**Mark Logic Corporation**

**MarkLogic Toolkit for Excel®**

**Developer’s Guide**

1

Release 1.0

May, 2009

Last Revised: 1.0-1, May, 2009

© Copyright 2002-2009 by Mark Logic Corporation. All rights reserved worldwide.

1. **Overview and Requirements**

This chapter introduces the MarkLogic Toolkit for Excel ® and lists the product requirements and supported platforms. It includes the following sections:

• Overview of the MarkLogic Toolkit for Excel

• System Requirements

**1.1 Overview of the MarkLogic Toolkit for Excel**

The MarkLogic Toolkit for Excel allows you to integrate Microsoft Excel 2007 directly with content stored in a MarkLogic Server database. You can use the MarkLogic Toolkit for Excel for the following:

• Access Content from MarkLogic Server in Microsoft Excel 2007

• Custom Browser-Based Add-In

• No C# Coding Required for Add-In

• Configurable Installer to Deploy to Excel Client

**1.1.1 Access Content from MarkLogic Server in Microsoft Excel 2007**

An *add-in* in Microsoft Excel 2007 is a panel that appears in the Excel interface that allows you to perform custom actions. The MarkLogic Add-in for Excel® uses Excel’s add-in facility to create a panel in Excel that allows you to run a MarkLogic Server application in the MarkLogic Add-in for Excel. You can use the MarkLogic Add-in for Excel to retrieve content from MarkLogic Server and use it in an Excel document, and you can use the Add-In to take content in an Excel document and use it in a MarkLogic Server application.

**1.1.2 Custom Browser-Based Add-In**

The MarkLogic Add-in for Excel exposes a browser in the Microsoft Excel interface, and the browser has access both to Microsoft Excel and to anything over HTTP. The Add-In communicates with an HTTP Server application running in MarkLogic Server. The MarkLogic

Server application can perform whatever tasks you want, and therefore allows for completely custom behavior of the MarkLogic Add-in for Excel. For example, the application can provide a search interface to content stored in MarkLogic Server, and can allow you to use content from that search in Excel. The application can have any interface that you can run in a browser, and it can do things like allow users to insert content into Excel, extract content from Excel, and so on.

**1.1.3 No C# Coding Required for Add-In**

The MarkLogic Add-in for Excel uses a JavaScript library to communicate with Microsoft Excel

2007, and it uses HTTP to communicate with MarkLogic Server. Consequently, you can create full-featured applications to run in the MarkLogic Add-in for Excel, and those applications run as a MarkLogic Server HTTP Server application.

The usual way to create a MarkLogic Add-in for Excel is to use the Excel software development tools and create a C# program that communicates with Excel. The MarkLogic Add-in for Excel requires no developer-written C# code; all of the communication with Excel 2007 is done via a JavaScript library. The JavaScript library communicates with Excel on the client side, and with MarkLogic Server on the server side, requiring no C# coding. You can make server-side code changes to your MarkLogic Server application which will change the behavior of your MarkLogic Add-in for Excel.

**1.1.4 Configurable Installer to Deploy to Excel Client**

The MarkLogic Toolkit for Excel includes a customizable Windows installer file (.msi). You can create a customized installer which points to your application. Once you have created the custom installer, you can deploy it on any number of Microsoft Excel 2007 instances, either by having individual users install the .msi file or by having a network administrator do a network installation.

**1.2 System Requirements**

The MarkLogic Toolkit for Excel has the following system requirements:

For the Microsoft Windows client:

* Microsoft Office 2007.
* .NET Framework 3.5
* Microsoft Visual Studio Tools for Office Runtime 3.0
* Windows Installer 3.1

(Note that the MarkLogic Add-in for Excel installer checks for the above, and provides a facility to install them).

* The 2007 Microsoft Office Primary Interop Assemblies, which are installed with Office 2007 and also available separately.
* Administrative privileges on the Windows machine.

For the Windows environment in which you customize the MarkLogic Add-in for Excel installer (.msi) file:

* All of the prerequisites for the Microsoft Windows client (above).
* Microsoft Windows SDK v6.0 or greater.

For the instance of MarkLogic Server with which the MarkLogic Add-in for Excel communicates:

* MarkLogic Server version 4.0-3 or later, on any supported platform. See the *Release Notes* and the *Installation Guide* in the MarkLogic Server documentation for details on therequirements for MarkLogic Server.
* Administrative privileges for the MarkLogic Server environment are required.

**2.0 Preparing the Installer and Installing the MarkLogic**

**Add-in for Excel**

11

This chapter describes the installation process for the MarkLogic Add-in for Excel. The procedures involve installing an application in an instance of MarkLogic Server, creating a custom installer for the Add-in, and installing the MarkLogic Add-in for Excel. The following sections are included:

• MarkLogic Toolkit for Excel Installation Package

• Installation Procedure

**2.1 MarkLogic Toolkit for Excel Installation Package**

You can download the latest MarkLogic Add-in for Excel installation package from developer.marklogic.com. The installation package is a zip file that expands to a folder with the name and version number of the MarkLogic Toolkit for Excel, and that folder contains the following:

* addin.deploy folder: Contains the MarkLogic Add-in for Excel installation packages (.msi and .exe files).
* config folder: Contains the config.idt file, which is used when customizing the installation package.
* docs folder: Contains the documentation for the MarkLogic Add-in for Excel, including the *Excel Add-In Developer’s Guide* (this document), the JavaScript API documentation, and the XQuery API documentation.
* js folder: Contains the JavaScript API source file to use in your applications.
* Samples folder: Contains a sample MarkLogic Server application that communicates with the MarkLogic Add-in for Excel.

There are three main parts to the installation process: installing the application in MarkLogic

Server, creating the custom installation package for the Add-In, and deploying the Add-In in

Microsoft Excel. The remainder of this chapter describes the installation process for the three main parts of the MarkLogic Add-in for Excel installation.

**2.2 Installation Procedure**

This section describes the various parts of preparing a MarkLogic Add-in for Excel installer for deployment. For information about system requirements, see “System Requirements” on page 4.

This section contains the following parts:

• Installing Your Application in MarkLogic Server

• Creating a Custom Installer for the MarkLogic Add-In for Excel

• Installing the MarkLogic Add-in for Excel on Clients with Microsoft Excel

• Uninstalling the MarkLogic Add-in for Excel

• Modifying the Registry to Change the URL during Development

**2.2.1 Installing Your Application in MarkLogic Server**

To use the MarkLogic Add-in for Excel, you need an application that runs on MarkLogic Server to which the Add-In communicates. The application can be any application that runs on a MarkLogic Server HTTP Server. The application uses the JavaScript API as its interface to communicate with Microsoft Excel. For a description of a simple application, see “Getting Started With the MarkLogic Add-in for Excel” on page 12.

After you install your application, make a note of the URL to that application, as you need to enter that in your .msi file before you deploy the application to your users. A sample URL might be http://marklogic.myserver.com:8123/, where the HTTP Server is running on the machine marklogic.myserver.com and is running on port 8123.

**2.2.2 Creating a Custom Installer for the MarkLogic Add-In for Excel**

You can create a custom .msi file for the MarkLogic Add-in for Excel. The customization modifies the following properties for the MarkLogic Add-in for Excel, and those properties end up stored in the Windows registry under HKEY\_CURRENT\_USER\MarkLogicAddinConfiguration\Excel:

* URL: The URL of a MarkLogic Server application to which the MarkLogic Add-in for Excel connects. The default value is <http://www.marklogic.com>.
* RbnBtnLbl: The label that appears on the custom ribbon button created by the Add-In. The default value is My Button.
* RbnGrpLbl: The label that specifies the Group in which the button appears in the ribbon (the ribbon group label). This appears in the ribbon underneath the button. The default value is My Group.
* RbnTabLbl: The label for the ribbon tab. The default value is My Tab.
* CTPTitle: The title for the Custom Task Pane that has the browser embedded. The default value is My Task Pane.
* CTPEnabled: A boolean that, when true, indicates that the MarkLogic Add-in for Excel pane is opened when Excel starts, or when false, indicates that the user must enable the Add-In using the button. The default value is true.

The following parts outline several different ways you can customize the .msi file. You can use whichever way makes is the most convenient in your development environment.

• Edit the config.idt and use MsiDb to customize the .msi File

• Use Orca.exe to customize the .msi File

**2.2.2.1 Edit the config.idt and use MsiDb to customize the .msi File**

Perform the following steps to edit the .msi file using the config.idt file supplied in the installation zip package.

1. Open the config/config.idt file supplied in the zip installation package.

**Note:** If you do not want to use the config.idt file, you can export the idt information from the .msi file by running the following command:

MsiDb -f "<directory where idt is to be exported>" –d "C:\MyAddin\MarkLogic\_ExcelAddin\_Setup.msi" -e Registry

1. Use a text editor to edit the URL, CTPTitle, and other values in the idt file. Make sure to save your changes to the idt file.
2. Use the MsiDb in the Windows SDK to run the following command to update the .msi file with your new values.

MsiDb -f "<directory where idt is located>" –d "MarkLogic\_ExcelAddin\_Setup.msi" -i config.idt

For example: C:\> cd "C:\Program Files\Microsoft SDKs\Windows\v6.0A\Bin"

C:\Program Files\Microsoft SDKs\Windows\v6.0A\Bin> MsiDb -f

"C:\MyAddin\MyConfig" –d

"C:\MyAddin\MarkLogic\_ExcelAddin\_Setup.msi" –I config.idt

This updates the .msi file with your new values.

**2.2.2.2 Use Orca.exe to Customize the .msi File**

You can use the Orca.exe utility to customize the .msi file. Orca.exe is part of the Windows SDK, but is not always installed by default. To modify the .msi file using Orca, perform the following steps:

1. Launch Orca.exe (for example, from the start menu). If you do not have Orca installed, install it. For details on Orca, see the Microsoft SDK documetation.
2. In the Orca window, select File > Open and navigate to the location of the MarkLogic Add-in for Excel .msi file (for example,

c:/tmp/MarkLogic\_ExcelAddin-1.0-1/addin.deploy).

1. Select the MarkLogic\_ExcelAddin\_Setup.msi file and click Open.
2. Select the Registry table on the left pane.

5. Edit the values for the URL, CTPEnabled, RbnGrpLbl, RbnTabLbl, CTPTilte, and/or RbnBtnLbl

names.

**Warning:** Do not edit any other values than these, otherwise the installation might not work correctly.

1. Use File > Save to save the .msi file, or use File > Save As... to save the file to a different .msi file.

This updates the .msi file with your new values.

**2.2.3 Installing the MarkLogic Add-in for Excel on Clients With Microsoft**

**Excel**

After you have prepared the .msi as described in the previous section, it is ready to install into

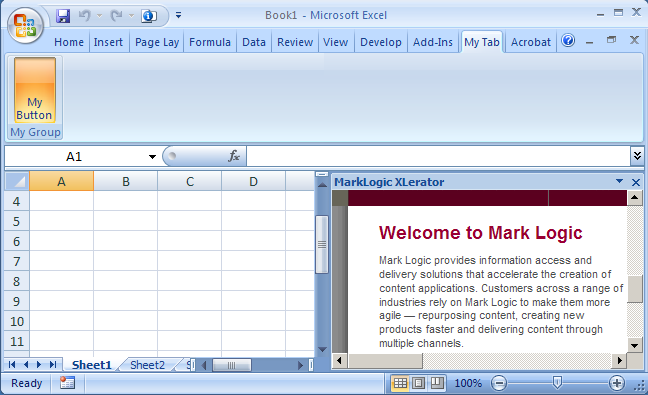
Excel on your Microsoft Windows clients. You can use the standard Windows techniques to deploy the installer across an enterprise, or you can make the .msi file and the setup.exe file available to your users and have them run the installer on their own machines using the steps below.

To run the installation, perform the following steps:

1. Update the .msi file as described in “Creating a Custom Installer for the MarkLogic Add-In for Excel” on page 7.
2. Run the addin.deploy/setup.exe file. This installer checks for all of the dependencies needed by the MarkLogic Add-in for Excel, and then installs them if they are not already installed.

**Note:** If the setup program determines that you need to install .NET 3.5, the .NET installation will take some time and requires a reboot.

1. When it is done checking for dependencies, and if all of the dependencies are met, it automatically launches the MarkLogic Add-in for Excel installation (the MarkLogic\_ExcelAddin\_Setup.msi file).
2. Follow the prompts for the installation, clicking Next until it installs the MarkLogic Add-in for Excel.
3. Click Close when the installer completes.
4. Launch Microsoft Excel to test the installation. You should see a new tab at the top of the Excel interface with the name you added in your customization procedure earlier, and the My Task Pane to the right. If you made no modifications to the .msi file, the Add-In looks like the following in Microsoft Excel:



**2.2.4 Uninstalling the MarkLogic Add-in for Excel**

If you need to change any of the parameters of the MarkLogic Add-in for Excel (the URL or the names of the customizable fields, for example), then you must uninstall the current version of the Add-In before installing a new version.

To remove the MarkLogic Add-in for Excel from a Microsoft Excel environment, perform the following steps:

1. On the Windows machine in which the MarkLogic Add-in for Excel is installed, open the Control Panel (for example, from the Start menu).
2. In the Control Panel, click the Uninstall a program link (Windows Vista) or use the Add/Remove Programs Control Panel (Windows XP or Server 2003) to uninstall the MarkLogic\_Excel program.

**2.2.5 Modifying the Registry to Change the URL during Development**

The URL with which the MarkLogic Add-in for Excel communicates with MarkLogic Server is stored in the Windows Registry, and the installer updates that registry key during installation. During development, it is sometimes convenient to change the URL (or some of the other parameters) without creating a new installer, uninstalling the old one, and reinstalling the new one. You can modify the registry directly to change these key values. For example, run the following command from a Windows command prompt to change the URL key to http://localhost:8888:

REG ADD HKEY\_CURRENT\_USER\MarkLogicAddinConfiguration\Excel /v URL /d

http://localhost:8888/

**3.0 Getting Started With the MarkLogic Add-in for Excel**

4This chapter describes how to quickly get started using the MarkLogic Add-in for Excel. It is a good starting place to understand what the MarkLogic Add-in for Excel does, and includes the following sections:

• Download and Unzip the Installation Package

• Create an HTTP App Server in MarkLogic Server

• Customize the MarkLogic Add-in for Excel Installer

• Install the MarkLogic Add-in for Excel

• Hello World Applications

• Sample Application

**3.1 Download and Unzip the Installation Package**

If you have not already done so, download the MarkLogic Add-in for Excel installation package from developer.marklogic.com. After you download it, unzip it into a convenient location (for example, c:/tmp).

**3.2 Create an HTTP App Server in MarkLogic Server**

Create an HTTP Server in MarkLogic Server to use as an App Server for the MarkLogic Add-in for Excel. For example, log into the Admin Interface, go to Groups > Default > App Servers and click the Create HTTP button. Note the server and port number, as you will need that to modify the installer with the URL of this server.

**3.3 Customize the MarkLogic Add-in for Excel Installer**

Use the procedure in “Creating a Custom Installer for the MarkLogic Add-In for Excel” on page 7 to customize your MarkLogic Add-in for Excel installation. You must customize the URL with the one to your MarkLogic Server App Server, and you can modify any of the other settings, too, if you want.

**3.4 Install the MarkLogic Add-in for Excel**

Install your customized MarkLogic Add-in for Excel as described in “Installing the MarkLogic Add-in for Excel on Clients with Microsoft Excel” on page 9.

**3.5 “Hello, Excel” Applications**

This section will show how to build a “Hello, Excel” app to get you started. (tbd)

**3.6 Sample Applications**

The sample applications are included in the Samples/ directory of the zip file. These applications allow you to search within Excel 2007 documents expanded and stored in MarkLogic Server. Sample code is provided on an as-is basis; the sample code is not intended for production applications and is not supported. For details, including setup instructions, see the README.txt file and the samples-license.txt file in the Samples directory of the zip file.

The sample application allows you to search arbitrary XML as well as Excel workbooks unzipped and expanded in SpreadsheetML format. When opening search results into Excel. If the XML is not SpreadsheetML, but appears to be in a tabular format, an attempt will be made by the code to generate a Workbook from the XML. If this is not possible, an error message is provided. (More details follow).

The samples also demonstrate how to save metadata in a custom XML piece to an Excel Workbook, as well as the ability to save Workbooks directly to MarkLogic Server from the task pane.

**3.6.1 Setup**

The following is a quick-start guide to installing and configuring the Add-in With Excel.

Assuming Office 2007 is already installed on your system, to install the Add-in for Excel, just double-click setup.exe. As with any good Windows app, click Next, Next, Next, and you’ll be installed and ready to use the Add-in. If any prerequisites are missing, you’ll be prompted to download and install them. Once the prerequisites are installed, the Add-in for Excel will install with a default configuration. A couple of simple updates (described below), and we’ll be up and running with the Samples.

The following will walk you through setting up an HTTP server and configuring the sample code to run within the task pane in Excel.

1. Copy spreadsheet-ml-support.xqy to <ServerInstallationDir>/Modules/MarkLogic/openxml
2. Create an HTTP Server in MarkLogic Server, and set its root directory to the Samples dir included in the .zip
   1. On MarkLogic Server, In the Admin interface, Navigate to "Groups" -> "Default" -> "App Servers"
   2. Click the "Create HTTP" tab
   3. Enter a servername: example: "excel-samples"
   4. Enter the root: this is the path of where the /Samples directory is found

example: "C:/Samples"

* 1. Enter a port: example: 9001
  2. Set database to the database where your expanded Excel files are located
  3. Click "OK" to save the above information

1. Set the URL for the Add-in to the server you just created
   1. In Regedit, navigate to

"HKEY\_CURRENT\_USER" -> "MarkLogicAddinConfiguration" -> "Excel"

* 1. set the URL property value to the url where the samples are located:

example: http://localhost:9001

1. Your almost ready to start using the Samples. Just make the quick updates required in the next sections **(3.6.1.1 , 3.6.1.2)** and you’ll be up and running. They’re quick, simple updates, but warrant a little detail.
2. Now start the Excel application, the Samples default page should be located in the pane on the right side of Excel. This default page includes links to all the samples, and provides brief descriptions.

Samples include: Search, Metadata, and Save. Click the link to navigate to the individual Sample app.

**3.6.1.1 Install the SpreadsheetML Process Pipeline**

You can think of a .xlsx document as essentially a bag of XML. It’s a .zip file, that holds various XML documents that are the serialization of an Excel Workbook in XML format. When you save to MarkLogic Server, you can extract the individual pieces of any Open XML package (.dotx, .pptx, .xlsx) by enabling the ‘Open XML Extract’ pipeline in Content Processing. (Note: Open XML Extract requires the Status Change Handling pipeline be enabled as well.)

Included in the toolkit is an additional pipeline that maps SharedStrings from the SharedStrings table in a .xlsx package, to the worksheet (sheet#.xml within the package), so you can search for text values within worksheets, instead of having to manage the relationships of the parts within your queries.

You’ll find the required files in the /cpf directory of the Toolkit .zip. To install the pipeline:

1. Copy map-shared-action.xqy to

<MarkLogicInstallDir>/ Modules/MarkLogic/conversion/actions/map-shared-action.xqy

On your MarkLogic Server.

Note: This relies on the spreadsheet-ml-support.xqy library you installed in section 3.6.1

1. Install the SpreadsheetML Process Pipeline
   1. Open the Admin Interface of MarkLogic Server
   2. Navigate to Databases -> <database-name> -> Content Processing -> Pipelines
   3. Click the “load” tab
   4. In the “directory” field, enter the path to the /cpf directory of the Toolkit .zip
   5. Click Ok

The pipeline will load. You now just have to enable it for your database.

1. Enable the SpreadsheetML Process Pipeline
   1. Navigate to Databases -> <database-name> -> Content Processing -> Domains -> Default Documents -> Pipelines
   2. Check the checkbox for the SpreadsheetML Process Pipeline
   3. Click “ok”

By default, the scope is root “/”, with a depth of infinity. This is just a quick guide to get you up and running. For more information on working with and configuring CPF (the Content Processing Framework), please see the CPF documentation on our developer zone: <http://developer.marklogic.com/pubs/4.0/default.xqy>.

**3.6.1.2 Update search.js, save.js**

The Samples include demonstrations of how to open an Excel document in MarkLogic Server into Excel, and likewise the ability to save a document directly to MarkLogic Server from Excel.

Note: There are a few ways to open documents from MarkLogic Server and save documents to MarkLogic Server by using the pane and/or Microsoft Office. In the sample we’ve provided one example. If you check the api documentation, you’ll find functions for opening/closing workbooks through the Add-in to/from webDAV. Also, if you setup a webDAV client, you can open and save workbooks directly from webDAV using Excel’s native open/save/close functionality, and no access through a pane is required.

To use the included Samples:

1. update search.js
   1. Change the uname/pwd for the following line to the credentials of the HTTP Server you configured above.

MLA.openXlsx(tmpPath, docuri, url, "uname","pwd")

1. update save.js
   1. Change the uname/pwd for the following line to the credentials of the HTTP Server you just configured above.

MLA.saveActiveWorkbook(tmpPath, doctitle, url, "uname","pwd");

Remember, these are Samples, and we aim to keep it simple. There are ways you can obfuscate the login information, but we leave that as a task/choice for the addin developer based on their requirements, and which functions they choose to use.

**3.6.2 Search Sample**

You are provided with a search box. Enter some text and hit “enter” or click “search” to perform a search against documents in MarkLogic Server.

1. A search will be performed in the Server for any Worksheets and/or other XML documents that contain the text you've entered. The results returned are the names of the documents that met the criteria for search, and are hyperlinks. If you click the link, the document will open in Excel. (see notes below on opening documents from the pane.)

If the document is a .xlsx workbook, metadata for the workbook is included below the name of the document

If a single text word is used for search, the text will be used to search for element names that may be equal to the text you've entered as well as text within worksheets and arbitrary XML.

For searching worksheets, the Search app assumes you've previously saved Excel workbooks (.xlsx) to MarkLogic, and that you have

1. unzipped the parts using the "Open XML Extract" pipeline.
2. mapped shared strings to worksheets in the .xlsx package using the "Excel Map SharedStrings" pipeline. (see section **3.6.1.1**)

For documents that are not spreadsheets, an attempt will be made to use the XML, either by the specified element name or by the parent elements of the text found, to transform the XML into an Excel spreadsheet.

If successful, a workbook will be generated and will open in Excel with the Workbook you've just created. If there is any error, a message is provided and you can click "try again" to try and open another XML file as an Excel document.

**3.6.3 Metadata Sample**

You can add a custom metadata document as a custom XML piece to the active workbook and save it within the .xlsx package.

1. Enter values in the fields, and click "Add Metadata" to add the metadata document to the package.
2. If you edit the metadata, and click "Add Metadata", the metadata will be updated for the document.
3. Click "Remove", to remove the custom piece entirely from the package.

The sample saves the information as dublin core metadata. But you can add any well-formed XML as a custom piece. You don't have to use dublin core. To do this, you just have to create your own form for editing/saving the metadata to the document. See the code for the example

While the metadata is saved with the package, it is only accessible to users using the Add-in who are able to expose the values using the pane. Once saved to MarkLogic, you can now search on that custom metadata to find your spreadsheets.

**3.6.4 Save Sample**

Enter a name for the Active Workbook and click the floppy disk icon. The workbook will be save to the MarkLogic Server you’ve configured with the Add-in. You can then use Search to find your document and open it back up into Excel.

You don’t need to add the .xlsx extension to the filename before saving, it will automatically be appended. Also, if you don’t enter a name for the Workbook, it will be saved as Default.xlsx.

**4.0 Developing Applications for the MarkLogic Add-in for**

**Excel**

This chapter describes some techniques for building XQuery applications that communicate with

the MarkLogic Add-in for Excel, and includes the following section:

• JavaScript Library

• XQuery Libraries

**4.1 JavaScript Library**

The JavaScript library is used to communicate beteen a MarkLogic Server application and

Microsoft Excel. It has functions to create basic building-blocks in a Microsoft Excel document. The library is included in the installation zip package under js/MarkLogicExcelAddin.js. For the function signatures of the APIs in the JavaScript library, see the JavaScriptDoc in the docs/jsdocs directory of the installation zip package.

**4.2 XQuery Libraries**

When you create a MarkLogic Server application that communicates with Excel, you can use any of the functions from the large MarkLogic Server XQuery library. For example, there are two XQuery libraries that have functions directly useful for Office 2007-based applications:

* The ZIP Package library: package.xqy
* The WordProcessingML library: word-processing-ml.xqy

These libraries are installed under the /Modules/MarkLogic/openxml directory in the MarkLogic Server installation directory.

Additionally, there is a library of XQuery functions designed to support the Microsoft SpreadsheetML format. This library is included in the Toolkit for Excel installation zip package as xquery/spreadsheet-ml-support.xqy. Likewise, copy it to /Modules/MarkLogic/openxml.

The documentation for this library is included in the zip package as docs/xquery-apidoc/spreadsheet-ml-support.html.