

Apex Support Bulletin: Installing User Content in Office 2016 for Mac

Revision 1.0 [March 2, 2016]

Contact eriksc@microsoft.com

Summary

Office 2016 for Mac applications are fully self-contained, code-signed, and sandboxed, so user content such as templates, add-ins, and proofing tools cannot be installed in the same locations supported by Office 2011 for Mac. This document details where installers and administrators should place content to extend the functionality of Office 2016 for Mac.

Background

Office 2011 for Mac shipped with a common “Office” folder that contained content shared by all the Office applications, and users and administrators could put additional content in that folder to extend the functionality of the Office applications. The “Office” folder no longer exists for Office 2016 for Mac; the applications are fully self-contained, code-signed, and sandboxed. This means that

- because the applications are fully self-contained, Microsoft-originated content is now located inside each application,
- because the applications are code-signed, their contents must not be altered in any way or the code signature will no longer be valid, and
- because the applications are sandboxed, they cannot consume content automatically from arbitrary locations.

Instead, Microsoft now specifies user-specific and machine-wide locations where users and administrators should place content that can be automatically discovered and used by Office 2016 for Mac.

Solution

Additional user content can be installed into a couple of locations. One folder tree is for user-specific content and is located relative to the user’s home folder. The other folder tree is in a non-user-specific location and is intended for use by all users. Although the nodes in the paths are not exactly the same, the root user content node and all folder structures inside that hierarchy are well defined. Each folder for user content can exist with or without the extension “.localized”. While the extension is optional and the Office applications will consume content in either form, Microsoft strongly recommends using that extension if you create these folders yourself. Folders created by Office applications inside the user content tree will have the extension and will also provide localized display names for these folders as per Apple guidelines at

<https://developer.apple.com/library/mac/documentation/FileManagement/Conceptual/FileSystemAdvancedPT/LocalizingtheNameofaDirectory/LocalizingtheNameofaDirectory.html>

User-specific content can be pre-installed into the following path and will be usable in all of the Office applications. This path is always readable and writable by the applications. The “UBF8T346G9” token is required by Apple, but is defined by Apple to represent Microsoft and will never change:

```
~/Library/Group Containers/UBF8T346G9.Office/User Content.localized/...
```

Machine-wide content can be pre-installed into the following path. This location is only readable by the applications; they cannot write to this tree except with direct user action such as “Save As...”. Installers can place content here as they wish. While Office doesn’t particularly care what ownership and permissions are set on the directories and files in this location, you should consider appropriate restrictions if you wish to prevent users from changing this content without administrator privileges. At a minimum the Office applications need +r+x access on all directories and +r access on files. Note that there is no space in the folder “Office365”:

```
/Library/Application Support/Microsoft/Office365/User Content.localized/...
```

Within each of those two locations, there is a full tree of folders that can contain various content. Each of these folders can have the “.localized” extension, although it is omitted here for readability.

.../User Content/	
Add-Ins/	For add-in files and compiled dynamic libraries
Apps for Office/	For modern Java and http-based extensions
Chart Templates/	For Chart-specific templates
Citations/	For Word citation styles
Dictionaries/	For custom dictionaries of uniquely-spelled words
Diagrams/	For custom SmartArt diagrams
Document Elements/	For Word textual snippets and formats
Proofing Tools/	For additional language spelling, hyphenation, and grammar checkers
Startup/	Files to be opened at startup by each app
Excel/	
Word/	
PowerPoint/	
Style Sets/	For Word formatting styles
Templates/	For all other kinds of templates
Themes/	For document themes
Web Queries/	For Excel web-based queries

File access

If you are authoring addins or other content with executable code, you need to be aware of the impact of sandboxing on your code. Because the Office applications are sandboxed, all code running in the context of the applications is constrained on where it can access data. While administrators and installer packages can install content anywhere, such as the above locations, addin code runs with the user’s access and underneath sandbox restrictions.

Read and write access:

The following paths are accessible for read and write access by add-in code. The first path, in the application-specific container, is only accessible by code running in that application and cannot be used to share data between different Office applications. The second path, in the Group Container folder, is intended for sharing data between applications or with a 3rd party application. Your code should not put its data inside the “User Content” folder in the group container; choose a new folder unique to your project and place it inside the UBF8T346G9.Office node:

```
~/Library/Containers/com.microsoft.<AppName>/Data/...  
~/Library/Group Containers/UBF8T346G9.Office/...
```

Read-only access:

The following paths are accessible for read-only access by add-in code. The application path is implicitly available but there is rarely a case that 3rd party code would need to consume content from Office applications and you should avoid doing so. The machine-wide path may contain content you install in order to support your add-in. You should not put installed content inside the “User Content” folder in the machine-wide path other than your add-in itself; choose a new folder unique to your project and place it inside the Microsoft node:

```
/Applications/Microsoft <AppName>.app/...  
/Library/Application Support/Microsoft/...
```

Q&A

1. Why did the Office folder go away? The apps are so large now!
The apps are packaged as required by Apple for delivery via the Mac App Store. While we have not yet released the apps via the App Store we are prepared to do so. Mac App Store rules require that each app be fully self-contained and cannot have externally installed components.
2. Why is modifying the applications so bad?
Each application is code-signed and can be proven to come from Microsoft without any external tampering. Any change to a code-signed application (adding, removing, or altering content inside the .app folder) will cause the application to no longer match its signature, meaning either it did not come from Microsoft or some possibly nefarious tampering occurred. At this time neither the OS nor the applications themselves do anything if they detect the application has been tampered with, but future OS or application releases could decide to prevent the application from being used if it has been altered.
3. Why is the machine-wide location read-only? I want users to be able to modify files in that location.
Apple rules about accessing content outside of the user’s folder, in particular for modifying machine-wide content, are quite strict for the Mac App Store. By default, it is simply forbidden. Even though the apps are not yet available in the Mac App Store, we have designed them to follow Apple’s guidelines and rules for the Mac App Store. Because of Office’s long-time support for extensible content, we compromised on allowing read-only access to this one Microsoft-specific folder tree.
4. That “.localized” extension looks ugly. Do we have to use it?
By default, the OS Finder will hide extensions, so users will usually not see it. The Finder uses that specific extension to trigger additional lookups for localized folder display names, which the Office apps use in now that the applications ship as multi-lingual.
5. Those user content subfolders are confusing. What type of content do I put where?
These folders and their usage has evolved over time. The most common content is likely to be regular file templates and those should go into the Templates folder. If you have questions about specific content and where to put it, let us know and we’ll work with the application teams to clarify what should go where.
6. I am using a volume licensed installation. Why does the machine-wide location have “Office365” in the path?
Branding. Just close your eyes and accept it. This path is rarely (if ever) exposed to the user in the application UI.
7. Can I put our templates on a network share instead?
You can for Word. You need to set the Workgroup templates location in Word’s preferences to point to them (just like in Office 2011) and will need to grant sandbox access to them. The Office applications do not have a predefined sandbox-accepted network path.
8. Can we hide all your built-in and online templates?
Not at this time. However, all user and machine-wide custom templates will always be shown ahead of all built-in templates in the New from Template window.