Fabasoft

White Paper

Fabasoft Integration for WebDAV

2026 November Release

Copyright © Fabasoft R&D GmbH, Linz, Austria, 2026. All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

Contents

1 Introduction 3

2 Software and Infrastructure Requirements 3

2.1 Infrastructure Requirements 3

3 Mapping the User's Desktop to a WebDAV Resource 3

3.1 Structure of Root URL 3

3.2 Overriding the Root Object 3

3.3 Creation of WebDAV Resources 4

4 Configuring WebDAV Integration 4

4.1 Object Classes for Newly Created Folder Objects 4

4.2 Object Classes for Newly Created Content Objects 5

4.3 Object Classes Allowed in Web Folders 8

4.4 Object Classes Not Allowed in Web Folders 8

4.5 File Size Limit 9

4.6 PROPFIND Depth Limit 9

5 Using the WebDAV Integration 9

5.1 Microsoft Windows Clients 9

5.1.1 Activate Basic Authentication 9

5.1.2 Client Configuration for Files Larger Than 30 MB 10

5.1.3 Other WebClient Parameters 10

5.1.4 Integrated Authentication over Proxy Server 10

5.2 Linux Clients 11

5.3 Apple macOS Clients 12

5.4 Programmatic Use 13

6 User Language 14

7 Known Issues 14

7.1 Objects of the Same Name 14

7.2 Objects without a Name 14

7.3 Nautilus on Ubuntu 14

7.4 Finder on macOS 15

8 References 15

1 Introduction

The WebDAV protocol is an extension of the HTTP/1.1 protocol. The extension defines additional HTTP methods for browsing and locking.

The Fabasoft Folio WebDAV integration maps a user's desktop to a hierarchical file view that can be browsed by a WebDAV enabled client. The client can do file system like operations such as copy, move, delete and rename. In addition, objects can be locked, edited and unlocked.

2 Software and Infrastructure Requirements

System environment: All information contained in this document implicitly assumes a Microsoft Windows environment or a Linux environment.

Supported platforms: For detailed information on supported operating systems and software see the software product information on the Fabasoft distribution media.

2.1 Infrastructure Requirements

Firewalls and proxies may need to be configured to enable the WebDAV protocol on the network.

3 Mapping the User's Desktop to a WebDAV Resource

3.1 Structure of Root URL

The root URL of a Fabasoft Folio Web Folder is as follows:

`http://<webserver>/<vdir>/webdav`

The placeholders <`webserver`> and <`vdir`> stand for the web server and the virtual directory of the Fabasoft Folio Web Service. Such a root URL is mapped to the current user's desk object.

3.2 Overriding the Root Object

By default, the root object is the user's desk object. This can be overridden to access any object in the domain with following URL:

`http://<webserver>/<vdir>/webdav/[<override object>-<override attribute>]`

The placeholder `<override object>` stands for the object address. The `<override attribute>` stands for the specific attribute on the object, which is to be displayed. The following example shows a URL that can be used for a HTTP PUT request to create or change a PDF object named "report" in the property `COO.1.1001.1.182590` of object `COO.1.16.4.19`.

Example

```
http://f1.fabasoft.com/fsc/webdav/[COO.1.16.4.19-COO.1.1001.1.182590]/report.pdf
```

Note: If configured correctly, the `<override attribute>` can be omitted, as there is usually only one list per object visible as a WebDAV resource.

`http://<webserver>/<vdir>/webdav/[<override object>]`

Note: The combination of `<override object>` and `<override attribute>` must be a list of objects, as the WebDAV root object is always considered to be a list of elements.

3.3 Creation of WebDAV Resources

The object name of a Fabasoft Folio object (attribute `COOSYSTEM@1.1:objname`) on the user's desktop is mapped to the resource name in the WebDAV resource. The resource names are key values and hence must be unique within a folder (In connection with this topic see chapter 6 "User Language".

Compound objects of the user's desktop map to folders in the WebDAV resource and content objects map to files. The classes of objects actually displayed in a web folder view can be configured in the web service configuration object (see next chapter).

If a compound object has more than one object list (or view) then the object lists are displayed as child folders. These folders are read-only and cannot be deleted, copied, or moved. The name of such a child folder corresponds to the name of the property which contains the object list.

4 Configuring WebDAV Integration

The configuration data for the WebDAV integration is stored in the configuration object `FSCOWS@1.1001:WebServiceConfiguration`. The "Web Folder" form page contains the following properties:

- *Object classes for newly created folder objects*
- *Object classes for newly created content objects*
- *Object classes allowed in web folders*
- *Object classes not allowed in web folders*

4.1 Object Classes for Newly Created Folder Objects

Reference: `FSCOWS@1.1001:webfoldernewfolderclassmappings`

When a user creates a new folder or imports a folder, a new compound object will be created in Fabasoft Folio. The object classes for the new compound objects are defined in this aggregate array. The aggregate consists of the following fields:

- *Object class of the parent folder* (`FSCOWS@1.1001:webfolderparentclass`)
  Specifies the object class of the parent folder.
- *View attribute* (`FSCOWS@1.1001:webfolderviewattr`)
  Specifies the view attribute of the parent compound object, for example `COOSYSTEM@1.1:objchildren`.

- *Object class for new folder objects* (`FSCOWS@1.1001:webfoldernewfolderclass`)
  Specifies the object class for new folder objects to be created.
- *Software component* (`FSCOWS@1.1001:webfolderswc`)
  Specifying a software component for customized entries ensures that the changes are kept beyond updates.

The meaning of an entry depends on the fields specified. You can define object classes for new compound objects but you can also prevent the creation of new folder objects. The following table lists the possible combinations depending on what fields are specified:

Parent class	View attribute	New folder class	Comment
		X	Defines a default folder class for all views in all parent folder classes.
	X		Prevents creation of folder objects in the specified view.
	X	X	Defines a folder class for new folders in the specified view.
X			Prevents creation of folder objects in all views of the specified parent folder class.
X		X	Defines a folder class for new folders in all views of the specified parent folder class.
X	X		Prevents creation of folder objects in the specified view of the parent folder class.
X	X	X	Defines a folder class for new folders in the specified view of the parent folder class.

If there is more than one matching entry for a given case the most suitable entry will be applied.

Note: The parent class takes precedence over the view attribute.

4.2 Object Classes for Newly Created Content Objects

Reference: `FSCOWS@1.1001:webfoldernewcontentobjectclassmappings`

When a user imports or saves a new file, a new content object will be created in the Fabasoft product. The class of the new content object will be chosen depending on the parent folder class, the current view attribute and the file extension of the new file. The aggregate consists of the following fields:

- *Object class of the parent folder* (`FSCOWS@1.1001:webfolderparentclass`)
  Specifies the object class of the parent folder.
- *View attribute* (`FSCOWS@1.1001:webfolderviewattr`)
  Specifies the view attribute of the parent compound object, for example `COOSYSTEM@1.1:objchildren`.
- *File extension* (`FSCOWS@1.1001:webfolderextension`)
  Specifies the file extension of imported or newly created documents.
- *Object class for new content objects* (`FSCOWS@1.1001:webfoldernewcontentobjectclass`)
  Specifies the object class for new content objects to be created.
- *Expression to determine Object Class* (`FSCOWS@1.1001:webfoldernewcontentobjectclassexpr`)
  Specifies the expression to determine the object class for new content objects to be created. If an expression is specified the property *Object class for new content objects* will be ignored. The return value of the expression should be desired object class.
  The global scope contains the following items:
  - `::parent (COOSYSTEM@1.1:OBJECT)`
    Contains the current parent object.
  - `::viewattrdef (COOSYSTEM@1.1:OBJECT)`
    Contains the current view attribute of the current parent object.
  - `::filename (COOSYSTEM@1.1:STRING)`
    Contains the file name of the new content object to be created.
  - `::fileextension (COOSYSTEM@1.1:STRING)`
    Contains the file extension of the new content object to be created.
  - `::skipline (COOSYSTEM@1.1:BOOLEAN)`
    Set the value of this item to `TRUE` if the expression cannot determine an object class but the creation of the new content object should not be prevented. In this case the configuration will be evaluated again skipping the current configuration line.
- *Software components* (`FSCOWS@1.1001:webfolderswc`)
  Specifying a software component for customized entries ensures that the changes are kept beyond updates.

The meaning of an entry depends on the fields specified. You can define object classes for new content objects but you can also prevent the creation of new content objects. The following table lists the possible combinations depending on what fields are specified:

Parent class	View attribute	File ext	New object class	Comment
			X	Default content object class for all views of all parent classes and any file extension.
		X		Prevents the creation of content objects with the given file extension in all views of all parent classes.

		X	X	Defines the content object class for all views of all parent classes for the given file extension.
	X			Prevents the creation of content objects in the specified view for all parent classes and any file extension.
	X		X	Defines the content object class for the given view in all parent classes and any file extension.
	X	X		Prevents the creation of content objects in the specified view for all parent classes and the specified file extension.
	X	X	X	Defines the content object class for the specified view for all parent classes and the specified file extension.
X				Prevents the creation of content objects in all views of the specified parent class for all file extensions.
X			X	Defines the content object class for all views of the specified parent class and any file extension.
X		X		Prevents the creation of content objects in all views for the specified parent class and file extension.
X		X	X	Defines the content object class for all views in the specified parent class and given file extension.
X	X			Prevents the creation of content objects in the specified view and parent class for any file extension.
X	X		X	Defines the content object class for the specified view and parent class and any file extension.
X	X	X		Prevents the creation of content objects in the specified view and parent class for the given file extension.
X	X	X	X	Defines the content object class for the specified view and parent class for the given file extension.

If there is more than one matching entry for a given case, the most suitable entry will be applied.

Note: The parent class takes precedence over the view attribute and the view attribute takes precedence over the file extension.

4.3 Object Classes Allowed in Web Folders

Reference: `FSCOWS@1.1001:webfolderincludedclasses`

This aggregate array lists all object classes that should be displayed in web folders. You may also specify specific views of compound objects. The aggregate consists of the following fields:

- *Object class* (`FSCOWS@1.1001:webfolderobjclass`)
  Specifies the object class to be included in web folders.
- *View attribute* (`FSCOWS@1.1001:webfolderviewattr`)
  You may select a specific child view of a compound object. If you leave this attribute blank, all views of the compound object are displayed in web folders.
- *Show PDF Summary* (`FSCOWS@1.1001:webfolderalternatecontent`)
  Set the value of this property to *Yes* to display instances of the desired object class as PDF documents. This setting makes sense for object classes without a content to display a PDF summary of the meta data.
- *Software component* (`FSCOWS@1.1001:webfolderswc`)
  Specifying a software component for customized entries ensures that the changes are kept beyond updates.

Note: An empty aggregate array means that all object classes should be included in the web folder view.

4.4 Object Classes Not Allowed in Web Folders

Reference: `FSCOWS@1.1001:webfolderexcludedclasses`

This aggregate array lists all object classes that should not be displayed in web folders. You may also specify specific views of compound objects. The aggregate consists of the following fields:

- *Object class* (`FSCOWS@1.1001:webfolderobjclass`)
  Specifies the object class to be excluded in web folders.
- *View attribute* (`FSCOWS@1.1001:webfolderviewattr`)
  You may only select a specific child view of a compound object that should not be displayed. If you leave this attribute blank, the entire compound object will not be displayed.
- *Software component* (`FSCOWS@1.1001:webfolderswc`)
  Specifying a software component for customized entries ensures that the changes are kept beyond updates.

Note: The exclusion list takes precedence over the inclusion list.

4.5 File Size Limit

The file size limit for uploading is defined in the *Default Import Limit (Bytes)* field in the virtual application configuration.

Note: Currently the *Default Import Limit (Bytes)* field in the virtual application configuration is only considered when using the Fabasoft Client via "Open <product> Folder" (see screenshot option marked in green). When opening a network drive with a local WebDAV Client, always a 4 GB limit applies. This can be done like in the screenshot marked in red "Open <product> Network Drive".

4.6 PROPFIND Depth Limit

The depth limit for a PROPFIND request is determined by the Registry Key (DWORD) `HKEY_LOCAL_MACHINE\SOFTWARE\Fabasoft\FSCOWS@1.1001\PROPFINDMaxDepth`.

The default value is 5.

5 Using the WebDAV Integration

5.1 Microsoft Windows Clients

The Fabasoft Folio desk can be mounted as a drive in the Microsoft Windows Explorer. It is also possible to connect to Fabasoft Folio as a Microsoft Windows share (e.g. `\\webserver\fsc\`).

To ensure a correct functionality, be sure to enable and start the "Web Client" Windows service.

Connecting to Fabasoft Folio via Network Drive

- Click "Map Network Drive" in the Windows Explorer and enter the server's address in the field "Folder". The address must be built like this: `https://webserver/fsc/webdav`
- Click "Finish" to map the network drive. Credential input may be required.

5.1.1 Activate Basic Authentication

Microsoft Windows 11

- Open the Microsoft Windows Registry editor.
- Navigate to the key: `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient\Parameters`
- Assign a value of 2 (dec) to the key `BasicAuthLevel`.
- Restart the computer.

Note: As standard you cannot open Office file types directly from a server that supports only basic authentication over a non-SSL connection. To resolve this problem, consult http://support.microsoft.com/kb/2123563/en-us.

5.1.2 Client Configuration for Files Larger Than 30 MB

To use files, which are larger than 30 MB with WebDAV the file size limit need to be changed on the client.

Microsoft Windows 11

- Open the Microsoft Windows Registry editor.
- Navigate to the key:
  `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient\Parameters`
- Assign a value of 300000000 (dec) to the key `FileSizeLimitInBytes`.
- Restart the computer.

5.1.3 Other WebClient Parameters

- `SendReceiveTimoutInSec`
  This parameter (DWORD) specifies the timeout in seconds that WebDAV the WebClient service uses after issuing a request, such as "GET /file.ext" or "PUT /file.ext". The default value is 60 seconds.
- `InternetServerTimeoutInSec`
  This parameter (DWORD) specifies the connection timeout for the WebClient service uses when communicating with non-local WebDAV servers. The default value is 30 seconds.
- `FileAttributesLimitInBytes`
  This parameter (DWORD) specifies the maximum size that is allowed by the WebClient service for all properties on a specific collection. This is necessary if a large number of files and directories are transmitted.

5.1.4 Integrated Authentication over Proxy Server

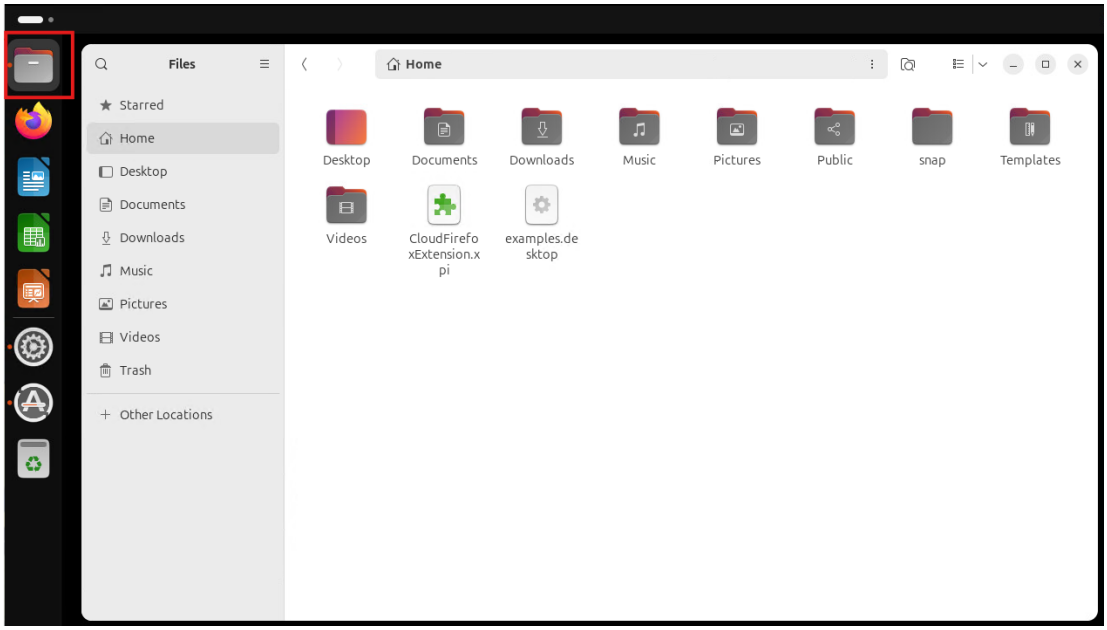
- Microsoft Windows 11
  - Open the Microsoft Windows Registry editor.
  - Navigate to the key:
    `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient\Parameters`
  - Create a new Multi-String Value `AuthForwardServerList`
  - Enter the URLs of the Fabasoft Folio webserver (e.g. "http://localhost")
  - Restart the computer.

Detailed information can be found at: http://support.microsoft.com/kb/943280/en-us

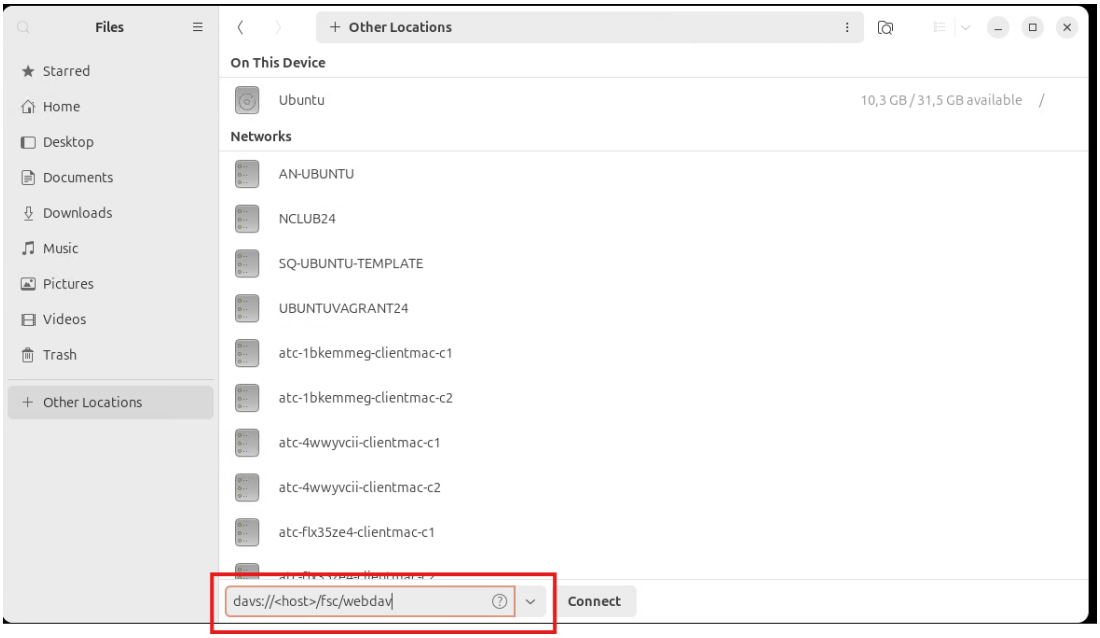
5.2 Linux Clients

To mount a Fabasoft WebDAV resource on a Linux system with the Gnome desktop environment, perform following steps:

1. Open the Nautilus file manager (Files application).



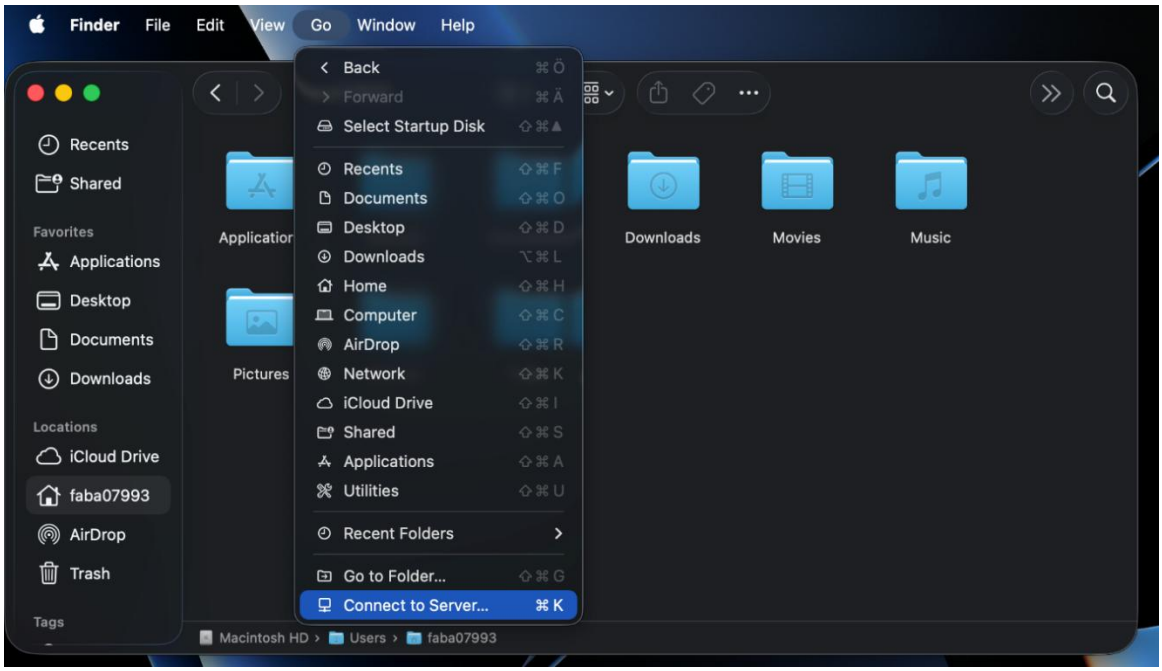
2. On the left panel, click “Other Locations”. In the new view, locate the address bar at the bottom, type the WebDAV server URL (`dav://<hostname>/fsc/webdav`), and press “Connect”. For secure connections, use **`davs`**`://<hostname>/fsc/webdav` instead.



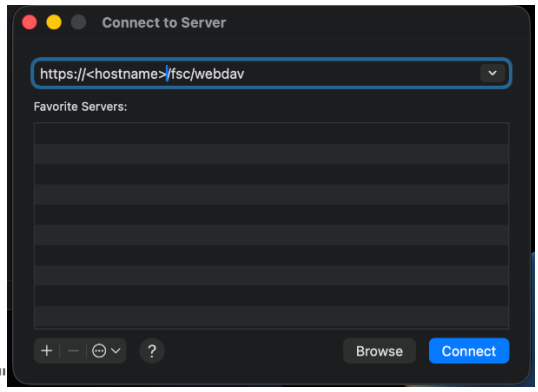
5.3 Apple macOS Clients

To mount a Fabasoft WebDAV resource on Apple macOS, perform the following steps:

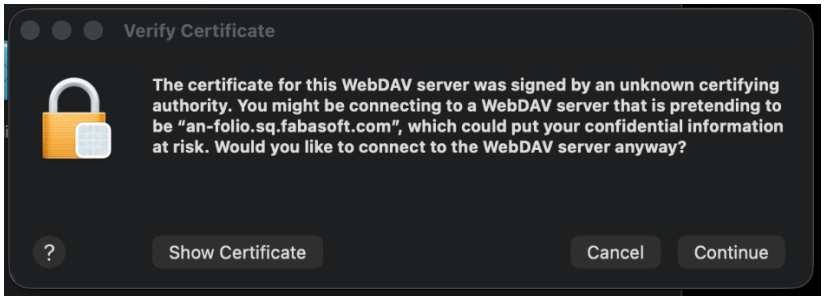
1. Open Finder and on the top bar click “Go” > “Connect to Server”.



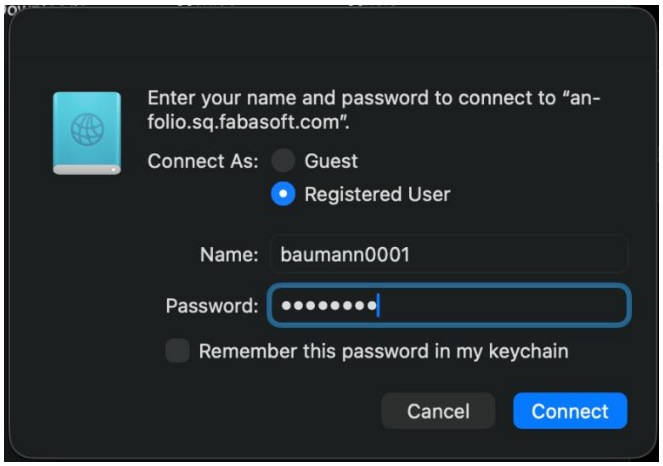
2. Enter the server address following this template **Error! Hyperlink reference not valid.** and click “Connect”.



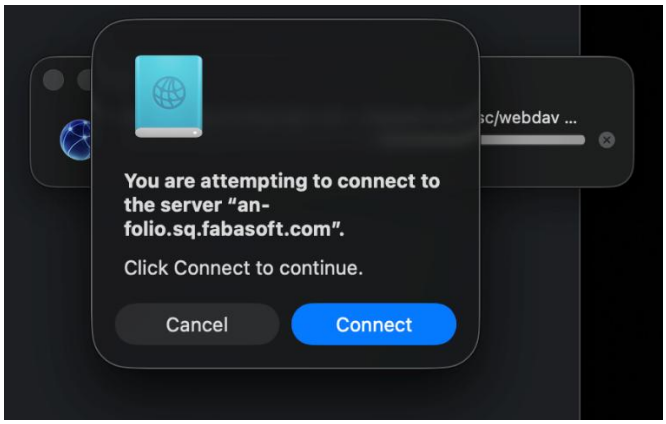
3. Verify the certificate and click “Continue”.



4. Enter your credentials and click “Connect”.



5. Confirm the connection by clicking “Connect”.



5.4 Programmatic Use

The following example shows how to get the metadata of all objects in an object list.

Example

```
curl -X PROPFIND -H "Depth: 1" -G -u user:passwd
'http://localhost/fsc/webdav/\[COO.1.506.1.1002293-objchildren\]'
```

Result:

```
...
<D:response>
  <D:href>/fsc/webdav/[COO.1.506.1.1002293-objchildren]/doc1.docx</D:href>
  <D:propstat>
    <D:prop>
      <lp2:COOSYSTEM_1_1_objaddress xmlns:lp2="http://schemas.fabasoft.com/swc/">
        COO.1.506.1.1002295
      </lp2:COOSYSTEM_1_1_objaddress>
      ...
      <D:iscollection>0</D:iscollection>
    </D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
</D:response>
...
```

The retrieved metadata can be used to download documents.

Example

```
curl -G -u user:passwd 'http://localhost/fsc/webdav/\[COO.1.506.1.1002293-
objchildren\]/doc1.docx' > doc1.docx
```

6 User Language

It has to be considered that in case of WebDAV requests the `Accept-Language` header of the HTTP request will be ignored. If the current user has not configured a user language in the user environment the default language of the Fabasoft Folio domain will be used.

7 Known Issues

7.1 Objects of the Same Name

The object name of a Fabasoft Folio object on the user's desktop specifies the name for the created resource in the WebDAV resource. Within a web folder the resource names are key values and hence must be unique.

In Fabasoft software products objects can be of the same name. When using web folders you should try to apply a name only to one object of a certain class. Otherwise a collision will occur.

7.2 Objects without a Name

Fabasoft Component objects without a name cannot be displayed correctly in the WebDAV resource because there is no name for the created resource. When using web folders you should try to avoid objects without a name on the user's desktop.

7.3 Nautilus on Ubuntu

The Nautilus client on Ubuntu has troubles by creating and opening documents and folders on WebDAV resources located on an Apache web server. In this case Fabasoft Folio can be accessed by mounting the root URL (`http://<webserver>/<vdir>/webdav`) with the option `davfs` as follows:

```
sudo mount -t davfs http://<webserver>/<vdir>/webdav <local directory> -o
uid=<local user>
```

Sample:

```
sudo mount -t davfs http://folioserver/fsc/webdav ~/foliowebdav -o uid=foliouser
```

Note: Due to synchronization issues, the davfs2 `delay_upload` option has to be set to `0` in `/etc/davfs2/davfs2.conf`.

7.4 Finder on macOS

Windows Server in combination with Windows Authentication can lead to errors, for example when uploading a file. The errors can be worked around by deactivating Windows Authentication in the Internet Information Services (IIS) Manager.

8 References

Goland, Yaron Y. et al: RFC 2518: HTTP Extensions for Distributed Authoring - WebDAV. February 1999. URL: http://www.ietf.org/rfc/rfc2518.txt.

Fielding, Roy T. et al: RFC 2616: Hypertext Transfer Protocol - HTTP/1.1. June 1999. URL: http://www.ietf.org/rfc/rfc2616.txt.