possible to generate outgoing electronic documents in OIOUBL format has been deployed in Microsoft Dynamics C5 2010.

In Microsoft Dynamics C5 2010 Service Pack 1 OIOUBL functionality is extended to be able to import documents from both customers and suppliers.

The following documents are released:

Document	Export	Import
Sales invoice	x	
Sales credit note	x	
Project invoice	x	
Project credit note	x	
Reminder	x	
Purchase order (export of purchase requisition)	x	
Sales order (import of purchase requisition)		x
Sales order (export of order confirmation)	x	
Purchase order (import of order response)		x
Purchase order (import of sales invoice)		x

Microsoft Dynamics C5 provides a standard OIOUBL Setup, which is based on OIOUBL schemas published by IT og Telestyrelsen. The default mapping between schemas and database in Microsoft Dynamics C5 is based on standard business logic.

The instructions in this document are based on a default installation of Microsoft Dynamics C5 where the data folder is located in **C:\ProgramData\Microsoft Dynamics C5 <version> Data**.

SETTING UP OIOUBL

Before you can use OIOUBL you must do the following:

1. Download schemas
2. Manually download Presentation style sheets and Schematron validation files (optional)
3. Import the mapping setup supplied with Microsoft Dynamics C5
4. Manually link Presentation style sheets and Schematron validation files to the schemas if they have been downloaded in step 2.
5. Manually set up actions (OIOUBL actions are not set up on customers or vendors action lists automatically)
6. Specify EAN numbers and VAT numbers for relevant Customers/Vendors
7. Set up data conversion
8. Specify which customers/vendors can accept incoming OIOUBL documents

These steps are described in detail below.

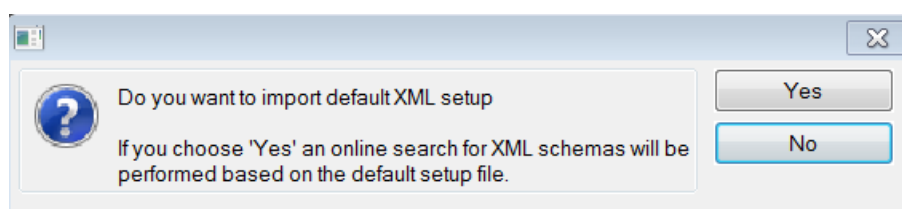
XML SCHEMA

The setup and mapping of existing schemas is performed in **General/Table/XML/XML Schema**.

Microsoft Dynamics C5 provides a standard setup for the selected OIOUBL version 2.0 schemas.

When you access the menu item **XML schema** and no XML schemas are set up yet, you can run an automatic standard setup that downloads XSD files (XML schema files), sets up the paths and loads mappings for the schema files.

The following dialog appears and if "Yes" is selected XSD files will automatically be downloaded from <http://rep.oio.dk/ubl/xml/schemas/2.0/maindoc/>.



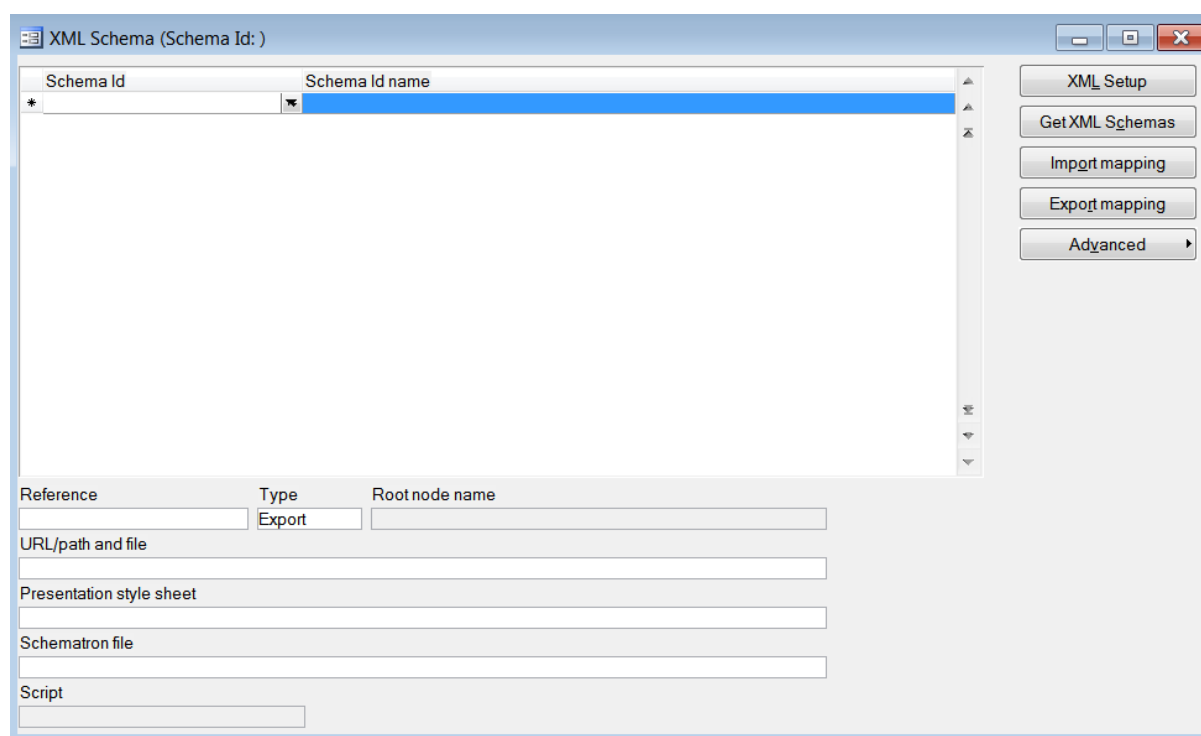
The files to be downloaded are defined in the file **OIOUBL. Schemas. c5c**, which is placed in the folder **C:\ProgramData\Microsoft Dynamics C5 <version> Data\Country\DK**

Two default paths are set up in the **XML Setup**: the path to the folder with the downloaded schema files and the path to the folder where all incoming XML document files must be placed in order to be imported.

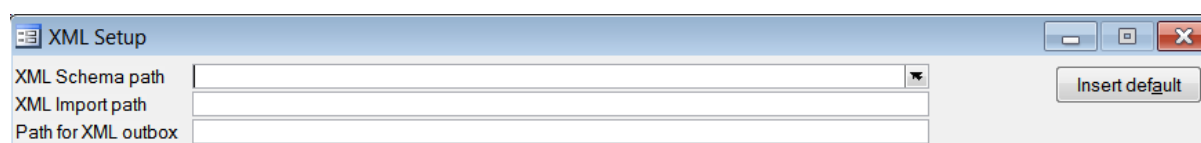
Both folders are created in the directory where C5 is installed. The folder with the schema files is by default named **XSDCache**. The folder for incoming XML files is by default named **<Database extension>_XMLFileImport**, e.g. **DAT_XMLFileImport**.

It should be noted that only Supervisor can perform this setup and that internet access is required during the download of schemas.

If you declined the automatic setup, all setup in the **XML Schema** window must be performed manually.



The menu item-XML Setup



XML Schema Path is the path to the folder where all schema files are located, schemas can be downloaded automatically — see more about this in the section **Get XML schemas**.

XML import path is the path to the folder where all incoming XML document files are collected in order to be imported.

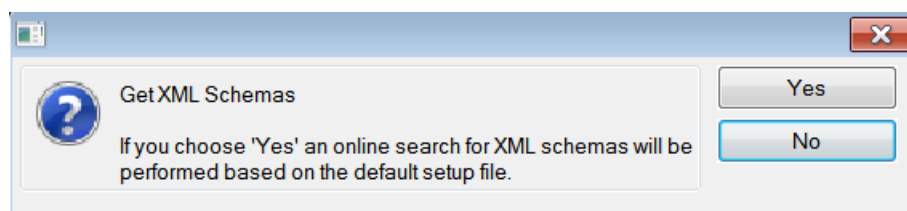
The menu item **Insert default** inserts the default values as described in the previous section. The default folders will then be created automatically.

NB - if the XML Import path is changed after some XML documents have already been imported, a warning about lost references to these documents will appear as the document management system does not save import paths with the saved documents, but dynamically retrieves the current path from the XML Setup.

In Microsoft Dynamics C5 version 2012 HF 1 and newer versions it is furthermore possible to specify a path to a folder where all outgoing XML files of all types must be collected together, for example, in connection with a centralised dispatch of files. This path can be specified in the **Path for XML outbox** field. If this path is specified in the XML Setup window a copy of all outgoing XML files that are generated in the current data file will be saved in the specified folder.

The menu item – Get XML schemas

This menu item activates the following prompt window:



To proceed with the schema download you must agree to continue and select the default schema download definition file in the Windows Explorer window that opens immediately after the prompt window.

The default schema download definition file **OIOUBL.Schemas.c5s** contains internet links to the supported OIOUBL schema files and is located under:

C:\ProgramData\Microsoft Dynamics C5 <version> Data\Country\DK

Download will start automatically from <http://rep.oio.dk/ubl/xml/schemas/2.0/common/> and from <http://rep.oio.dk/ubl/xml/schemas/2.0/maindoc/>

The schema files will be downloaded to a number of new subfolders that will be created automatically in the XML Schema path folder specified in the XML setup.

The manual schema download can be used to update the already downloaded schema files without affecting the related mappings.

The menu item – Import mapping

This menu item opens Windows Explorer where you must select an XML mapping file to be imported. The default XML mapping file is named **OIOUBL.Mapping.c5m** and is located under:

C:\ProgramData\Microsoft Dynamics C5 <version> Data\Country\DK

If the mapping import is called from an empty line in the XML Schema window, the mapping import proceeds automatically.

If the mapping import is called from an existing line in the XML Schema window, the mapping import script will prompt you to select one of the following options: Import mapping for the line's Schema Id only if it is available in the selected file or Import all mappings that are available in the selected file.

If a mapping being imported already exists you can choose to skip the import or to overwrite the existing mapping.

NB – The default mapping file supplied with Microsoft Dynamics C5 has been changed since the original version 2010 release and may also be changed in the future. It is therefore recommended to check if a newer version of the file is available and to manually update the mapping setup by overwriting all existing mappings. It is also recommended to back up the existing mappings prior to performing the update as all user modifications will be lost after the update.

The menu item – Export mapping

This menu item can only be called from an existing line in the XML Schema window. The menu opens Windows Explorer where you must select a location and specify a name for the file that will contain the mapping export.

The mapping export script will then prompt you to select one of the following options: Export mapping for the line's Schema Id only or Export all existing mappings.

After a mapping has been exported it can be imported using the **Import mapping** menu item.

Schematron and Presentation style sheets files

Presentation style sheet files that are used to display the contents of XML files in a more user-friendly layout are not included in the Schema download, but they can be downloaded manually from this link:

<http://www.oioubl.info/oioubltools/da/oioubltools.html> - Stylingpakke (new)

It is recommended to create a new sub folder "Stylesheets" in the folder with XML Schema files and to download the style sheets to this folder.

After the style sheets have been downloaded you will need to link them to appropriate schemas. This is done manually by specifying an appropriate style sheet file for a Schema in the field Presentation style sheet.

Schematron files that are used to validate the contents of XML files are not included in the Schema download either; they can be downloaded manually from this link:

<http://www.oioubl.info/oioubltools/da/oioubltools.html> - Validationpackage with testeksamples (new)

It is recommended to create a new sub folder "Schematron" in the folder with XML Schema files and to download the Schematron files to this folder.

After the Schematron files have been downloaded you will need to link them to appropriate schemas. This is done manually by specifying an appropriate Schematron file for a Schema in the field Schematron file.

The table below shows the correlation between Schemas, Representation style sheets and Schematron files.

Schema Id	Presentation style sheet	Schematron file
CustomerReminderUBL	ReminderHTML.xsl	OIOUBL_Reminder_Schematron.xsl
Proj_CreditNoteUBL	CreditNoteHTML.xsl	OIOUBL_CreditNote_Schematron.xsl
Proj_InvoiceUBL	InvoiceHTML.xsl	OIOUBL_Invoice_Schematron.xsl
Purch_In_InvoiceUBL	InvoiceHTML.xsl	OIOUBL_Invoice_Schematron.xsl
Purch_In_OrdConfUBL	OrderResponseHTML.xsl	OIOUBL_OrderResponse_Schematron.xsl
Purch_OrderReqUBL	OrderHTML.xsl	OIOUBL_Order_schematron.xsl
Sales_CreditNoteUBL	CreditNoteHTML.xsl	OIOUBL_CreditNote_Schematron.xsl

Sales_InvoiceUBL	InvoiceHTML.xsl	OIOUBL_Invoice_Schematron.xsl
Sales_In_PurchReqUBL	OrderHTML.xsl	OIOUBL_Order_schematron.xsl
Sales_OrdConfirmUBL	OrderResponseHTML.xsl	OIOUBL_OrderResponse_Schematron.xsl

The menu item Advanced

See the technical guide.

Detailed description of fields in XML Schema

See the technical guide.

SETTING UP XML CONVERSION

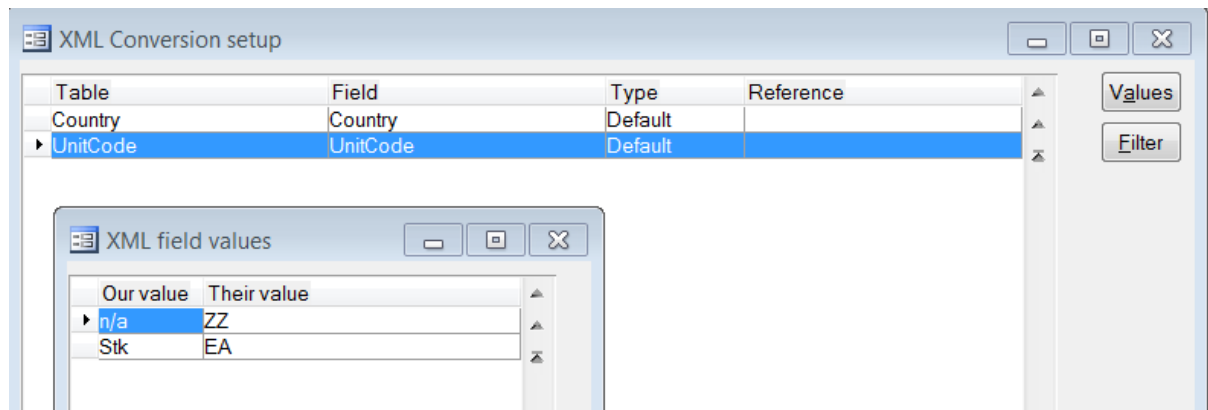
XML conversion may be necessary in case some values in C5 do not comply with OIOUBL requirements or if the company's base data entries (e.g. item numbers) are different from the OIOUBL customer's or vendor's.

XML conversion can be set up under **General/Table/XML/XML conversion**.

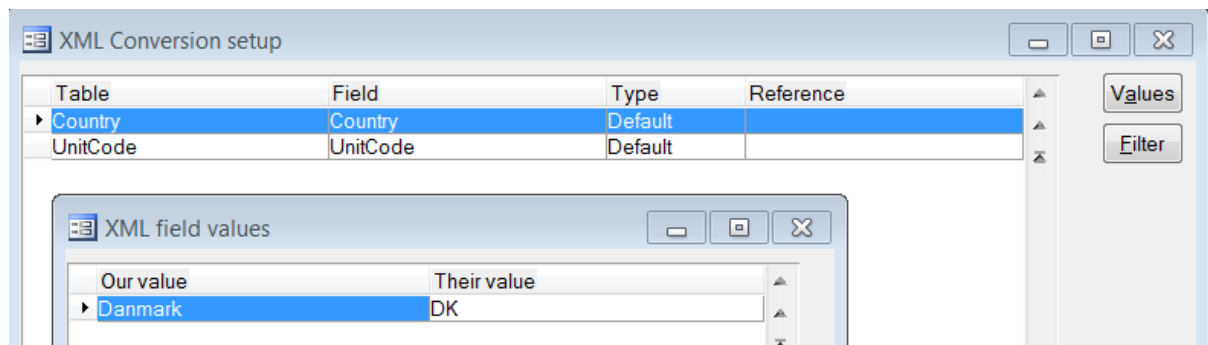
To specify what type of value must be converted - first select a table and then a field. Then specify conversion values using the menu item **Values**.

If a table is not selected you can, in the field **Field**, select an ENUM type whose values must be converted - for instance **Language_**. In **Values** select individual Enum value(s) and specify the conversion value.

Conversion of unit codes is necessary if the unit codes used in the C5 database do not comply with OIOUBL requirements, for example, it will be necessary to convert "*Stk*" or "*Pcs*" as they are specified as "*EA*" in OIOUBL files. A blank unit code must be converted to "*ZZ*".



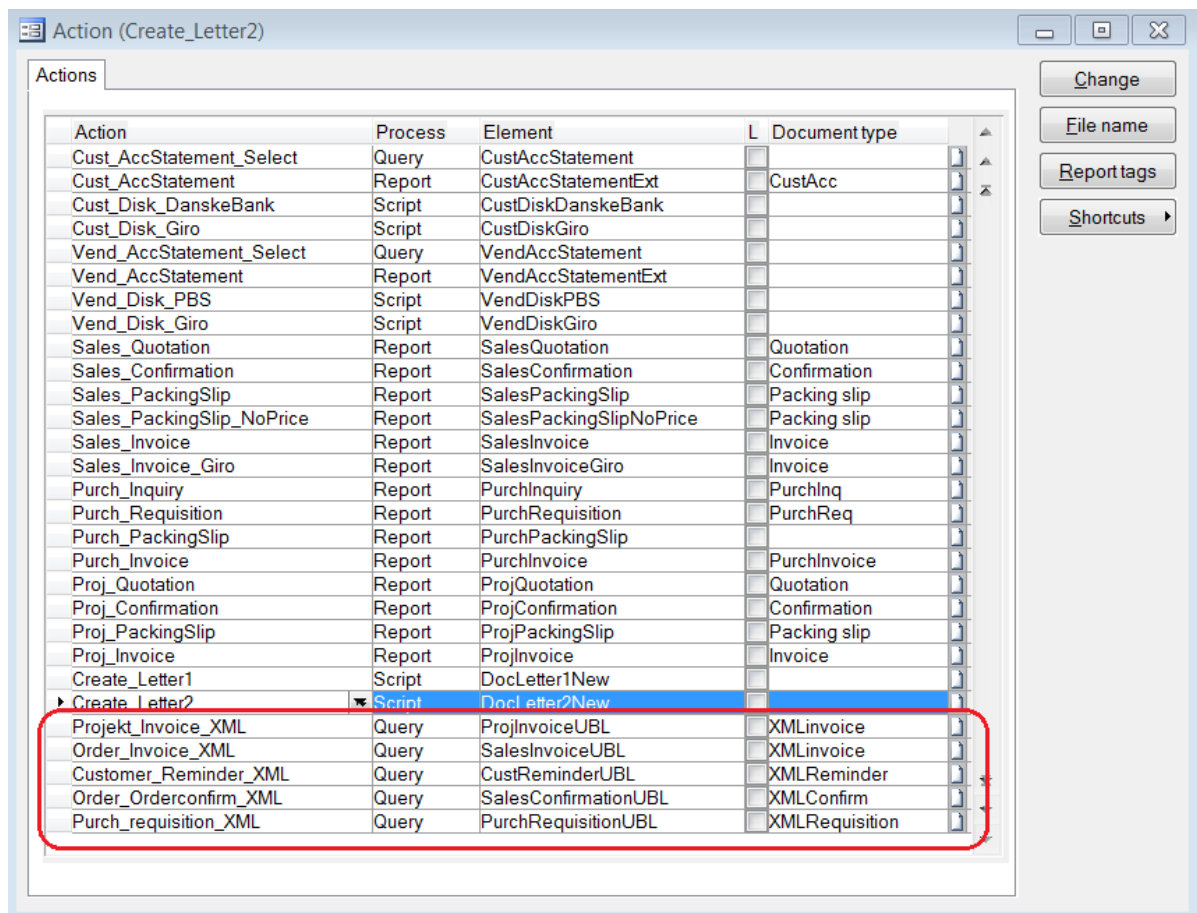
It is also necessary to convert Country names, for example, *Denmark* must be converted to *DK*



The Conversion table is also used to establish and store links between related documents, e.g. an incoming XML purchase requisition and a C5 sales order that has been created from this requisition.

SETTING UP ACTIONS

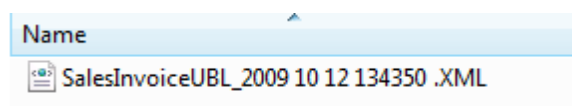
Under **General/Table/Actions** the following new actions are created automatically. These actions are used when generating and saving outgoing XML documents.



All outgoing XML documents follow the usual logic used in the Document management module: any outgoing XML document will be saved in a document type subfolder to the company's document management folder. If, for example, a DAT Company uses the Danish functionality and the default document management folder, all OIOUBL invoices will be saved in:

`\DocMgmt\DAT\XMLFaktura`. Default XML file names are composed as follows: <Name of the

script that has generated the file>_<Date as YYYY MM DD and Time as number of seconds elapsed since midnight when the document was created>, e.g.

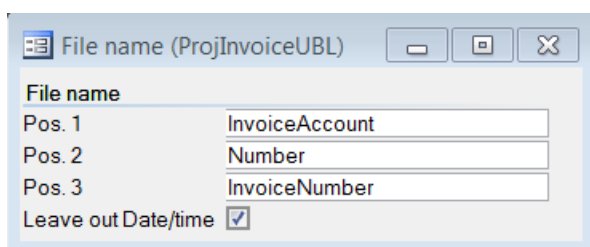


You can redefine the composition of file names using the local menu item **File name**.

The file name setup is stored with the element that generates the files. An element can have only one file name setup at a time.

All outgoing XML files in C5 are generated by Query elements (QTX). In the File name setup window it is possible to select up to 3 fields from the table that the element's query is based on and use the values of these fields as part of the file name. It is also possible to omit Date and Time specification in the file names.

It is important to ensure that the values of the selected fields together produce a unique file name for a document so that overwriting of other documents with the same file name is avoided.



The setup for OIOUBL Project invoice shown above and based on Customer invoice journal table will produce the following file name: <Invoice account> <Project number> <Invoice number>, e.g. 9999991 670 89. xml

SETTING UP ACTION LISTS FOR CUSTOMERS AND VENDORS

The output of all outgoing business documents such as invoice, reminder, order, confirmation and purchase requisition is controlled by action lists. This means that in order to be able to output a document you must set up an action associated with the document on an action list.

Action Lists in customer and vendor modules can be created on two levels:

- The default action lists are automatically applicable to all customers/vendors

- Local action lists are only applicable to the customer/vendor for whom the local action list is created and override the default action lists.

To add a new action to an action list select a system function that must trigger the document output and then select an action that is associated with the output of the desired document.

If, for example, a customer should receive invoices in OIOUBL format, a new Order_Invoice_XML action for SalesInvoice function must be added to the customer's action list.

Actions on customer (Account: 45823423)

Function	Action	L	O	Copy	Message	Document type
CustReminder	Customer_Reminder_XML	<input type="checkbox"/>	<input checked="" type="checkbox"/>			XMLReminder
SalesConfirmation	Order_Orderconfirm_XML	<input type="checkbox"/>	<input checked="" type="checkbox"/>			XMLConfirm
SalesInvoice	Order_Invoice_XML	<input type="checkbox"/>	<input checked="" type="checkbox"/>			XMLInvoice
* ProjInvoice	Projekt_Invoice_XML	<input type="checkbox"/>	<input checked="" type="checkbox"/>			XMLInvoice

Reset
Shortcuts

Actions on vendor (Account: 98992618)

Function	Action	L	O	Copy	Message	Document type
* PurchRequisition	Purch_requisition_XML	<input type="checkbox"/>	<input checked="" type="checkbox"/>			XMLRequisition

Reset
Shortcuts

SPECIFYING EAN NUMBERS FOR CUSTOMERS AND VENDORS

The OIOUBL format requires unique identification of the document's sender and recipient. EAN number is one of the accepted identification formats.

To set up an EAN number for a customer or a vendor you must create an EAN address in the customer's or vendor's address table: **Customer/Table/ (Menu) Setup/Addresses**.

It is possible to create multiple EAN addresses in the address table, however only the EAN number that is specified in the field **EAN Number** in the Customer table will be used as the default EAN Address for the customer.

Invoice

OIOXML ☐

EAN number 579xxxxxxxxxx

A/c dimension

SPECIFYING VAT NUMBER FOR CUSTOMERS AND VENDORS

VAT number specification is mandatory when using OIOUBL.

VAT number is specified in the field **VAT No** in the customer or vendor table. VAT numbers must be specified in numeric format. All characters for country identification must be omitted.

VAT No 11111111

XML import ☒

SPECIFYING WHICH CUSTOMERS AND VENDORS CAN ACCEPT INCOMING OIOUBL DOCUMENTS

In order to be able to receive and import OIOUBL documents from a particular customer or a vendor, the customer or the vendor must be marked as a valid XML sender.

VAT No 11111111

XML import ☒

Approve a customer or vendor as a valid XML sender by setting the check mark in the customer's or vendor's **XML import** field. As soon as a customer or a vendor has been approved it will be possible to import incoming OIOUBL documents that are sent by the customer or the vendor.

USER GUIDE

OUTGOING XML FILES

As already mentioned, all outgoing XML documents follow the usual logic used in the Document management module which means that the output of these documents like that of any other outgoing business document is triggered by action lists.

If the OIOUBL setup has been done correctly all OIOUBL documents will be generated automatically as soon as a relevant standard system function (such as invoicing, order confirmation etc.) has been activated.

If however the OIOUBL setup contains errors or omissions the generation of an OIOUBL document may be aborted or you may be prompted to specify missing mandatory information, e.g. missing VAT numbers.

Any outgoing XML document will be saved in a document type subfolder to the company's document management folder. If, for example, a DAT Company uses the Danish functionality and the default document management folder, all OIOUBL invoices will be saved in:
\DocMgmt\DAT\XMLFaktura.

If a **Path for XML outbox** has been set up, a copy of the document will be placed in the outbox folder. (This functionality is only available in version 2012 HF1 and newer versions)

When an OIOUBL document has been generated and before it is saved it is automatically validated against the document's schema and Schematron (if a Schematron file has been specified for the document's Schema).

If the document fails to validate it will be saved in a special folder that is created automatically in the company's document management folder. This folder is named **XML OUTGOING VALIDATION ERROR**.

Any outgoing OIOUBL document will also automatically be saved in the document management module and attached to the document's source record. An OIOUBL sales invoice made for a sales order, for example, will be saved with the invoiced archived sales order and can be accessed at any time from this archived order using Ctrl N.

NB Microsoft Dynamics C5 only provides the functionality for generating OIOUBL files. The dispatch of the generated files to the recipients is not a part of this functionality.

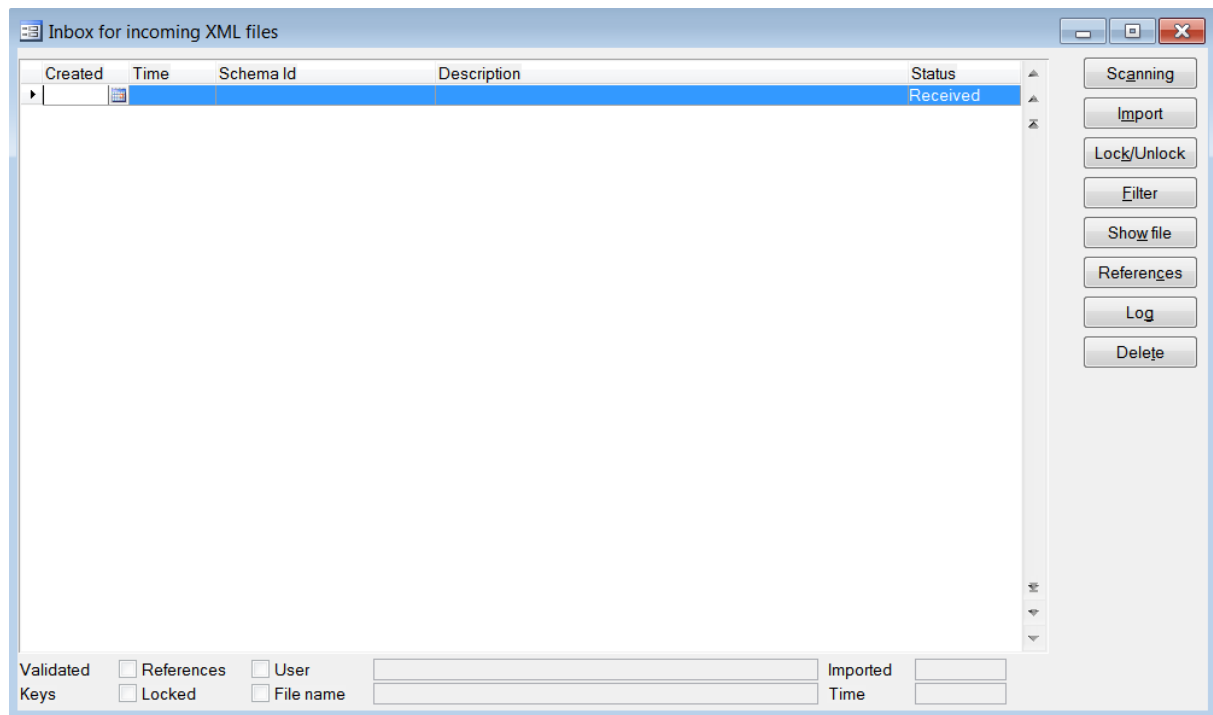
INCOMING XML FILES

Processing of all incoming XML documents of any type is done in the **XML inbox (General/Daily/XML Inbox)**.

All incoming XML files that can be imported into C5 must be collected in the folder that is specified in the **XML import path**.

XML INBOX

XML inbox is designed to be used by one user at a time. If the XML inbox is already in use when you try to access it you will receive a warning. It is then only recommended to proceed if you believe that the XML Inbox has been locked by mistake or if you just want to look at the contents of the XML inbox without performing any other actions.



The menu item - Scanning

This menu item is used to scan the XML import folder for new XML files or to rescan the already received XML files.

When called from a blank line the scanning proceeds automatically and only new files in the XML import folder will be scanned.

When called from an existing line the scanning script will prompt you to select one of the following options: To rescan the line's XML file or To scan the import folder for new XML files. Note that only lines with status *Received* can be rescanned.

Any new XML file found during the scanning is moved to the automatically created subfolder **RECEIVED** and a new line for the file with status *Received* is created in the XML Inbox.

When an XML file is scanned it is simultaneously validated as follows:

- Against the appropriate XML schema if the file contains a Schema reference. If the file has been validated with success the field **Validated** for the file's line is updated with a check mark, if the validation fails the field **Validated** is cleared.

If the file does not contain any Schema reference, any further validation is aborted and the file's line in the XML Inbox will be created without indication of Schema Id.

Any processing of files without Schema Id is not possible, you can however specify a Schema Id manually and then rescan the file.

- Against the Schematron file if it has been specified for the file's Schema. If the file has been validated with success the field **Validated** for the file's line is updated with a check mark, if the validation fails the field **Validated** is cleared.

- The key values according the file's Schema mapping are retrieved from the file and validated. The validation script attempts to find the values in the C5 database (after they have been converted if any conversion setup exists). If the key values have been found, the field **Keys** for the file's line is updated with a check mark, if the Keys have not been found the field Keys is cleared.

The existence of key values is crucial for the further processing of the file. For example, the unique Customer identification must be specified in the file and it must exist in the C5 database either directly in the Customer table or in the conversion setup before the file from the customer can be imported.

- The reference values according the file's Schema mapping are retrieved from the file and validated. The validation script attempts to find the values in the C5 database (after they have been converted if any conversion setup exists). If the reference values have been found, the field **References** for the file's line is updated with a check mark, if the references have not been found the field References is cleared.

The reference values are similar to the Key values but they are not crucial for the file import and can be corrected after the file has been imported. For example, an item number specified in the XML file may not exist in the C5 database, but an order for the item can still be created. This order cannot be invoiced before the item is created in the database, but this can be done during manual processing of the order.

All errors and discrepancies found during the scanning and validation are saved in the file's Scanning log that can be accessed using the local menu item **Log**. A scanning log may be divided into 5 different sections depending on the log's subject:

- **Technical log** contains general XML errors and errors found during Schema validation
- **Schematron** contains errors found during Schematron validation
- **Key fields** contains information about missing mandatory key fields
- **Application log** contains information about missing reference values
- **XML Import – comparison log** contains information about differences between file values and corresponding values in the C5 database found during file import. See the detailed description in the next section.

The menu item - Import

When an XML file has been successfully validated and all key values exist the file can be imported into the C5 database. Note also that only lines with status *Received* can be imported.

If some reference values have not been found during the scanning the import script will warn you about the missing references. If you agree to continue with the file import the XML Inbox line becomes locked with your user name until the file import is finished.

The import script will either create a new document in the C5 database (such as a Sales or a Purchase order) from the file's document or update a document in the C5 database (such as a Sales or a Purchase order) with any changes present in the file if the file contains reference to an already existing document in the C5 database. Note, that during import the data from an XML file may either be ignored or imported depending on the value of the **Import** field in the mapping setup for the particular XML data. (See Technical guide for further information)

All documents in the C5 database are first of all created according to the standard application business logic and not entirely from the file's data as it may not be directly compatible with the application logic. It is therefore necessary to compare the created or updated document with the file's data and notify users about any differences that may require manual resolution. The differences are logged in the **XML Import – comparison log**.

If any differences in document totals have been found during the import the import script will show these differences in a new window and ask you either to confirm or to decline the import.

If you agree to continue the import script will create or update the document, mark it as created from XML and attach the source XML file together with the import log to the document's document management archive. The created or updated document number will then be shown in a separate window with the possibility to open the document directly from this window.

When the import is finished:

- the line's XML file is moved to the automatically created subfolder IMPORTED
- the line in the XML Inbox becomes unlocked
- the line status is changed to *Transferred*
- the line is updated with the date and the time of the import
- the line's log is updated with the XML import – comparison log
- the link between the file's reference number and our document number is updated in the XML conversion table if necessary

The menu item - Show file

This menu item displays the line's XML file. If any Presentation style sheet has been specified for the file's Schema it will be used for display. If no Presentation style sheet has been specified for the file's Schema the file will be displayed by the default program associated with the file type.

The menu item - References

This menu item shows references to all C5 documents (such as Sales or Purchase orders) that have been created or updated from the line's XML file

The menu item - Log

This menu item shows the scanning and import log for the line

The menu item - Filter

This menu item is used to filter the display of the lines in the XML Inbox according to the selected line status

The menu item- Delete

This menu item is used to delete the lines with status *Transferred* and *Rejected*

If necessary the lines with status *Received* must be deleted manually

The menu item – Lock/Unlock

This menu item is used to lock or unlock the line manually. This function can be used to lock some lines in order to prevent other users from e.g. importing the line

The field – Status

This field showing the line's status is normally updated automatically. In some situations however this field can and must be changed manually:

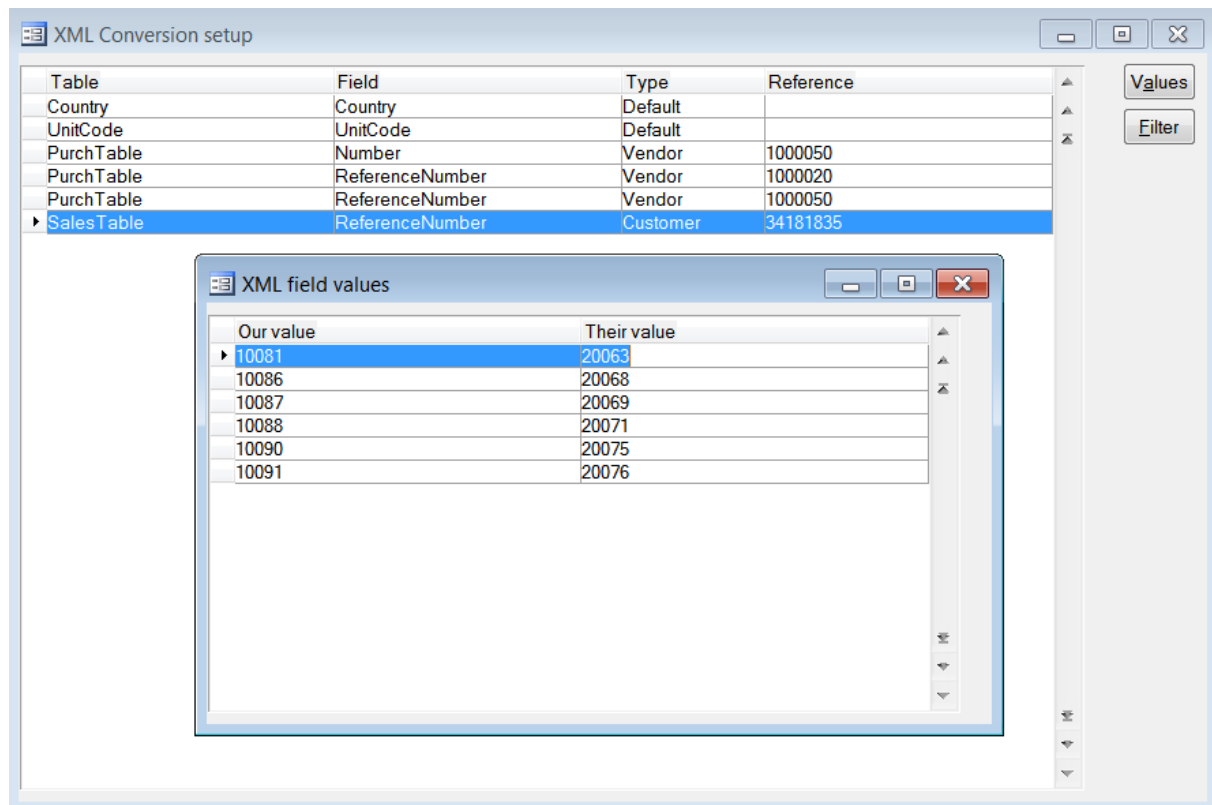
- if a line in the XML Inbox cannot be approved for import (e.g. due to unknown origin or similar) it is recommended to manually change the line's status to *Rejected* or to delete the line. When a line's status is changed to *Rejected* the line's XML file is moved to the automatically created subfolder REJECTED
- manually change the line's status from *Imported* back to *Received*, if a line has already been imported but for some reason needs to be imported again and/or rescanned.

Note that once a line has been imported, a reference between the file's reference number and the corresponding C5 document was created in the XML conversion table. When you change the status and import the line again it won't create any new document because of the existing reference in the XML conversion table, but only update the existing document. If you want to create an entirely new document you must manually delete the existing reference in the XML conversion table. See also next section.

XML CONVERSION (AUTOMATIC UPDATE)

As already mentioned the import of incoming XML documents creates a link between the reference number from an XML file and a C5 document number that was created from this file.

These links are stored and maintained in the XML Conversion table. To view the existing links or conversion between a customer's or vendor's reference numbers (such as Sales order numbers or Purchase order numbers) use the local menu item **Filter** to view conversion lines for individual customers and vendors.



The example above shows that a company has received and imported an XML file from the customer 34181835 with the Purchase order number 20063 that corresponds to the Sales number 10081 in the company's database. If the company later receives another XML file from the same customer and with the same reference number (i.e. Purchase order number) no new Sales order will be created in the company's database by the file import script as the corresponding Sales order already exists according to the XML conversion table and the import script will only update the Sales order with changes.

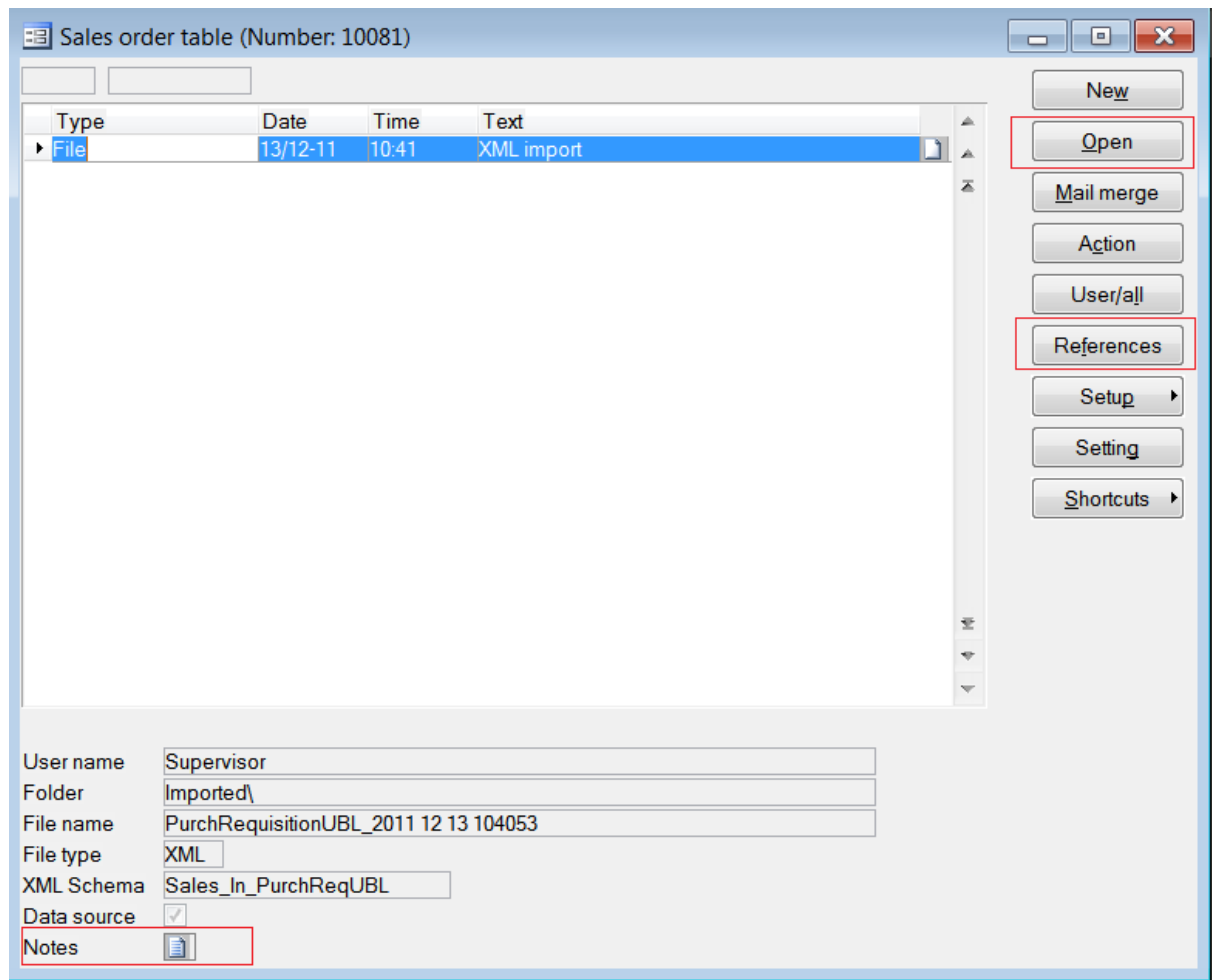
Conversion lines for individual customers and vendors can be accessed directly from the Customer and Vendor table using the local menu items Setup and XML conversion.

FURTHER PROCESSING OF SALES/PURCHASE ORDERS CREATED BY OIOUBL IMPORT

All purchase and sales orders created by OIOUBL import are automatically marked as XML orders which can be seen in the field **From XML**. The value of FromXML cannot be changed by users and users are warned when making any changes to XML orders.

Note that the field **Invoice** on Purchase orders is by default updated with the vendor's sales invoice or confirmation number.

As mentioned before the original XML source files are saved in the document management module and can be accessed directly from the related Sales or Purchase order by using the Ctrl+N shortcut.



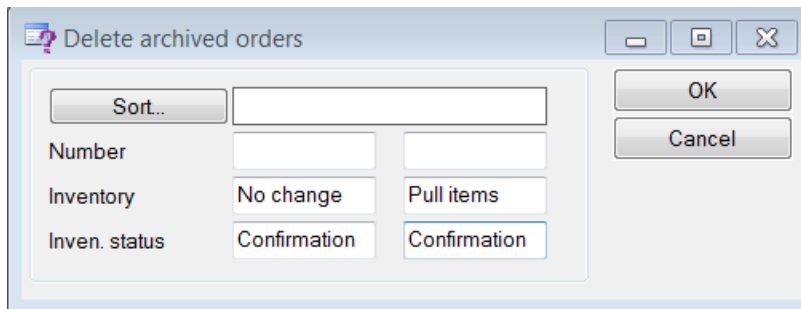
In the example above the line in the document management window for the sales order number 10081 shows that the Sales order has been created from an OIOUBL purchase requisition. The menu item **Open** is used to open and view the original XML file. The **Notes** for the line contain the XML import comparison log while **References** menu item is used to view all references to the XML file in the C5 database.

REQUIREMENTS FOR EXTENDED OIOUBL FUNCTIONALITY

Save order confirmation/save purchase requisition

In order to be able to generate sales order confirmations and purchase requisitions in OIOUBL format they must be saved in the Sales order archive and Purchase order archive respectively. Ensure that the relevant Sales and Purchase order parameters are set correctly.

If you don't want to keep the sales confirmations and purchase requisitions in the archives after they have been generated you can delete them using Periodic clean-up functionality (Sales/Periodic/Clean-up/ Delete Sales order archive and Purchase/Periodic/Clean-up/ Delete Purchase order archive)



Delimit the deletion to the required document type in the field **Inven. status**

NB It won't be possible to re-create the documents in their original form after they have been deleted from the archives.

OIOUBL – KNOWN RESTRICTIONS

- Delivery information is used from order headers only as OIOUBL does not accept delivery information on the header level and line level simultaneously
- Account order lines (i.e. lines without item number) are not accepted by Schematron
- Zero prices are not accepted by Schematron
- The solution supports only the following payment methods: Bank transfer and Inpayment form.
- All sales and purchase orders that originate from XML files are created with Inventory = No change
- The field "Country code in VAT no." must be specified for each OIOUBL relevant country in the Country table
- Bank a/c no. in the Company table must be specified as follows: [BankRegNo] <single space>[BankAccountNo] e.g. 1234 12345678901 (hyphens are not permitted)
- All VAT numbers must have numeric format e.g. 12345678
- The mandatory OIOUBL node "BuyerContact" is filled with the value of the "Your contact" field from Sales and Purchase orders. This field is therefore also mandatory when generating OIOUBL documents
- The OIOUBL node "Buyers (Order) Id" is filled with the value of the "Order" field from Sales and Purchase orders.
- OIOUBL documents support only the two following VAT types: 25% and "ZERORATED" (tax exempt).
- Microsoft Dynamics C5 only provides the functionality for generating OIOUBL files. The dispatch of the generated files to the recipients is not a part of this functionality.
- All notes (multi line free text fields) are only supported in CDATA format
- It is not recommended to use lines of Blank type
- It is important not to change the sequence of the sales/purchase order lines after an OIOUBL document for the order has been generated.
- The fields Deliver now and Receive now on sales and purchase orders are not updated when the quantity is updated
- If, according to the existing mapping, the quantity field should not be used when importing and creating new purchase or sales order lines all quantities in the created lines will by default be set to 1
- If, according to the existing mapping, the price/discount fields should not be used when importing and creating new purchase or sales order lines all prices/discounts in the created lines will by default be retrieved from the C5 database
- Positive discount amounts at the sales and purchase header level are not supported

ONLINE VALIDATION OF OIOUBL DOCUMENTS

It is possible to validate OIOUBL files online following the instructions found at:

<http://www.oioubl.info/validator/default.aspx>

The validation tool validates OIOUBL documents against the relevant schemas and Schematron style sheets. The validation page also contains links to the latest Schematron files that can be downloaded directly from the page.

See <http://www.oioubl.info/classes/en/index.html> for further information and latest updates on the OIOUBL.

TECHNICAL GUIDE

The XML functionality is developed as a generic solution and not just specifically for the generation of OIOUBL documents. The XML functionality can thus be set up and used for the generation of any kind of XML documents. As the present solution is schema driven it is a requirement to have an XML schema for every XML document that is intended to be generated by this functionality.

XML Schema

XML Schema (Schema Id: Sales_InvoiceUBL)

Schema Id	Schema Id name
CustomerReminderUBL	urn:oasis:names:specification:ubl:schema:xsd:Reminder-2
Proj_CreditNoteUBL	urn:oasis:names:specification:ubl:schema:xsd:CreditNote-2
Proj_InvoiceUBL	urn:oasis:names:specification:ubl:schema:xsd:Invoice-2
Purch_In_InvoiceUBL	urn:oasis:names:specification:ubl:schema:xsd:Invoice-2
Purch_In_OrdConfUBL	urn:oasis:names:specification:ubl:schema:xsd:OrderResponse-2
Purch_OrderReqUBL	urn:oasis:names:specification:ubl:schema:xsd:Order-2
Sales_CreditNoteUBL	urn:oasis:names:specification:ubl:schema:xsd:CreditNote-2
Sales_InvoiceUBL	urn:oasis:names:specification:ubl:schema:xsd:Invoice-2
Sales_In_PurchReqUBL	urn:oasis:names:specification:ubl:schema:xsd:Order-2
Sales_OrdConfirmUBL	urn:oasis:names:specification:ubl:schema:xsd:OrderResponse-2

XML Setup
Get XML Schemas
Import mapping
Export mapping
Advanced

Reference: Type: Root node name:

URL/path and file:

Presentation style sheet:

Stylesheets:

Schematron file:

Script:

All setup in connection with the XML functionality is done in the XML Schema window, that contains the following fields:

- Schema Id** Human readable unique short identification of the line's schema. Schema Id can only be specified when creating a new line.
- Schema id name** No user input. The name is retrieved automatically from the schema file.
- Reference** Main script calling the line's script. This field is used when the line's script cannot be called independently but only in connection with another script. For example in Microsoft Dynamics C5 it is normally one and the same script that creates both invoices and credit notes and therefore only one action calling the script has to be set up in order to be able to create both invoices and credit notes. OIOUBL, however, requires 2 different schemas and thus 2 different setups and scripts that create OIOUBL invoices and OIOUBL credit notes respectively. In order to comply with the general application design and to avoid multiple setups, the credit note script has to be associated with the invoice script which is the main script used in

	setups etc. This is done by specifying the invoice script <i>Sales_InvoiceUBL</i> in the reference field for the credit note. NB. The reference script cannot contain a reference to yet another script. The reference script must, on the other hand, contain a code calling the scripts that are set up with the reference script.
Type	What the line's setup and script are used for. Possible values are Import and Export. Note that by default Import is normally associated with XAL scripts while Export is normally associated with QTX scripts.
Root node name	No user input. The root node name is retrieved automatically from the schema file.
URL/Path and file	Sub path to and name of the lines' schema file. The full path to the schema is dynamically constructed as: XML Schema path (from XML setup) + URL/Path and file. This field can be modified at any time.
Presentation style sheet	See description Schematron and Presentation style sheets files
Schematron file	See description Schematron and Presentation style sheets files
Script	The script that imports or generates the line's XML document. All export scripts can be generated automatically. All import scripts must be developed manually.

To setup a new XML document based on an XML schema you must create a new line in the XML Schema window:

- Specify a short name or identification for the new document in the field **Schema Id**.
- In the field **URL/Path and file** select an XML schema that must be used with the new document. The schema file must be placed in the folder that is specified for **XML Schema path** in the **XML setup** window.
- As soon as a schema file has been selected in the previous step, the file is automatically loaded in memory and validated. If the file has been successfully loaded and validated the fields **Schema Id name** and **Root node name** are automatically filled with the corresponding information retrieved from the schema file.
- Specify whether the line's setup is intended for outgoing or incoming XML documents by selecting *Import* or *Export* in the field **Type**.

Before the new line can be used for generation or import of XML documents it must be additionally set up according to the line's type. The additional setup is done by using the menu **Advanced**

- **Mapping**

The mapping tool is used to easily implement XML data mapping to the C5 database with customized data processing. The data mapping setup must be done for both import and export types of documents. When accessing the mapping tool the line's schema is loaded in memory and validated.

XML document mapping (Schema Id: Sales_InvoiceUBL)

Path: /Invoice/cbc:IssueDate

Shortcuts

Element	Table	Field name	Fixed value	Group
- Invoice				
+ UBLExtensions				
+ UBLVersionID			2.0	CustJournal
+ CustomizationID			OIOUBL-2.02	CustJournal
+ ProfileID			Procurement	CustJournal
+ ID	CustJournal	InvoiceNumber		CustJournal
+ CopyIndicator				
+ UUID				
▶ IssueDate	CustJournal	Date_		CustJournal
IssueTime				
+ InvoiceTypeCode			380	CustJournal
+ Note	SalesTableArch	RecID		CustJournal
+ TaxPointDate				
+ DocumentCurrencyCode	CustJournal	Currency		CustJournal
+ TaxCurrencyCode				
+ PricingCurrencyCode				
+ PaymentCurrencyCode				
+ PaymentAlternativeCurrencyCode				
+ AccountingCostCode				
+ AccountingCost	SalesTableArch	DimAccountCode		CustJournal
+ LineCountNumeric				
+ InvoicePeriod				

Function: Fixed value: Return type: Change sign: Conversion: Key field: Reference:

Import: NS: Leve: Never: cbc: 2

Fields in the Mapping window

Path

The full XPATH path for the current element. The XPATH is automatically retrieved from the schema and cannot be edited.

+/-/<blank>/a

You can expand or collapse the current element by clicking on this field.

+ indicates that the current element contains other elements and that it is collapsed

- indicates that the current element contains other elements and that it is expanded

<blank> indicates that the current element does not contain any other elements and that it can neither be collapsed or expanded

a indicates that the current element is an attribute of the expanded parent element and that it can neither be collapsed or expanded.

Element

The name of the current element. The values of this field are automatically retrieved from the schema and cannot be edited.

Table

This field is used to select a C5 table to be used as source or destination for XML data.

Field name

This field is used to select a field from the C5 table selected in the field **Table** to be used as source or destination for XML data.

Fixed value

This field is used to specify a value that must be used directly as XML data or indirectly as a parameter in customized data processing and that cannot be retrieved from the C5 database. The field can also

	contain a variable that is declared under Macroloads/Variables ; use field help (Alt+H) to view the list of declared variables.				
Group	<p>This field is used to specify a data group for the current XML data mapping line. An XML data mapping line will only be used during import or export of XML data if it contains a valid group specification. A data group determines a time/sequence point for the use of the mapping line during import or export of data. Normally a line's group name will be the same as the line's table, if however the same table is used in more than one setup under Table structure the group name must be the same as that of the corresponding table setup sequence.</p>				
Function	<p>This field is used to specify a function to be used as a part of customized data processing of XML data. When a line contains a function the line's data fields will not be used directly, but only as parameters for the function. The value that will be used as XML data is the function's return value. A function must exist in the C5 application and it must accept the required number of parameters. The type and order of parameters within a function must be as follows:</p> <ol style="list-style-type: none"> 1) STR 12 containing the handle to the current DOM document. 2) Variable accepting the value of the line's Table.Fieldname if any 3) Variable accepting the value of the line's Fixed value field if any 4) Variable accepting the value of the line's Fixed value 2 field if any <p>Any table that is in scope at the point when the function is called and that is set up as an EXTERN table under Table structure will also be used as a parameter for the function.</p> <p>If the line's Fixed value field contains a variable this variable will be updated with the function's return value if the line's Return type field is not specified.</p> <p>In export scripts all function return values are converted to strings using Any2Str function so any special conversions should be done directly in the function's script. Ensure also that the string return values are at least 20 characters long.</p>				
Fixed value 2	This field is used only with functions and only when the line's Fixed value field is not empty. A value specified in this field will be used as an additional parameter for the line's function.				
Return type	<p>This field is used to specify the processing of the return value for the line's function:</p> <table> <tr> <td><Blank></td><td>No special processing. If a variable is specified in the line's Fixed value field it will be updated with the function's return value</td></tr> <tr> <td>String</td><td>Only used to disable update of the variable</td></tr> </table>	<Blank>	No special processing. If a variable is specified in the line's Fixed value field it will be updated with the function's return value	String	Only used to disable update of the variable
<Blank>	No special processing. If a variable is specified in the line's Fixed value field it will be updated with the function's return value				
String	Only used to disable update of the variable				

	specified in the line's Fixed value field								
BSTR handle	Used to specify that the function returns a handle to a Big String object								
Integer/Real	Not used								
Date	Not used								
CDATA handle	Used to specify that the function returns a handle to a CDATA Big String object (typically when processing notes)								
Change sign	This field is used when the line's numeric data must be used with the opposite sign (e.g. when processing credit notes)								
Conversion	This field is used to specify whether the line's value must be checked prior to import or export and converted to another value according to the existing XML data conversion setup.								
Key field	This field is used only during import. When the line's table field is specified as a key field the corresponding XML data value will be validated for existence in the C5 database prior to import. If the XML data value does not exist in the C5 database the import of the XML file may be denied.								
Reference	This field is used only during import. The reference mark means that each occurrence of the line's XPATH must be treated by the import script as a signal to create a new record in the line's table. It is therefore important to specify only one reference line per table.								
Import	<p>This field is used only during import. The value of the field determines the processing of the line's XML data during import. Possible values are:</p> <table> <tr> <td>Never</td><td>The line's XML data will be ignored during import</td></tr> <tr> <td>Insert</td><td>The line's XML data will be inserted into the C5 database during import but only when the import creates a new document in the C5 database (e.g. when a completely new purchase or sales order is created from an XML file)</td></tr> <tr> <td>Update</td><td>The line's XML data will overwrite/update the corresponding C5 data but only when the import updates an existing document in the C5 database (e.g. when an existing purchase or sales order is updated from an XML file)</td></tr> <tr> <td>Always</td><td>Depending on whether the import creates or updates a document in the C5 database the line's XML data will either be inserted or used to overwrite/update the corresponding C5 data</td></tr> </table>	Never	The line's XML data will be ignored during import	Insert	The line's XML data will be inserted into the C5 database during import but only when the import creates a new document in the C5 database (e.g. when a completely new purchase or sales order is created from an XML file)	Update	The line's XML data will overwrite/update the corresponding C5 data but only when the import updates an existing document in the C5 database (e.g. when an existing purchase or sales order is updated from an XML file)	Always	Depending on whether the import creates or updates a document in the C5 database the line's XML data will either be inserted or used to overwrite/update the corresponding C5 data
Never	The line's XML data will be ignored during import								
Insert	The line's XML data will be inserted into the C5 database during import but only when the import creates a new document in the C5 database (e.g. when a completely new purchase or sales order is created from an XML file)								
Update	The line's XML data will overwrite/update the corresponding C5 data but only when the import updates an existing document in the C5 database (e.g. when an existing purchase or sales order is updated from an XML file)								
Always	Depending on whether the import creates or updates a document in the C5 database the line's XML data will either be inserted or used to overwrite/update the corresponding C5 data								

NB For all lines that are mapped to a temporary C5 table, this field can be set up only once for the entire table group, the setup should be done for a line with *NodeList* specified in the Fixed value field.

NS This field shows the current element's Namespace. The field cannot be edited.

Level This field shows the current element's level in the XML file's structure counted from the root node (Level 1). The field cannot be edited.

All lines for which a mapping exists become highlighted to provide a better overview.

Changes in the mapping window can normally be made without having to re-generate the schema's import or export script.

When working with XML data mapping it is important to know the requirements that are dictated by the particular XML schema (such as mandatory and optional nodes, values etc.). This information can normally be found in the documentation for the schema. Some information about a particular element can be retrieved from the schema and shown in a separate window by using the ALT+X key combination directly in the mapping window.

- Macroloads/Variables

This menu is used only with export scripts that must be generated automatically. As mentioned above all export scripts (QTX) can be generated automatically as opposed to import scripts that must be developed manually.

A QTX script consists normally of multiple triggers and can include multiple searches based on multiple tables. The current automatic script generation procedure has, however, the following restrictions: only INIT trigger is used on the entire QTX level, only Pre-Body trigger is used on the table level and only one table can be used as QTX search table. The automatic script generation procedure also requires an additional technical setup for each export script. This setup is divided into 2 different steps with corresponding menus: **Macroloads/Variables** and **Table structure**.

Macroloads/Variables window is used primarily to set up an export script's INIT trigger.



Type	Le	Enum name	Name
MacroLoad			SALESPURCH
INT			LineNo
INT			RefLineNo
INT			VatTempld
INT			AllChargeTempld
INT			LineDlvTempld
Macro run			OIOXmlVariables
Macro run			SetOIOTemplds
STRING	25		InvEanNo
STRING	25		DlvEanNo
ENUM		NoYes	Dialog
INT			Repro
STRING	25		AccountDimension
Pre-Body Macro ru			UpdateAddressOkParm
Pre-Body Macro ru			GetCheckEanNo

Fields in the Macroloads/Variables window

Type	The setup type for the line. The pre-defined setup types are:	
	MacroLoad	Macroload of a macro library in INIT trigger
	STRING	Declaration of a STRING variable in INIT trigger
	REAL	Declaration of a REAL variable in INIT trigger
	INT	Declaration of an INT variable in INIT trigger
	DATE	Declaration of a DATE variable in INIT trigger
	ENUM	Declaration of an ENUM variable in INIT trigger
	Macro run	Call/execution of a macro in INIT trigger
	Pre-Body Macro	Call/execution of a macro at the beginning

	run Top	of the QTX table's Pre-Body trigger
	Pre-Body Macro	Call/execution of a macro at the end of the
	run Bottom	QTX table's Pre-Body trigger
L	This field is only used with STRING types to specify the length of a string variable.	
Enum name	This field is only used with ENUM types to specify the type of enumeration for an ENUM variable.	
Name	Depending on the line's type: the name of a macro library, macro or variable.	

Any changes made in this window after a script has been generated will require re-generation of the script.

- **Table structure**

This menu is used only with export scripts that must be generated automatically.

In order to be able to generate an export script (QTX) automatically, it is necessary to define a QTX search table (i.e. the main delimitation table on which the QTX will be based) as well as to describe the order in which other tables that are mapped to XML data must be searched within the QTX table's Pre-body trigger.

Sorti	Type	F	Table	Index	Group	Counter
1	QUERY	<input checked="" type="checkbox"/>	CustTable	AccountIdx	Customer	
2	INTRODUCE	<input checked="" type="checkbox"/>	TmpSum	SesKeyIdx	Customer	
3	INTRODUCE	<input checked="" type="checkbox"/>	TmpFrmVirtual	SesFileRecIdx	Customer	
4	INTRODUCE	<input checked="" type="checkbox"/>	CompanyInfo	RecID	Customer	
5	INTRODUCE	<input checked="" type="checkbox"/>	TmpPaySum	RecID	Customer	
6	SEARCH	<input checked="" type="checkbox"/>	CustTrans	AcOpenInvIdx	PreCustTrans	
7	SEARCH	<input checked="" type="checkbox"/>	CustTrans	AcOpenInvIdx	CustomerTrans	
8	INTRODUCE	<input checked="" type="checkbox"/>	TmpAccountSum	AccountIdx	CustomerTrans	Balance03
9	SEARCH	<input checked="" type="checkbox"/>	TmpAccountSum	AccountIdx	TaxTotals	

Table	Field	Ref. table	Field
CustTrans		CustTable	
Field 1	Account	Field 1	Account
Field 2	Open	Field 2	1
Field 3	Currency	Field 3	&CurrencyCode
Field 4		Field 4	#XMLRemCode

Fields in the Table structure window

Sort The order in which a table is used in the automatically generated export script

Type The setup type for the line. The pre-defined setup types are:

QUERY Indicates that the line's table is the main table for the QTX script. There can only be one QUERY table in a Table structure setup. The QUERY table should also always be defined on the first setup line.

	<p>INTRODUCE Indicates that the line's table should be accessed in the QTX table's Pre-Body trigger using the INTRODUCE command. If a line of INTRODUCE type is preceded by a line of SEARCH type, the line's table will be accessed by INTRODUCE within the SEARCH from the previous line.</p> <p>SEARCH Indicates that the line's table should be accessed in the QTX table's Pre-Body trigger using the SEARCH command. Note that it is not possible to define a SEARCH within another SEARCH.</p> <p>EXTERN Indicates that the line's table should be accessed in the QTX table's Pre-Body trigger using the EXTERN command. If a line of EXTERN type is preceded by a line of SEARCH type, the line's table will be accessed by EXTERN within the SEARCH from the previous line.</p>
F(unction)	This field is used to specify whether the line's table buffer can be used with the functions from the mapping setup. If the field is set to <i>Yes</i> the table's buffer can be accessed in the functions by using the EXTERN command.
Table	This field is used to specify a table name for the current setup line.
Index	This field is used to specify an index to be used with the table. The first index defined for the table is used by default.
Group	This field is used to specify a data group name for the current setup line. A data group name creates a link between the data search structure and the XML data mapping. A data group determines, therefore, a time/sequence point for the use of the mapping line during export of data. Normally a line's group name will be the same as the line's table, if however, the same table is used in more than one setup in Table structure the group name must be different for different uses of the table.
Counter	This field is used to specify an alternative record counter field for the line's table. When an export script is generated an integer variable named <i><Group name>Counter</i> (e.g. <i>&SalesLineCounter</i>) is automatically created for each data group search performed in the script's Pre-Body trigger. These variables count each table record found during the defined table/group search. The variables can be used to output a sequence number of a record if this is required by an XML Schema. However if a different sequence is needed and this sequence cannot be achieved by using an index the record sequence can be redefined by using an alternative counter field that then will be used for sorting records. Note that any counter sequence must be unbroken and that a valid record must exist for each redefined counter value.

Table	This field shows the table name for the current setup line. This is just a copy of the Table field described above and it is only used only for information purposes.
Field 1 – Field 4	These fields are used to specify up to 4 index components from the table's index or search delimitation fields that must be used when accessing the line's table by using the command specified in the line's Type field.
<Operator>	This field is used to specify an operator that must be used with the table's index component or with the table's search delimitation field
Ref. table	This field is used to specify another table from the Table structure setup whose data values must be used as delimitation when accessing the line's table. The reference table must be in scope when the line's table is accessed, i.e. the reference table's setup sequence must precede the current line's setup sequence.
Field 1 – Field 4	These fields, displayed under the Ref. table field, are used to specify up to 4 delimitation values to be used when accessing the line's table. These values can be the fields from the reference table, declared variables, fixed values or macros.

If a **Field <n>** in the **Table** column is blank while the corresponding **Field <n>** field in the **Ref. table** column contains a macro this macro will be used as the delimitation value directly without any operators. For example if the **Field 2** field in the **Table** column is blank while the **Field 2** field in the **Ref. table** column contains the macro #Macro the automatically generated code will be as follows:

*WHERE Field 1 == Field 1
AND #Macro*

Any macro used in this way is expected to contain a valid alternative delimitation that can be used when accessing the line's table.

Any changes made in this window after a script has been generated will require re-generation of the script.

- **Create query**

This menu is used only with export scripts that must be generated automatically. All required setups as described above must exist prior to using this menu item.

When the menu is activated you will be prompted to specify a QTX name to be used as the system name for the new export script. If a name already exists you must either specify another name or agree to overwrite the already existing QTX with the new one. The new export QTX is then automatically created from the existing export setups: **Macroloads/Variables**, **Table structure** and **Mapping**. If the automatically generated script compiles with errors you must either correct the setups or manually correct the errors directly in the generated QTX. Note that the new QTX will be created in the USR application layer.

All export QTX scripts will be created by using the following template:

INIT trigger

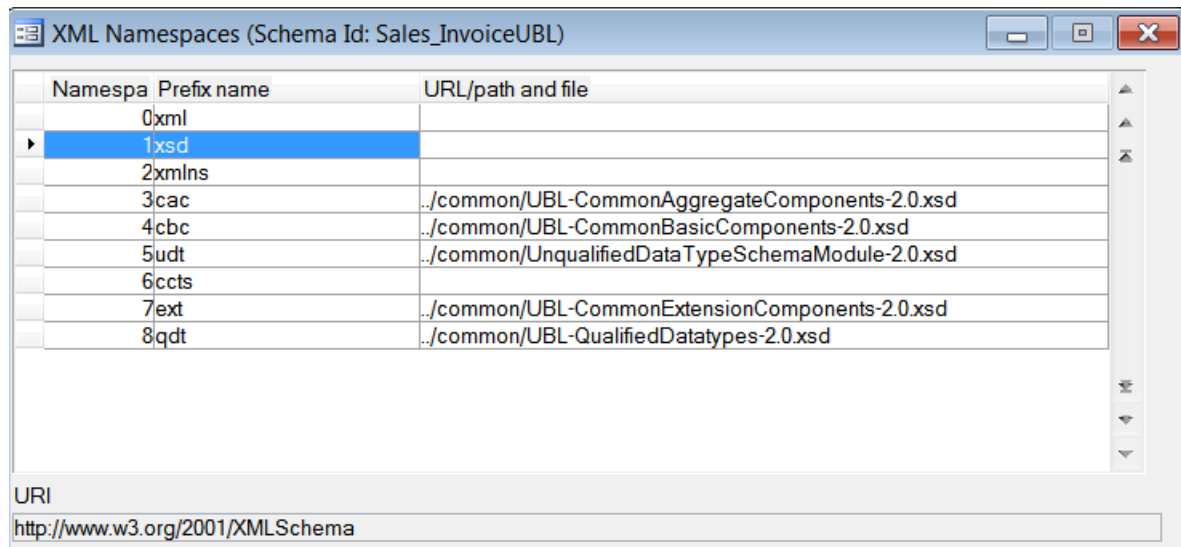
Macroloads	//from Macroloads/Variables
Variable declaration	//from Macroloads/Variables
Group counter variable declaration	//Built in functionality
Load Schema cache	//Built in functionality
Execute local macros	//from Macroloads/Variables

Pre-body (QUERY table)

Create DOM XML document (Temp XML)	//Built in functionality
Update QUERY table group counter variable	//Built in functionality
Execute Pre-body TOP macros	//from Macroloads/Variables
INTRODUCE/EXTERN Tables*	//from Table structure
SEARCH Tables*	//from Table structure
Update SEARCH table group counter variable	//Built in functionality
INTRODUCE/EXTERN Tables*	//from Table structure
SEARCH Mapping (with the same Group as the SEARCH Table)	//Built in functionality
Write data to the temp XML using XPath according to the mapping line's setup (either directly from database or fixed value or after processing in a function)	//Built in functionality
END	
END	
SEARCH Mapping (with the Group defined for QUERY table)	//Built in functionality
Write data to the temp XML using XPath according to the mapping line's setup (either directly from database or fixed value or after processing in a function)	//Built in functionality
END	
Create DOM XML Document (Final XML) based on the Schema	//Built in functionality
Retrieve data from the Temp XML file using XPATH	
Validate the Final XML file	//Built in functionality
Save the Final XML file	//Built in functionality
Execute Pre-body BOTTOM macros	//from Macroloads/Variables
/* As many times as there are tables of the respective type and sequence in the Table structure	

- XML Namespaces

This menu item activates a window with information on the current schema's namespaces. The information shown in this window is retrieved directly from the schema and cannot be edited.



KNOWN RESTRICTIONS

- All automatically generated QTX scripts create an XML file for each record found when searching the main QTX table
- Only schemas containing XML namespaces are supported

Microsoft Dynamics is a line of integrated, adaptable business management solutions that enables you and your people to make business decisions with greater confidence. Microsoft Dynamics works like and with familiar Microsoft software, automating and streamlining financial, customer relationship and supply chain processes in a way that helps you drive business success.

U.S. and Canada Toll Free 1-888-477-7989

Worldwide +1-701-281-6500

www.microsoft.com/dynamics

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, this document should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2012 Microsoft Corporation. All rights reserved.

Microsoft, the Microsoft Dynamics Logo, BizTalk, FRx, Microsoft Dynamics, .NET Framework, SharePoint, Visual Basic, Visual C++, Visual SourceSafe, Visual Studio, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation, FRx Software Corporation, or Microsoft Business Solutions ApS in the United States and/or other countries. Microsoft Business Solutions ApS and FRx Software Corporation are subsidiaries of Microsoft Corporation.

Microsoft®