Microsoft Office and Easy Software Deployment

# Introduction

You may have noticed we are not offering Microsoft Office in our Package Store. Not because it’s impossible but because it’s not allowed to distribute Microsoft products to third parties. However, there’s no reason why we can’t tell you how to easily do this yourself!

In this guide we’ll explain how to distribute Microsoft Office through Easy Software Deployment, and how to customize the settings according to your specific needs.

Prerequisites:

* Microsoft Office Deployment Tool
  + This can be found [here](https://www.microsoft.com/en-us/download/details.aspx?id=49117).
* Easy Software Deployment (request a free 30 min. demo [here](https://calendly.com/dereckbreuning/esd-demo))

Open the downloaded .exe file from the Office Deployment Tool, read the license terms and agree to it. Select a folder inside the Applications share you have set up when installing Easy Software Deployment. After that you can safely delete the downloaded .exe as the extracted files are all we need.

The downloaded .exe file will extract just a few files: a setup.exe and some .xml files. You will need the .xml files to specify how and where you want the installer to install Microsoft Office. We’ll explain how you can pre-download the required files and save those in the Application share so you can use these for distributing Office.

# XML configuration

Open an Explorer window and navigate to the path where the files are extracted. Depending on the version of Microsoft Office, select the .xml file for either Office 365 or Office 2019 Enterprise and change the name to something you can easily identify, for example ‘configure-O365-x64.xml’. Open the configuration file with your favorite text editor.

Since we do not want every single computer to download Microsoft Office from the Microsoft servers, we will set up a local share for the installation files. For this guide we’ll use the installation of Office 365 Pro-Plus Retail as an example. The steps described are the same for any version, just the Office version may be different.

Start by taking a look at the following line of code:

<Add OfficeClientEdition="64" Channel="Monthly">

Depending on the architecture (either 32 bits or 64 bits) and from what update channel the version needs to be, you need to change the setting accordingly. For now, we leave it as is.

This line also tells the installer where the installation files are located. We can do this by specifying the “SourcePath” argument. The line will look like this:

<**Add SourcePath="UNC path"** OfficeClientEdition="64" Channel="Monthly">

For simplicity we will use another folder in the application share where the installation files will be cached. In this case that would be ‘[\\ESD-SRV\esd\Applications\OfficeDownload](file://ESD-SRV/esd/Applications/OfficeDownload)’. Change “UNC path” to the UNC path of the new folder you just created, and make the line look something like the following.

<Add SourcePath**="\\Esd-srv\esd\Applications\OfficeDownload**" OfficeClientEdition="64" Channel="Monthly">

Note: as Easy Software Deployment copies the source folder and its contents to a local cache on the agent computer, we’ll want to download the Microsoft Office installation files to a different folder from where the setup.exe and the xml files are stored. This prevents Easy Software Deployment from copying the files while getting the installer. For this example, this will be **\\Esd-srv\esd\Applications\OfficeDownload**.

The next lines determine which version of Office is going to be installed and what language.

**<Product ID="O365ProPlusRetail">**

**<Language ID="en-us" />**

**<Product>**

This example tells the installer to install the US English version Office 365 Pro Plus. You can match the language to that of the OS or another installed Office product:

<Product ID="O365ProPlusRetail">

**<Language ID="MatchInstalled" />**

**<Language ID="MatchOS" Fallback="en-us" />**

</Product>

“MatchInstalled” tells the installer to install the same language if it finds another Office program installed. If no other Office program is installed, it will install the same language as the OS. Alternatively, if the installed OS is in a language Office is not compatible with, it will install English United-States as a fallback option. All these settings are optional.

In the next step we need to specify which applications in the Office suite need to be installed and which ones should be skipped. For this example, we’ll keep the main Office applications but exclude OneDrive. This is done by adding the “ExcludeApp ID” keyword.

<Add SourcePath="\\Esd-srv\esd\Applications\OfficeDownload" OfficeClientEdition="64" Channel="Monthly">

<Product ID="O365ProPlusRetail">

<Language ID="en-us" />

**<ExcludeApp ID="OneDrive" />**

</Product>

</Add>

You need to specify each application you don’t want to be installed on a separate line:

<Product ID="O365ProPlusRetail">

<Language ID="en-us" />

**<ExcludeApp ID="OneDrive" />**

**<ExcludeApp ID="Publisher" />**

</Product>

If you want to install another Office solution like Visio, you will need to create a separate “Product ID” section:

<Product ID="O365ProPlusRetail">

<Language ID="en-us" />

<ExcludeApp ID="OneDrive" />

<ExcludeApp ID="Publisher" />

</Product>

**<Product ID="VisioProRetail">**

**<Language ID="en-us" />**

**</Product>**

Let’s look at a complete example of how to create an install configuration file.

Suppose you want to:

* Install Office 365
  + x64 edition
  + Download the source files in \\Esd-srv\esd\Applications\OfficeDownload
  + Install the same language as the installed OS.
  + Exclude OneDrive

Make the necessary changes and additions to the XML file so it looks like this:

<Configuration>

<Add SourcePath="\\Esd-srv\esd\Applications\OfficeDownload" OfficeClientEdition="64">

<Product ID="O365ProPlusRetail">

<Language ID="MatchOS" Fallback="en-us" />

<ExcludeApp ID="OneDrive" />

</Product>

</Add>

<Updates Enabled="TRUE" Channel="Monthly" />

<Display Level="NONE" AcceptEULA="TRUE" />

<!-- <Property Name="AUTOACTIVATE" Value="1" /> -->

</Configuration>

The last three options determine how updates are handled, whether the EULA is automatically accepted, and activation is done automatically. You need to remove the comment indicators (<!-- … -->) as we did for the first two options in this example.

There are many more configuration options available, and you can find these in [this document](https://docs.microsoft.com/en-us/deployoffice/office-deployment-tool-configuration-options) on the Microsoft Documentation site.

# Downloading the files

This XML file is now ready to be deployed to client computers, but we need to take a few more steps as we want to download the files first. The same XML file we just created comes in handy. Make a new copy and change the name to something like “Download-Office365-x64.xml”.

This file can serve as a template for other configurations, and in it we are going to specify every language we want to download. Just add an extra line for each language, in this example English and Dutch. This will download the installation files for Office in each of the languages.

<Add SourcePath="\\Esd-srv\esd\Applications\OfficeDownload" OfficeClientEdition="64" Channel="Monthly">

<Product ID="O365ProPlusRetail">

**<Language ID=" en-us" />**

**<Language ID=" nl-nl" />**

</Product>

</Add>

To download the installation files needed for Microsoft Office, we need to open Command Prompt or PowerShell and navigate to the **local** path where the Office Deployment Tool files are located. Make sure you are in the right folder and then run the following command:

**setup.exe /DOWNLOAD Download-Office365-x64.xml**

This tells the setup to download the files according to the options in the XML file. The setup will now only **download** the source files for Office and **not install** them. Note that there won’t be any progress bar to show you the download status, just a new line showing up in the command prompt once the files are downloaded. Now we can move on to Easy Software Deployment, where of course the fun starts!

# Adding the Office installation to Easy Software Deployment

Adding the setup.exe for Microsoft Office to Easy Software Deployment works the same as for any other software. In short: add an Installation Item, navigate to the UNC path where the setup.exe and the XML files are found.

After adding the Office Installation Item by adding the appropriate setup.exe we need to change the installation command to be as follows:

**setup.exe /CONFIGURE configure-O365-x64.xml**

This command line tells the installer to install and configure Office according to the specified settings.

For the Installation Item to be installed successfully by the Easy Software Deployment Agent it’s important to make sure the Verify File/Folder Registry check is added. Edit the Installation Item, navigate to Deployment Options and then Verify File/Folder Registry (last two items on the page). Depending on what version of Office and architecture you’re installing you need to choose the appropriate registry to check. Your version can be found in one of these registry keys:

For 32 bits:  
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\<Version>\DisplayName

For 64 bits:  
HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Uninstall\<version>\DisplayName

If you want to make sure you have the right key, deploy the installation package to a test environment. This creates the registry key, after which you can use RegEdit to look it up.

# Uninstalling Office through Easy Software Deployment

Now that we have a working installation of MS Office through Easy Software Deployment, it is necessary to set up a similar way to **uninstall** Office. Again, we can use an XML file to achieve this.

Create a text file with the XML extension for uninstalling in the folder where the Office setup.exe is located and name it accordingly. Open the XML file with your favorite editor.

Copy and paste the following lines in it.

**<Configuration>**

**<Remove>**

**<Product ID="O365ProPlusRetail">**

**</Product>**

**</Remove>**

**<Display Level="None"/>**

**<Property Name="FORCEAPPSHUTDOWN" Value="True" />**

**</Configuration>**

As you can see the uninstall configuration is more or less the same as the install configuration. Instead of adding a product we tell the install to remove a product. The lines really speak for themselves: it will delete Office 365 Pro Plus silently, and any open apps will be forcefully shut down.

Save the file and give it an easy to recognize name, for example **uninstall-O365-x64.xml**. Edit the installation item in Easy Software Deployment and add the following command to the uninstallation command:

**setup.exe /CONFIGURE uninstall -O365-x64.xml.**

That’s it! Easy peasy.