EWS Editor

An Exchange Mailbox Explorer built on the Exchange Web Services Managed API

# Overview

EWS Editor has three goals:

1. Demonstrate the [Exchange Web Services Managed API](http://msdn.microsoft.com/en-us/library/dd633709.aspx) functionality and simplicity to developers
2. Demonstrate the [Exchange Web Services](http://msdn.microsoft.com/en-us/library/bb204119(EXCHG.140).aspx) traffic used to perform actions initiated through an explorer user interface to aid any Exchange Web Services developer.
3. Assist non-developers in debugging and understanding Exchange stores by exploring items, folders, and their properties in depth.

# Contents

1. Release Notes
2. Feature Notes
3. Source Notes

# Release Notes

## 1.5 Release – 11/6/2009

The first public release of EWSEditor! The basic feature set is still be flushed out but it is still a very useful tool. There are three feature areas of EWSEditor: General UI, Support Tools, and API Coverage. The focus of this release was to establish the UI design and flow of the application, solidify some useful support tools, and illustrate the direction of the application.

There are some gaps in CRUD tasks across item and folder properties and items themselves. The major focus of upcoming releases will be to fill in these gaps before moving on to new features.

## 1.6.1 Release – 3/5/2010

API coverage is a major focus of the 1.6.\* releases of EWSEditor. The 1.6.1 release adds an OOF Settings and Availability form to this end. There were several bugs addressed in this release was well as some changes in the error dialog and about information. These changes will make the information displayed and logged by EWSEditor more useful.

## 1.6.2 Updates – 8/16/2011

Converted project to Visual Studio 2010.

Reference Exchange Managed API 1.1 instead of 1.0.

Code changes to support Exchange 2010 SP1.

Finished Autodiscover dialog.

Added code to allow redirects when logging in – this is needed for accessing outlook.com

Added wait cursor to Autodiscover dialog.

## 1.7.3 – 12/17/2013.

## Updates – 5/16/2013 - 12/17/2013

* Started 1.7.3
* Updated to latest Managed API released – 2.0
* Building with Visual Studio 2012 and .NET 3.5
* Can specify versions up to 2013 now.
* Added ability to set options when loading an item which was exported prior.
* Generated proxies have been updated to 2013 RTM.
* Added a screen for TimeZone information from the local machine and Exchange server.
* Set a default of the Exchange version to Exchange2007\_SP1instead of blank.
* The TimeZone can now optionally be set on login.
* I merged-in a lot of calendaring and diagnostics code I wrote for a parallel app. However, it is not yet accessible from the menu.
* Updated Timezone info screen.
* For the folder notifications pull subscriptions form I added the ability to subscribe to all folders, the timeout can be set, you can double-click on an event in the listview and it will display info on the event in an all text window. The formatting of the screen layout was also altered slightly.
* Streaming notifications window can now subscribe to all folders. If you double-click on the listview it will present the event you click on as text.
* Fixed formatting in streaming and pull notifications windows.
* Increased amount of info presented when looking at events in pull and streaming subscriptions windows. Now you can get a better idea of which items and folders were involved in the events.
* Importing MIME will now use the item specific objects for importing MIME for appointments/messages/Contacts
* Item Export menu now can display MIME in a window. This greatly developer review of Exchange generated MIME.
* Mime for each item can also be pulled and displayed. This new item only shows when you right-click on a calendar folder. A properties grid and list of attachments has not be hooked-up yet; however its planned.
* CalendarView integration. Provides a view of a calendar folder which allows you to look at instances/exceptions/singles by selecting a day on a calendar view control. It will display the info for each item in either a master calendar form or in an instance/exception form.
* Added ContactsForm with basic functionality.
* InboxRules enumerations windows was added.
* Added a feature to add MIME to a folder via a window with a text box.
* Fixed tab order in several screens
* Override ssl certificate verification is now defaulting to true
* Added Options window button the Service (logon) screen.
* Some minor changes to the Service (logon) screen to allow for longer values and did some minor formatting.
* Adding code to add "X-AnchorMailbox" when the version is set above Exchange 2010. This should be set for 2013 and Exchange online 15.
* Clean-up for Servicedialog - now easier to use.
* Added ability to show changes for SyncFolderItems.
* Added the ability to load properties in Sync Folder Items- if you double click it lists some properties and shows errors if it runs into any. Added buttons also to properties on every item in the list in one sweep and also to do individual item loads of properties. Any errors are displayed in a text window. Extensive error catching was added. This should help with identifying issues with loading properties on items as they can point out which items are of issue while getting props. Note that the two new buttons may take a while for their code to execute.
* Autodiscover errors now show more details on errors during service login and captures Autodiscover specific error events. Autodiscover viewer now shows error and response data - double click on item to see multivalued data - also captures Autodiscover specific error events.
* Connected Contacts form and adjusted layout to look better. Now you can right click on a contact and edit it. Minimal number of fields are displayed and some are not editable. Some properties such as phone numbers will take a bit of code to provide proper editing ability - this will need to be done at a later date.
* Added the Distribution List Expansion window on the Tools menu - it allows you to drill down through the distribution list hierarchy and see which mail addresses are under each DL. This is the last primary functionality piece EwsEditor has been needing.
* Delegate windows was showing an error while entering - fixed it.

## 1.7.4 – 12/27/2013 - 12/31/2013.

* Added task item edit screen. So, there is now a basic set of client like edit screens for Messages, Contacts, Appointments and Tasks. Additional features can be added to those windows as needed in the future.
* Can now create a new Email message, Contact and Calendar item by right clicking on a folder of that type and selecting New...
* Default button for user's mail address added to service window

Added play on phone window

* Login service windows can now default in user's SMTP address,
* Can now name folder and set folder class when creating a new folder
* New window under “Other” dropdown menu which displays information about the current window user's account.

## 1.7.5 – 1/1/2014 - 1/21/2014.

* The code has been updated to Visual Studio 2013 and .NET 4.5.
* A new EWS POST window has been added and it allows users to POST EWS XML and get back responses. I’ve added the ability to save the settings on the window (minus the password). There are also a lot of EWS XML samples included – you can load them via a menu and can also save your own requests.
* Menu items under the Tools menu which require a service object are now all disabled unless there is an active service object (ie you signed in).
* Fixed address saving in Contacts form (minus the multiline adress issue which has to be worked-around by a raw post or by using modified proxy classes or by swapping out the lf for clrl).
* Autodiscover viewer now breaks-down alternative mailbox information in autodiscover viewer.   
  Autodiscover viewer's maximize button is now enabled.

## 1.7.6 – 1/22/2014 - 2/5/2014.

* The EWS POST window had some minor fixes one and also many more EWS samples have been added - there are now well over 100.
* A window which can send email using System.Net.Mail sends email to an SMTP port or to an SMTP pickup folder. This can be used with testing EWS new mail notifications.

# Features

### Exchange Service Profiles

Exchange Service Profiles describe not only the connection information for calls specified in the [ExchangeService](http://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.exchangeservice.aspx) class but also the root folder to use in the tree view of EWS Editor. Multiple ExchangeService objects can be displayed at one time using unique configurations. This allows folders on multiple servers and server versions to be displayed at one time or displaying the same folders using different credentials.

Exchange Service Profiles can be saved to an XML file which contains the configuration information (sans specified passwords) so that it can be reloaded easily. If a profile requires credentials then a prompt will appear when loading the Services Profile to enter the credentials.



### View properties of ExchangeServices, Folders, and Items

The PropertyDetailsGrid display first class properties for all nodes of the tree view starting with the root, ExchangeService node and all the way to the folder contents form, displaying first class properties for Items in a folder. Under the View menu the [PropertySet](http://msdn.microsoft.com/en-us/library/microsoft.exchange.webservices.data.propertyset.aspx) used to retrieve item or folder data can be configured to customize which schema properties and extended properties are displayed in the PropertyDetailsGrid.



**Extended Property Lookup**

A number of known extended MAPI properties have been indexed in EWS Editor. Extended properties can be added to the PropertyDetailsGrid through a look up using a known constant name for the property such as PR\_SUBJECT, PidTagSubject, etc. An Extended Property Lookup tool provides a way to quickly get the property information for a known constant.



### Request and response logging

While EWS Editor is running, all requests and responses are logged to a file. The Chatter Log tool is used to view the history of requests and responses.



# Source Notes

### Understanding the PropertyInformation Namespace

The PropertyInformation namespace’s main purpose is to standardize the textual representation of object values in EWS Editor. This is done through the PropertyInterpretation class which uses TypeValues, SmartViews, and KnownExtendedProperties to return text data describing a given object property.

TypeValues standardize the text to display for the value of specific object types. SmartViews apply Exchange/Outlook business logic to property values to translate the raw value into a more meaningful one. The KnownExtendedProperties class contains lookup methods and data related to documented extended properties.

TypeValues and SmartViews conform to a simple interface which associated the logic implemented in a specific TypeValue or SmartView with specific types (ITypeValue) or known properties (ISmartView). A TypeValueFinder class and SmartViewFinder class retrieve the appropriate class based on an input type or property. To add new ITypeValue or ISmartView implementations to the project simply implement the interface and add the new class to the TypeValueFinder or SmartViewFinder’s Initialize() method. Then anytime the PropertyDetailsGrid or other controls which already use the PropertyInterpretation class will automatically benefit from the translation logic for the type or property referenced.