



pdfaPilot

**callas
pdfaPilot**

Table of Contents

1. Installation, activation, deactivation, updates.....	6
1.1 Installation of pdfaPilot Desktop or Server on Mac	7
1.2 Installation of pdfaPilot Desktop or Server on Windows.....	15
1.3 Activation and Deactivation of pdfaPilot	20
1.4 Multi-user system (Activating additional users on the same system).....	31
1.5 Migrating the pdfaPilot to another computer (Deactivation).....	32
1.6 Uninstalling the pdfaPilot Acrobat plugin	37
1.7 Change language of user interface (desktop only)	38
1.8 Using pdfaPilot and Preflight in Acrobat Pro parallel.....	40
1.9 Notifications about available updates	49
1.10 Update path to pdfaPilot CLI in Switch when installing new version of pdfaPilot.....	55
1.11 Run as a service (Windows)	56
1.12 Create a Daemon using Mac	63
1.13 Run as an autostart systemd service (Linux).....	64
2. callas pdfaPilot - First steps.....	68
2.1 Analyzing and converting files using “PDF/A in one click”	69
2.2 Preferences	77
2.3 The Switchboard: An overview	82
2.4 Switchboard: PDF/A & PDF/UA	97
2.5 Switchboard: Embedded files - Create file package	101
2.6 Switchboard: Office/E-mail to PDF	105
2.7 Switchboard: Document – Overlay	114
2.8 Switchboard: Pages - Scale and enlarge.....	123
2.9 Create Booklet	134
2.10 Switchboard: Arrange – Reader spreads/Split in half	137
2.11 Switchboard: Tagged PDF – HTML export	144
2.12 Switchboard: Metadata – Browse metadata	148
2.13 Convert entire folder into an archive PDF	151
2.14 Organize pages.....	154

2.15 PDF/UA-1 validation for standards-compliant universally accessible PDFs	160
2.16 Improved merging of adjacent heading (H1..H6) tags	172
2.17 Explore Fonts	177
2.18 Explore PDF	179
2.19 Keyboard shortcuts for pdfaPilot.....	193
3. callas pdfaPilot Basics	197
3.1 Fixups, Checks, Profiles, Process Plans and Libraries	198
3.2 Protect Fixups, Checks, Profiles, Process Plans and Libraries against changes	204
3.3 Duplicate and edit Profiles, Checks and Fixups	207
3.4 Creating Profiles	222
3.5 Deleting multiple Fixups or Checks at once	243
3.6 Actions in Process Plans	245
3.7 Batch processing for Actions	254
3.8 Libraries - Overview	257
3.9 Create new, Manage, Import or Export Libraries	261
3.10 Search in or across Libraries	271
3.11 Examining page content: The Object Inspector	276
3.12 Examining page content: Filter in the Object Inspector	279
3.13 Requirements for conversions to PDF	282
3.14 Checking the ISO standards using callas Desktop products	296
3.15 Modifying structures in a Process Plan.....	299
3.16 Process Plan to create bookmarks from headings	300
3.17 PDF reports (using masks or layers)	303
3.18 HTML based custom reports	310
3.19 HTML based custom reports – generate optional information	316
4. Decorate pages by adding content.....	331
4.1 Actions: Decorate	332
4.2 Place folding marks	344
4.3 Add text	346
4.4 Place page number	349
4.5 Place date.....	353

4.6 Add logo/image.....	359
4.7 Add watermark	362
4.8 Place mail stamp	364
4.9 Place letter background	366
4.10 Add summary page.....	370
4.11 Add divider page.....	372
4.12 Move objects	374
5. Working with XMP Metadata and PDF/A files.....	376
5.1 Extension schema editor.....	377
5.2 Inserting XMP Metadata into PDFs using callas pdfaPilot Desktop.....	392
6. callas pdfaPilot CLI.....	401
6.1 Introduction to pdfaPilot CLI	402
6.2 Installation and activation of pdfaPilot Server/CLI	405
6.3 Hints and trouble shooting & Displaying program information.....	411
6.4 Converting office documents to PDF or PDF/A.....	416
6.5 General command line options.....	419
6.6 Additional command line options and response files	424
6.7 PDF/A specific command line options	428
6.8 File content command line options.....	431
6.9 Font embedding command line options.....	435
6.10 Creating file packages	438
6.11 Converting email files to PDF	441
6.12 Creating HTML or EPUB.....	446
6.13 Creating ZUGFeRD / Factur-X invoices.....	448
6.14 Report creation and command line output.....	451
6.15 Results (Return codes - Error codes - Reason codes).....	458
6.16 Run as a server.....	465
6.17 Distributed processing	469
6.18 Running pdfaPilot via Webservices (SOAP)	480
6.19 Handling Licensing through the License Server.....	483
6.20 Modifying structure of bookmarks and DPart.....	484

7. callas pdfaPilot Server	492
7.1 Working with hot folders in pdfaPilot Server	493
7.2 Start a Server / connect with Remote Server	495
7.3 Job settings.....	500
8. callas pdfaPilot SDK.....	503
8.1 Activation and Deactivation of pdfaPilot SDK.....	504
8.2 callas pdfEngine SDK: First steps.....	507
8.3 Help? Displaying program information for pdfaPilot SDK.....	509
8.4 File components and their use in the SDK	510

1. Installation, activation, deactivation, updates

1.1 Installation of pdfaPilot Desktop or Server on Mac

You can download the latest version of pdfaPilot Desktop or Server from our website by requesting a trial:

1. pdfaPilot Desktop: [pdfapilot-desktop](#)
2. pdfaPilot Server: [pdfapilotserver](#)

You can also [register on our website](#) for easy access to all pdfaPilot download links. Once registered and logged in, you can access the list of download links here:

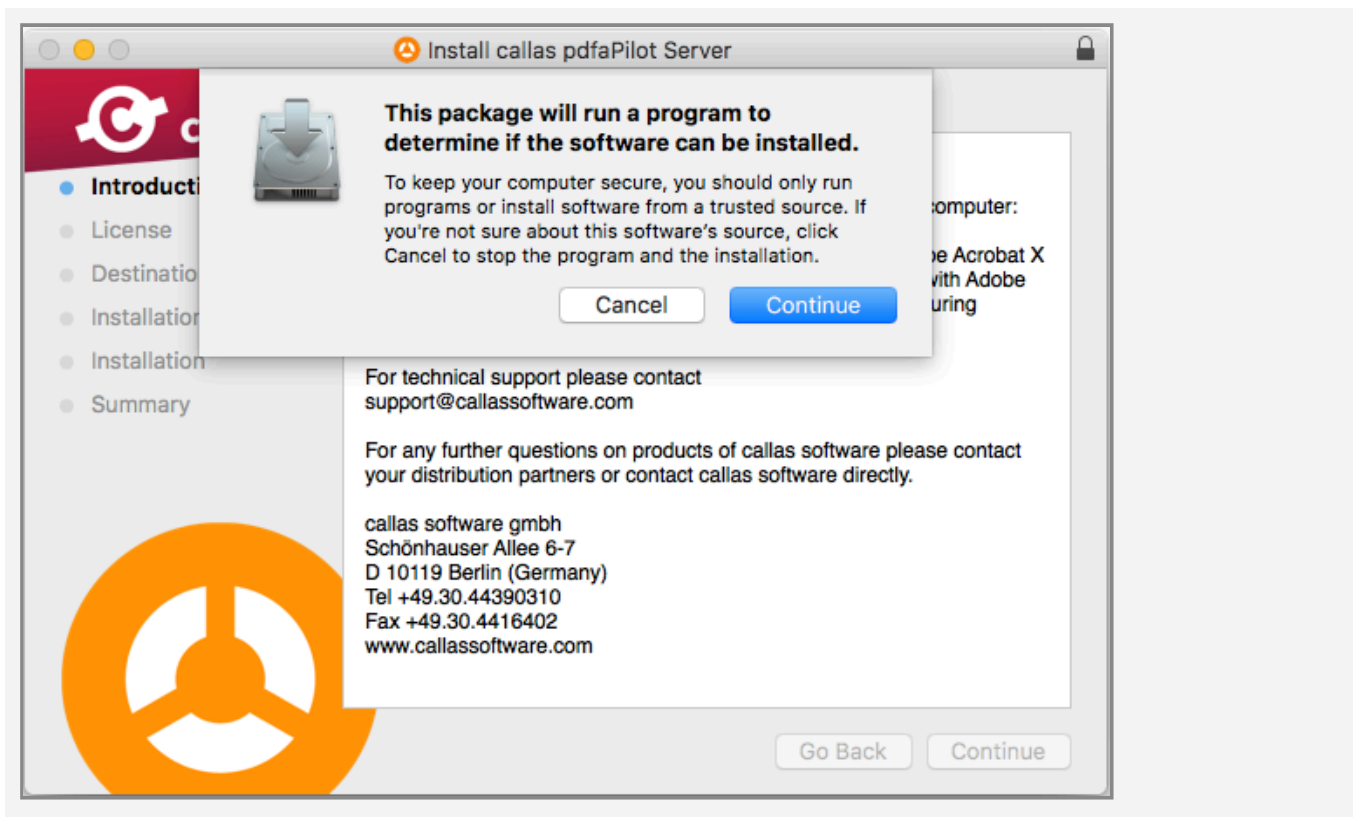
1. pdfaPilot Desktop: [pdfapilot-desktop](#)
2. pdfaPilot Server: [pdfapilotserver](#)

Installation on Mac



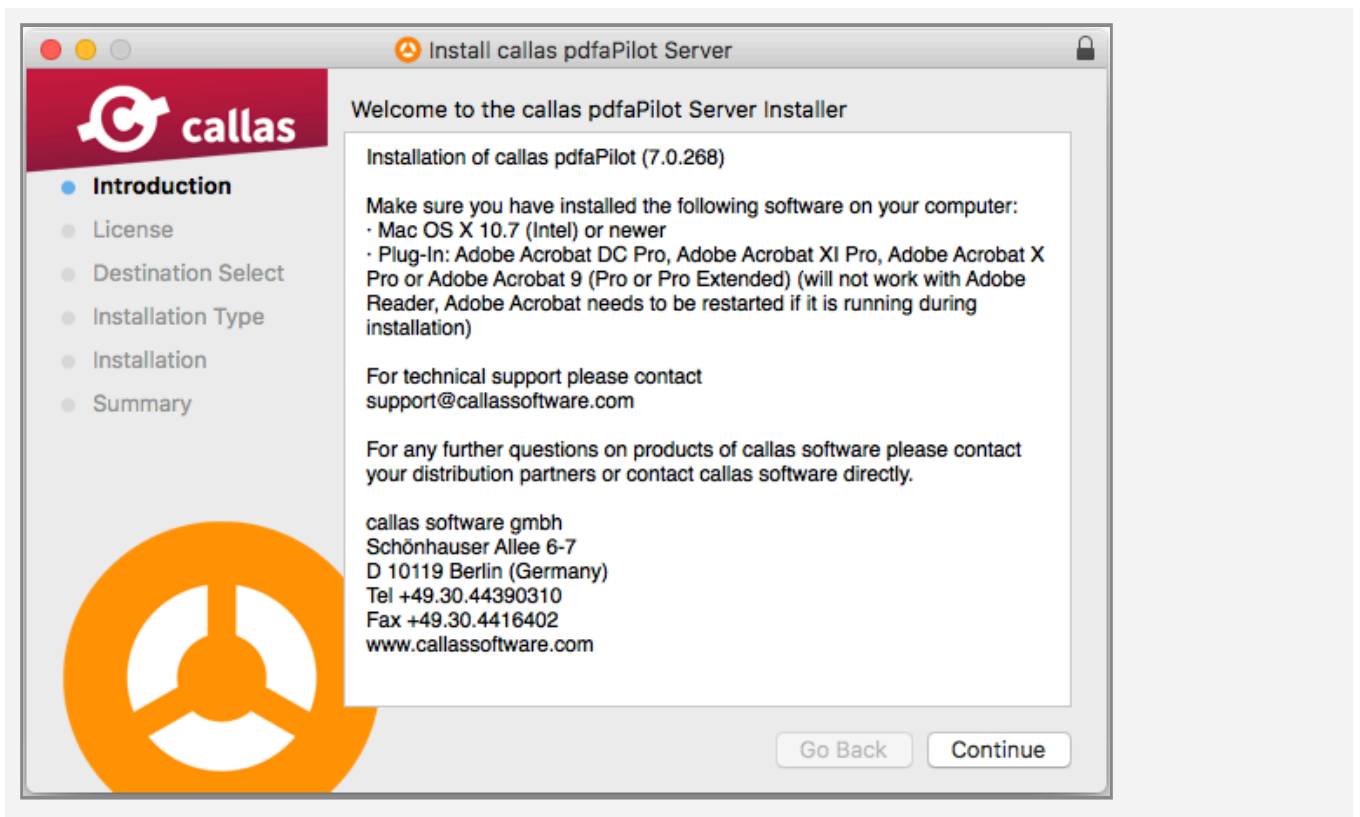
To install the application, double click the pdfaPilot icon.

System Check



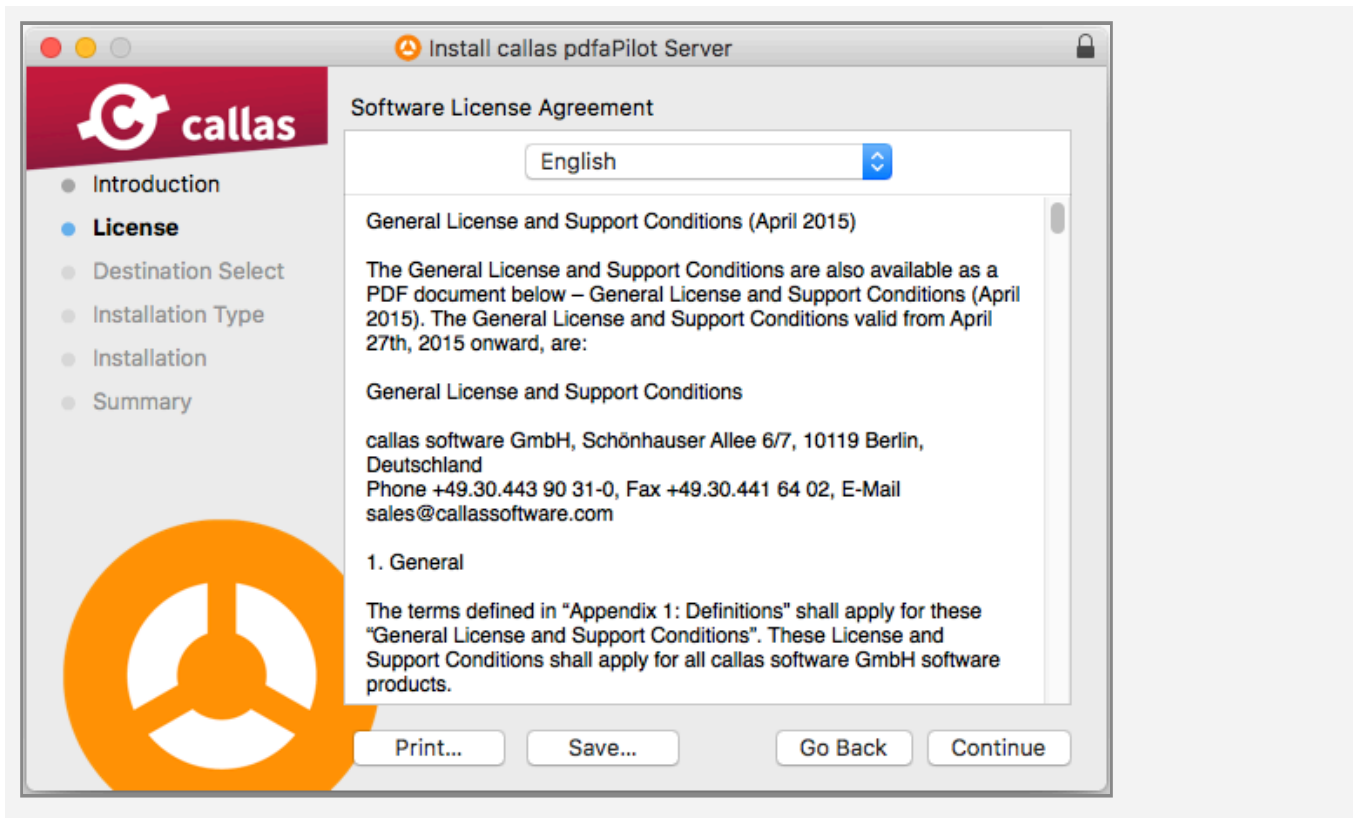
The package will then check the integrity of the installer.

Introduction Screen



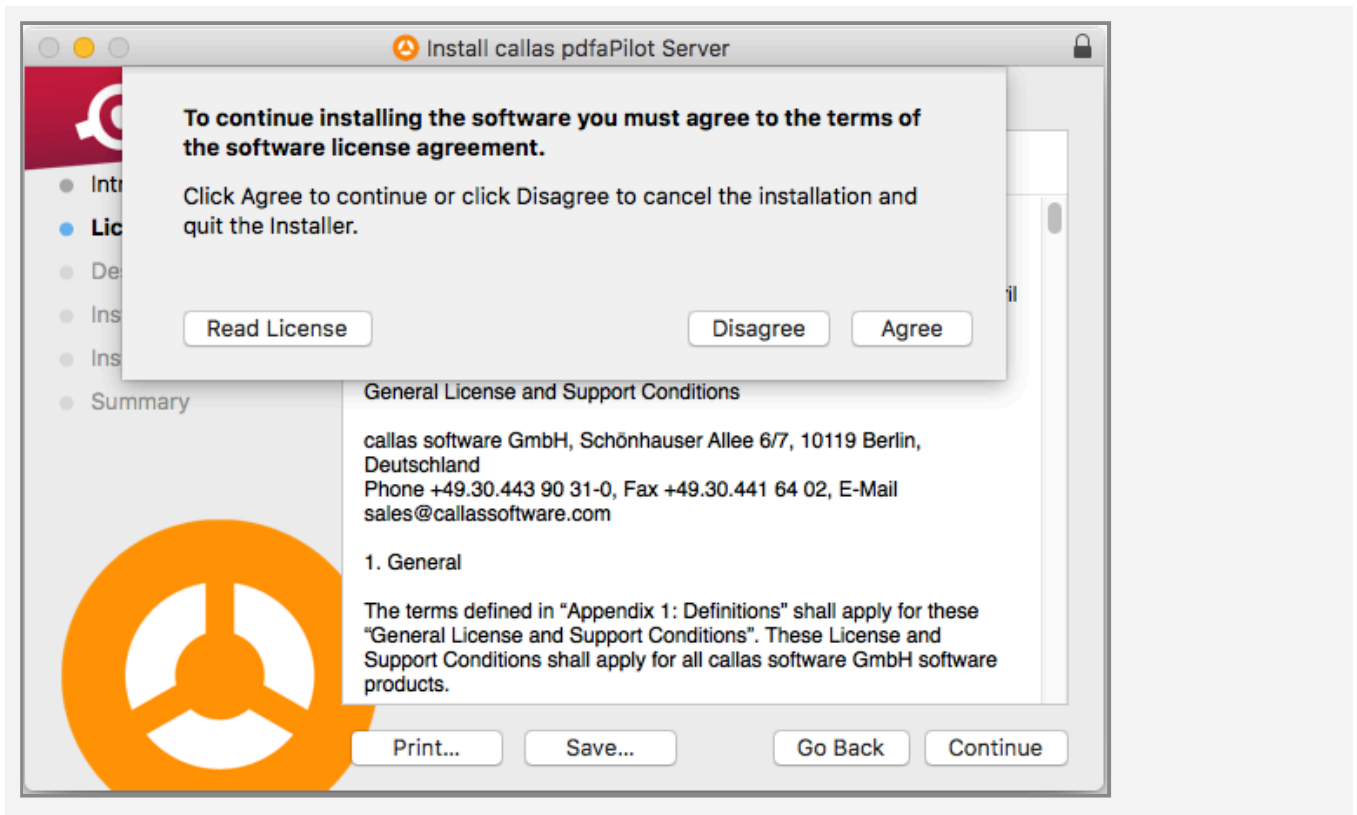
Please check that your system supports the requirements

License Dialog



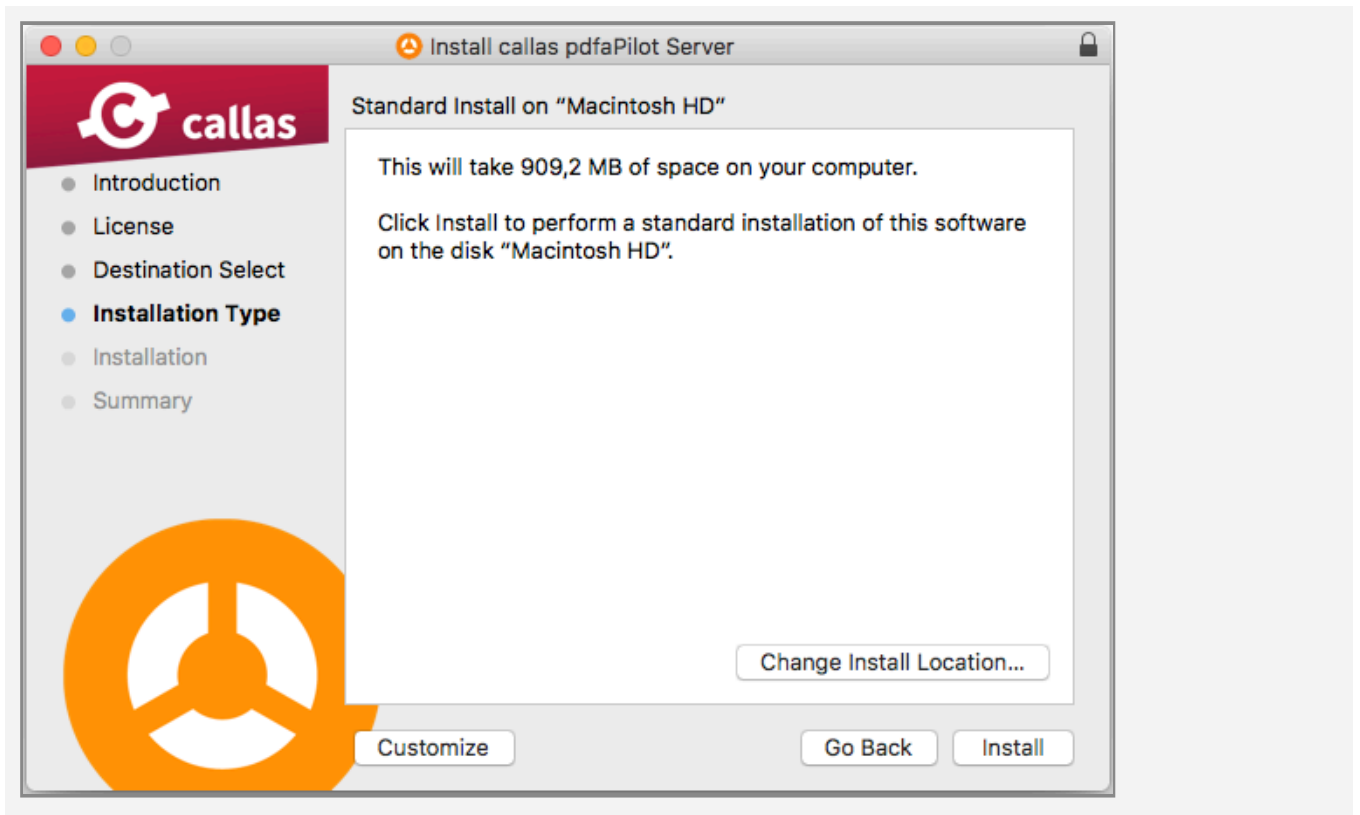
Please read the License agreement

Agree to the terms of agreement

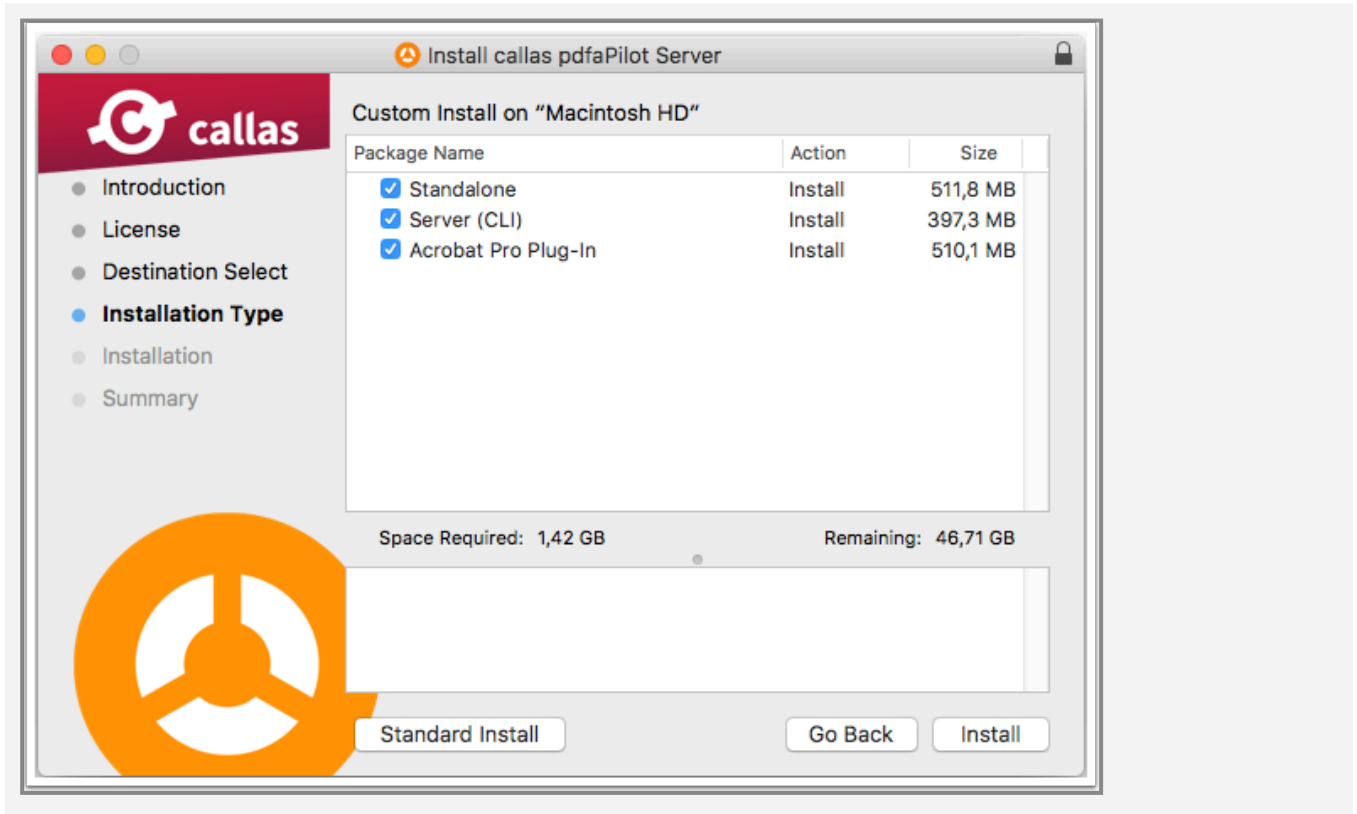


Click on "Continue" if you wish to accept the terms and and install the software.

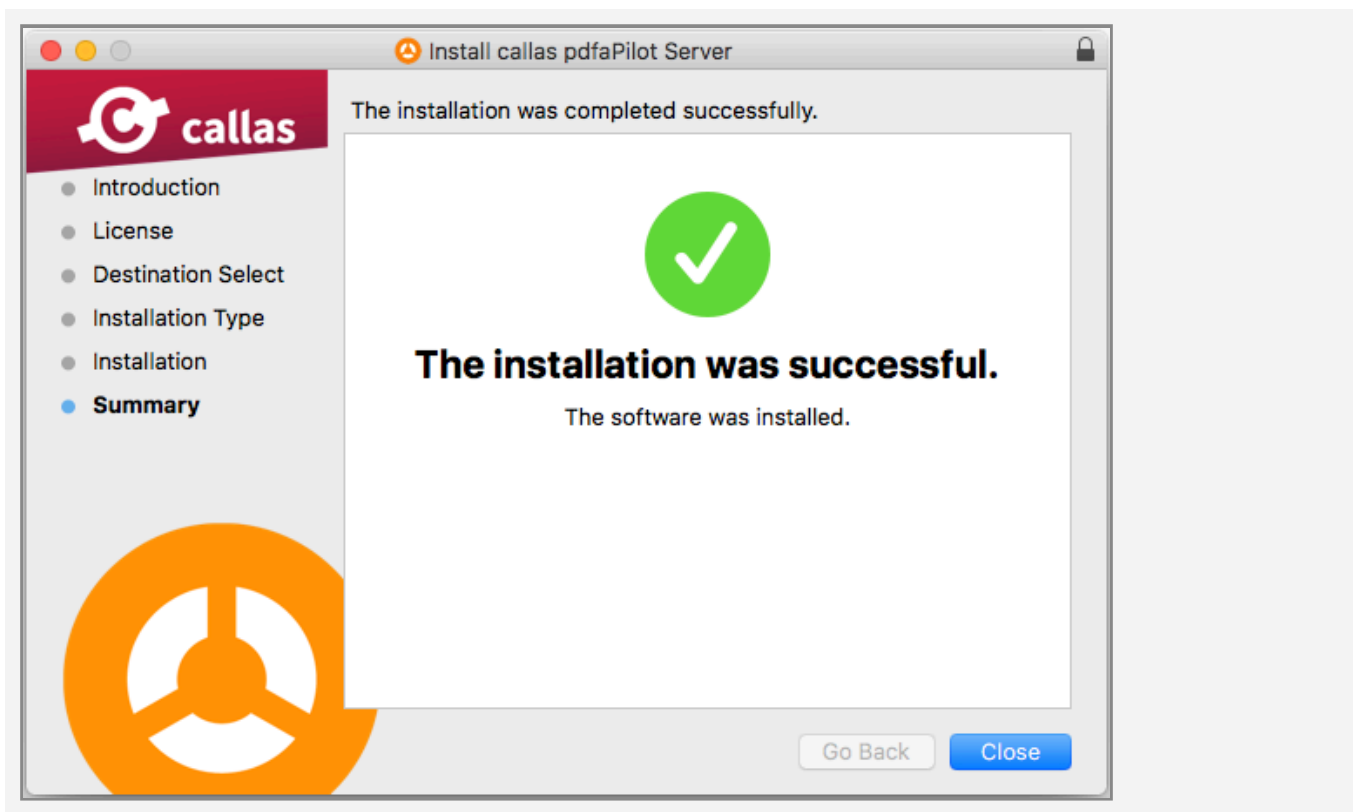
Installation type



It is possible to Install the Desktop version and the Plugin version together or only the Standalone version. If you do not wish to install the Plugin, click on "Customize" and uncheck the "Plugin" installation.



Installation was successful



1.2 Installation of pdfaPilot Desktop or Server on Windows

You can download the latest version of pdfaPilot Desktop or Server from our website by requesting a trial:

1. pdfaPilot Desktop: [pdfapilot-desktop](#)
2. pdfaPilot Server: [pdfapilotserver](#)

You can also [register on our website](#) for easy access to all pdfaPilot download links. Once registered and logged in, you can access the list of download links here:

1. pdfaPilot Desktop: [pdfapilot-desktop](#)
2. pdfaPilot Server: [pdfapilotserver](#)

Unattended installation

It is also possible to run the installation without any user interaction.

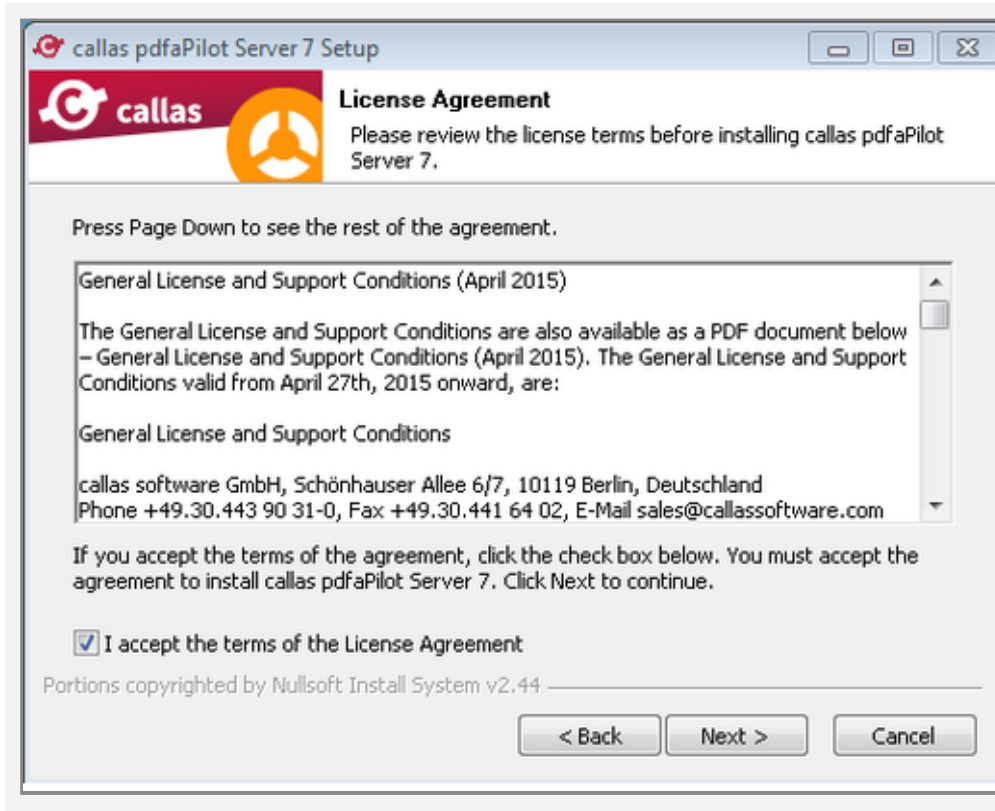
With the additional parameter /S the application will be installed at the default location on Program Files.

```
C:\>"C:\Users\Administrator\Desktop\callas  
pdfaPilot 7.exe" /S
```

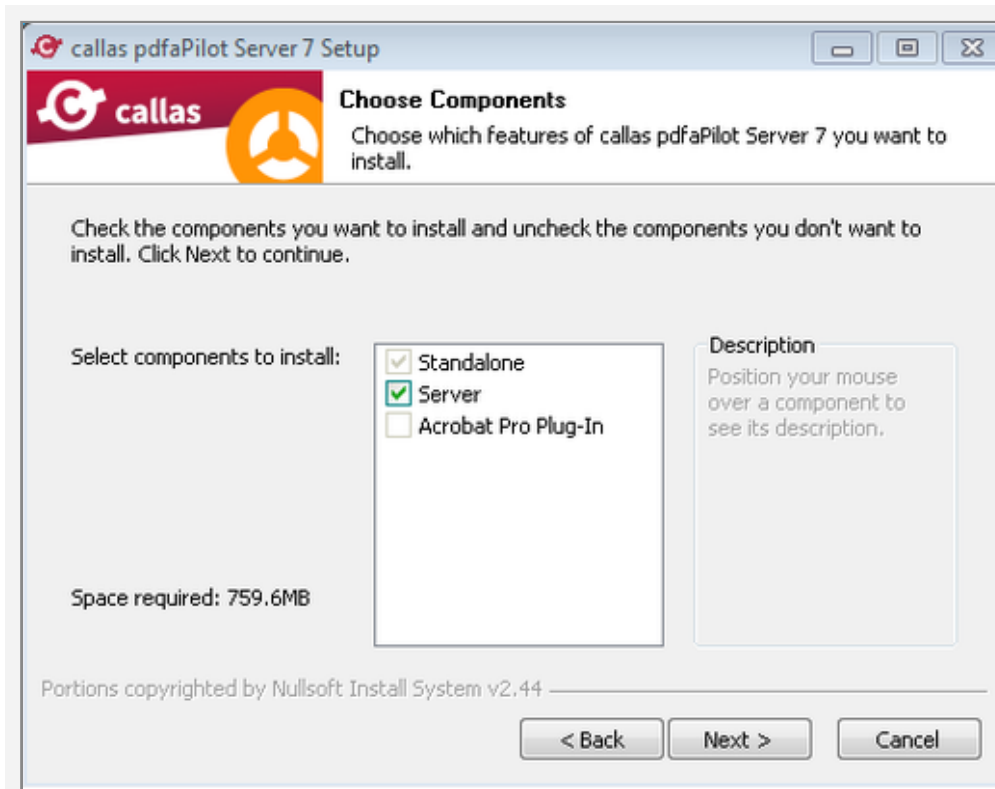
Installation with user interaction



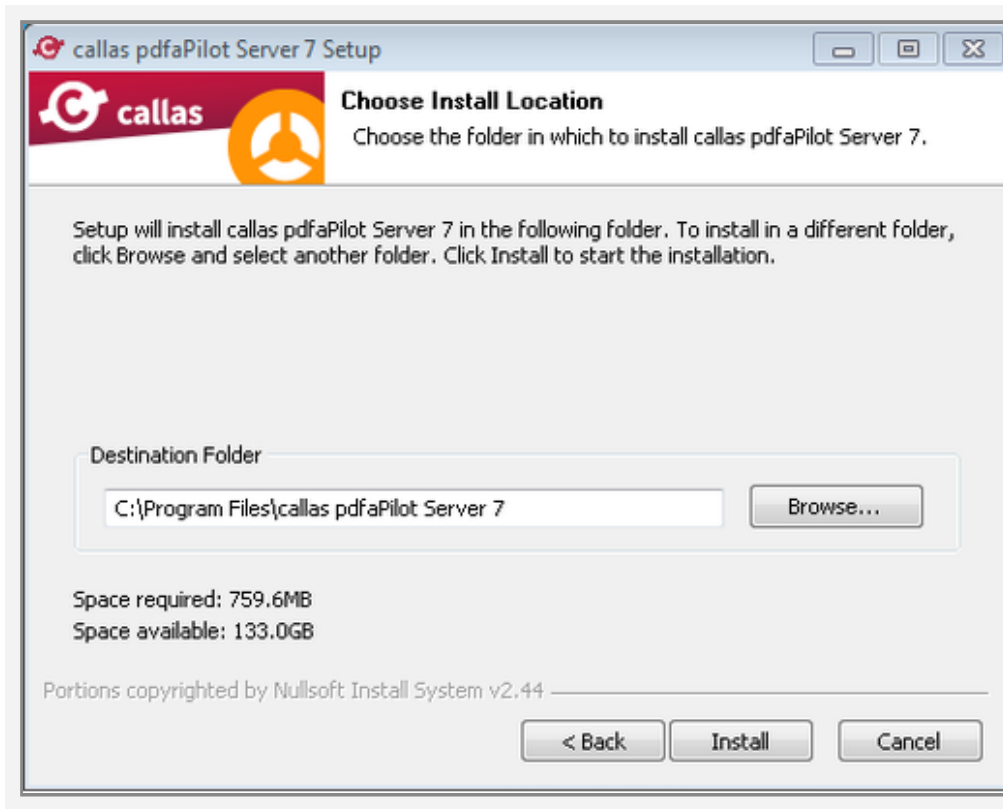
Short Introduction, continue with "Next".



Please read the License and Support Conditions carefully and click on "Next" if you agree.



Now select the components you want to install.



You can install the application on the default location in program files, else this can be changed by clicking on "Browse..." and setting path to the custom location.



Installation of pdfaPilot is now finished. Clicking on "Finish" will run pdfaPilot Desktop Standalone.

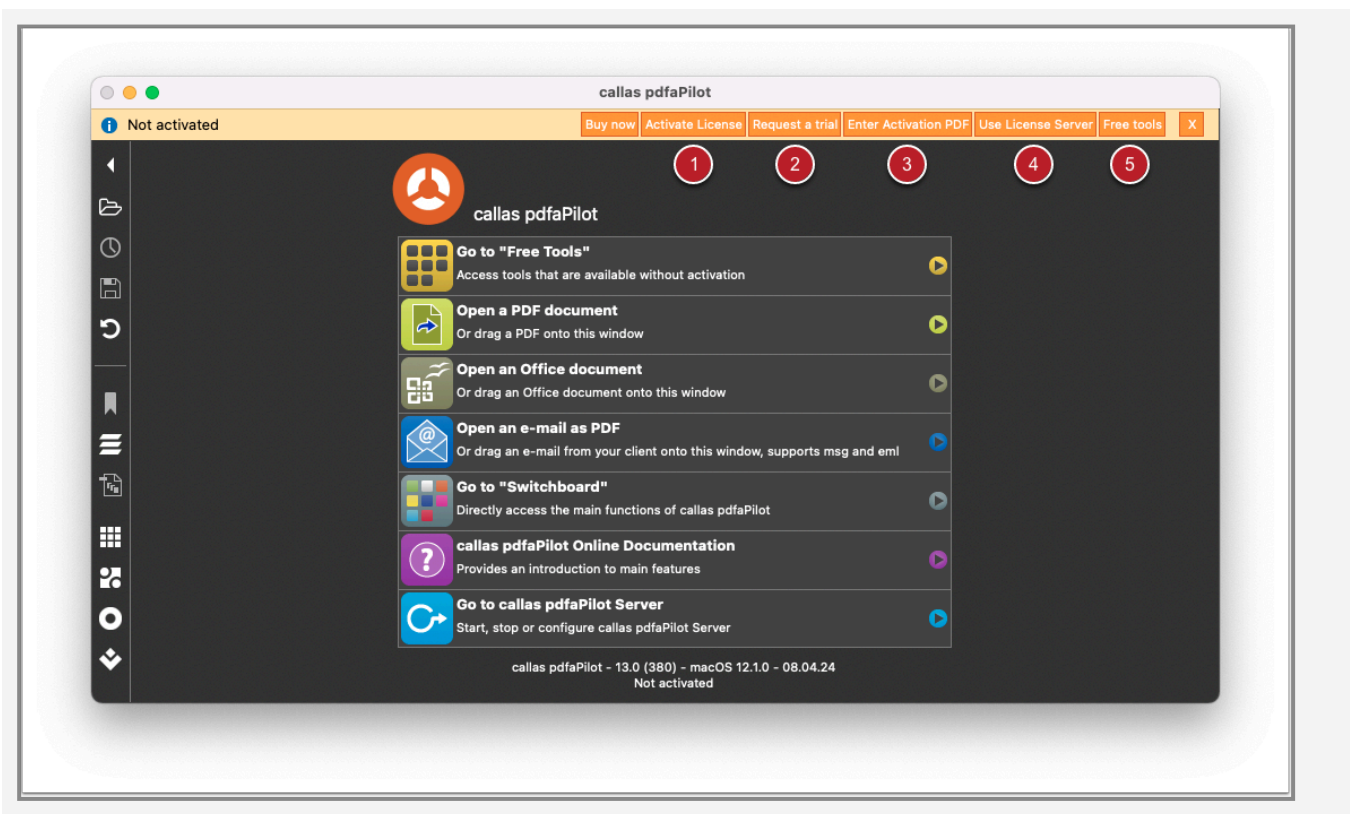
1.3 Activation and Deactivation of pdfaPilot

Main application window

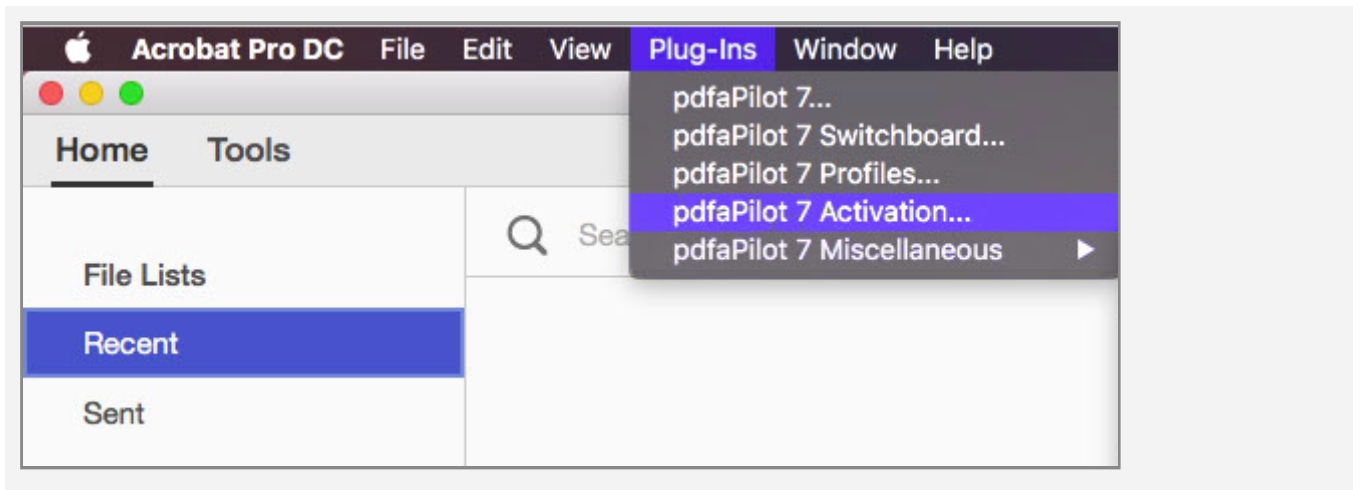
To use any callas product on a computer, you need to activate it first. This is true if you have bought a license key but also if you want to run the trial software. This article explains how the normal activation procedure for pdfaPilot Desktop and DeviceLink Add-on Desktop works and which steps you have to go through.

Launch the application. You will see the main window with a blue bar at the top with the following options:

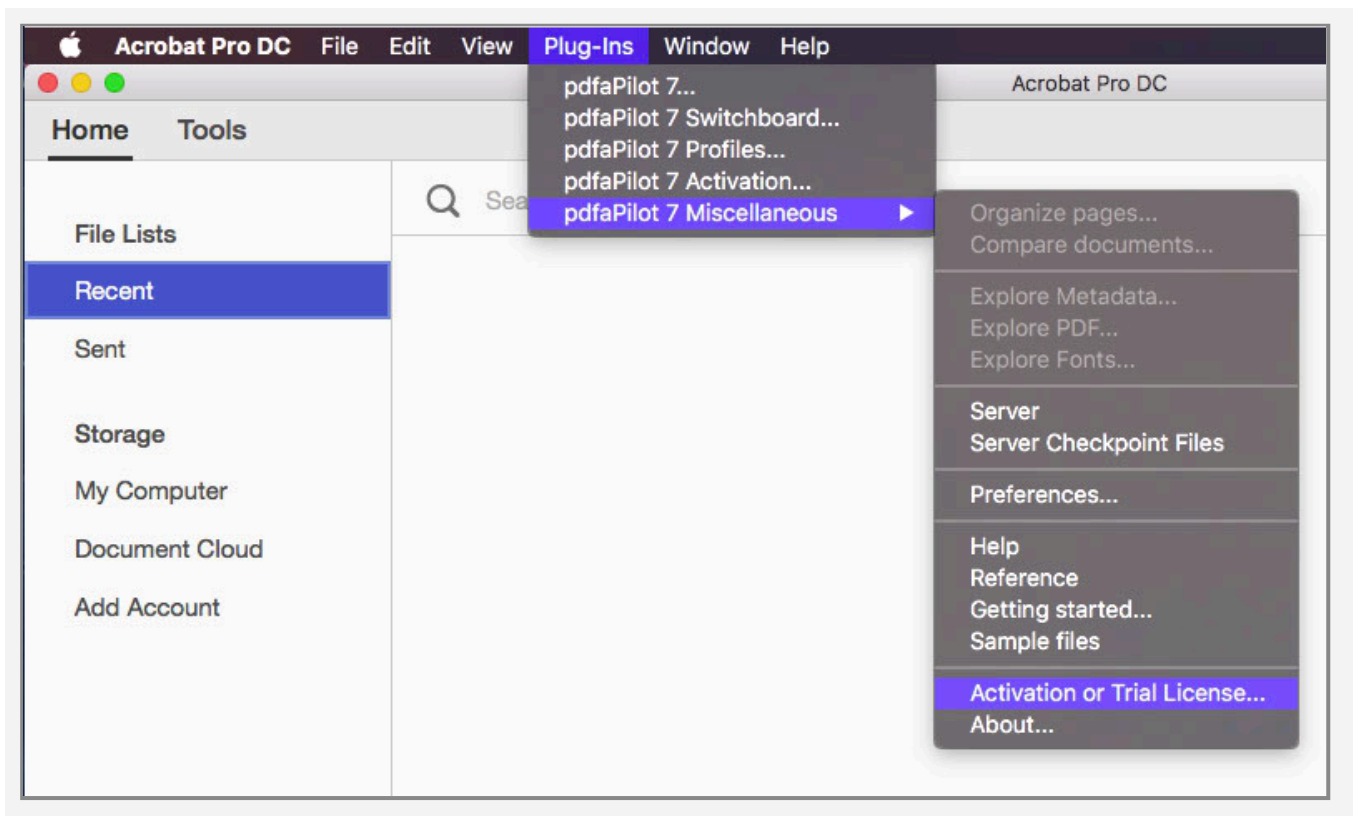
1. [Activate License \(if you have purchased a license key\)](#)
2. [Request a trial \(2-week trial license\)](#)
3. [Enter Activation PDF](#)
4. [Use License Server](#)
5. [Free tools](#)



Activation with the Acrobat Plugin



Alternatively go to this sub menu



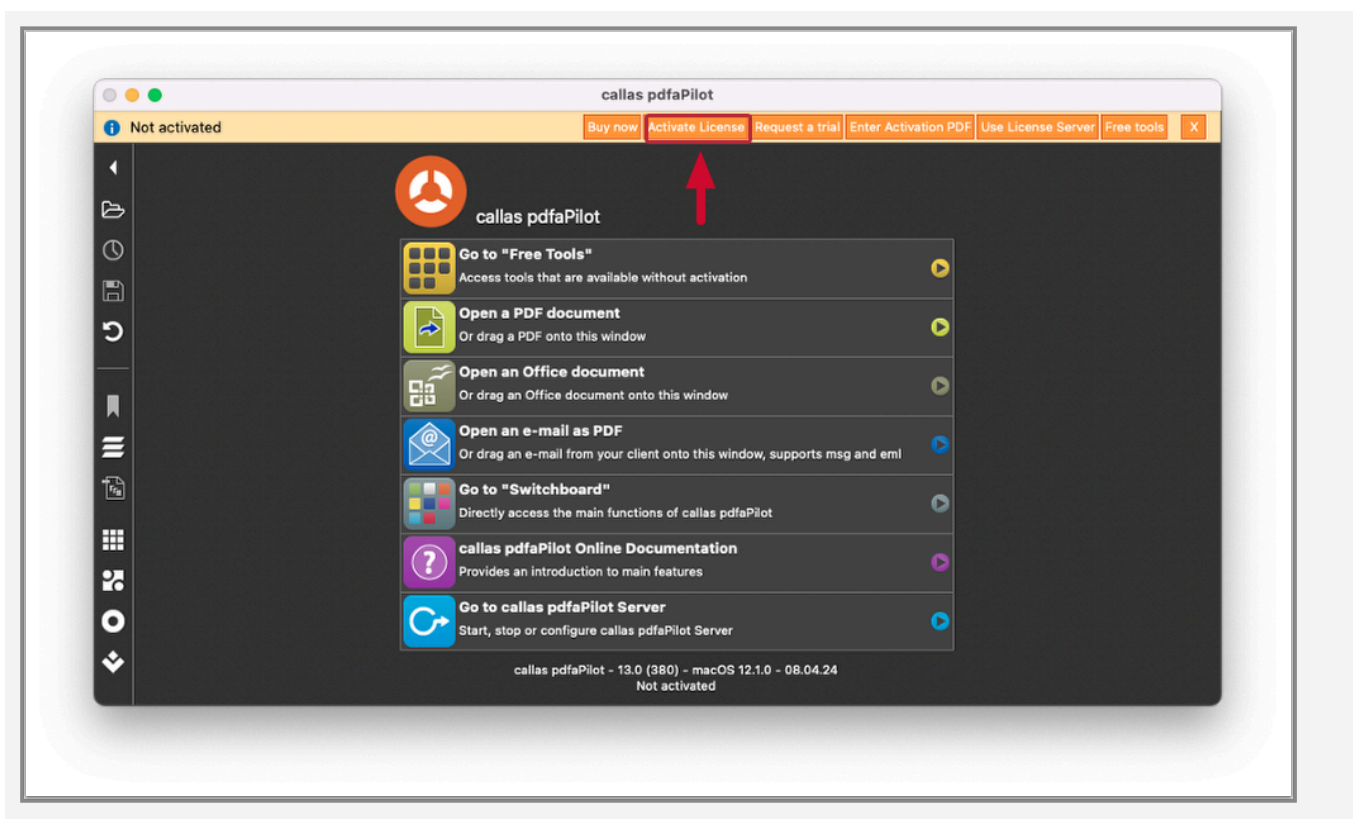
1. Activate the full version

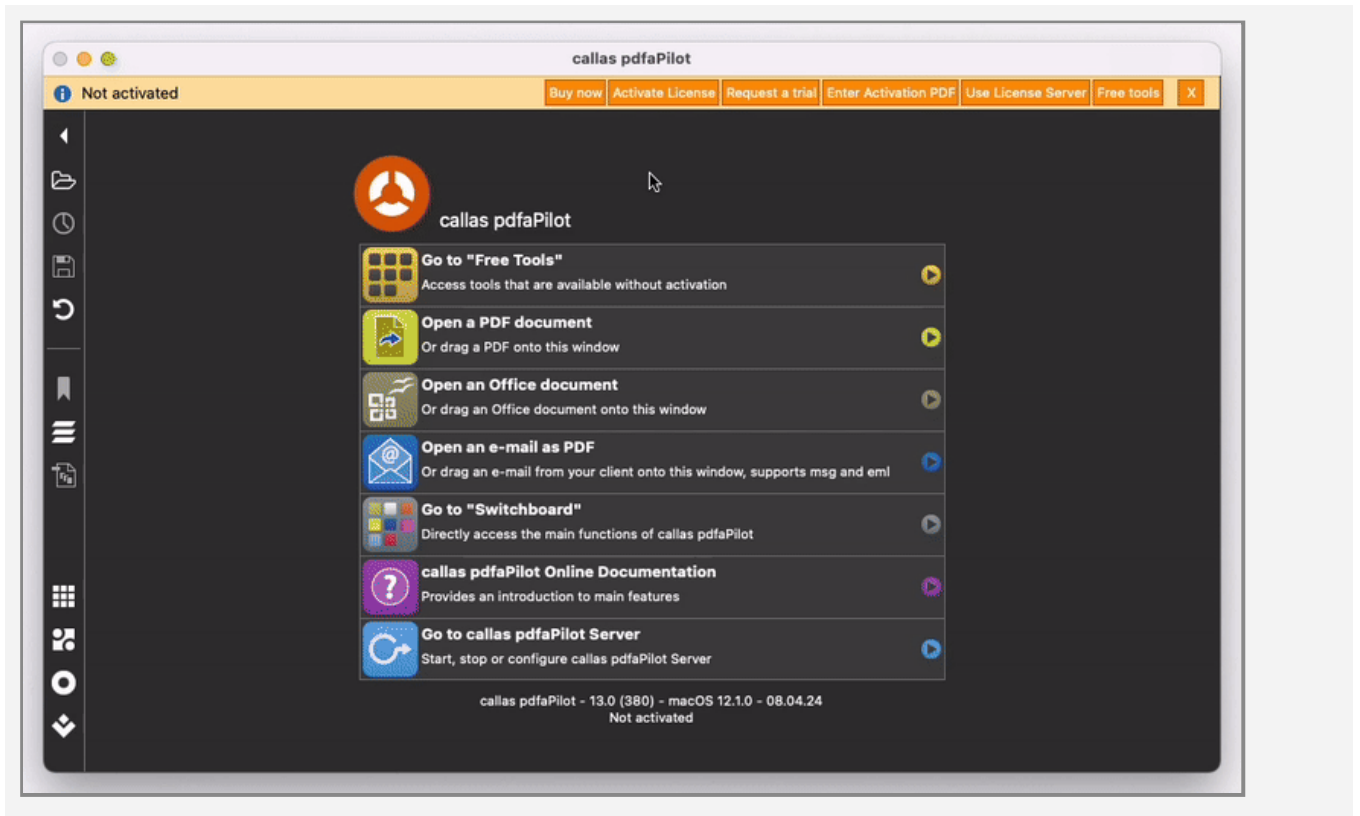
If you have bought a license for callas pdfaPilot Desktop, click on the "Activate License" button. To activate the full ver-

sion you will need a valid license key that can be found on your License.pdf. You'll need to fill out the window in step 2.

After clicking "Activate License" the Activation Window appears. Here you have to fill out your Name, Company, Email address and a valid License key. It is important to enter a valid email because the Activation PDF will be sent to that email address. You can enter the License Key by dragging and dropping your License PDF onto the lighter gray area.

Afterwards click the "Next" button. This will send your activation request to the activation server. Please continue now with [step 3](#).

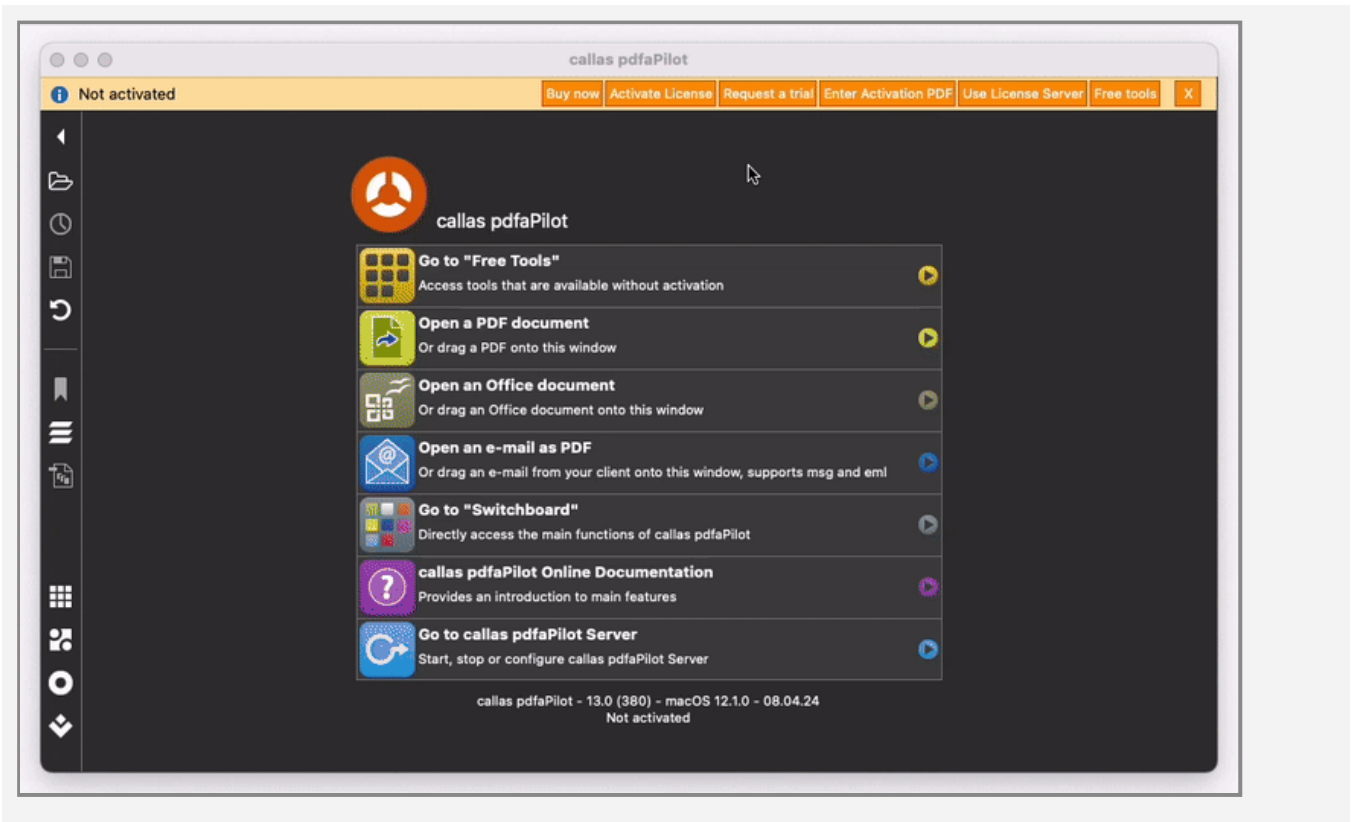
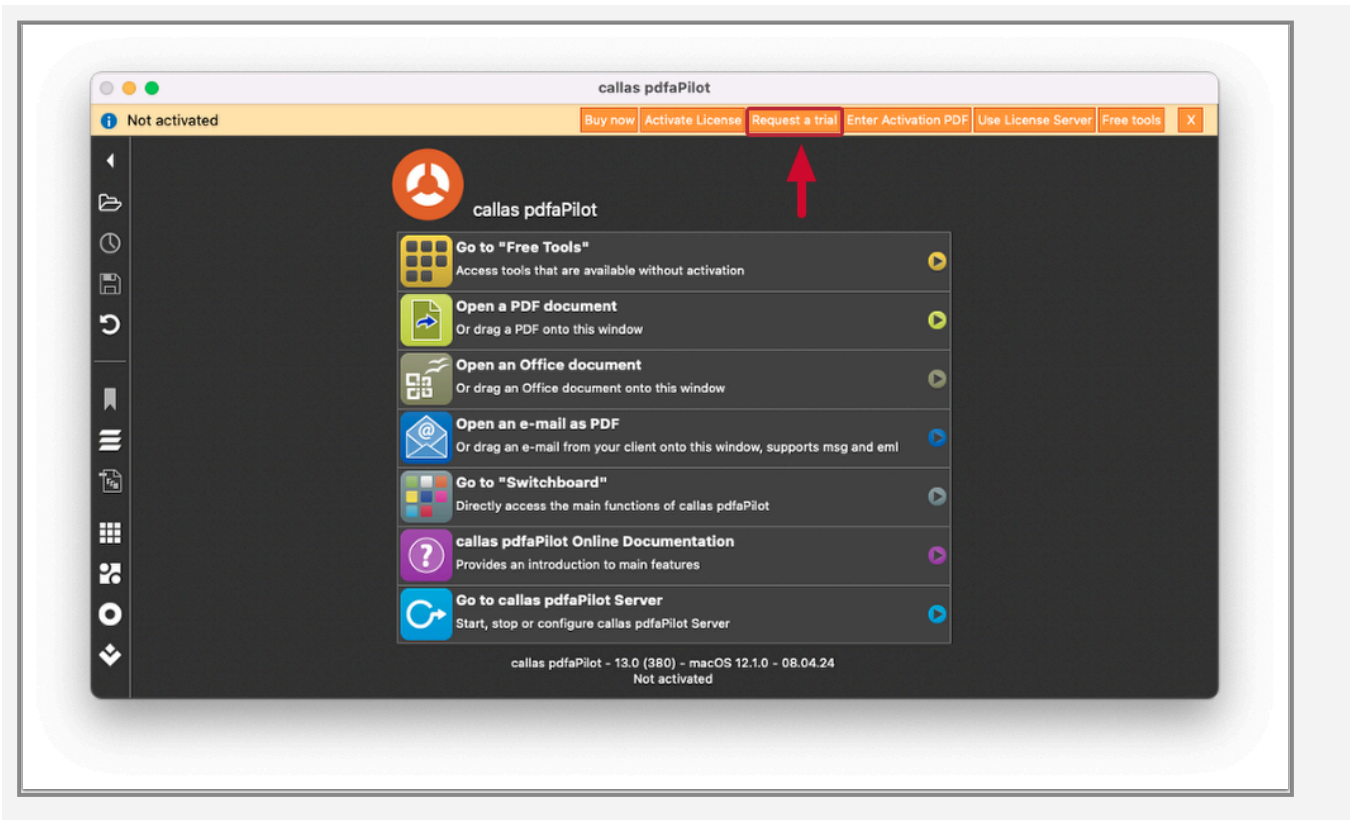




2. Request a trial version

If you simply want to try the product, click on "Request a trial". The Activation Window appears. Here you have to fill out your Name, Company and Email address. You can select which version of the software you want to activate a trial for. It is important to enter a valid email because the Activation PDF for your trial will be sent to that email address.

Afterwards click the "Next" button. This will send your activation request to the activation server. Please continue now with [step 3](#).



3. Activate pdfaPilot

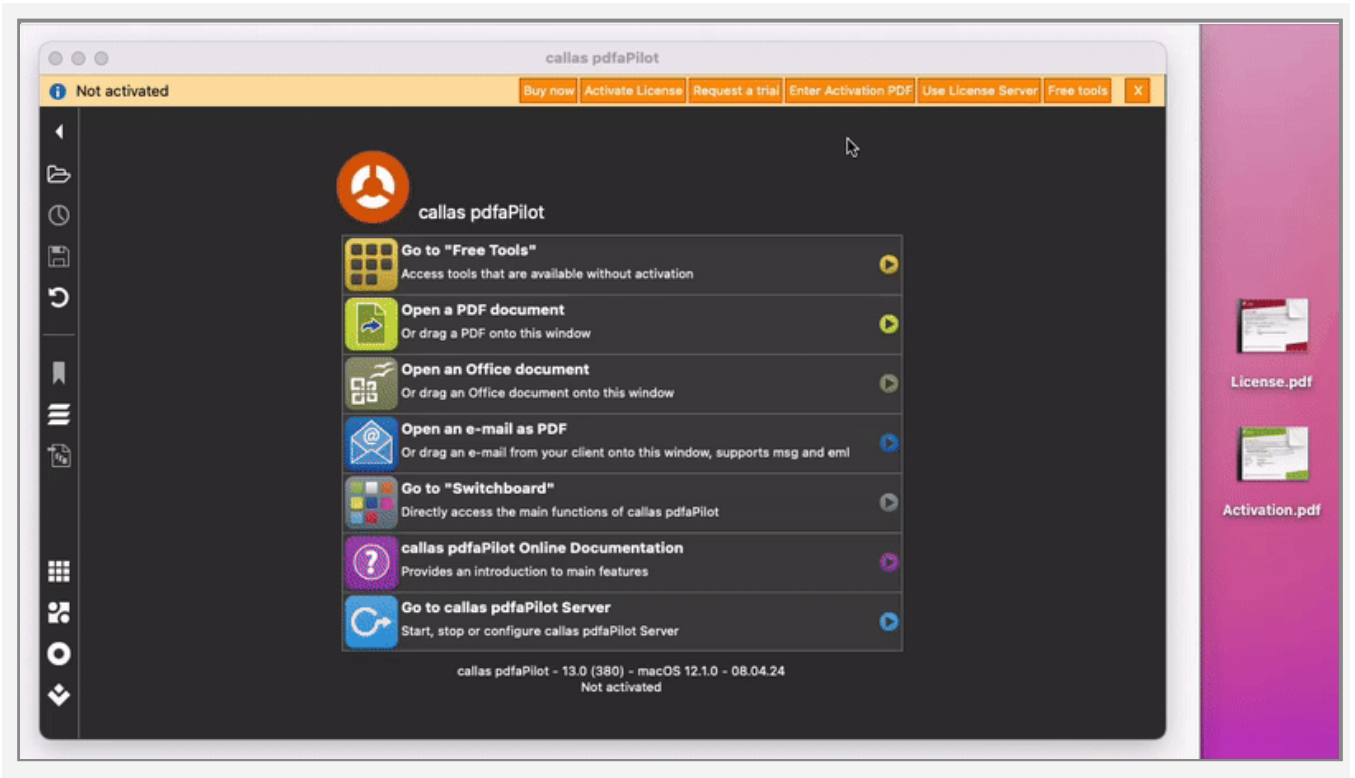
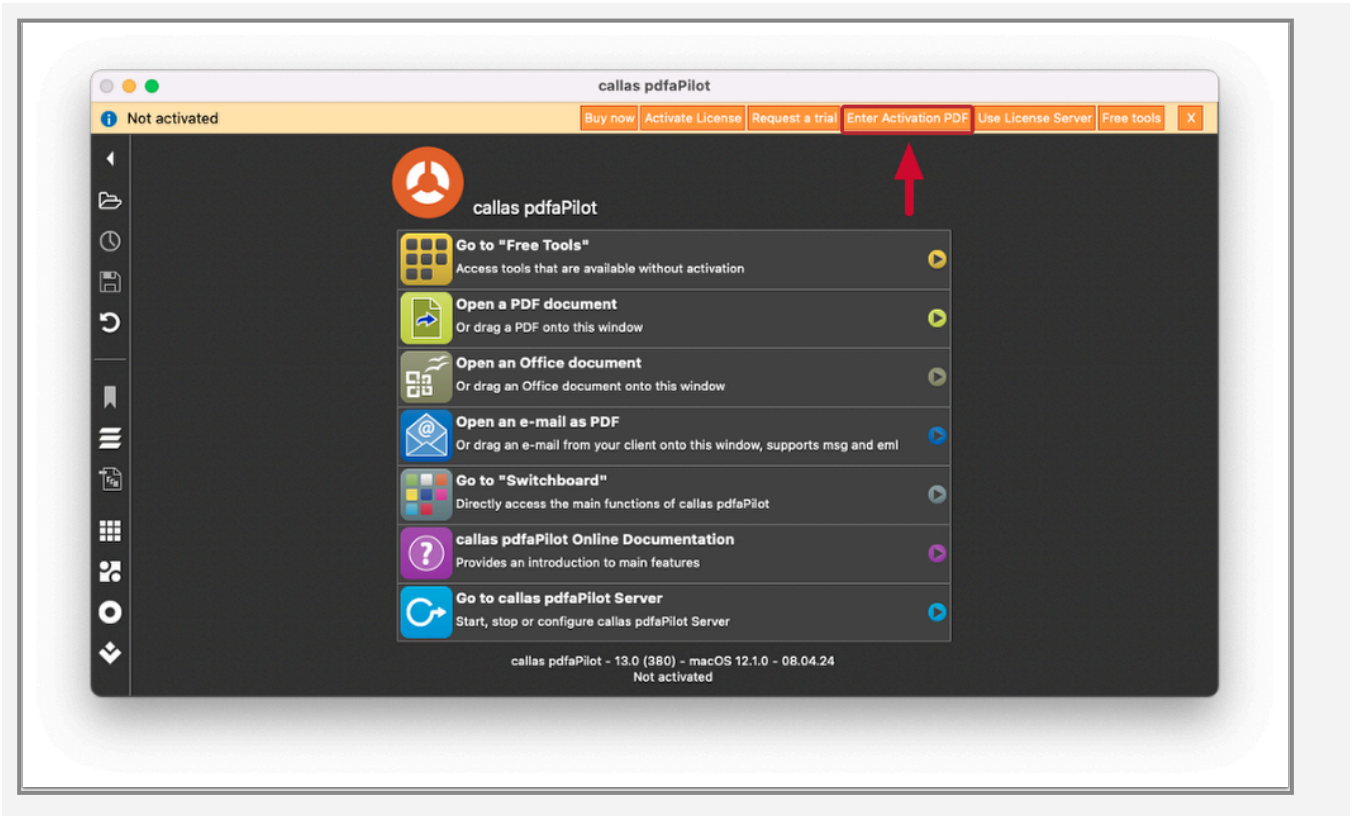
After submitting your details (and clicked on the "Next" button), it will typically take only a few seconds to receive an email from the callas activation server containing an attachment called "Activation.pdf".

If no response is received or in the event of an error, please contact support@callassoftware.com to determine the exact cause.

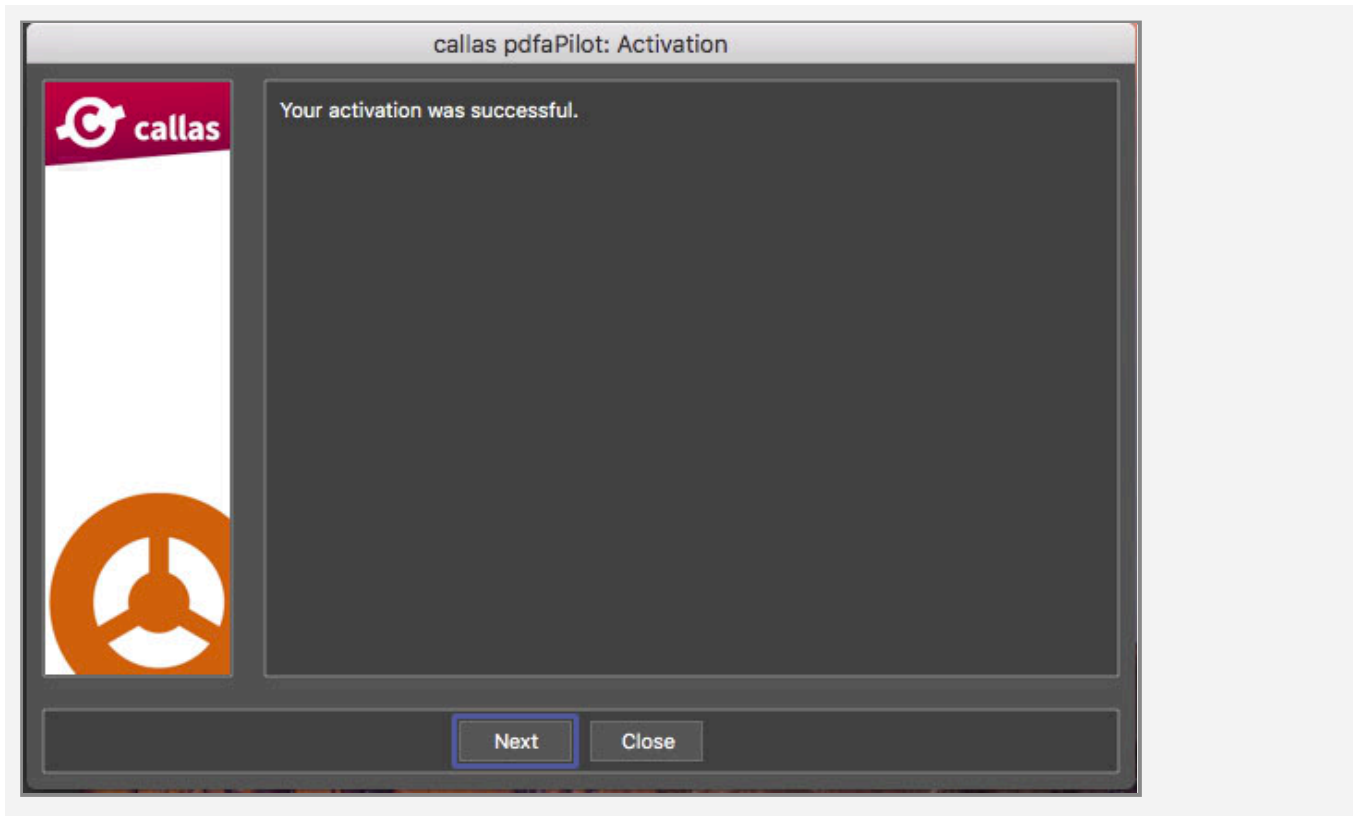


NOTE: The Activation.pdf is only **valid for 48 hours** after being requested. After this timeframe, a new Activation.pdf has to be requested from the activation server.

You can either drag and drop the Activation.pdf file onto the lighter gray area or copy and paste the information from the email into the field below. It is generally safer to drag and drop the file than to copy and paste the information. Dragging and dropping the file will automatically fill in the information in the field.



Click "Next" and your callas pdfaPilot will be activated. You can click "Close" and start using the software.



Activate pdfaPilot without internet connection

You can also install and activate callas software on a system that doesn't have an Internet connection. In order to make this work, you can do an offline activation. Follow to normal activation procedure until the point where you have entered your details:

1. Click on the "Details" button to get the activation information.
2. Click on the "Send via E-Mail" button to reveal the full email text that needs to be sent to the activation server.
3. Copy and paste this text into a text document and take it to a system where you have email access. Create a new email with the subject, content and "To" address as described.
4. Once you receive the response email with the attached "Activation.pdf", take this file back to the system without Internet access and use it to go through the rest of the procedure.

Deactivation

As the activation (and the resulting license file) is bound to the hardware, it is necessary to deactivate a license on one machine before an activation takes place on a new machine.

1. Launch the application.
2. Go to "Help > About callas pdfaPilot".
3. Click "Deactivate".
4. Choose the product "callas pdfaPilot (Desktop)".
5. Fill out your email address.
6. Click "Next".
7. Click "Yes".

After the procedure, your callas pdfaPilot Desktop license will be deactivated.

NOTE: The deactivation procedure is the same for DeviceLink Add-on Desktop (in step 3 you have to select the product "callas DeviceLink Add-on (Desktop)").



Problem with product activation on Windows?

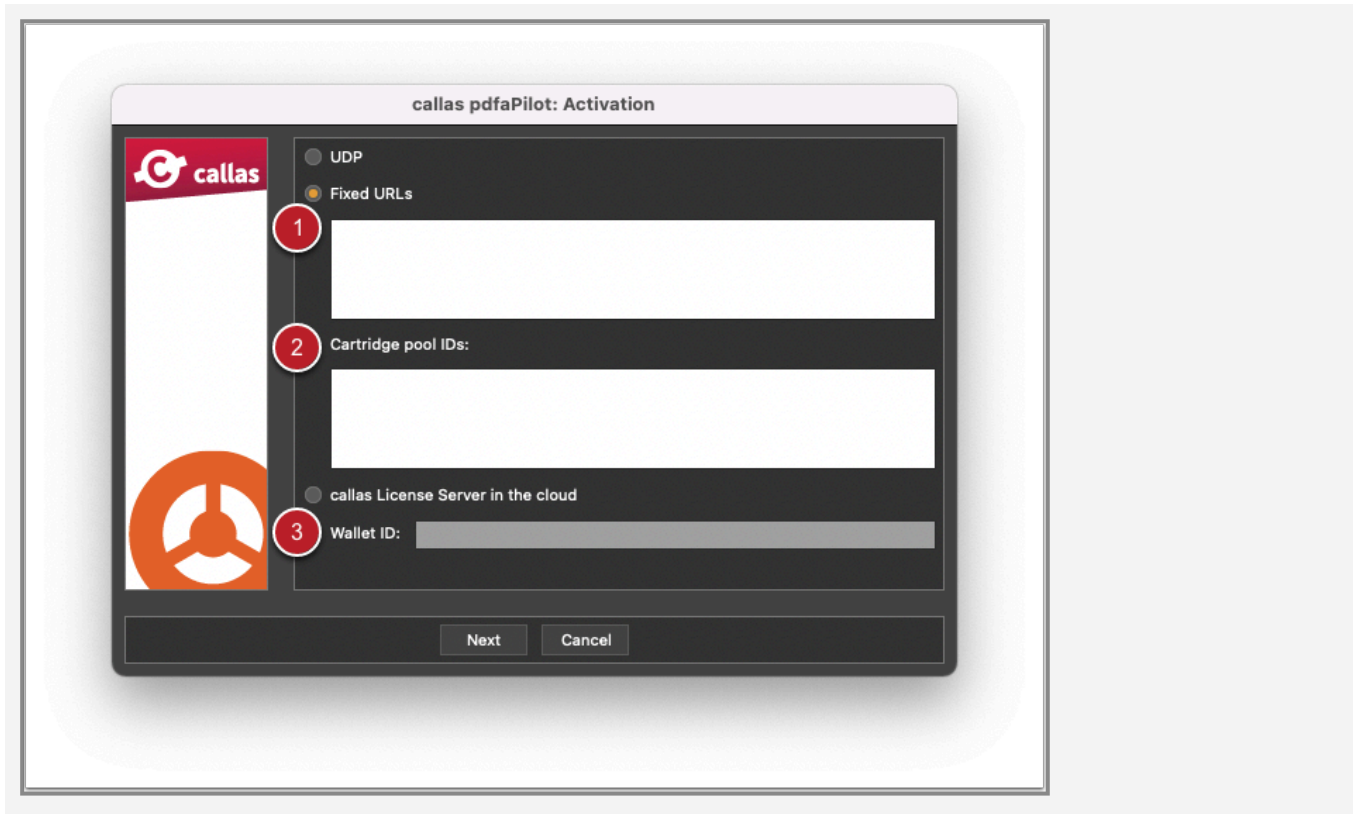
- Open Windows Explorer and type "%appdata%" in the navigation area.
- Copy the "License.txt" file you received from callas into the folder for pdfaPilot.
- Restart pdfaPilot

4. Use License Server

To run pdfaPilot Desktop without activating a hardware-bound licence, you can use the Licence Server. Here you have several options:

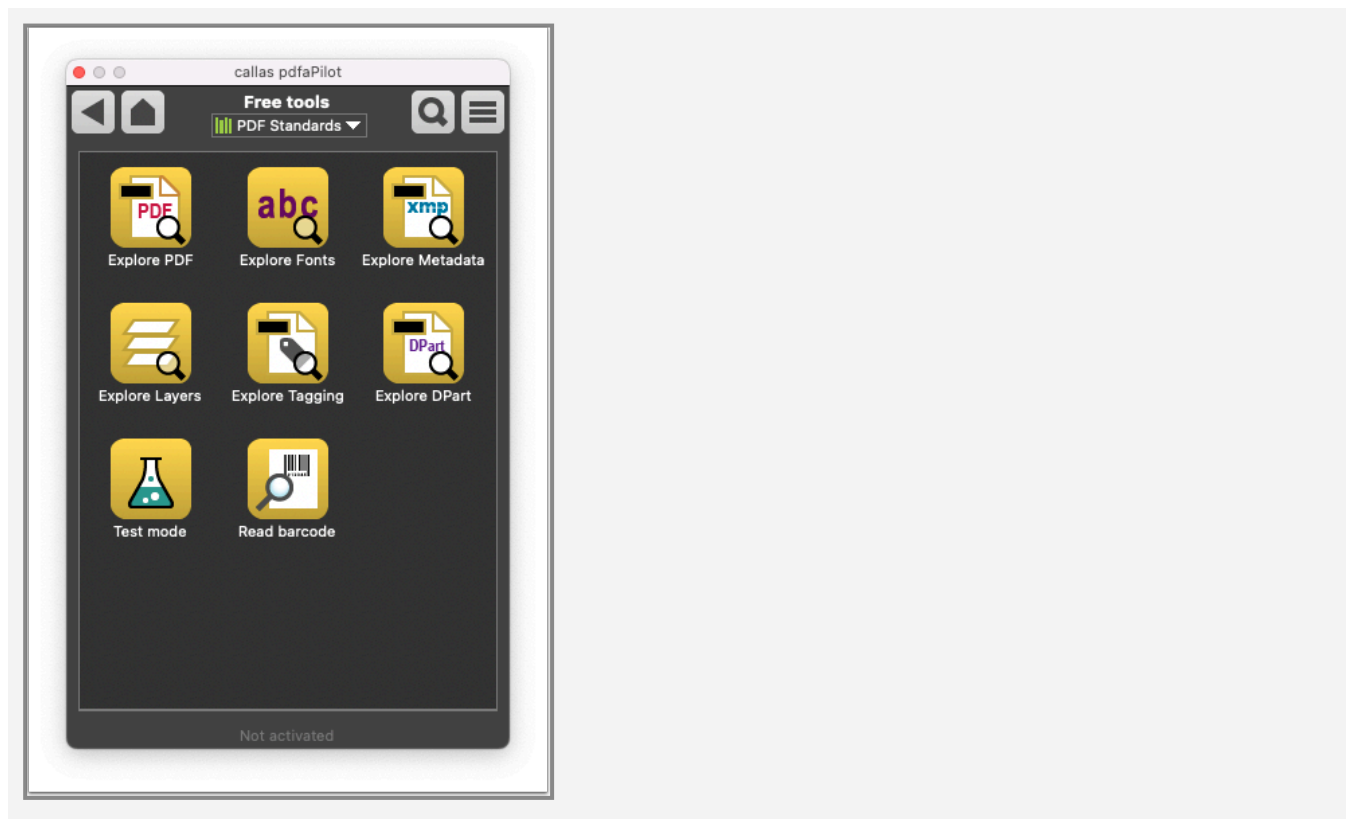
1. To use the License Server on-premise to activate pdfaPilot Desktop, you must enter the IP address of the server here. More information on how to activate the License Server on-premise can be found [here](#).

2. Input field to specifying a cartridge pool. More information can be found [here](#).
3. To use the License Server (SaaS solution) to activate pdfaPilot Desktop, you must enter the Wallet ID here. More information on how to use the License Server can be found [here](#).



5. Free tools

pdfaPilot also includes free tools that can be used without activation. The tools allow detailed insight into PDF files, for example into the internal structure, fonts, metadata, existing DPart or tagging structures or the reading of barcodes and QR codes. pdfaPilot also contains free tools that can be used without activation. The tools allow detailed insight into PDF files, e.g. the internal structure, fonts, metadata, existing DPart or tagging structures or the reading of barcodes and QR codes.



1.4 Multi-user system (Activating additional users on the same system)

pdfaPilot can be operated by multiple users on the same computer.


We'll show you where to copy the license file to in order to allow multiple users to access pdfaPilot.

Copying the license file

pdfaPilot licenses are only bound to a given system, not to specific users.

The license file can be copied under **Activate** in the User Preferences for the currently active user.

To allow other users to work with this license, you must manually copy the `License.txt` file into the User Preferences for the other user.

 The folder path from which the `License.txt` file can be retrieved and to which it should be copied is as follows:

MacOS:

`/Users/<USERNAME>/Library/Preferences/callas software/callas pdfaPilot <VERSION>`

Windows:

`C:\Users\<USERNAME>\AppData\Roaming\callas software\callas pdfaPilot <VERSION>`

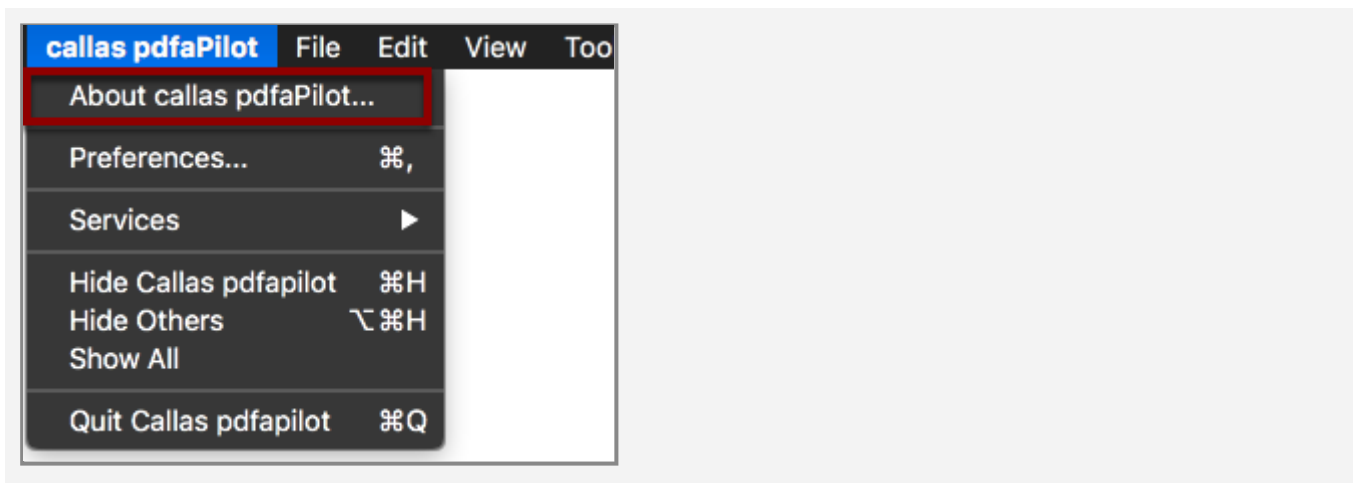
CLI/Server activation for all users on one system

If the pdfaPilot CLI/Server version shall be activated for all users on a system, there are various possibilities. Details can be found in the CLI manual under "[Activating pdfaPilot CLI](#)".

1.5 Migrating the pdfaPilot to another computer (Deactivation)

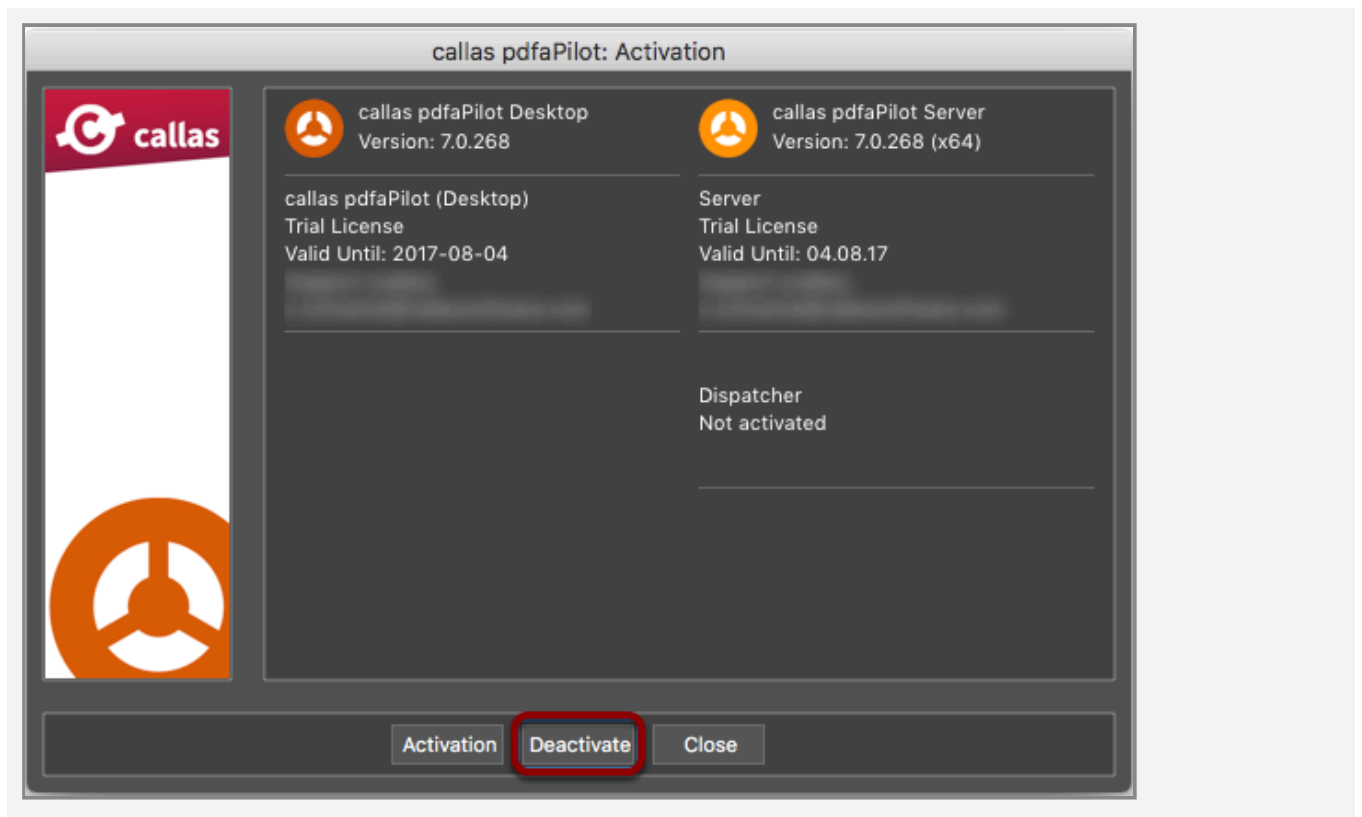
If you wish to use pdfaPilot on another computer - for example, because you have acquired a new device - you must first deactivate the application on the last computer used.

Open the “About pdfaPilot” menu item



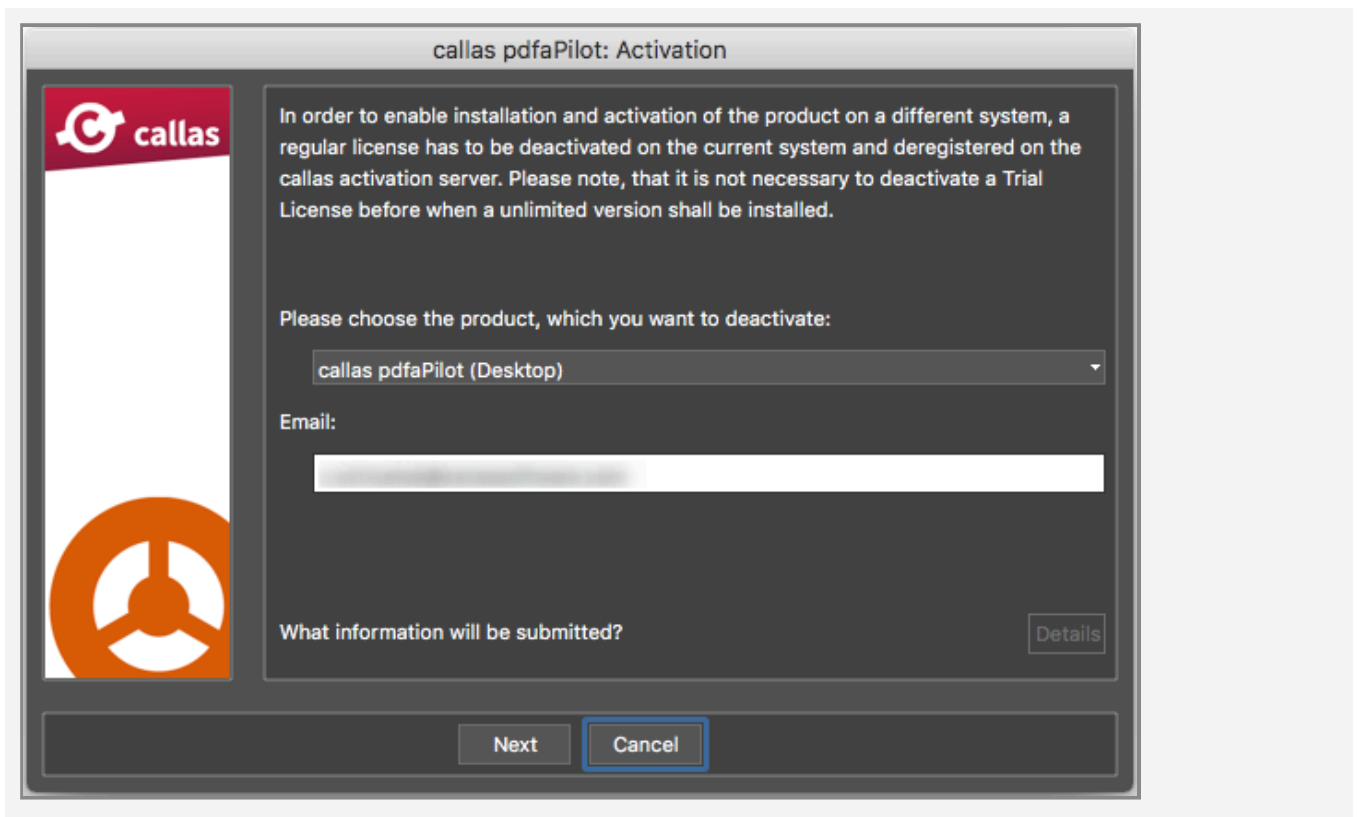
Options to **Activate** and **Deactivate** pdfaPilot can be found under the following menu item: **About callas pdfaPilot...**

The “Activate” window



In the Activate window, you can deactivate the pdfaPilot by clicking **Deactivate**.

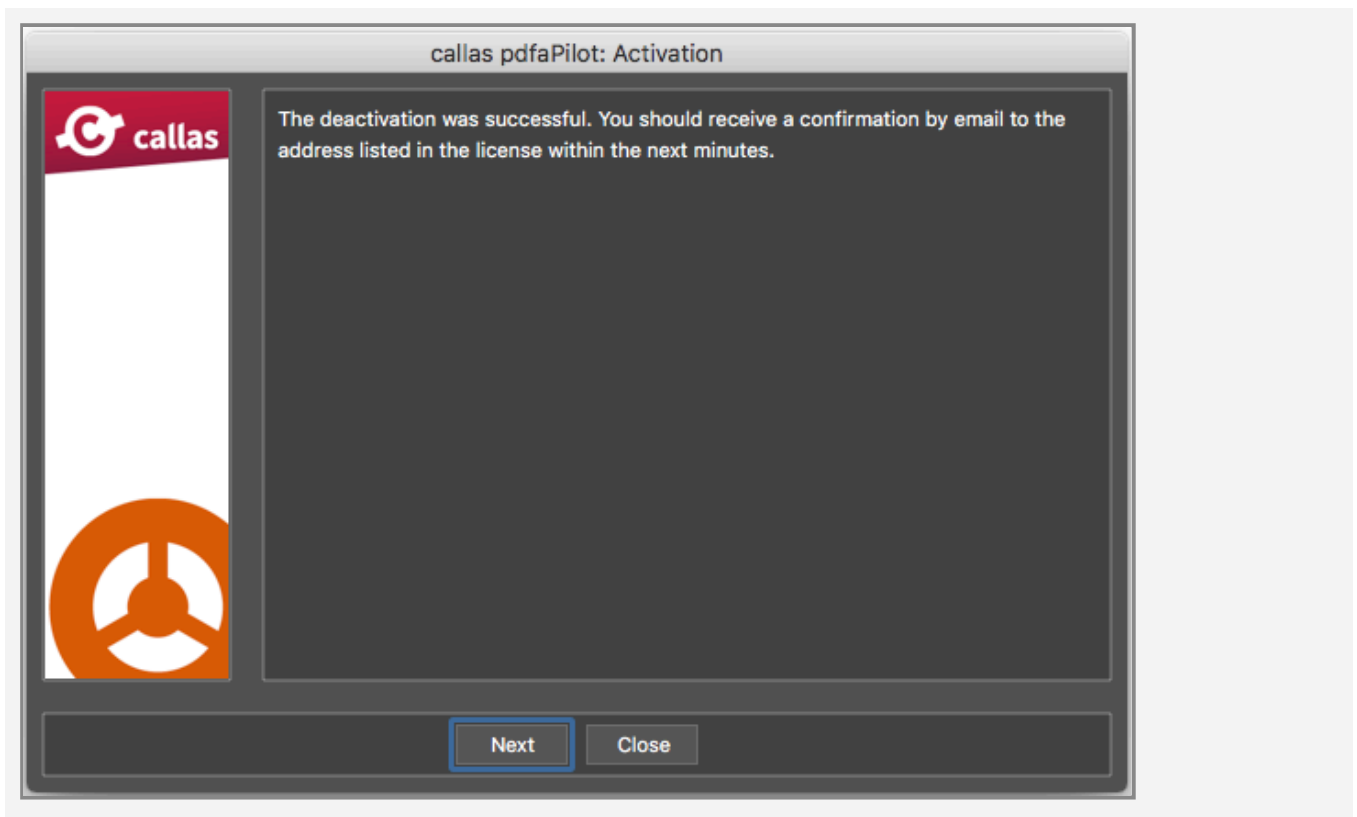
Deactivating license: Product and email



In the dialog which opens, you can select the **software to deactivate** (if you use multiple callas software products) and, if necessary, change the **email address** to which the confirmation message will be sent.

Click **Continue** to start the process.

Notification and deactivation email



pdfaPilot will report **Successful deactivation** and inform you that an **email confirmation** has been sent.

After you have received this deactivation notification by email, you can activate pdfaPilot on another machine using the original "License.pdf" which has to be valid for the installed version.

Please note: a previously received "Activation.pdf" can only be used on the machine where it has been requested from and only within 48 hours after receiving it.

To learn how to activate pdfaPilot, visit the support article [Activation procedure: callas pdfaPilot Desktop](#).

Move your pdfaPilot settings to another machine

If you have created a number of own Profiles etc., the most easiest way to move your settings is to export the respective Library as described here and copy the created .kfpl file to the new machine.

However, there might be cases, where a direct access to the stored user data is needed.

Your pdfaPilot settings are stored in the following location:

Windows:

C:\Users\USERNAME\AppData\Roaming\callas software\
callas pdfaPilot ...

Mac OS X:

/Users/USERNAME/Library/Preferences/callas software/
callaspdfaPilot...

Replace "USERNAME" in the path above with your username, and copy the contents of this folder to the same location on other computer.

1.6 Uninstalling the pdfaPilot Acrobat plugin

The pdfaPilot plugin can be easily deleted in just a few steps. We'll show you where the relevant packages can be found depending on the operating system and Acrobat version.

Removing the pdfaPilot Plug-in on MacOS

The pdfaPilot plug-in for Acrobat can be easily removed by deleting the pdfaPilot___.acroplugin package in the callas software subfolder in the plug-ins directory for Acrobat, under Mac.

Note: The file name contains the respective pdfaPilot version number at the "__" position.

Acrobat DC

Please note that as of Acrobat DC, Adobe has changed the folder in which it stores third-party plug-ins:

/Library/Application Support/Adobe/Acrobat/DC/Plug-ins/callas software

Acrobat XI and older

For Acrobat XI and older, meanwhile, the plug-in can be found in the following folder:

/Applications/Adobe Acrobat XI Pro/Adobe Acrobat Pro.app/Contents/Plug-ins/callas software

Removing the pdfaPilot Plug-in on Windows

On Windows, the plug-in can be found in the subfolder of the plug-in directory for the current Acrobat version:

Acrobat\plug_ins\callas software

The folders and files contained within can be identified by their name, e.g. "pdfaPilot 8" or "pdfaPilot 10".

Delete them to remove the plug-in.

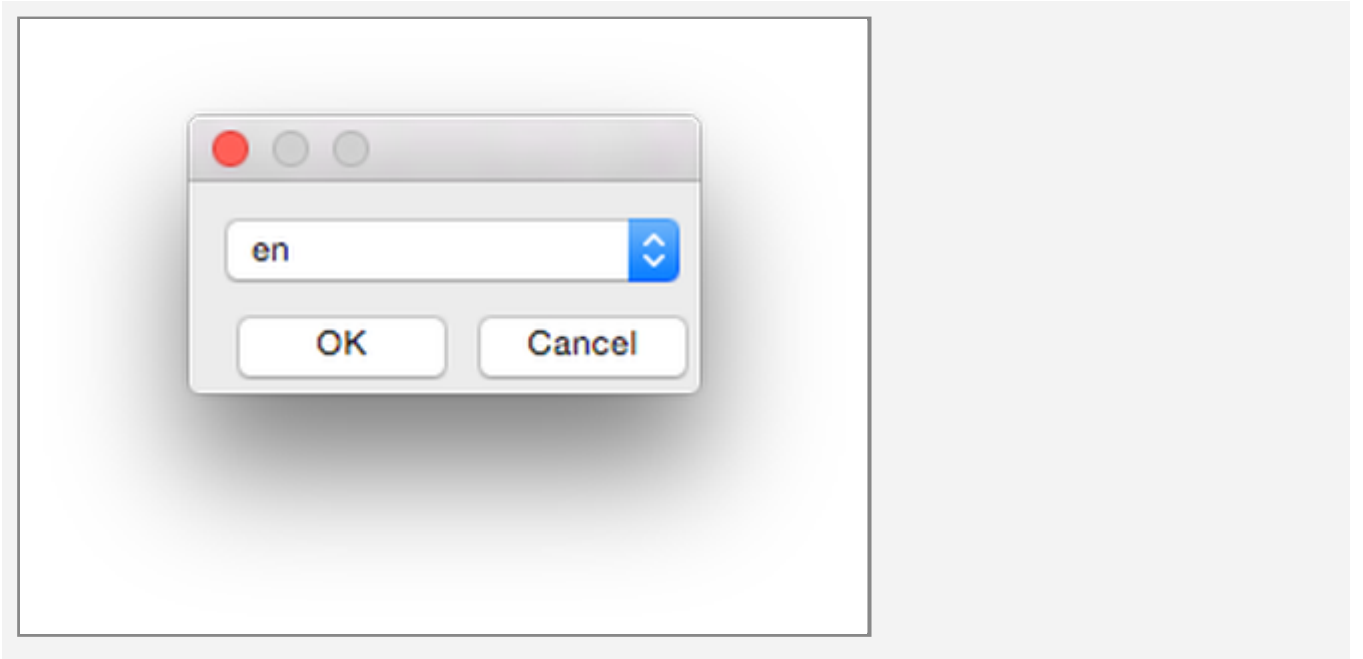
1.7 Change language of user interface (desktop only)

Normally, pdfaPilot Desktop will launch in the same language in which the the operating system is running (or English if the language in which the operating system running is not supported by pdfaPilot). It is possible to switch the language of the user interface to any other language supported by pdfaPilot Desktop. In order to do so, **press the Cmd/Ctrl key while launching pdfaPilot.**

A small dialog will appear, with a popup menu indicating the codes of the available languages, for example "de" for German or "fr" for French.

After selecting the desired language, simply click OK, and pdfaPilot Desktop will launch in the selected language.





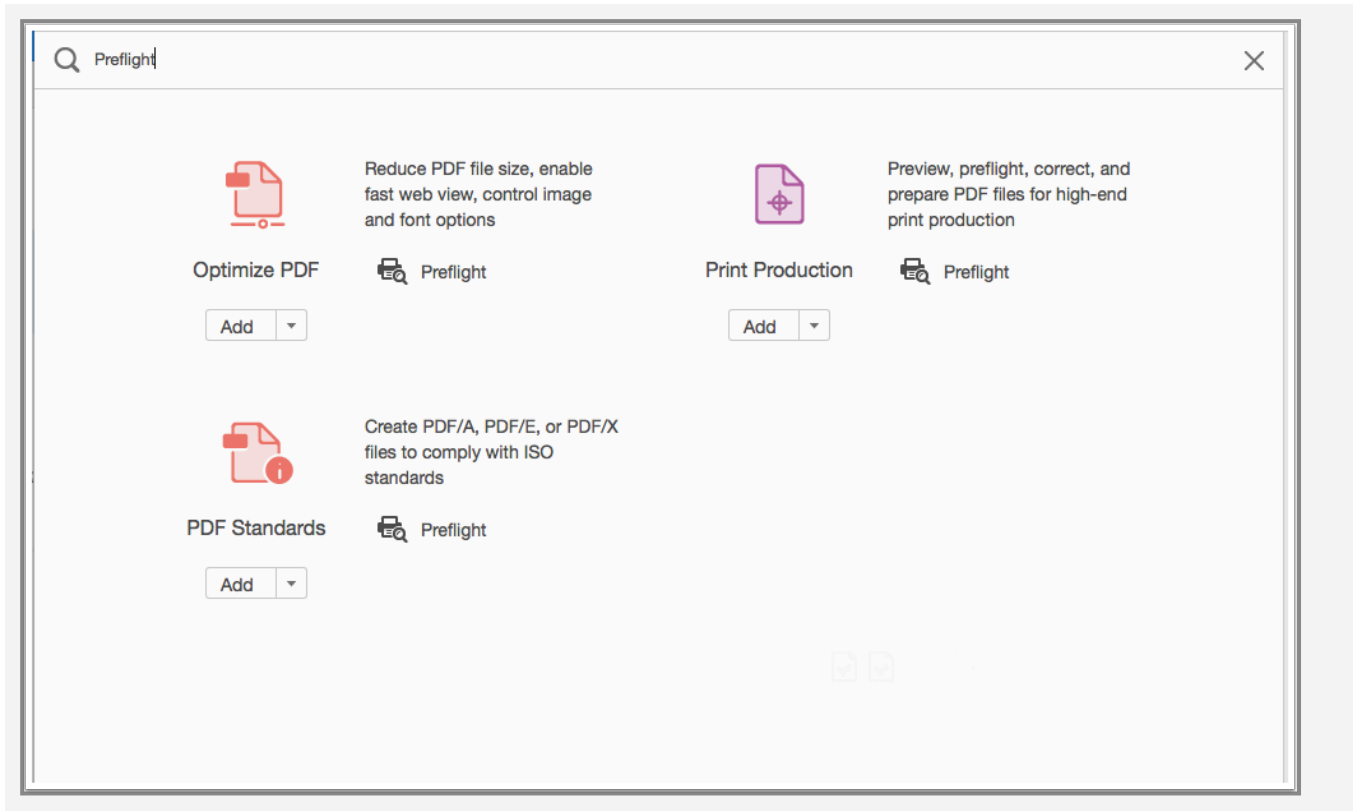
1.8 Using pdfaPilot and Preflight in Acrobat Pro parallel

A lot of users want to use the Acrobat Preflight functionality as well as the high number of additional tools from pdfaPilot simultaneously.

This tutorial explains, how this parallel usage of Acrobat Preflight (a development of callas software, which has been integrated by Adobe as a part of Adobe Acrobat Pro since 2003) and callas pdfaPilot works.

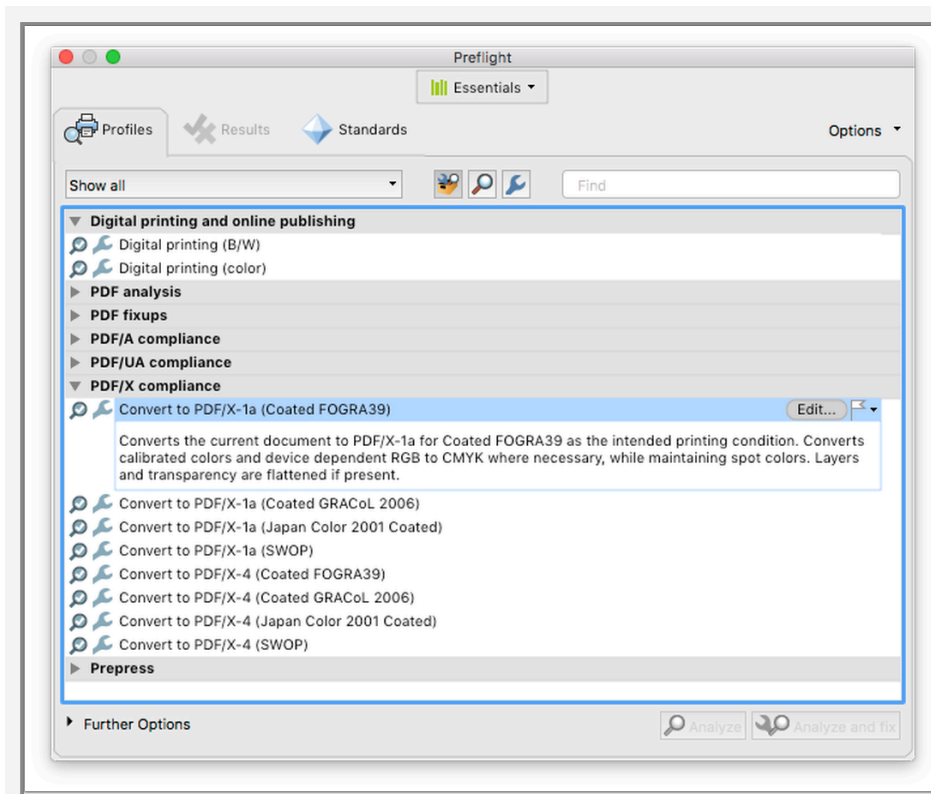
Open Acrobat-build-in Preflight

The Preflight functionality in Acrobat Pro can be found using the "Tools" bar for example or with the shortcut "CMD, Shift + X" (in MacOS) or "CTRL, Shift + X" (using Windows).



The Preflight Profiles window

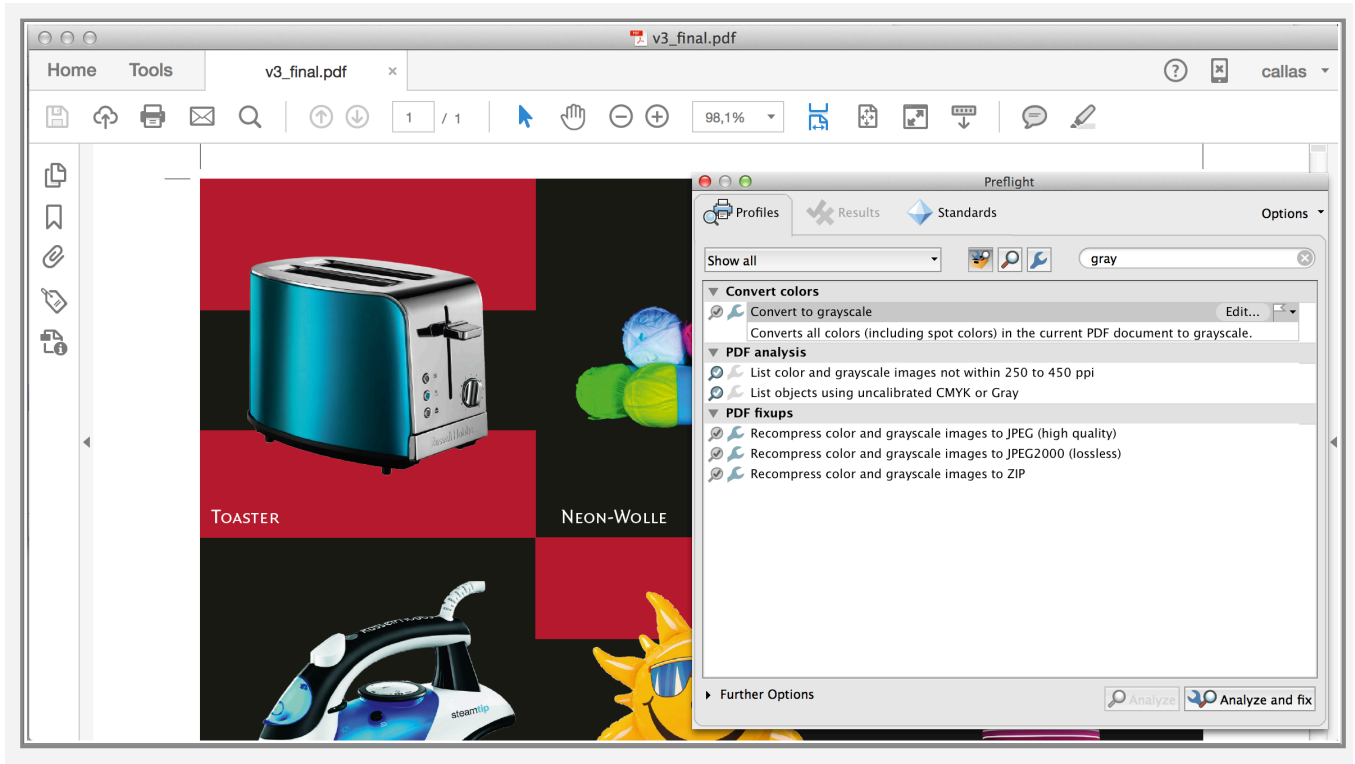
The Profile window lists a number of Profiles, Checks and Fixups. Also a separate "Standards" area gives easy access for conversion of PDF files into PDF-based ISO standards.



Process a PDF

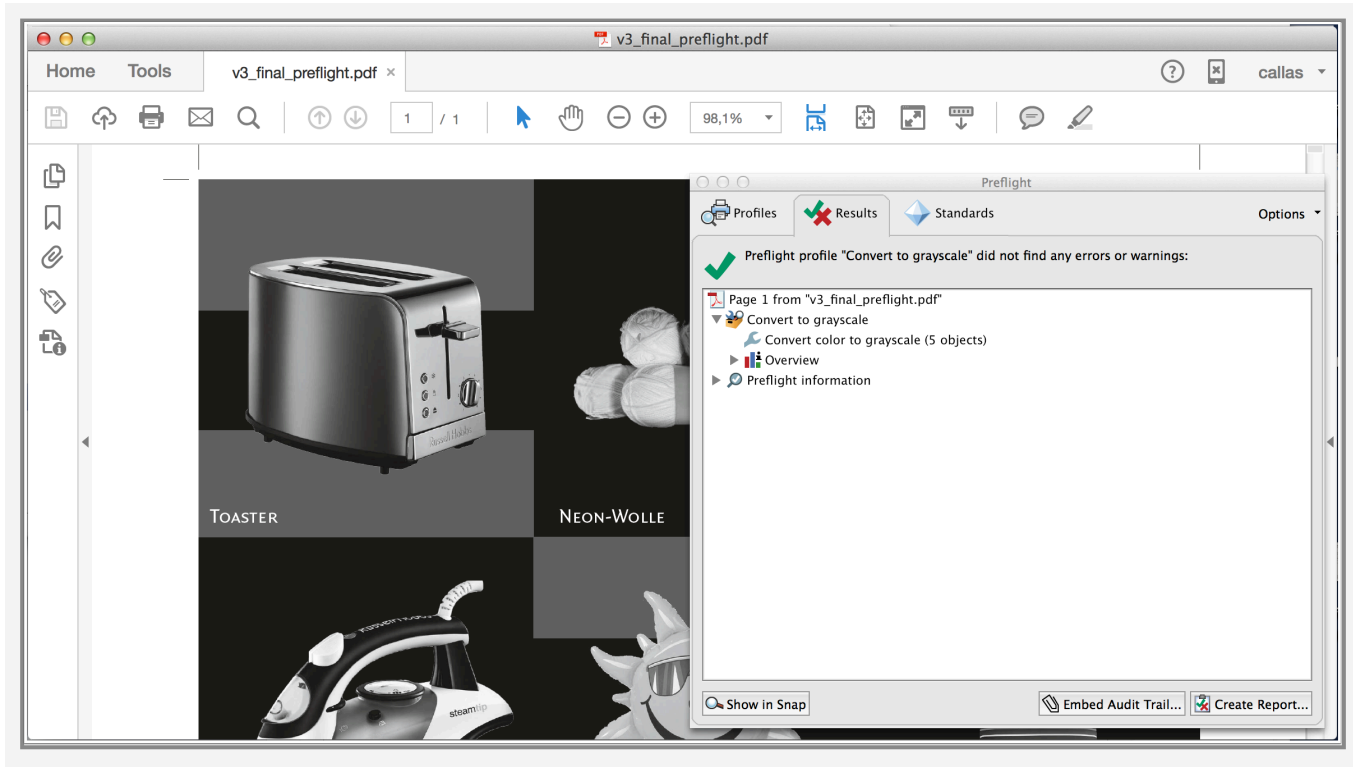
When a PDF is open, a conversion or analysis can be executed on that file.

Using Profiles, which only contain checks and/or validation of standards (and with Checks within the respective group), only the "Analyze" button is active. When a Profile contains Fixups and conversions into a PDF standard (and for single Fixups of course), also kann auch "Analyze and fix" can be used.



Result dialog

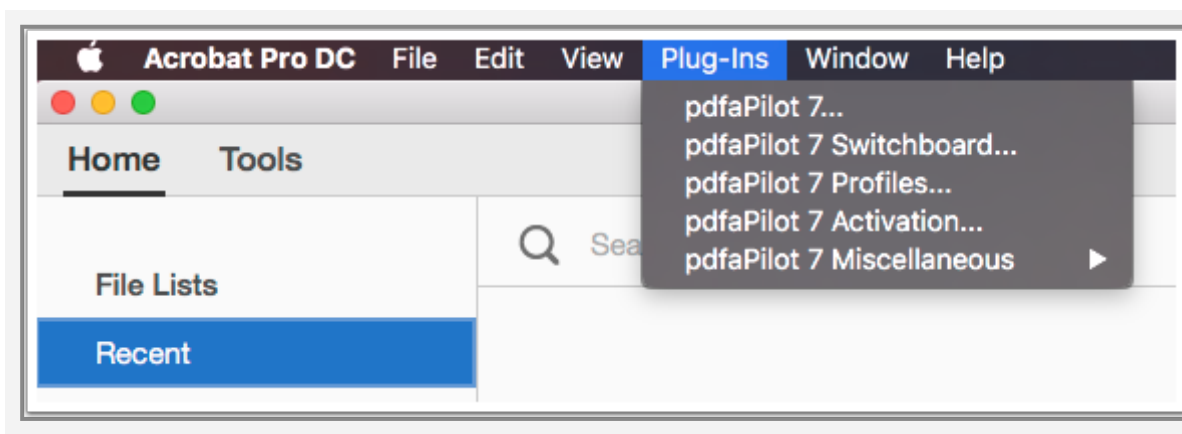
After processing, the result is shown and can be reviewed. A click on the "Profiles" tab within this window allows further processing or analysis.



pdfaPilot as an Acrobat Pro Plug-In

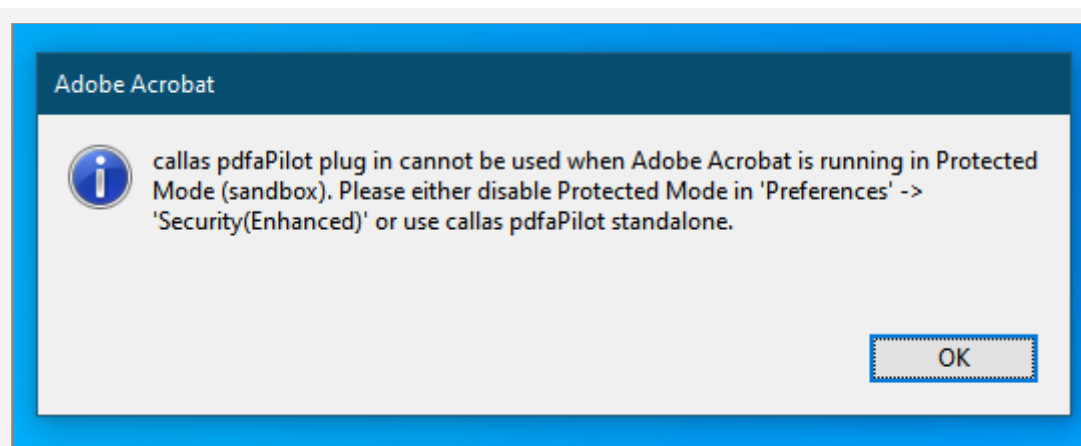
Using the pdfaPilot installer, a Desktop version and a Plug-In for Acrobat Pro will become installed (the latter of of course only if Acrobat Pro is installed on the respective system).

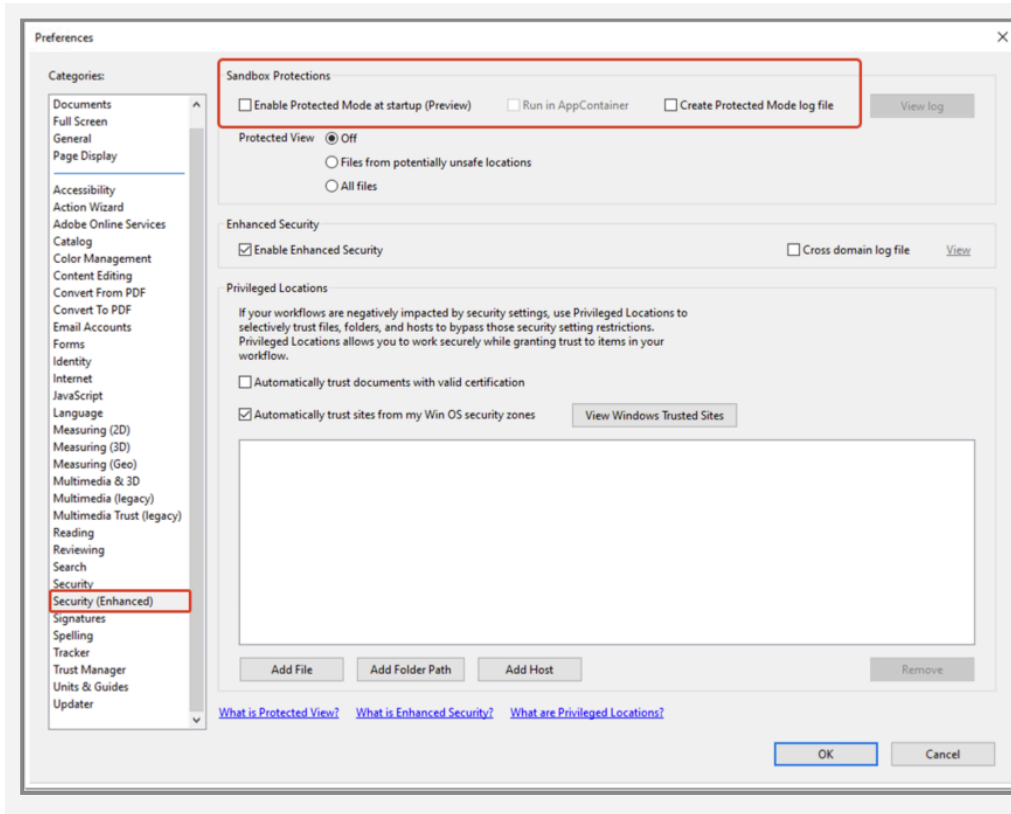
pdfaPilot can be found using the "Plug-Ins" menu or within the "Tools" sidebar.



Using the pdfaPilot Plug-In on Windows

If the pdfaPilot Plug-In is not displayed in Adobe Acrobat Pro after installation, the security settings should be checked. **The Plug-In for Acrobat cannot be used in protected mode (sandbox mode).** This mode must first be deactivated under "Preferences" > "Security (Enhanced)" (see 2nd screenshot). Then Acrobat must be restarted so that the Plug-In can be used.



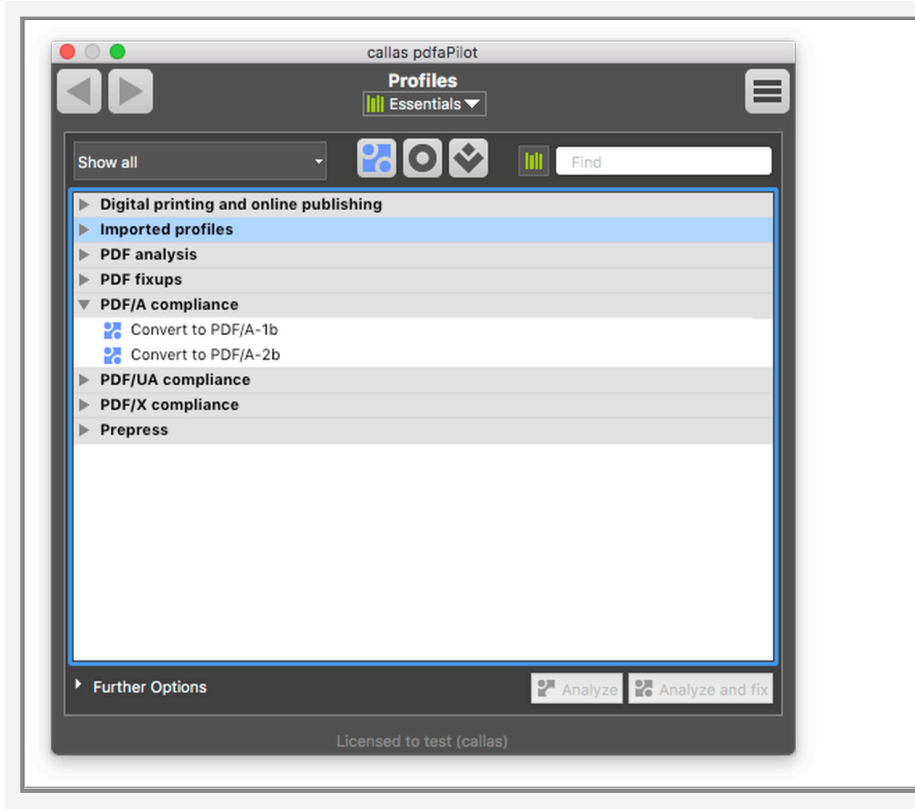


Open the pdfaPilot Profiles window

A click on the "pdfaPilot Profiles" entry opens a window with a number of predefined Profiles, Checks and Fixups (as well as ProcessPlans).

You will notice that the Preflight Profile window looks quite similar to the one of pdfaPilot.

The reason for this is quite simple: Adobe uses technology from callas software for the Preflight functionality in Acrobat Pro.



Why using pdfaPilot AND Acrobat Preflight?

There are several functions available in pdfaPilot (Plug-In and Standalone) only, which are not part of Acrobat Preflight:

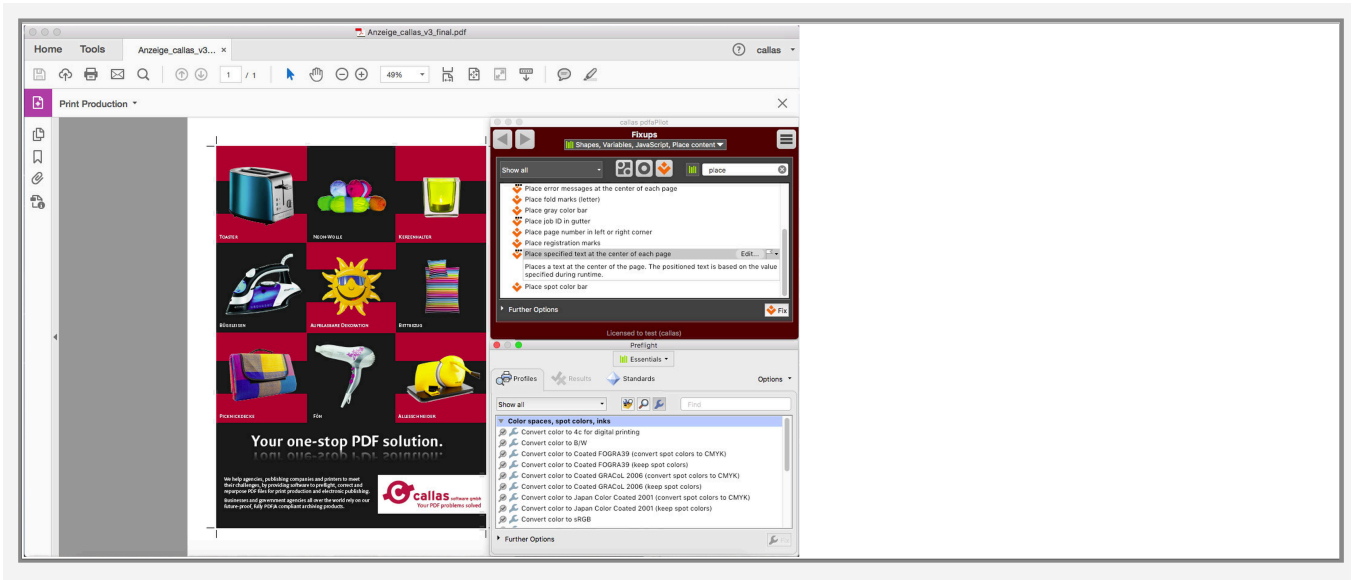
- ProcessPlans
- Custom reports (based on HTML-Template)
- **Several Fixups like:**
 - Place text
 - Place barcode
 - Place page number
 - Place content (based on HTML-Template)
 - Convert colors to n-channel

In general, pdfaPilot has a more frequent update interval than Acrobat Preflight.

Therefore new features, adjustments and fixes are earlier available in callas software products.

Running Preflight and pdfaPilot in parallel

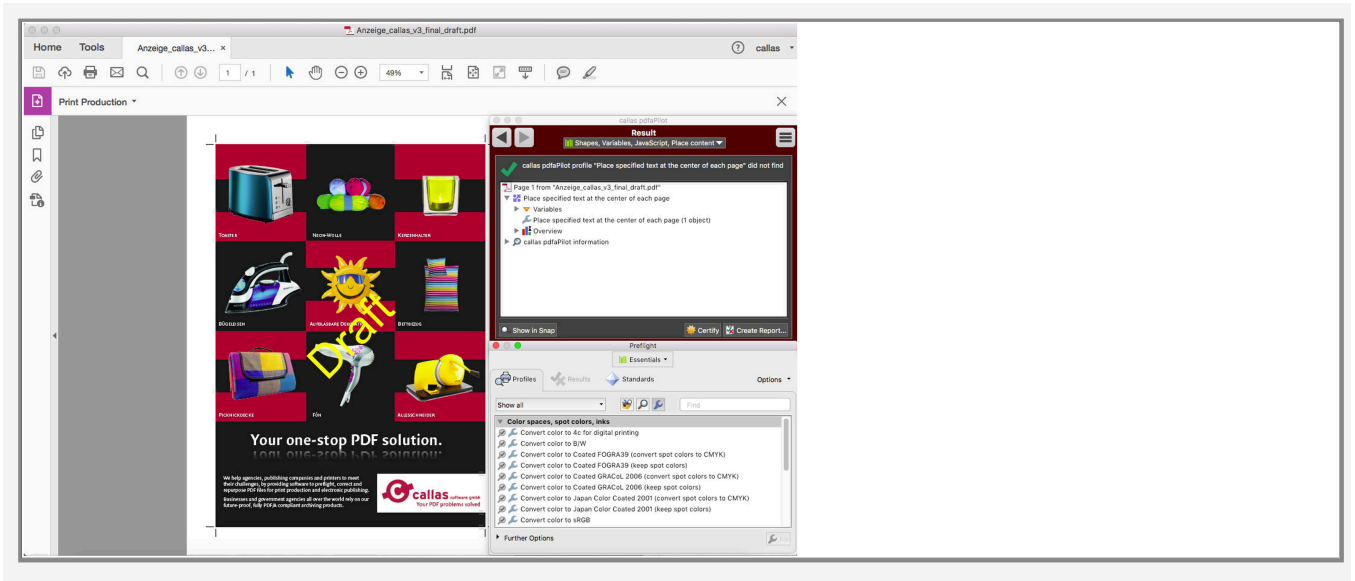
You can use Acrobat Preflight and callas pdfaPilot Plug-In in parallel.



Using pdfaPilot-only features

Using the pdfaPilot Plug-In allows to use pdfaPilot-only features like placing custom content using a Fixup.

For example placing a text, a barcode, numbering pages or a custom content (using a HTML template) as additional content into the PDF.

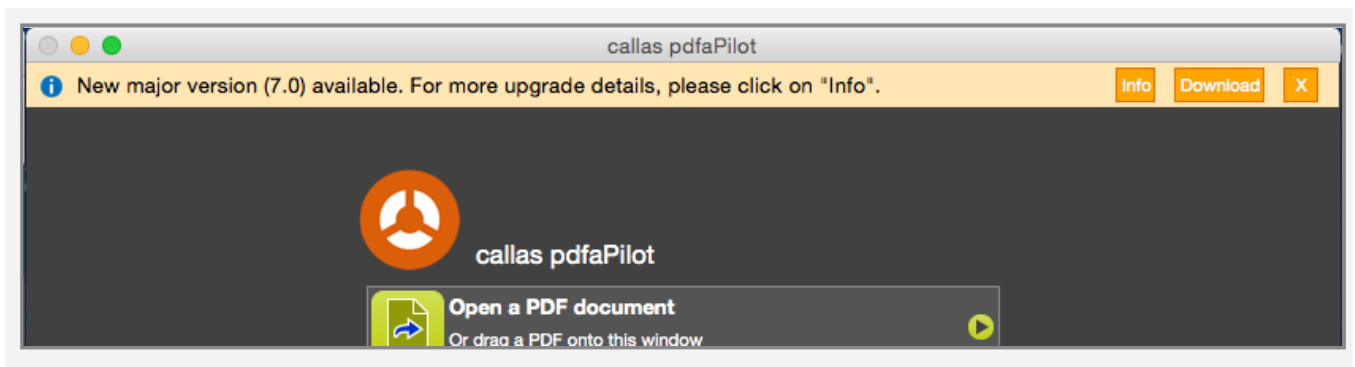


1.9 Notifications about available updates

In pdfaPilot Standalone we inform you about available software updates.

How can you customize these notifications? We will show you in this tutorial.

Update notification for major versions



As soon as a new major version is available for download, a notification bar in the main window of the Standalone version is shown. This functionality is not available in the Acrobat Pro PlugIn.

You can either download this update directly using the "Download" button or hide this notification using the "X". When you hide the notification you'll be asked if you want to be notified with the next start of the software or if you do not want any information about this update anymore. These settings can also be set in the preferences (more information further down in this tutorial).

Additionally, you can switch to the callas software homepage for more information about the update.

Additional information about the installation of new major versions

New major versions contain a bigger number of new functions as well as significant improvements regarding processing.

Major versions are installed in parallel to existing versions - so you can continue to use them.

A new license key is necessary for new major versions as well as a new activation.

Owners of a valid software maintenance agreement (SMA) will receive a new keycode from your reseller or integration partner.

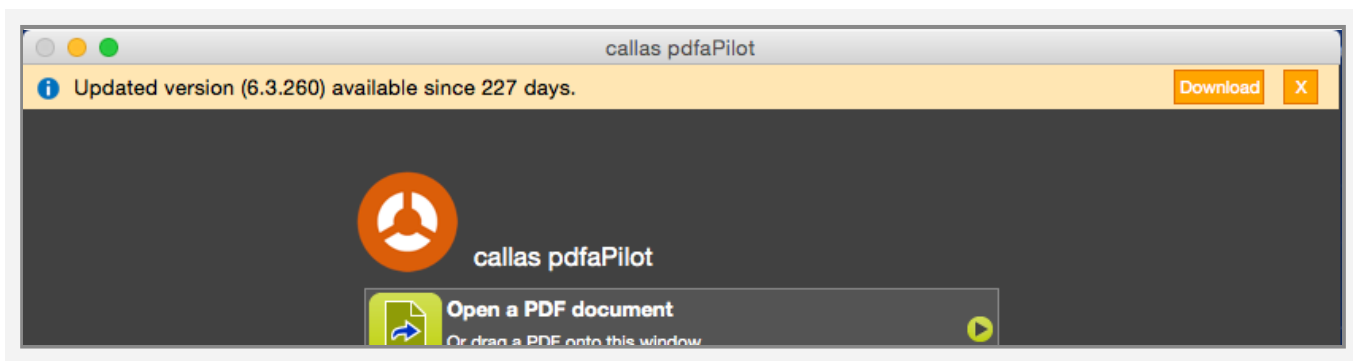
If no SMA exists, you can buy an update on the callas software website or at a local reseller. More information is shown using the "Info" button.

After the first start of the software after the update you'll be asked if you want to import your settings (e.g. own Profiles, ProcessPlans, Checks and Fixups) into the new version.

Please note: Profiles or Libraries exported from new major versions cannot be imported in older software versions. Profiles or Libraries from older versions can be imported into newer versions of course.

If you want to have the possibility to switch back to a previous version, we recommend a backup of your Profiles. More information about this feature in "Export Libraries" further down.

Update notification for minor versions



As soon as a new minor version is available for download, a notification bar in the main window of the Standalone version is shown. This functionality is not available in the Acrobat Pro PlugIn.

Minor versions are software updates, which not only contain bugfixes and performance improvements but also new functions.

User of the respective major version of the Desktop software can update free of charge. User of the Server/CLI version needs to have a valid SMA.

You can either download the version using the "Download" button or hide the notification with the "X" button.

In case you hide the notification, you can choose if you want a reminder at the next software start or if you want to skip this version update.

These settings can also be defined in the preferences of the software (more about this feature later in this tutorial).

Additional information about the installation of new minor versions

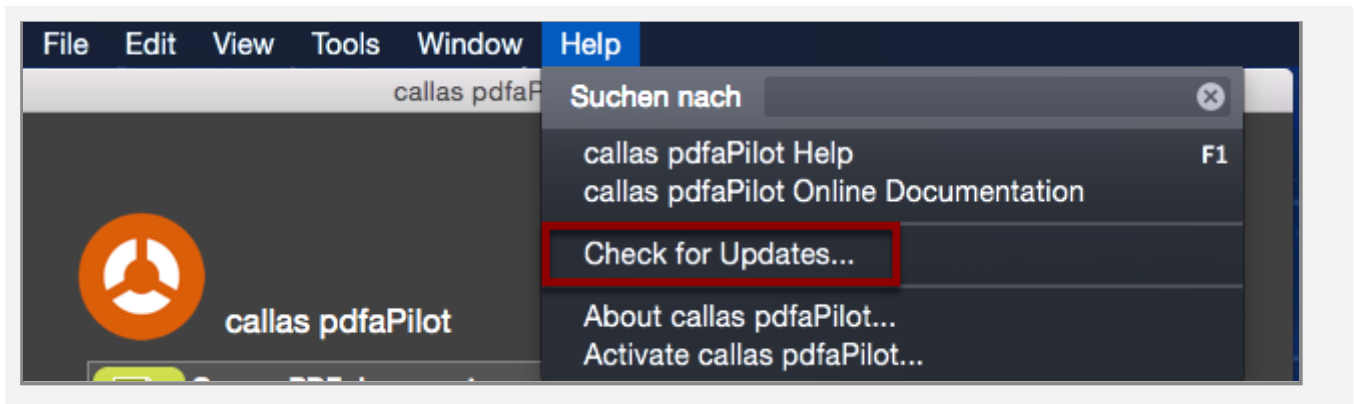
New minor versions can offer new Checks and Fixups as well as new settings for existing Checks and Fixups.

Sometimes also new, predefined Profiles, Checks and Fixups are contained (can easily be identified by a suffix behind the name - e.g. "v6.3").

Therefore it is not possible in most cases, to switch back to a previous minor version (e.g. from version 6.3 back to 6.2). Intensive internal regression tests ensure that a customer should not have any problem with an update.

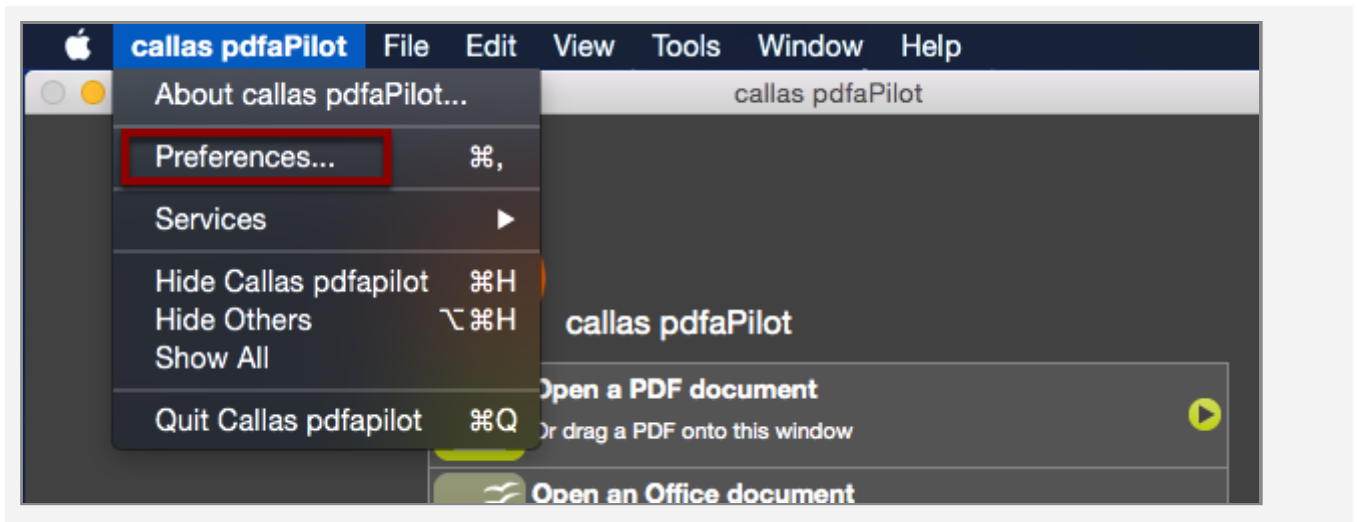
If you want to keep the possibility to switch back to a previous minor version, we recommend a backup of our Profiles. More information in the chapter "Export Libraries".

Manual check for new updates



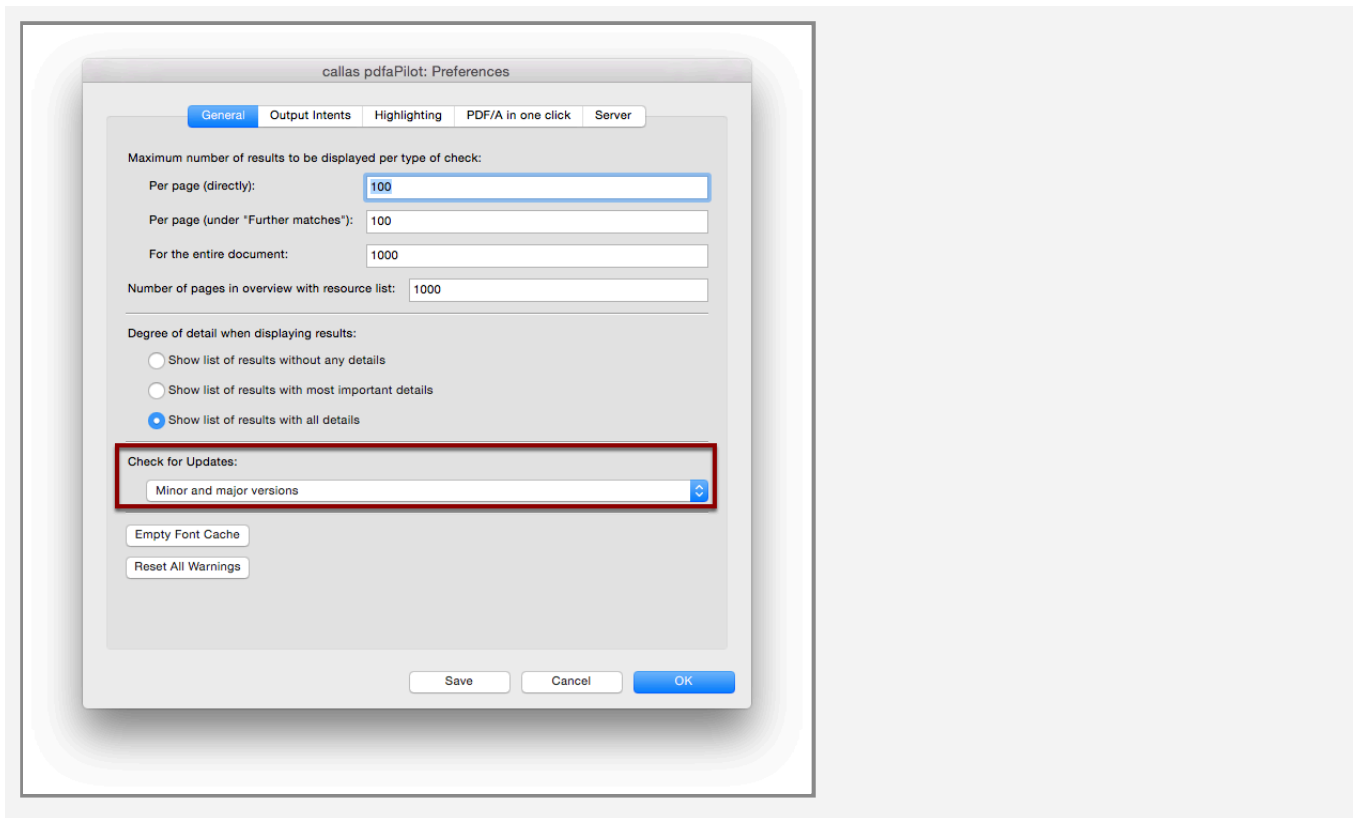
In the "Help" menu you'll find "Check for updates", which will check for the latest available versions.

Go to the update check in the preferences



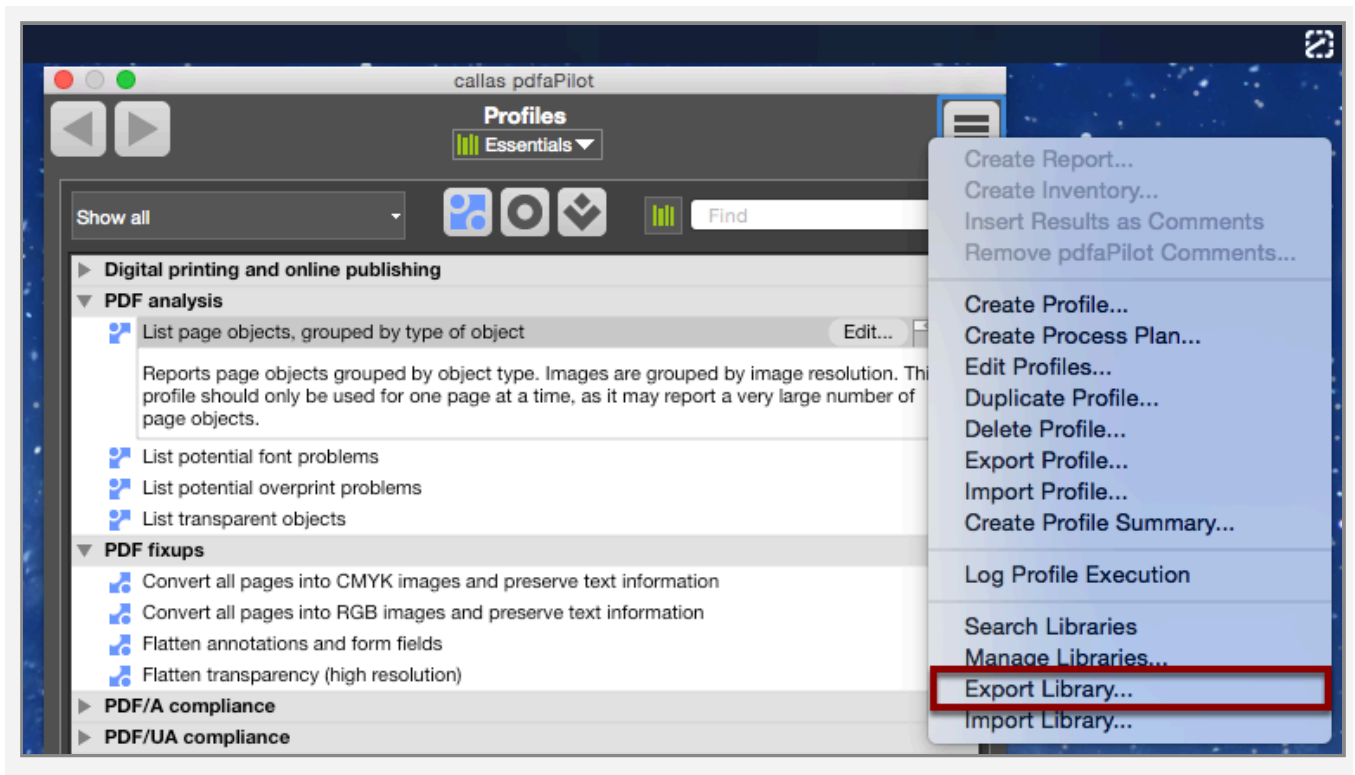
Using the menu bar, you can open the software preferences.

Configure the update check in the preferences



In the "General" tab in the preferences, you can easily define if the software shall check for updates. And if yes: whether it should happen for major versions, minor versions or both.

Export Libraries



You can Backup your set of Profiles using the Profiles window, options button in the upper right corner, "Export Library". This will save your Library into an external file. If you have more Libraries, each Library must be saved separately.

An exported Library will be stored into the selected folder. You can either import this Library on another computer or use this backup for an import later.

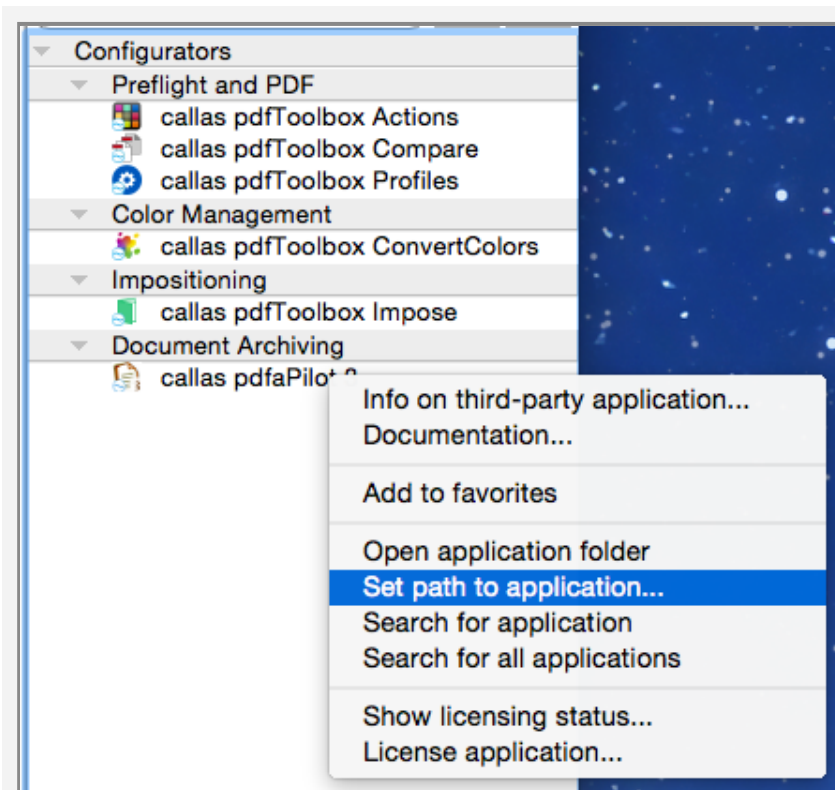
1.10 Update path to pdfaPilot CLI in Switch when installing new version of pdfaPilot

Using callas pdfaPilot Server / CLI in Enfocus Switch

When using pdfaPilot in the Enfocus Switch workflow system pdfaPilot Server/CLI has to be installed on that system.

When an existing version pdfaPilot is updated, Switch won't automatically update the path to the pdfaPilot Server/CLI application.

In order to adjust the path, you have to select the pdfaPilot Configurator element and click on "Set path to application...". Then, set the path to the CLI Version in the file system dialog shown.



1.11 Run as a service (Windows)

Start server, dispatcher or satellite as a service

The callas pdfaPilot Services application is only available for Windows at the moment. For Mac or Linux a Deamon can be used:

- [Create a Deamon using Mac](#)
- [Create a Deamon using Linux](#)

Installation

1. Ensure there is an installation of callas pdfaPilot Server/CLI on the system and the application has been activated successfully.

2. A special executable, which is needed to run pdfaPilot as a service, is located in `/cli/var/Service`.

Copy the executable into the subfolder `"cli"` of the application folder of the server installation.

3. To install, the following command has to be executed on the command line:

```
pdfaPilotService.exe --install
```

- Note: Please confirm the security question of Windows if shown.

4. Open the "Services dialog" of Windows. This dialog can easily be opened by typing the following string into the search field of the Windows start menu or use the following command on the command line:

```
Services.msc
```

5. There should show up 3 new services:

- callas pdfaPilot Server
- callas pdfaPilot Satellite
- callas pdfaPilot Dispatcher

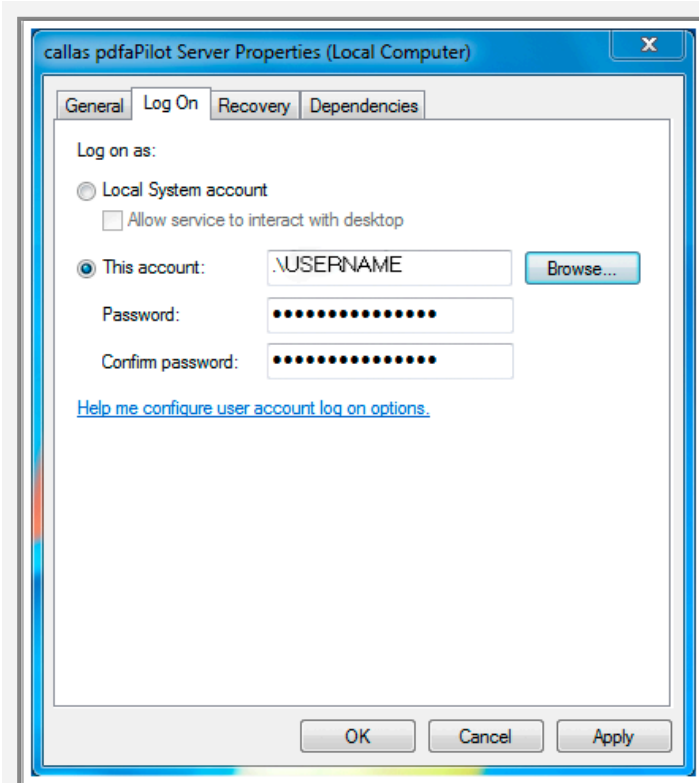
 callas pdfaPilot Dispatcher	Manual	Local System
 callas pdfaPilot Satellite	Manual	Local System
 callas pdfaPilot Server	Manual	Local System

6. Select "pdfaPilot Server" and use the right-click menu item "Action" [or "Properties" depending on the Windows-version] in order to use this service and set in "General" the "Startup Type" to "Automatic" or "Manual".

When "Automatic" is chosen, every started job will continue processing, even when no user is logged on the system. It will even start processing, when the operating system is started.

When using "Automatic", also user details have to be entered into the "Log On" tab.

It must be ensured, that all folders used in the jobsettings can be accessed by the defined user (especially when network paths shall be used by the job).



Configuration of a job

Now a job can be configured using the ServerUI, which can be accessed using the Standalone version (Menu: Tools - Server).

When a job is started, pdfaPilot Standalone can be closed. The services application ensures, that the processing will continue even when the user is logging off.

If a job is using network paths for the used folders, it is recommended to use UNC paths (e.g. "\\192.168.1.22\hotfolder\...") as an assigned drive letter (e.g. "H:\hotfolder\...") is user-specific.

Access by remote

It is possible to connect to a Server running as a service by remote via the local network.

Start pdfaPilot standalone, select menu: "Tools" - "Server" and choose

"Connect with remote server".

Enter the IP of the remote server where the service is running.

After connecting all jobs on the remote server are shown and can be started, stopped or even modified. (Hotfolder paths of any server jobs (IN, OUT, etc.) have to be configured so that they are valid from the service's perspective (the system where the service is running) - and not from the perspective of the controlling standalone application).

Uninstall or change to a new major version

If the Service is no longer needed and the entry shall be removed or when the Service entry has to be updated because of a new major version (e.g. when upgrading from version 7 to version 8), it is recommended to uninstall the service using the following command:

```
pdfaPilotService.exe --uninstall
```


The pdfaPilotService.exe has to be in the same directory like the pdfaPilot.exe (like when installing the Service).

Troubleshooting

If network paths are used for processing jobs, the user should have sufficient rights to access them.

There may be special requirements for converting Office files to PDF when using pdfaPilot as a service. Check http://www.callassoftware.com/goto/tbx_ENG_topdf for the latest details.

Note: In general it is recommended to grant the respective service user administrator privileges. If this level of rights can not set due to internal regulations, some additional settings within the operating system are recommended. [Additional settings with limited user rights](#).

Additional settings for Office conversion with restricted user permissions

If Office conversion is needed out of services using Windows it is recommended to grant the respective service user administrator privileges. If this level of rights can not set due to internal regulations, some additional settings within the operating system are recommended.

The following folders should allow the user the respective access right:

For 64-bit machines	
C:\Windows\Temp	Modify
C:\Windows\syswow64\config	Read
C:\Windows\syswow64\config\systemprofile	Read
C:\Windows\syswow64\config\systemprofile\AppData	Modify
C:\Windows\syswow64\config\systemprofile\Desktop	Modify (Create it, if it does not exist)
C:\Windows\syswow64\config\systemprofile\AppData\Local\Microsoft\Windows\INetCache	Modify (Create it, if it does not exist)

For 32-bit machines	
C:\Windows\Temp	Modify
C:\Windows\system32\config	Read
C:\Windows\system32\config\systemprofile	Read
C:\Windows\system32\config\systemprofile\AppData	Modify
C:\Windows\system32\config\systemprofile\Desktop	Modify (Create it, if it not exist)
C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\INetCache	Modify (Create it, if it not exist)

Additional settings:

- Set the 32-bit folder preferences (for details see above) in addition to the 64-bit preferences on 64-bit systems running 64-bit versions of Microsoft Office
- Set the default printer to XPS Document Writer

DCOM settings:

- Launch DCOMCNFG by using:

```
C:\WINDOWS\SysWOW64> mmc comexp.msc /32
```

- Go to Computers MyComputer DCOM Config.
- Right-click the application that you want to automate.
The application names are listed in the table below:

Application	DCOM Name
Microsoft Access 2007/2010/2013/2016	Microsoft Access Application
Microsoft Excel 2007/2010/2013/2016	Microsoft Excel Application
Microsoft Office Word 2007	Microsoft Office Word 97 - 2003 Document
Microsoft Word 2010/2013/2016	Microsoft Word 97 - 2003 Document

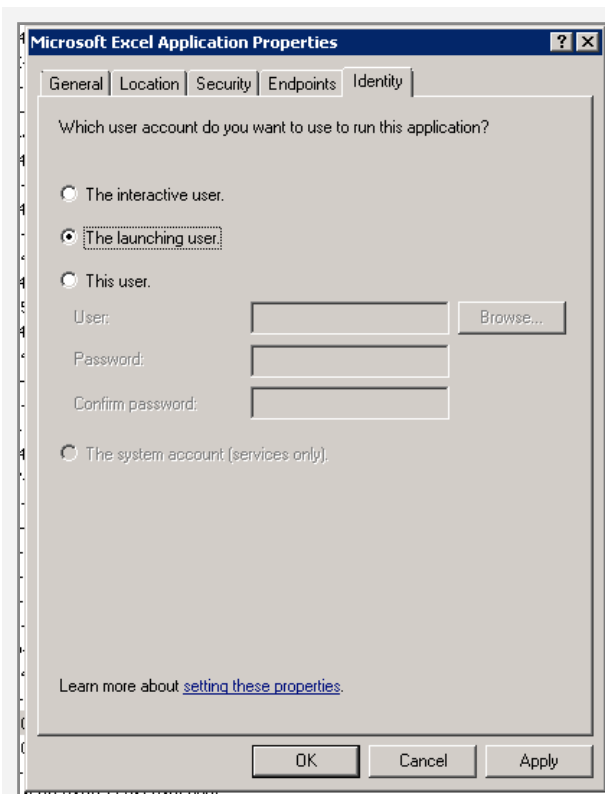
- On some systems Microsoft Word is not displayed and you will have to use
`{00020906-0000-0000-C000-000000000046}`
instead.

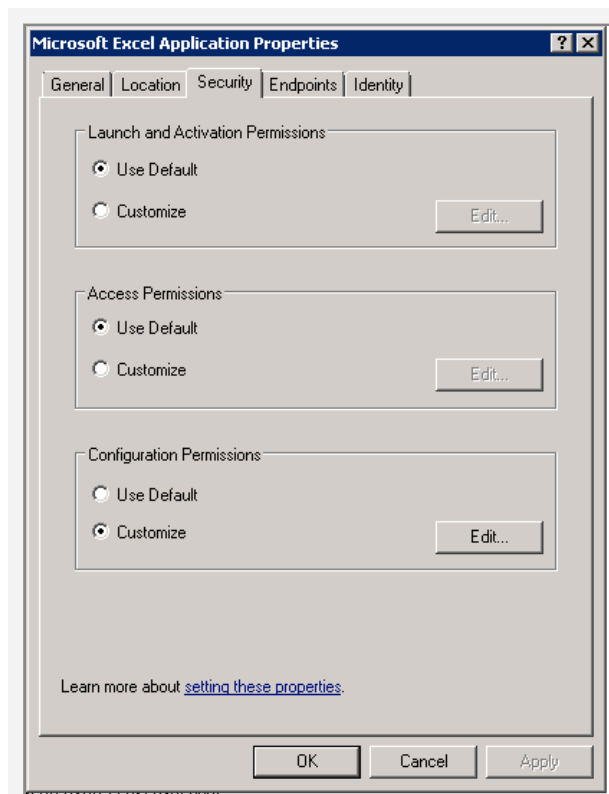
- Click Properties to open the property dialog box for this application.
- Verify Identity and Security tabs

Note

For Microsoft Excel, Microsoft Word and Microsoft PowerPoint the DCOM Identity "The launching user" must be selected.

For Microsoft Visio the DCOM Identity "The interactive user" needs to be selected.





1.12 Create a Daemon using Mac



com.callassoftware.pdfapilotserver.agent.plist

Other than a Service using Windows, a so-called Daemon using Mac can either be started when a User is logged in or (alternatively) when the system is started:

Copy the attached plist file to the following location:

/Library/LaunchAgents

(for run at Login)

or to

/Library/LaunchDaemons

(for run at Boot)

The agent will be activated with the following call:

Terminal:

```
launchctl load /Library/LaunchAgents/  
com.callassoftware.pdfapilotserv-  
er.agent.plist
```

Using the following call, the agent can be deactivated:

Terminal:

```
launchctl unload /Library/LaunchAgents/  
com.callassoftware.pdfapilotserv-  
er.agent.plist
```

If a Server-Job is active, this Job will start as soon as the system is started and files in the respective hot folders will be processed.

1.13 Run as an autostart systemd service (Linux)

The ReadMe as well as the template for running pdfaPilot as an autostart systemd service under Linux can also be found in the installation package in the following folder:

etc/systemd

systemd callas service usage/installation example

Note: Requires root permissions.

Note: The service description must *not* contain a privateTemp attribute.

Note: The whole <install_dir> path including all higher level directories need to have at least read-and-execute permissions.

Recommendation: Unpack the installer below a non-userspecific directory (such as e.g. /opt/callas).

Note: A `--cachefolder` option *must* be used, *even* if the user executing the service actually has a home directory. That is because per default a subdirectory of '/usr/share/callas software' is used as the storage folder and creating the needed directory structure would require permissions to run with the root user account.

Note: The <install_dir> and the <cache folder> must be owned by the user executing the service.

Recommendation: Choose a top level folder to contain the <install_dir> and the <cache folder>. Then change ownership for the whole top level folder, e.g. for pdfaPilot running in dispatcher mode:

```
<p>sudo mkdir -p /opt/callas/dispatcher  
  
sudo chown -R nobody:daemon /opt/callas/dispatcher</p>
```

Recommendation: to make future updates easier it is recommended to have a download folder, e.g. /opt/callas/dispatcher/download. A symlink would then point to the concrete pdfaPilot version to be used:

```
<p>sudo -u nobody bash
cd /opt/callas/dispatcher
mkdir downloads
cd downloads
wget https://www.callassoftware.com/extranet/callas_pdfaPilotCLIandServer/
callas_pdfaPilotCLI_x64_Linux_12-3-369.tar.gz
tar xvpf callas_pdfaPilotCLI_x64_Linux_12-3-369.tar.gz
cd ..
ln -s downloads/callas_pdfaPilotCLI_x64_Linux_12-3-369 callas_pdfaPilot_CLI</p>
```

Preparation

1. Use one of the prepared service templates such as ...
 - callas-dispatcher.service
 - callas-hotfolder.service
 - callas-satellite.service
2. Adjust the service description to fit your needs (e.g. change real installation directory and/or specify a `--licenseserver` option)
3. Setup a cache folder (Note: The cache folder must be owned/writable by the executing user, e.g. by running 'sudo -R chown nobody:daemon <given_cache_folder>')
4. Activate the license. This needs to be done with the user account that is executing the service and the specified `--cachefolder`

```
<p>sudo -u nobody bash
cd downloads/callas_pdfaPilotCLI_x64_Linux_12-3-369
./pdfaPilot --keycode myname mycompany mylicense.pdf --cachefold-
er=/opt/callas/dispatcher/cache
./pdfaPilot --activate myactivation.pdf --cachefolder=/opt/callas/
dispatcher/cache</p>
```

Installation

Copy the service description to the systemd service directory (as an example we are going to use the callas-dispatcher.service)

Note: /lib/systemd/system/callas-dispatcher.service must be a regular file and *not* a symbolic link.

Note: A 'sudo systemctl daemon-reload' is needed whenever /lib/systemd/system/callas-dispatcher.service is changed.

```
<p>sudo cp callas-dispatcher.service to /lib/systemd/system </p>
```

Check if the service is working as expected:

```
<p>sudo systemctl daemon-reload
sudo systemctl start callas-dispatcher
systemctl status callas-dispatcher </p>
```

Controlling the service

Check the service status:

```
<p>systemctl status callas-dispatcher</p>
```

Stop the service:

```
<p>sudo systemctl stop callas-dispatcher
systemctl status callas-dispatcher</p>
```

Start the service:

```
<p>sudo systemctl start callas-dispatcher
systemctl status callas-dispatcher</p>
```

Restart the service:

```
<p>sudo systemctl restart callas-dispatcher
systemctl status callas-dispatcher</p>
```

Set callas dispatcher service to auto start on reboot:


```
<p>sudo systemctl enable callas-dispatcher</p>
```

To disable the callas dispatcher service on next reboot:

```
<p>sudo systemctl disable callas-dispatcher</p>
```

2. callas pdfaPilot - First steps

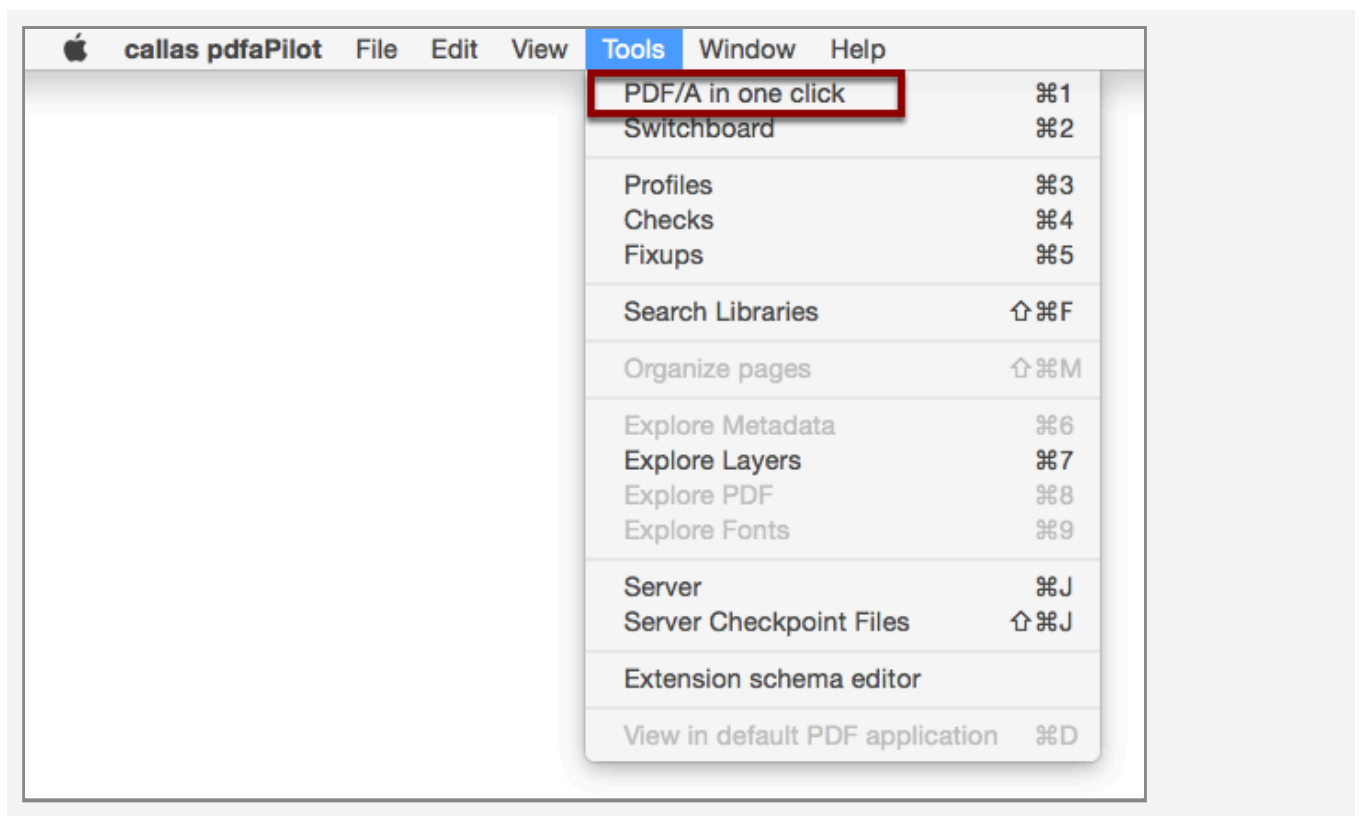
2.1 Analyzing and converting files using “PDF/A in one click”

Validating and creating PDF/A files is incredibly easy with pdfaPilot. Open the “PDF/A in one click” dialog to do exactly what it says: convert files to PDF/A or validate them with a single click.

Opening “PDF/A in one click”

“PDF/A in one click” is opened differently depending on whether you are using pdfaPilot as a standalone solution or as an Acrobat plug-in.

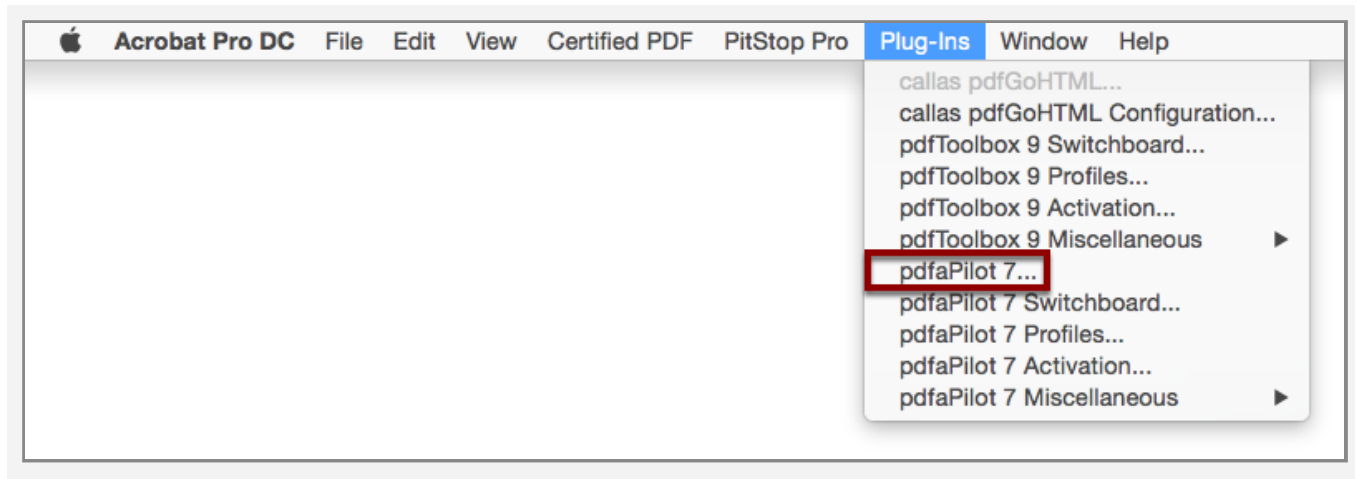
Starting “PDF/A in one click” in the standalone pdfaPilot



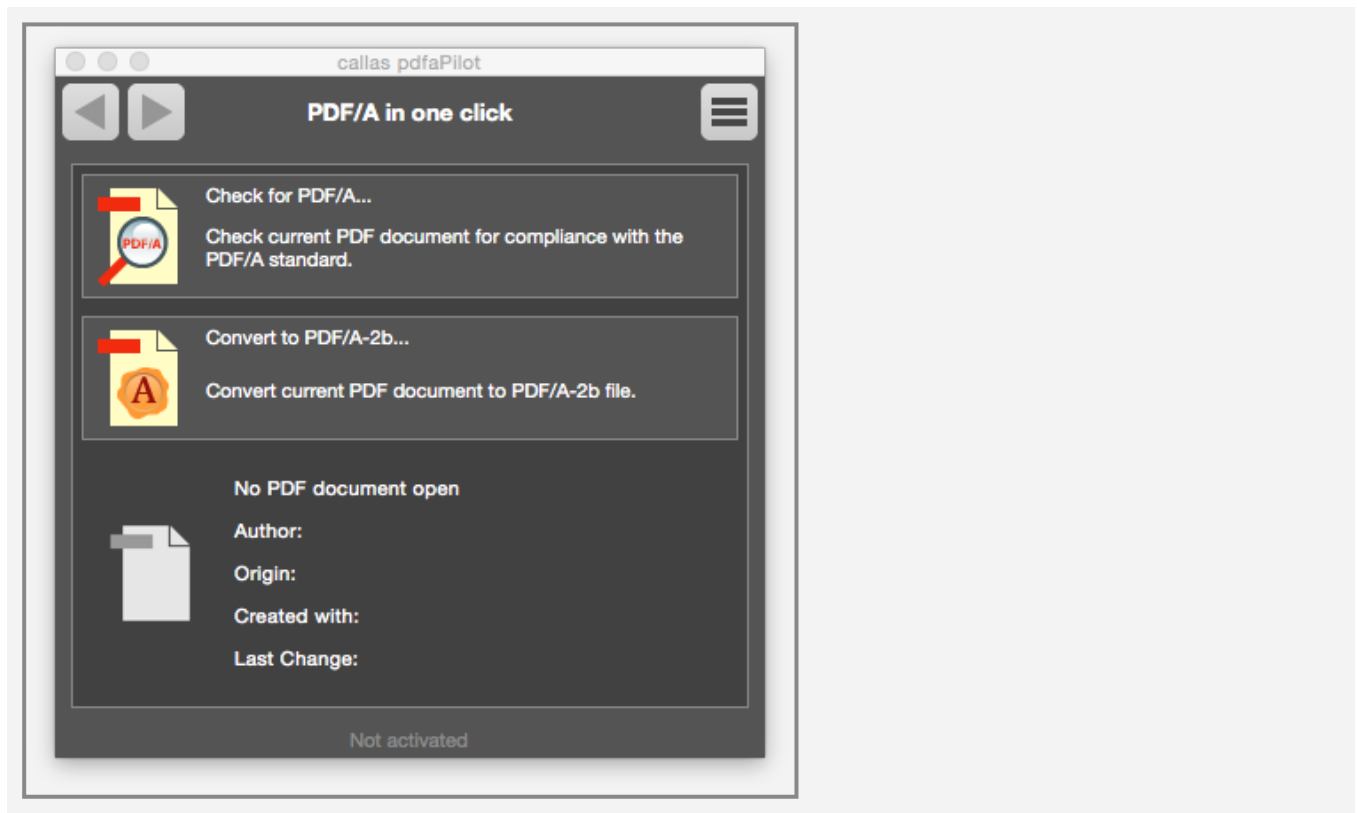
You can open PDF/A in one click by opening Tools > PDF/A in one click in the menu.

Alternately, you can use the keyboard shortcut Cmd+3.

Opening “PDF/A in one click” in the pdfaPilot Acrobat plug-in



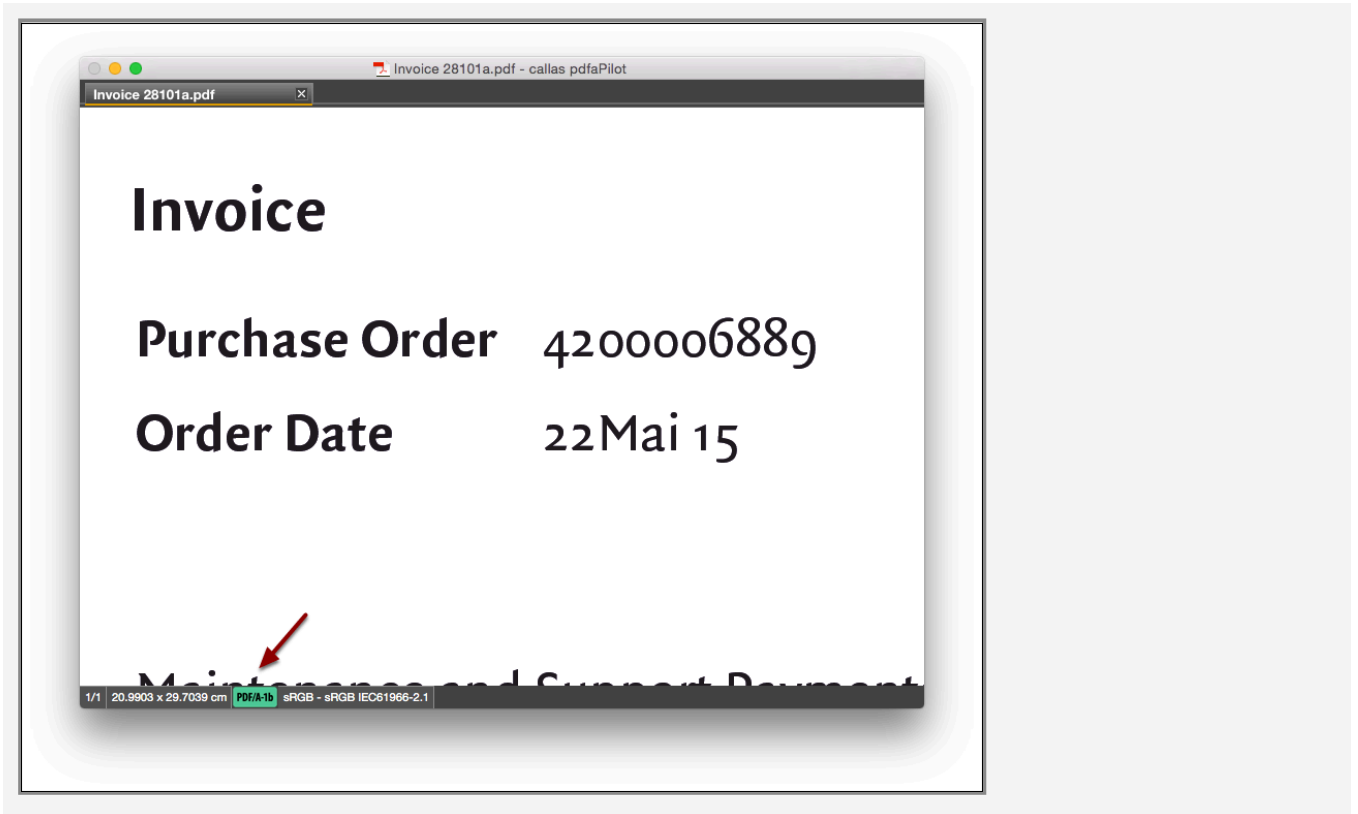
In Acrobat, you can access pdfaPilot’s PDF/A in one click function via Plug-Ins > pdfaPilot <version number> in the menu.



The PDF/A in one click window will then be displayed.

Case 1: The validated PDF file is PDF/A compliant

Open a PDF file in pdfaPilot.



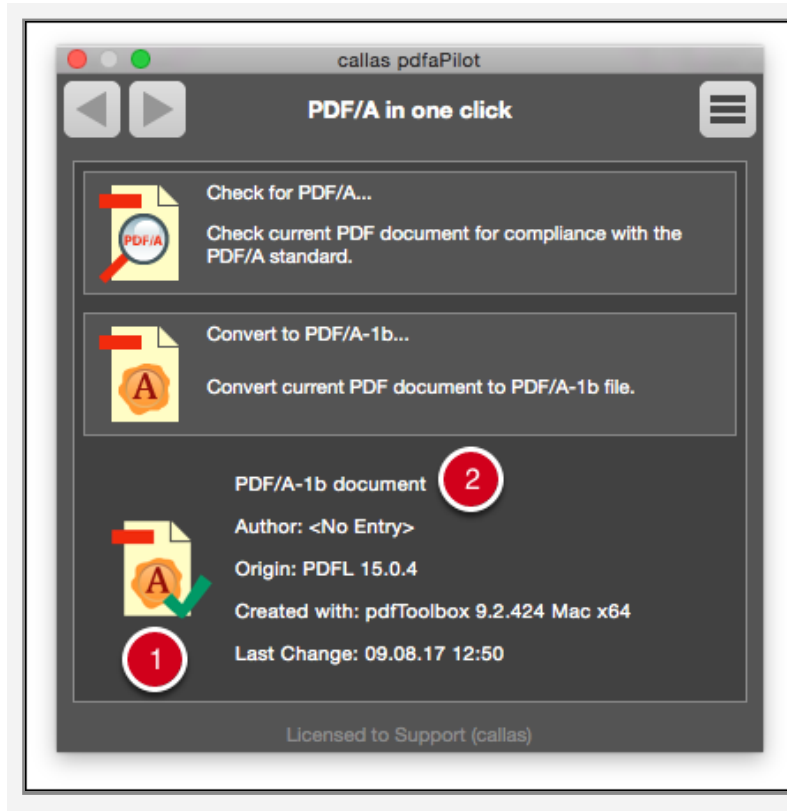
At the bottom of the file window, you will immediately be shown a notification if the current PDF document is a PDF/A document.

Here, pdfaPilot is reporting that the PDF document has been identified as a PDF/A-1b file.

The following test will inspect the file thoroughly.



1. The lower section of the dialog shows general file information as well as its PDF/A status. The current file has not yet been verified.
2. If you want to find out whether your file already complies with the PDF/A standard, click "Check for PDF/A..."

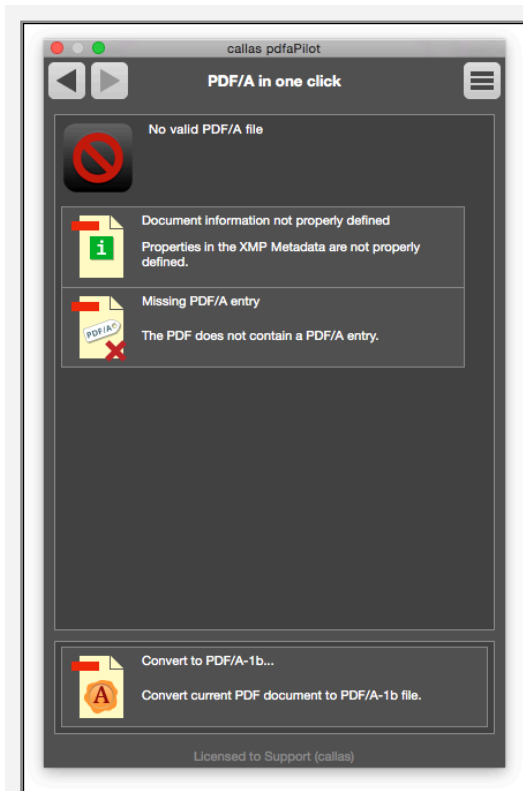


If the PDF file is checked and found to be PDF/A-compliant, pdfaPilot will report this in the window that opens.

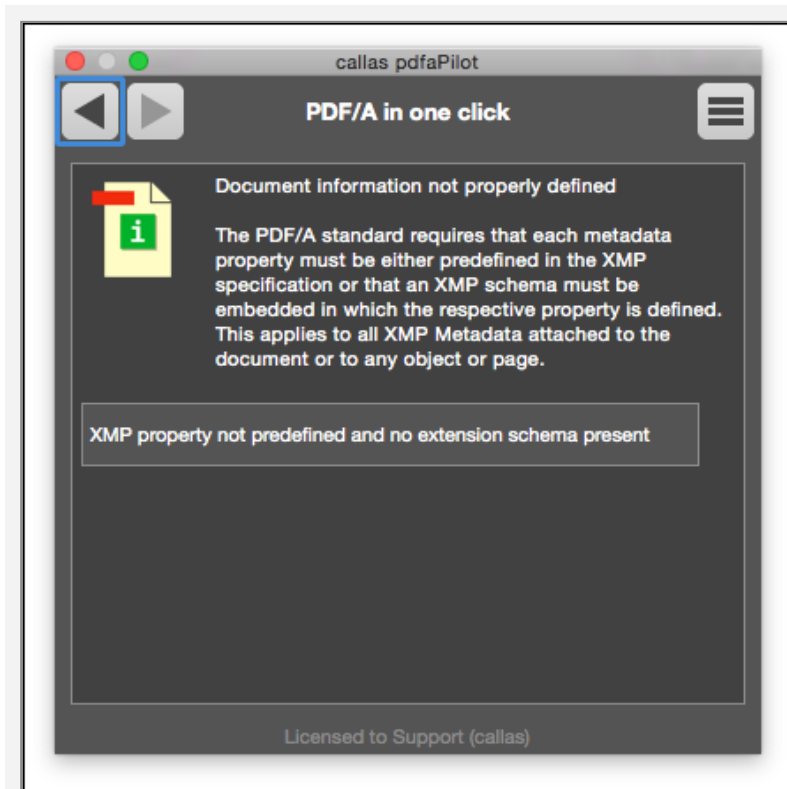
1. A green check mark indicates that the PDF/A check returned a positive result.
2. pdfaPilot will also inform you of the document's PDF/A level.

Case 2: The validated PDF file is not PDF/A compliant

If the file features any properties which are not permitted under PDF/A, these will be indicated in a brief overview at the end of the validation process.



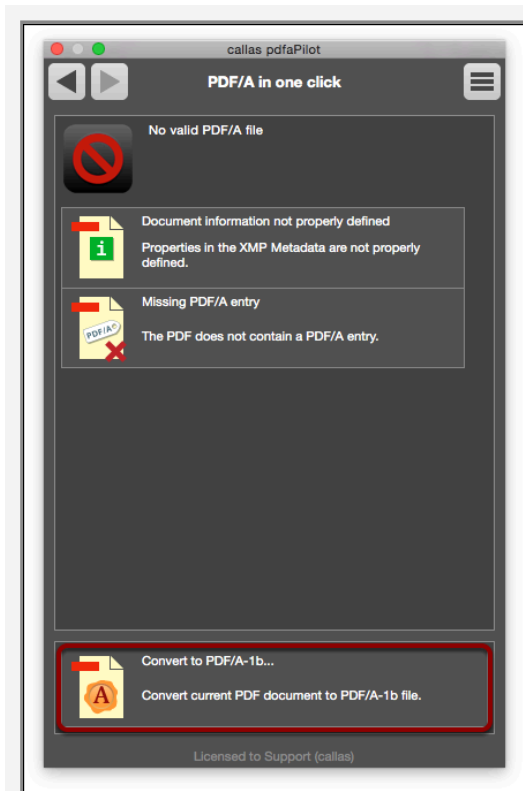
Click on an entry in the list to see the details of the problems that have been found.



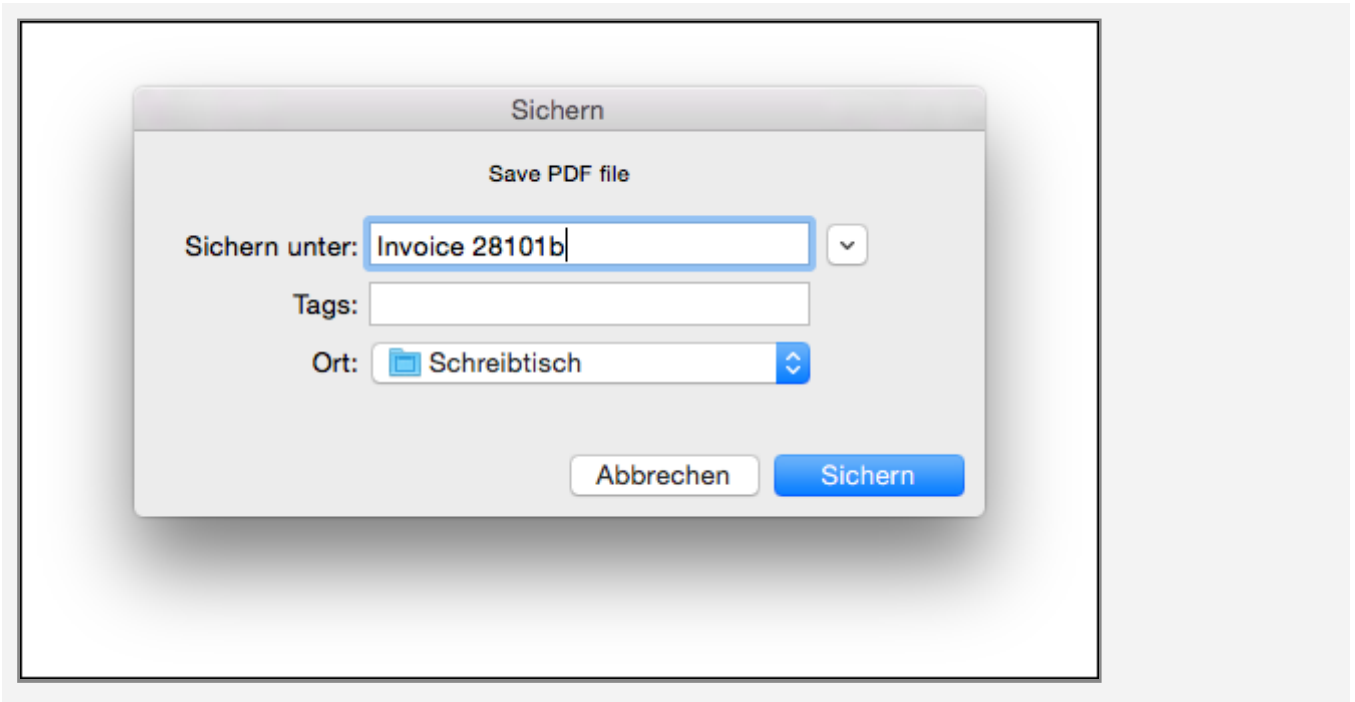
These types of PDF files can usually be converted to PDF/A.

The preset level is PDF/A-b.

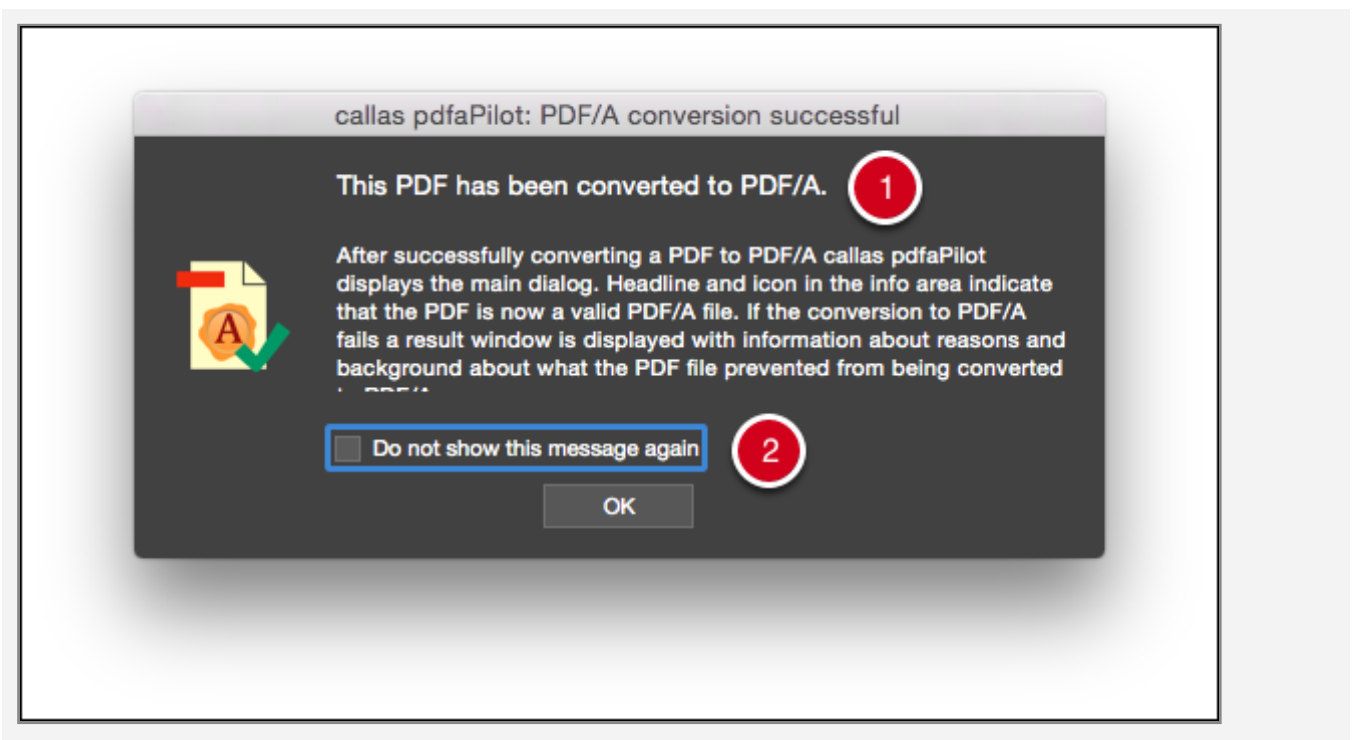
This can be changed in the [Preferences](#) menu.



Most problems can be easily solved by clicking on the **Convert to PDF/A-1b** button towards the bottom of the window.



A dialog will appear, allowing you to enter a name and location for the newly created PDF file.

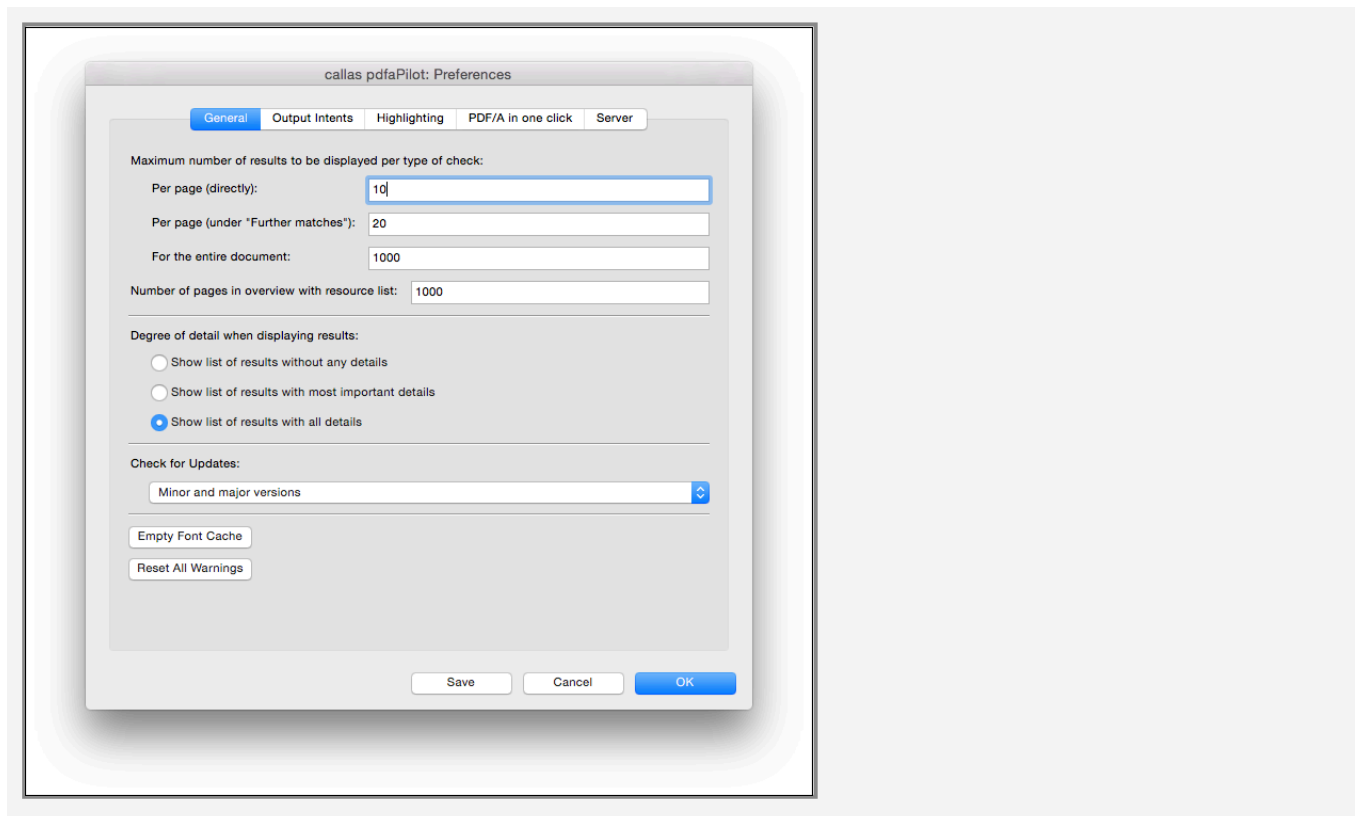


1. A message will appear informing you that the file was successfully converted.
2. You can check the box at the bottom to prevent the message from being shown again.

2.2 Preferences

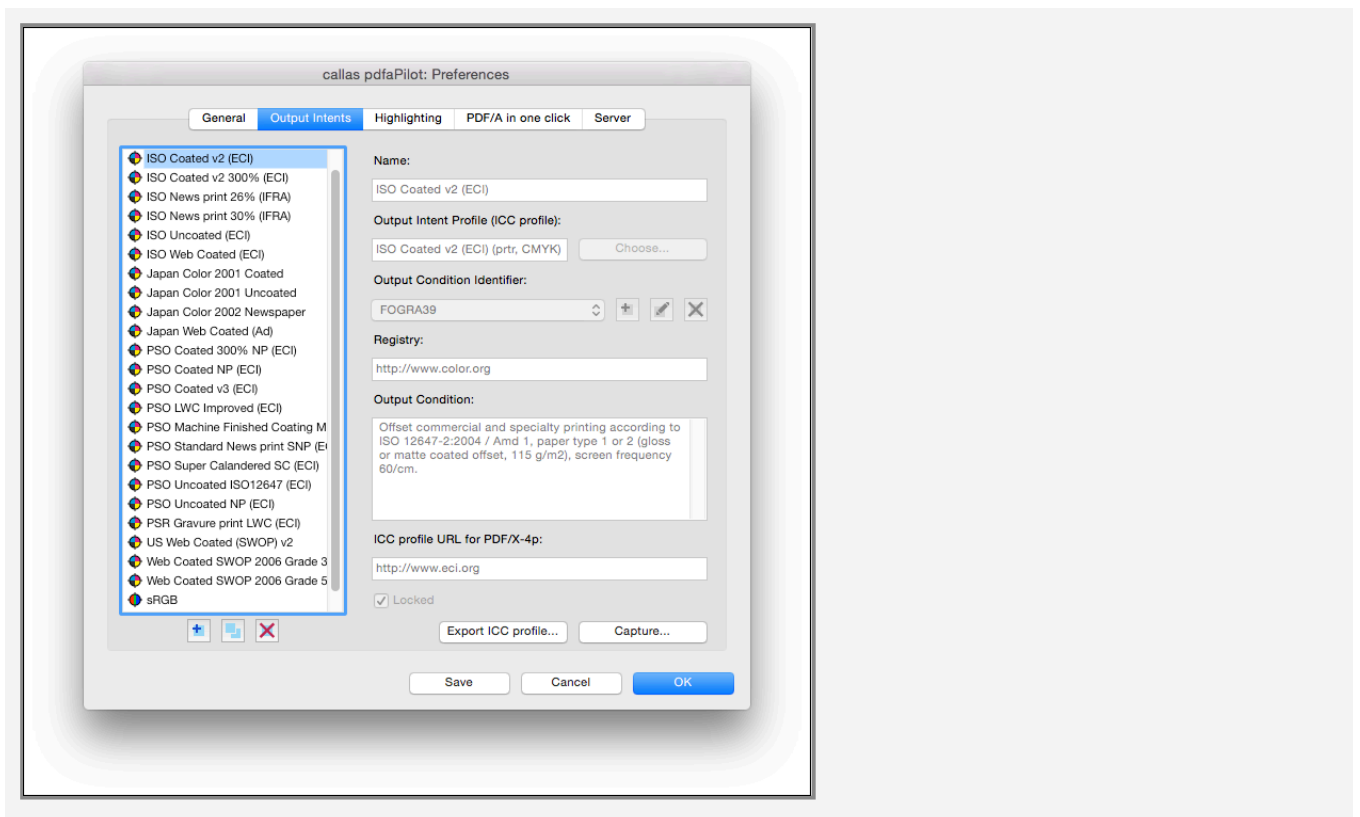
The Preferences dialog offers level-of-detail settings for customizing power-tool results, lets you specify how to highlight problems when generating a report, and provides an editor for creating and adjusting output conditions (OutputIntents). It also allows you to specify the PDF/A level to which PDF files will be converted if they fail to pass the initial check.

General



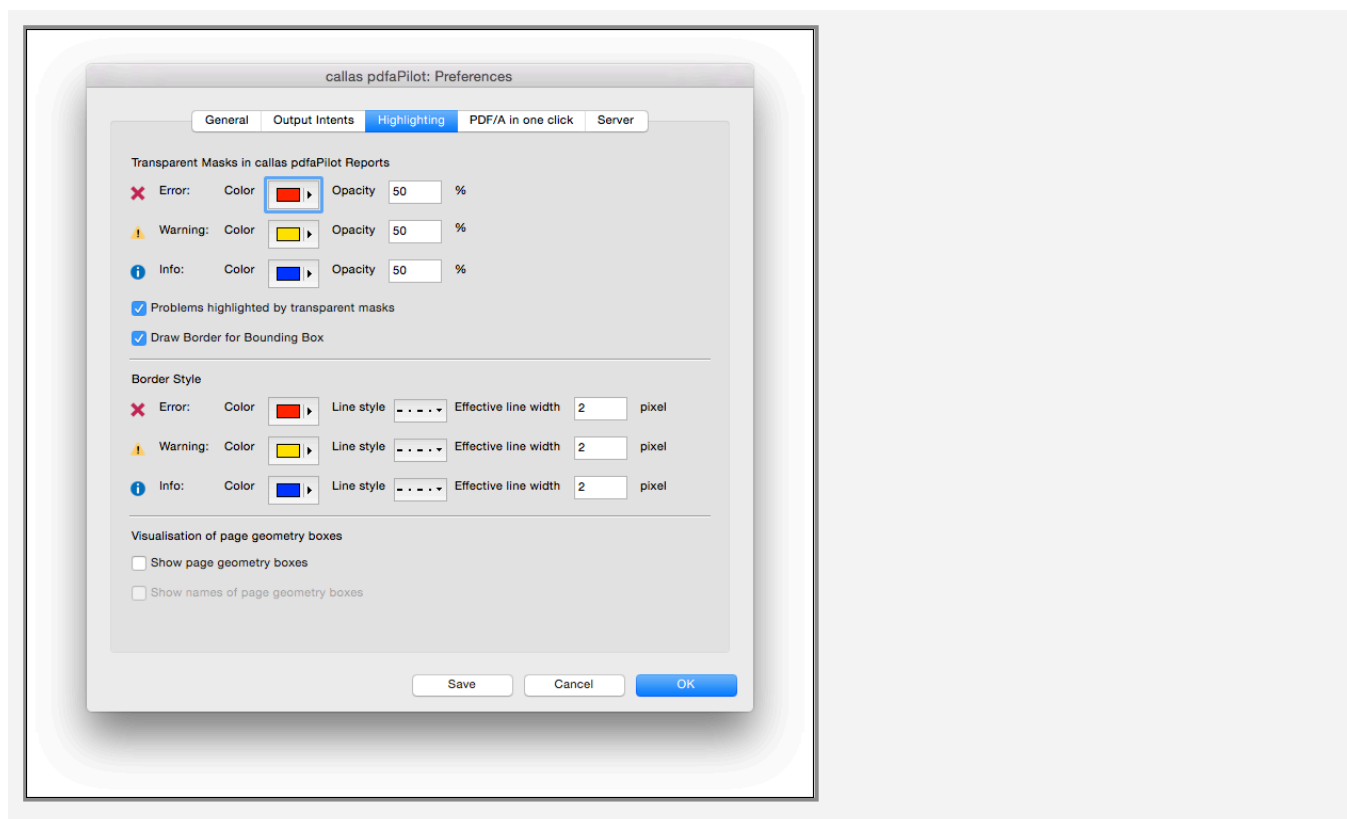
1. You can limit the maximum number of results to show per page or per document.
2. The level of detail when displaying results can be customized.
3. You can choose whether or not to check for updates.
4. You can also empty the font cache and reset all warnings within pdfaPilot.

OutputIntents



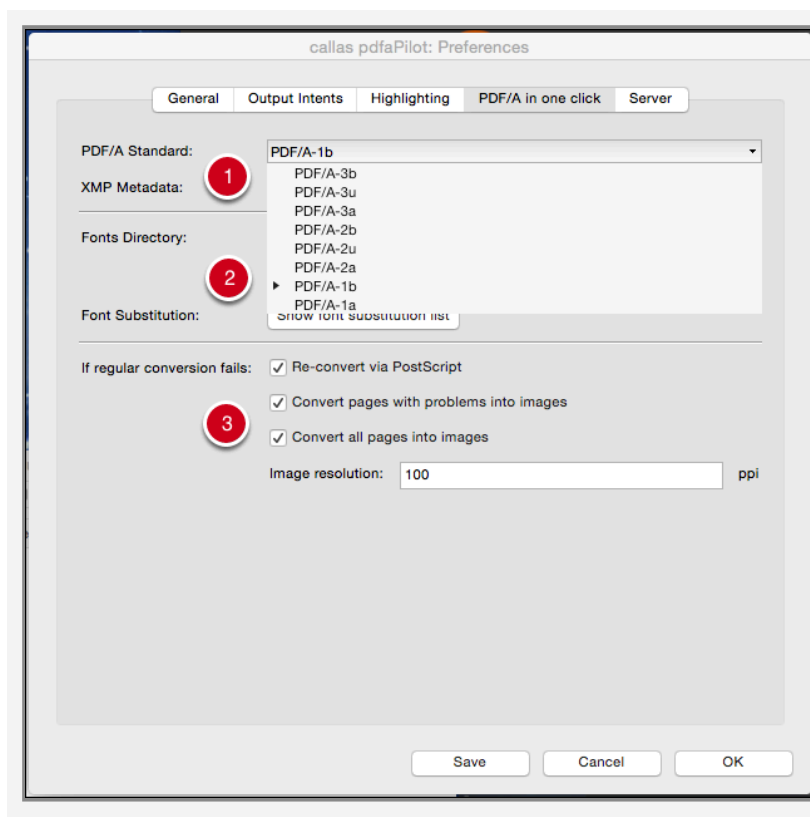
Here you can create, edit or export OutputIntents (output conditions) as well as extracting ICC profiles.

Highlighting



This tab lets you specify the color and style for highlighted objects within a PDF document or the transparent masks in a PDF mask report.

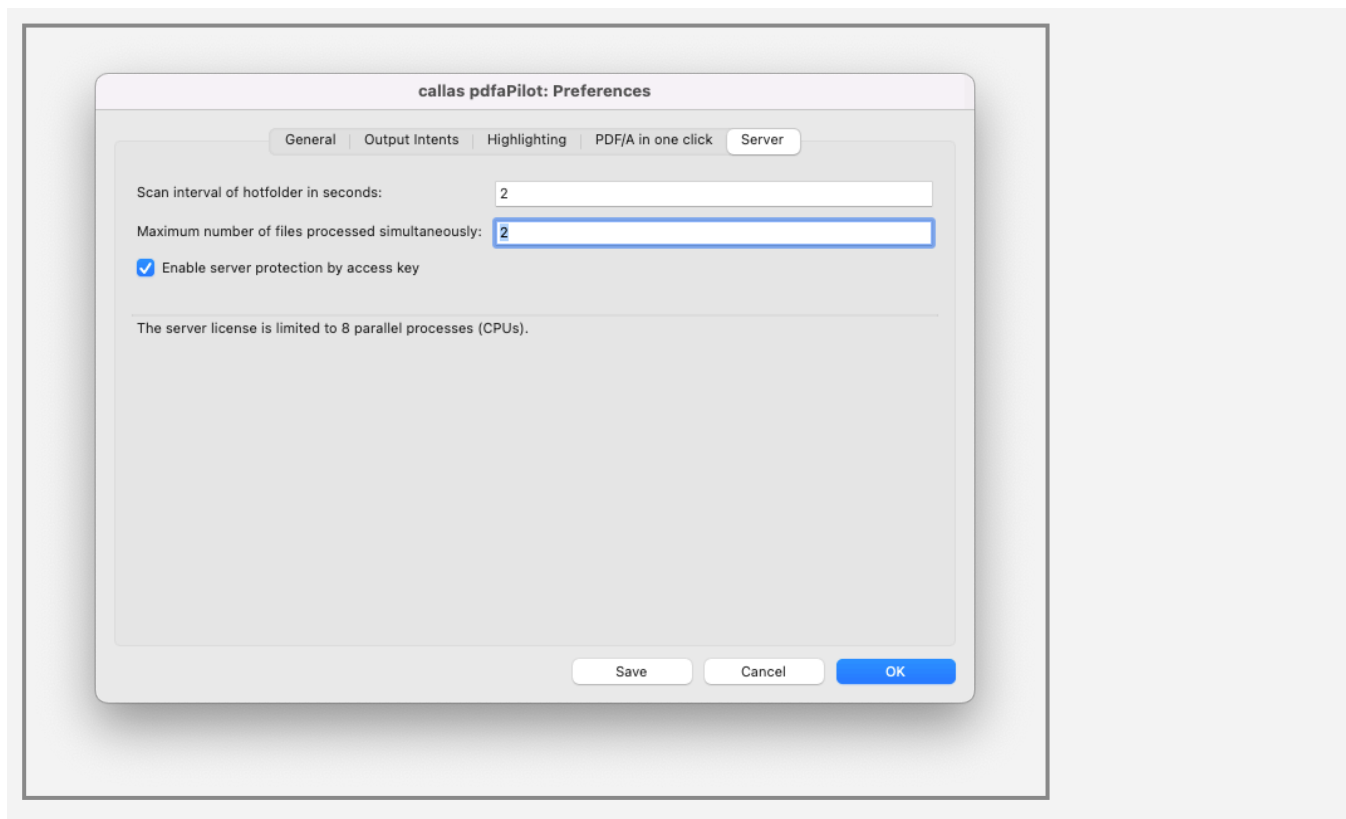
PDF/A in one click



1. This window lets you set the **desired PDF/A standard** as well as other detailed conversion settings when using the “PDF/A in one click” function.
2. The next section in this tab deals with fonts and font substitution.
3. You can also choose which steps to take if regular conversion to PDF/A fails (Re-convert via PostScript, Convert all pages with problems into images, Convert all pages into images.)

Server

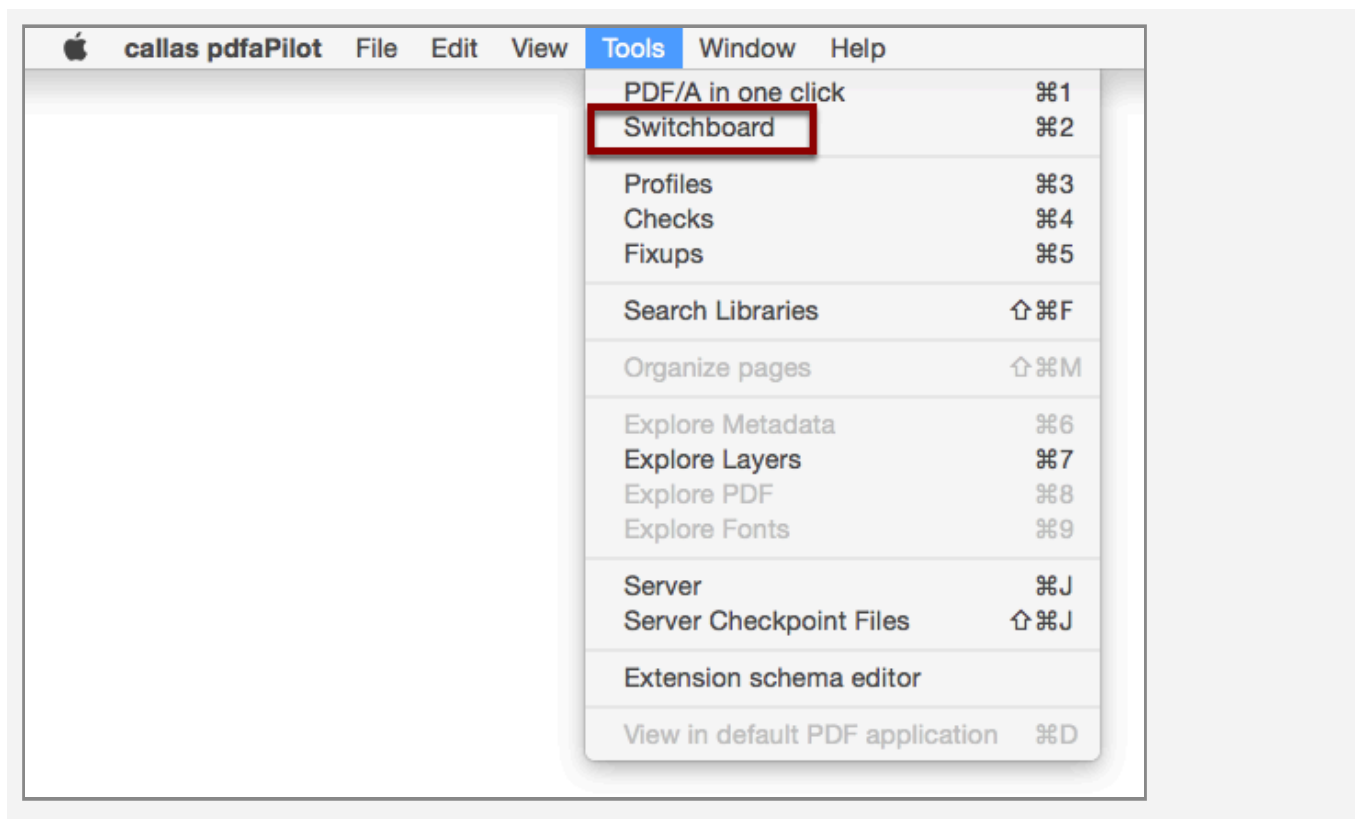
This tab lets you configure the settings for the server version of pdfaPilot. These server settings will only be applied when a server is started from pdfaPilot Desktop, they will not affect the settings of an already running server.



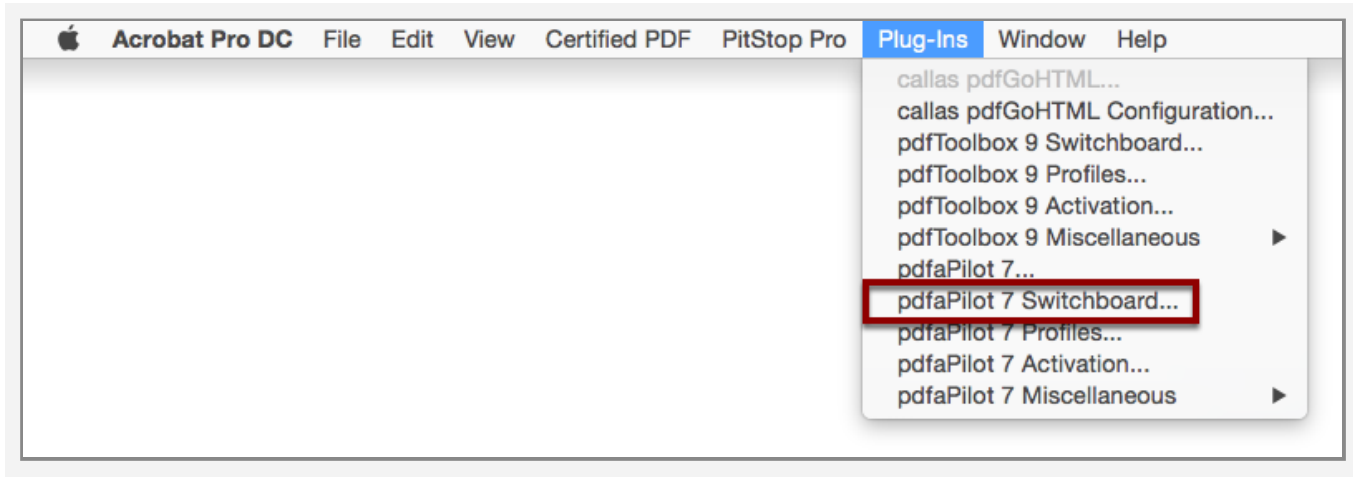
2.3 The Switchboard: An overview

As well as the automatic Fixups applied by pdfaPilot when converting files to PDF/A, the software also offers a wide range of other actions for manipulating a PDF file. Many of these can be accessed via the Switchboard.

In pdfaPilot's standalone edition, you can open the Switchboard via Tools > Switchboard in the menu or use the keyboard shortcut Cmd+2 (Windows: Ctrl+2).



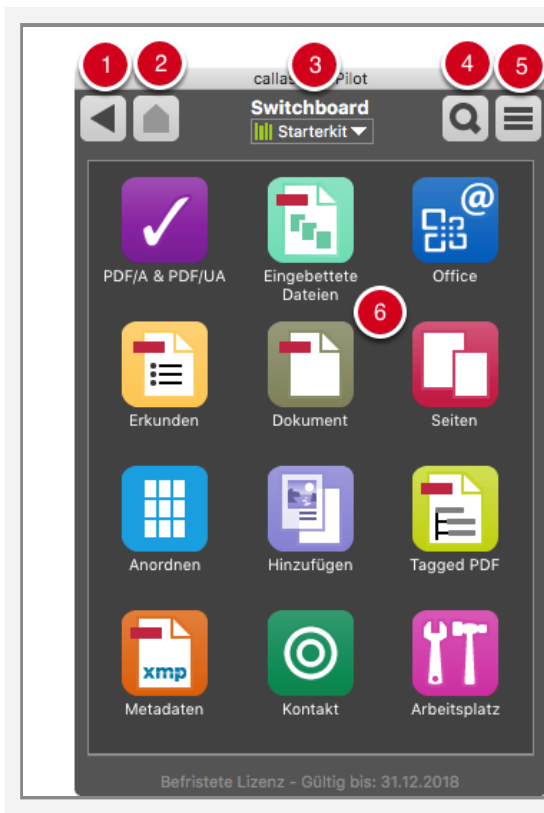
If you are using pdfaPilot as an Acrobat plug-in, open the switchboard via Plug-Ins > pdfaPilot <version number> Switchboard.



The pdfaPilot Switchboard window will be displayed.

The Switchboard - Structure

The Switchboard has a very simple, clear user interface, letting you get straight to work.



1. The arrow button will always take you back to the previous section.

2. The home symbol opens the main window.
3. The central drop-down menu at the top lets you choose between a range of libraries (such as PDF standards and Prepress.)
4. The magnifying glass opens the search tool for when you want to find a specific Action by name.
5. The button in the top right can be used to manage libraries.
6. The main section contains the various groups for different types of tasks.

The groups

The main window shows the various **groups** of Actions.

Each group, in turn, contains a range of **Actions**. These can be accessed by clicking on the group's symbol.

“PDF/A & PDF/UA” group



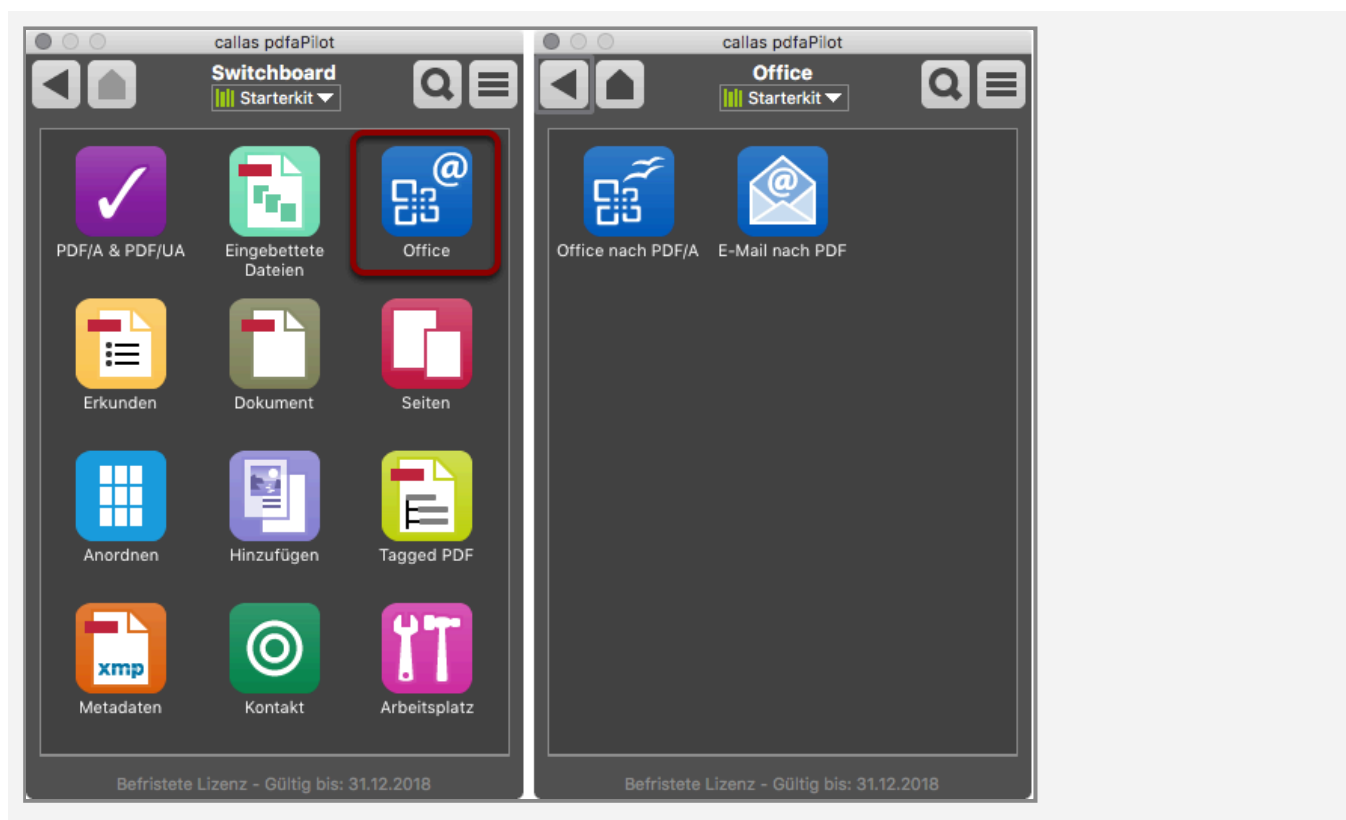
The Actions in this group are used to convert and optimize the PDF/A (long-term archiving) and PDF/UA (universally accessible PDF) standards.

“Embedded files” group



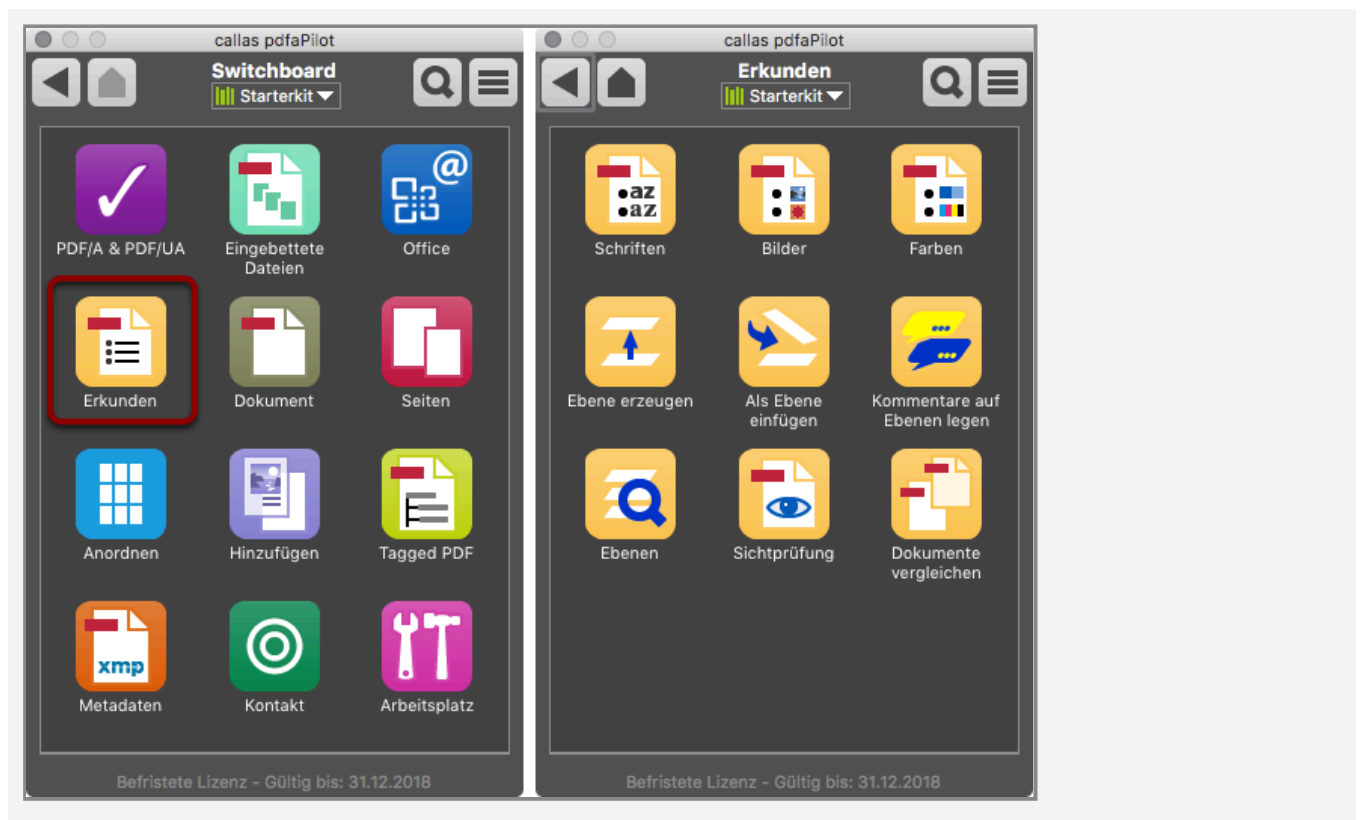
The Embedded files group offers a range of Actions for embedding and extracting files.

“Office” group



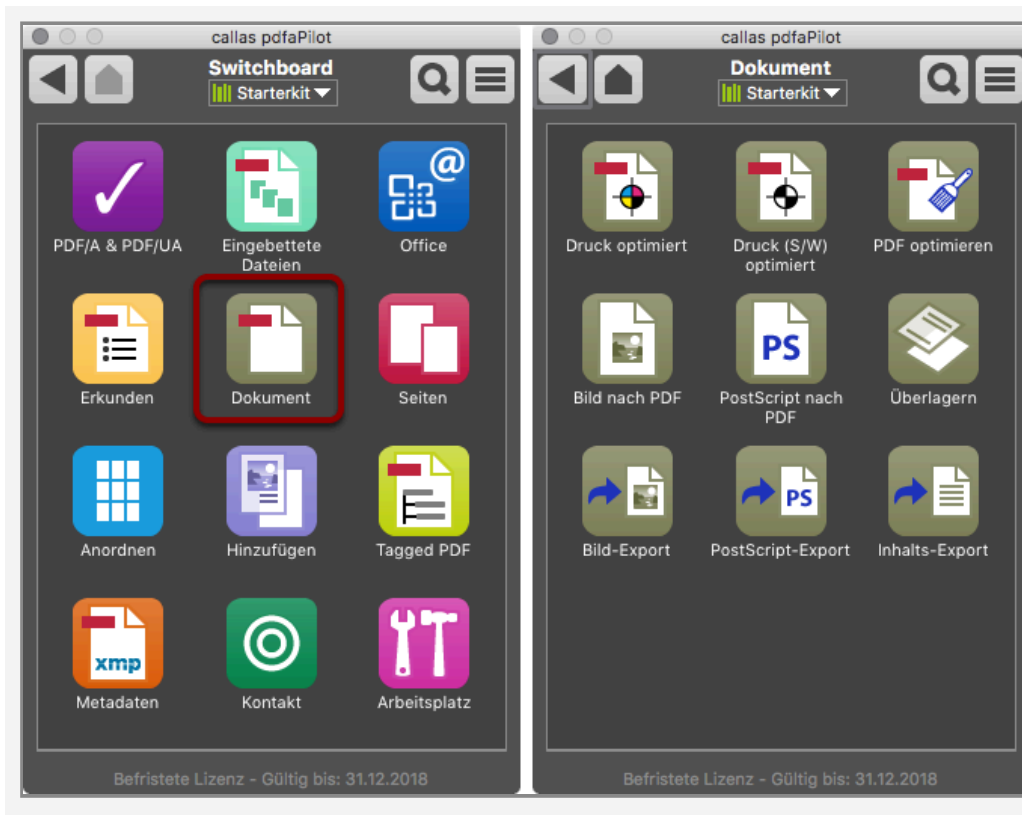
The actions in the **Office** group let you convert Office files and emails into PDF/A format.

“Explore” group



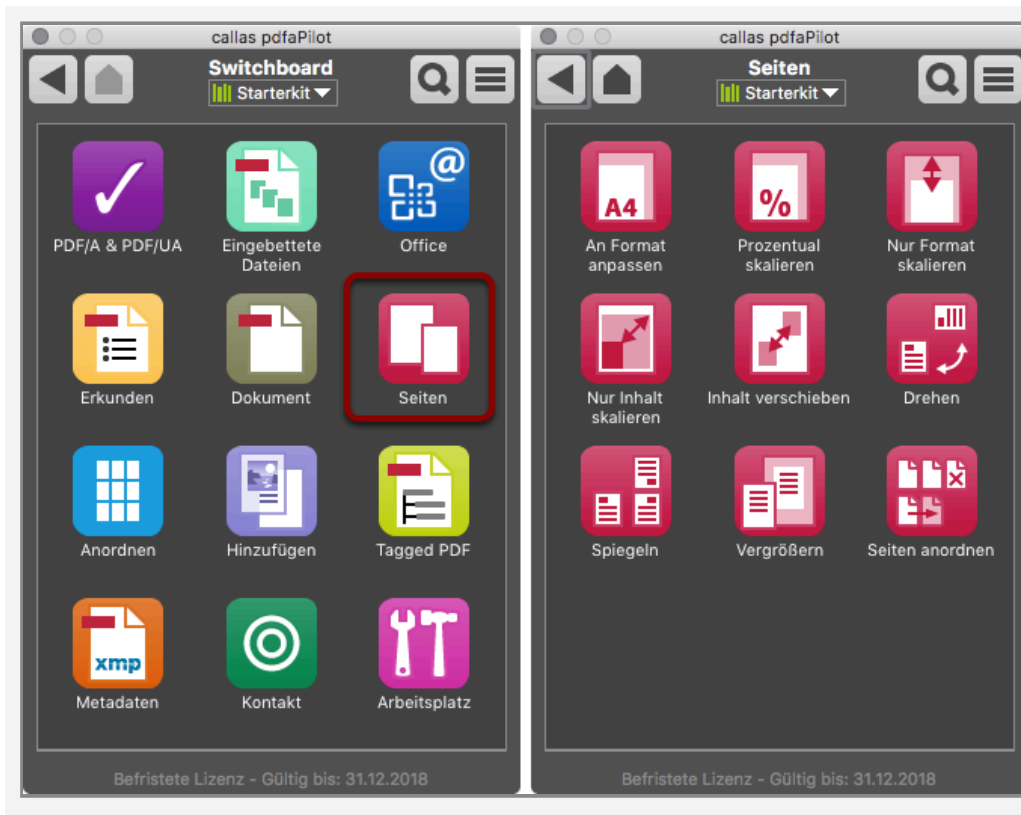
The Explore group provides a range of actions used to create different types of reports based on your files. Other actions involve layers and comments.

“Document” group



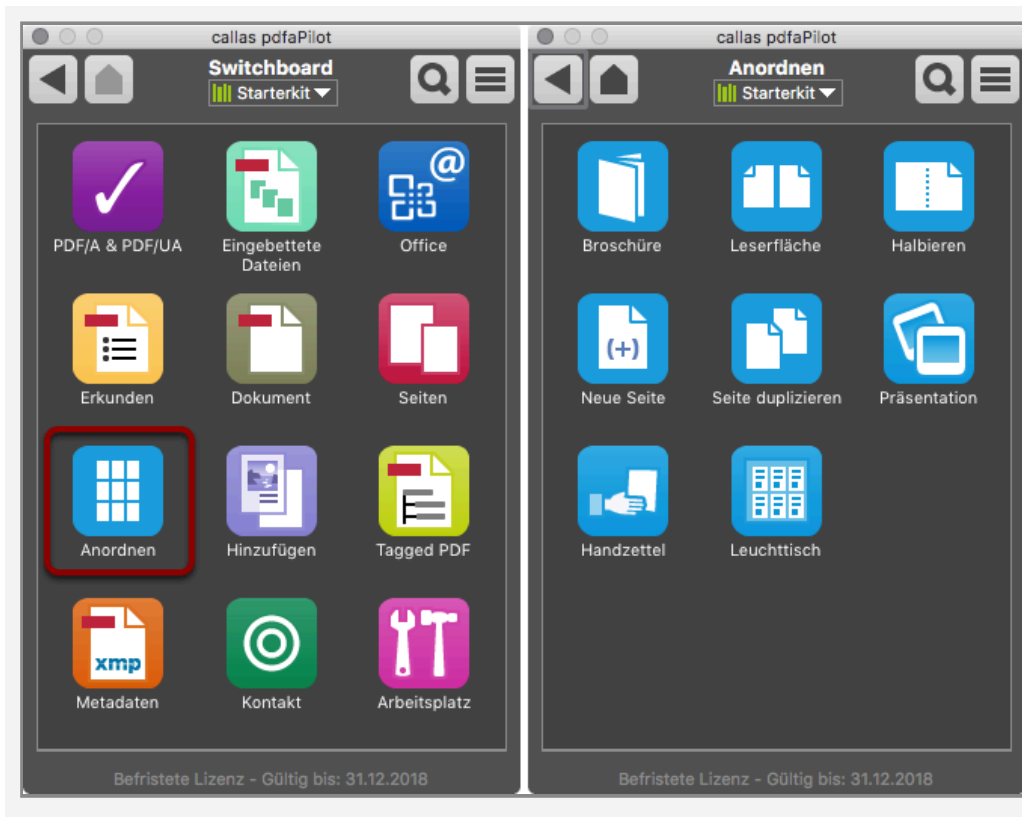
The Document group includes Actions used to change the properties of a document.

“Pages” group



The Pages group provides Actions used to change the page layout or order.

“Arrange” group



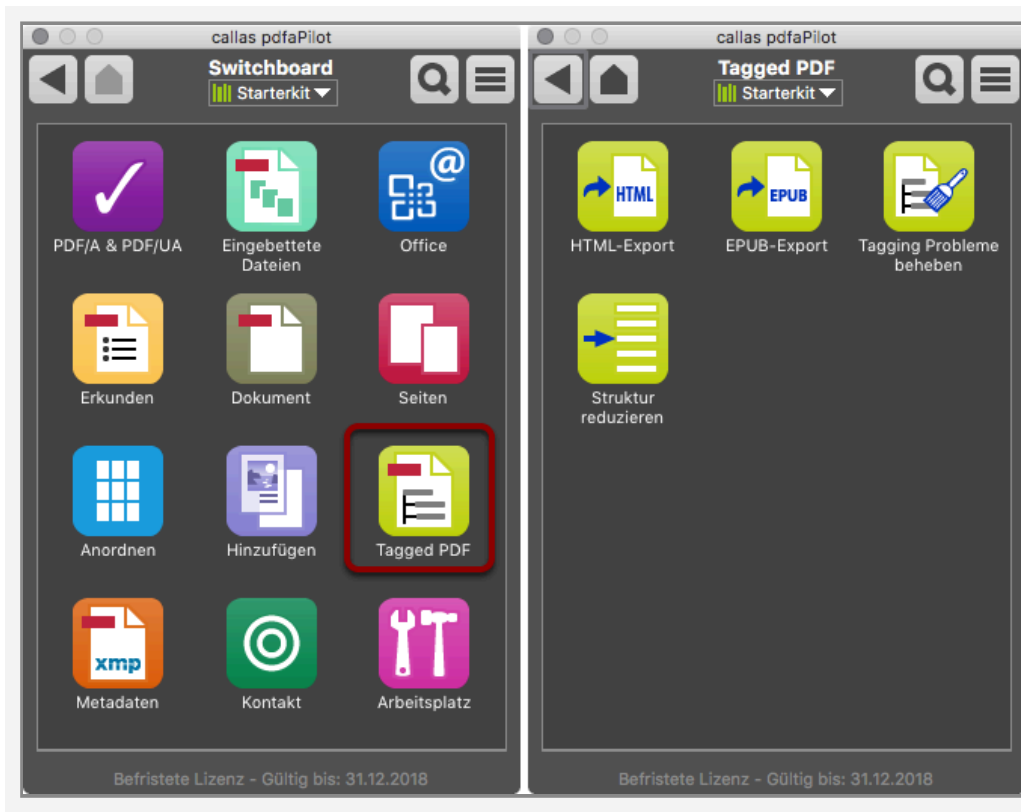
The Arrange group contains Actions used for imposition.

“Decorate” group



The Decorate group contains Actions used to add additional content to PDF files.

“Tagged PDF” group



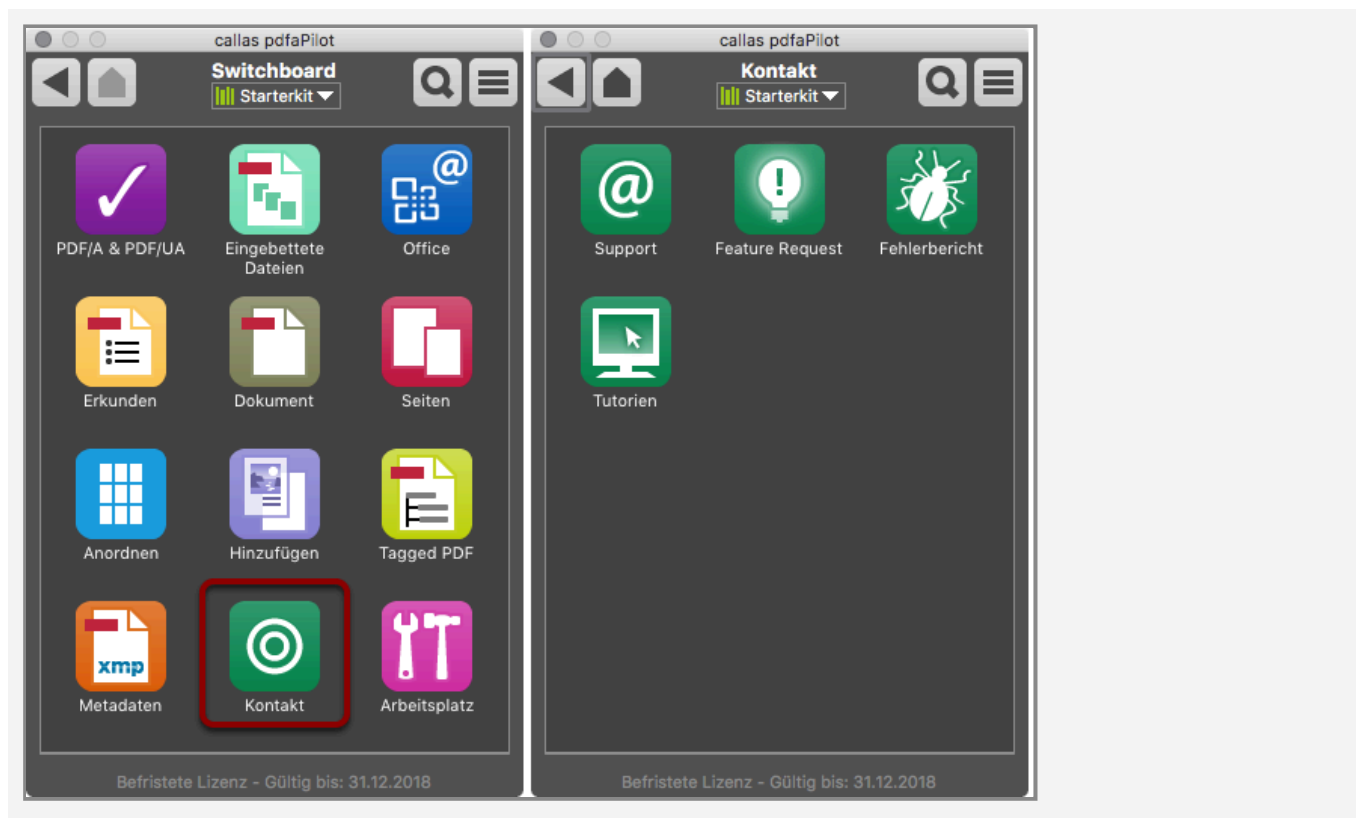
The **Tagged PDF** group provides Actions for exporting other file formats from structured PDFs or for optimizing structured PDFs.

“Metadata” group



The **Metadata** group includes Actions used to work with XMP metadata.

“Get in touch” group



The Get in touch group provides Actions used to contact callas support, report bugs or suggest improvements. It also provides access to various tutorials.

“Workspace” group

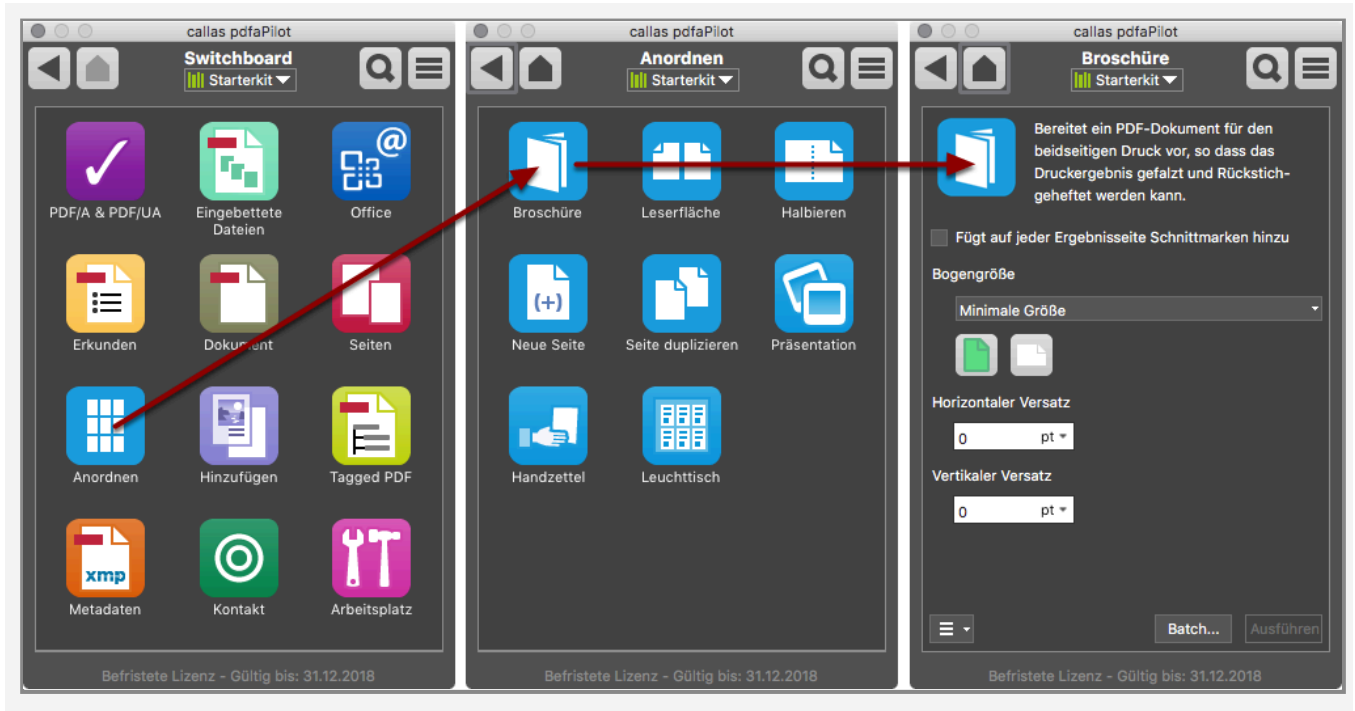


The **Workspace** group lets you collect your most frequently used Actions together in one place.

To add a Switchboard Action to the Workspace, click on the menu located in the bottom-left corner of the Action window.

An example of an action in use: Create Booklet

Click on an Action to open the relevant settings used to execute the Action.



Example: In the **Arrange** group, you will find the **Booklet** Action, which you can use to turn a multi-page file into a booklet.

You can specify the set size and the horizontal/vertical offset, as well as specifying whether cut marks should be added to each resulting page.

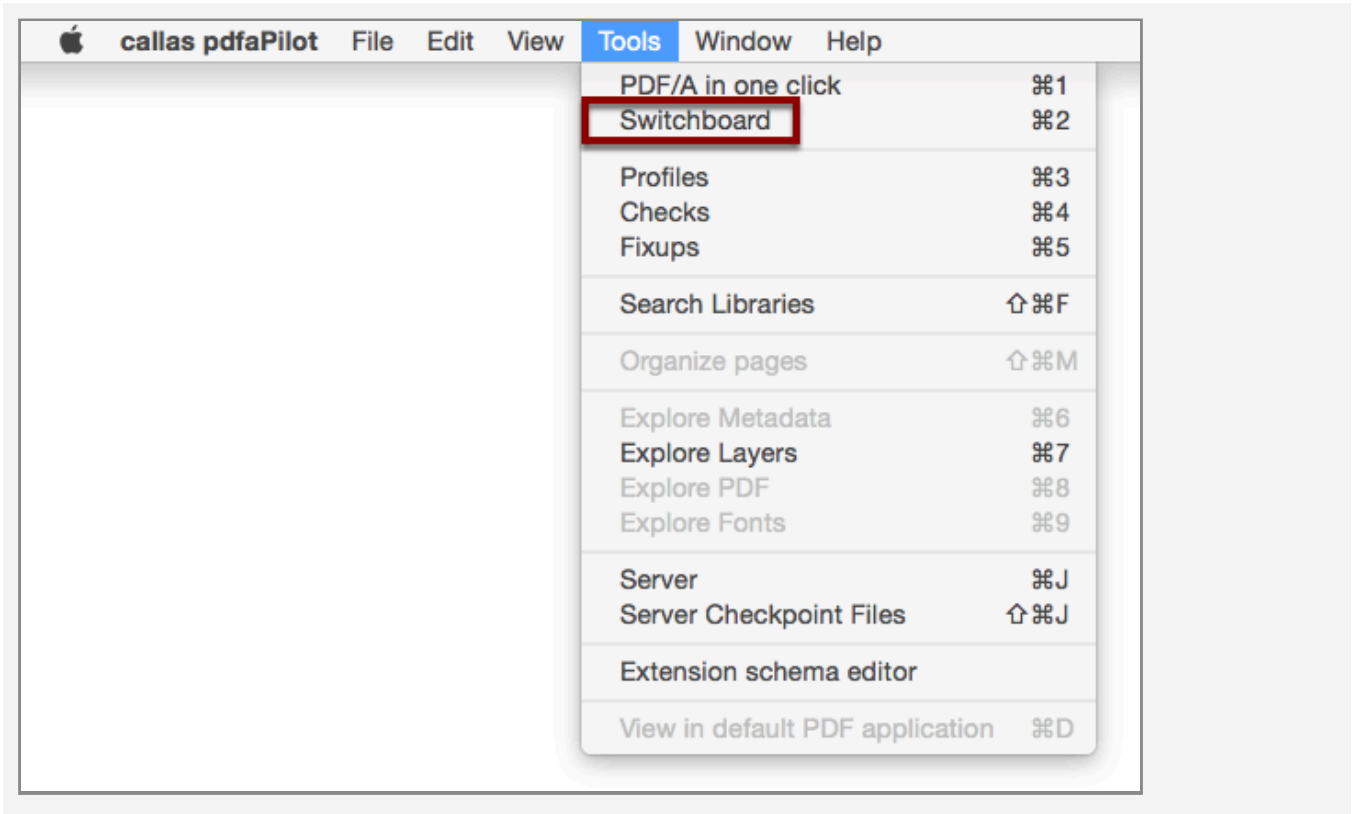
2.4 Switchboard: PDF/A & PDF/UA

The Actions in the PDF/A & PDF/UA group are used to convert and optimize the PDF/A (long-term archiving) and PDF/UA (universally accessible PDF) standards.

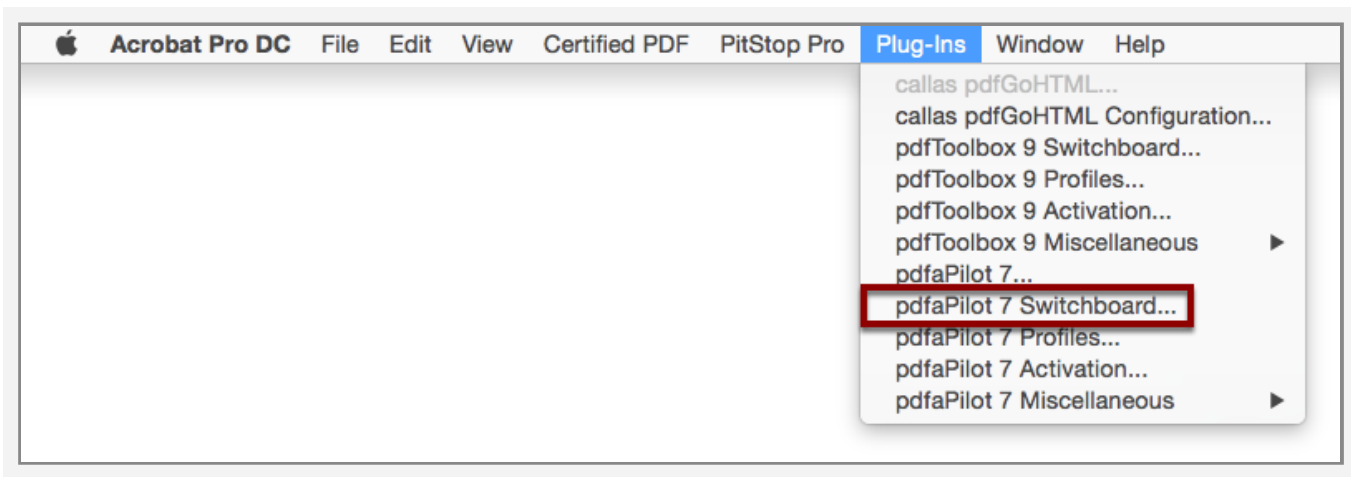


The “PDF/A” Action

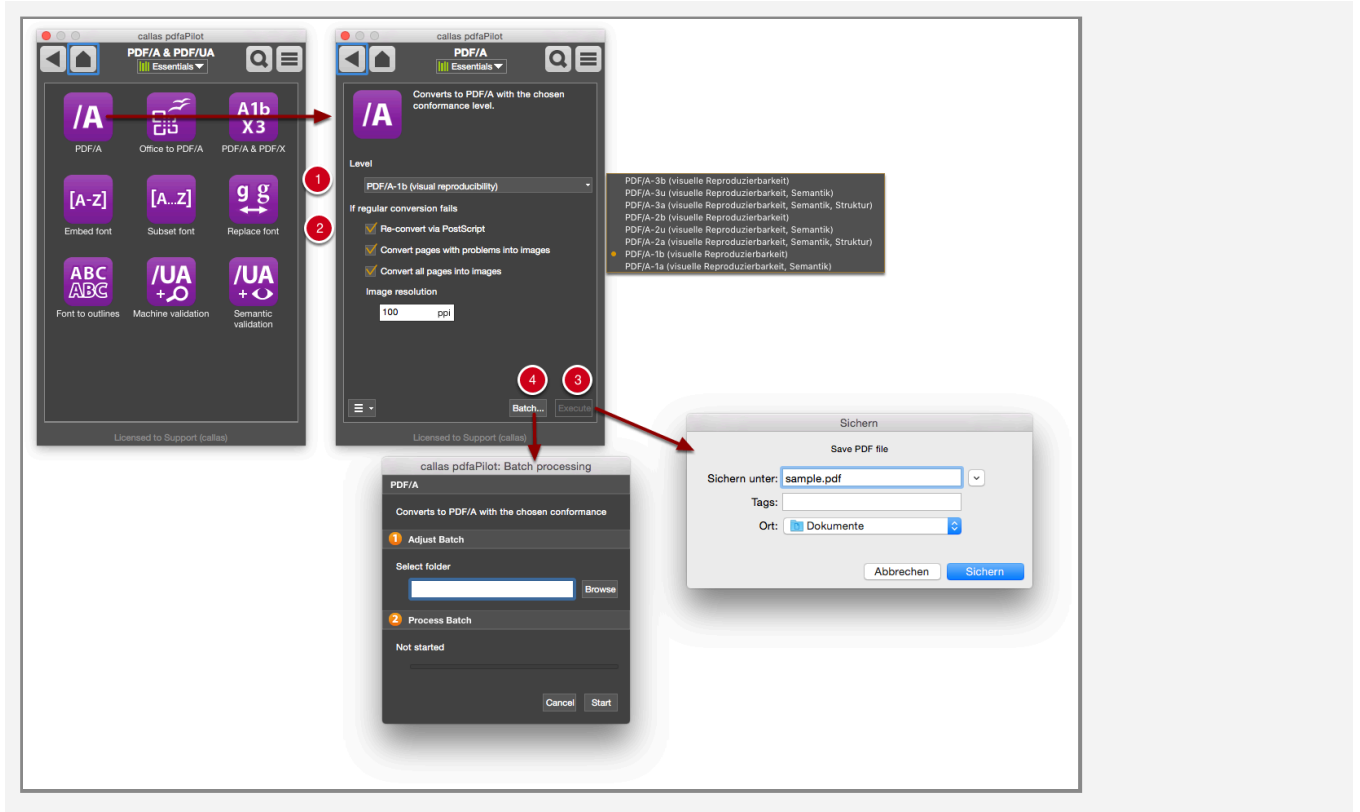
The PDF/A Action lets you convert PDF files to PDF/A in line with your desired compliance level.



In pdfaPilot standalone, open the switchboard via **Tools > Switchboard** in the menu or use the keyboard shortcut **Cmd+2**.



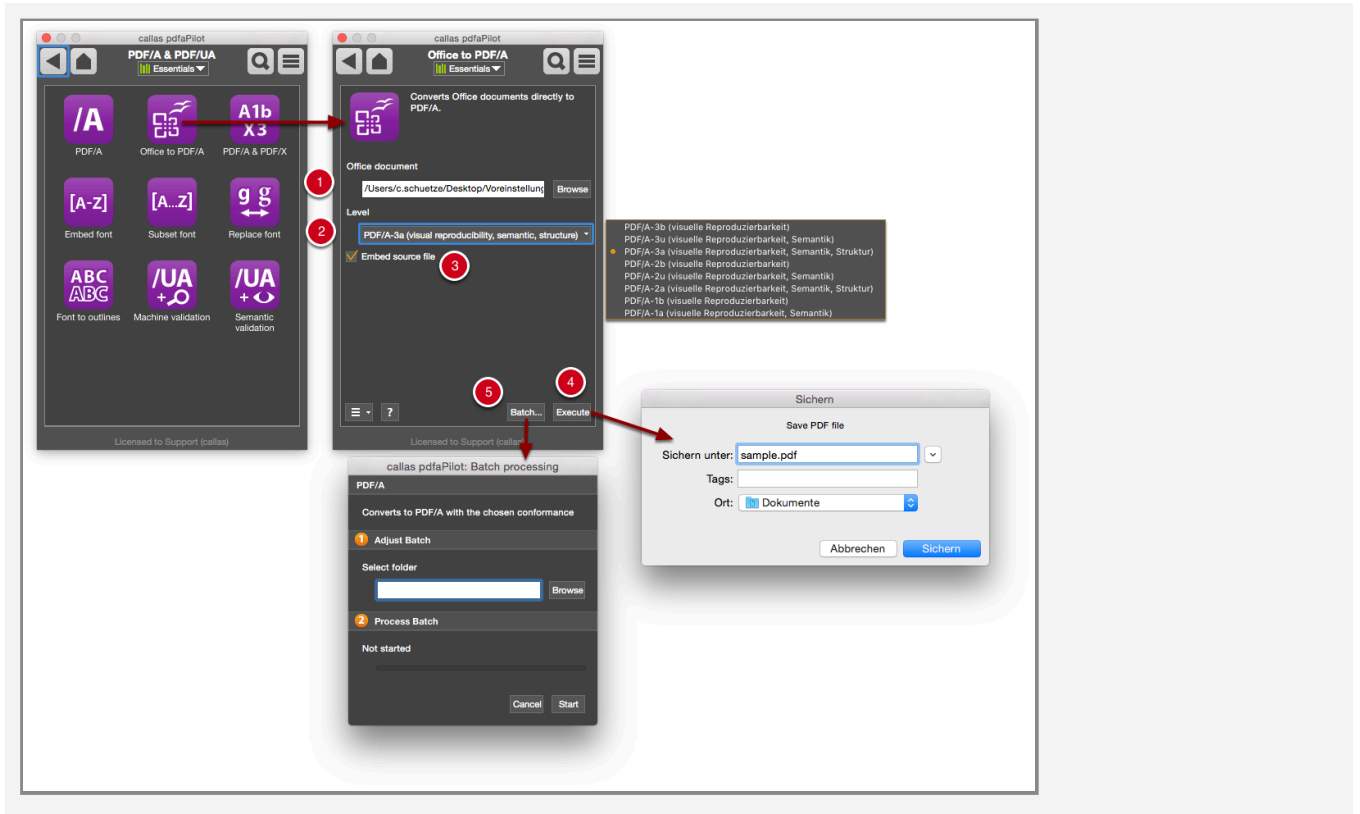
If you are using pdfaPilot as an Acrobat plug-in, you can open the switchboard via **Plug-Ins > pdfaPilot <version number> Switchboard**.



1. Under Level, you can specify the compliance level of the PDF/A file to be created. The effects of each setting are shown in brackets after the name.
2. The If regular conversion fails section offers a number of alternatives for processing (Re-convert via PostScript; Convert pages with problems into images; Convert all pages into images.)
 - If pages are to be converted into images, the desired Image resolution can also be set in ppi.
 - When converting pages into images, the program automatically generates an invisible text copy (where possible) to ensure that the PDF remains searchable.
3. Click Execute to convert the current PDF document using the specified settings.
4. Click Batch to process batches of PDF files within a single folder using the specified settings.

The “Office to PDF/A” Action

pdfaPilot also allows you to convert Office documents to PDF/A.

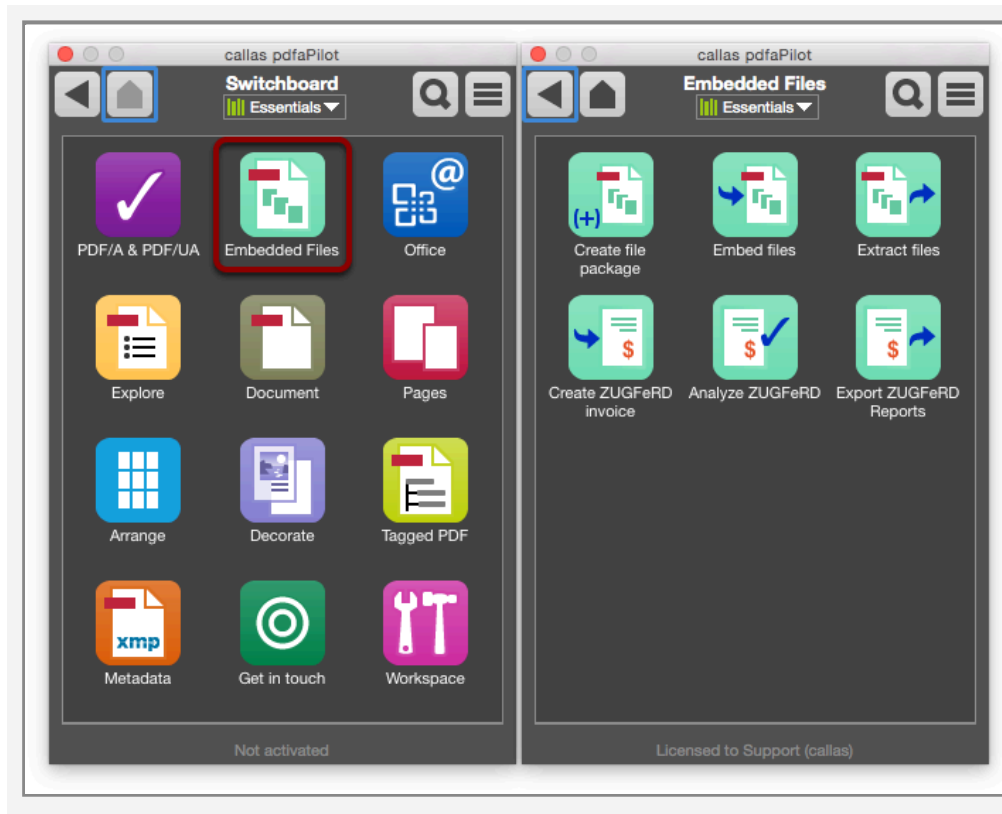


1. Under **Office document**, select the desired file by clicking **Browse** and using Finder or Explorer to locate the file.
2. Under **Level**, you can specify the compliance level of the PDF/A file to be created. The effects of each setting are shown in brackets after the name.
3. The **Embed source file** option is available when converting to PDF/A-3. This means that the original Office document will be embedded into the PDF/A-3 file.
4. Click **Execute** to convert the current PDF document using the specified settings. A **Save** dialog will ask you to enter the name and storage location for the new file.
5. Click **Batch** to process batches of Office files within a single selectable folder using the specified settings.

2.5 Switchboard: Embedded files - Create file package

Acting like containers, PDFs of version 1.7 and above can contain embedded files.

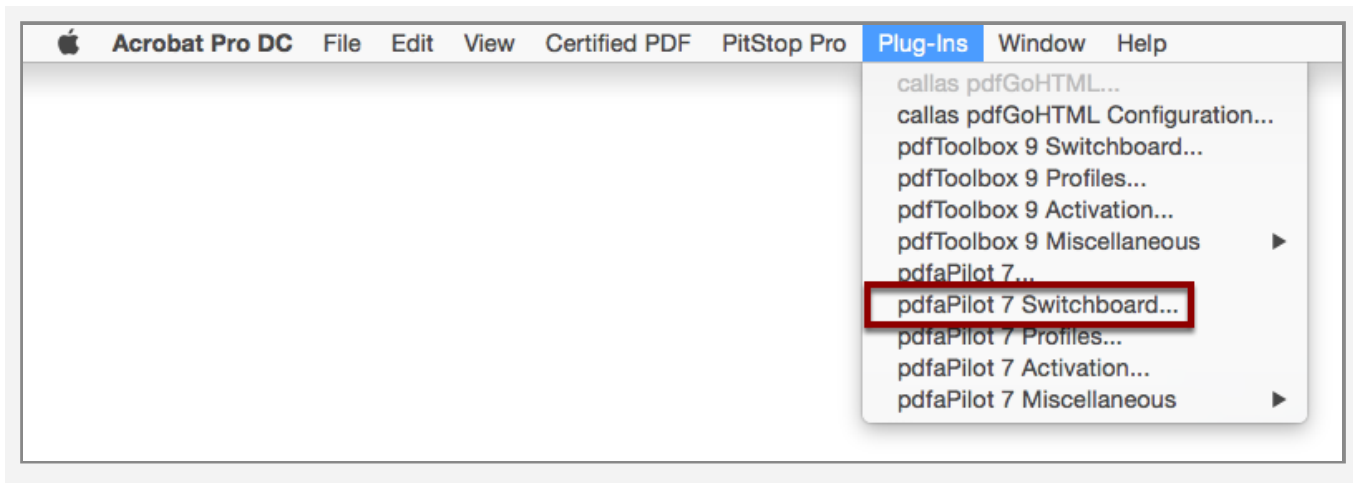
PDF/A-2 allows you to embed other PDF files; PDF/A-3 allows you to embed (source) files in other formats such as Office documents.



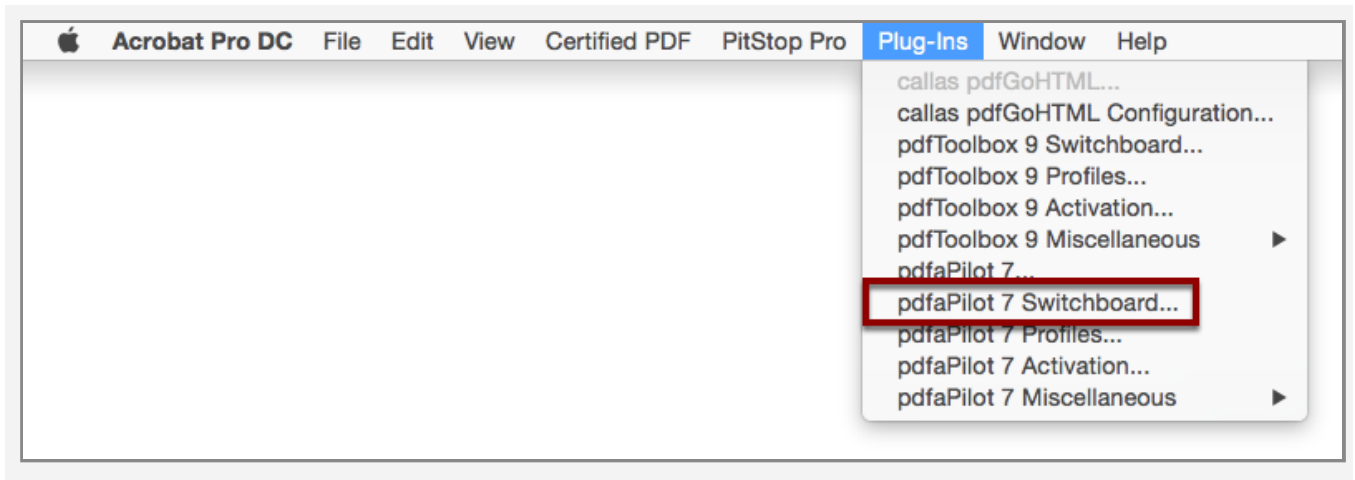
The Embedded files group lets you create file packages, embed files and extract files.

Embedded files are of interest for fields such as digital invoicing (ZUGFeRD).

Open the Switchboard



In pdfaPilot standalone, open the switchboard via **Tools > Switchboard** in the menu or use the keyboard shortcut **Cmd+2**.



If you are using pdfaPilot as an Acrobat plug-in, you can open the switchboard via **Plug-Ins > pdfaPilot <version number> Switchboard**.

The “Create file package” Action

The **Create file package** Action lets you create PDF and PDF/A files which in turn act as containers for additional files.



Click Create file package to open a settings dialog.

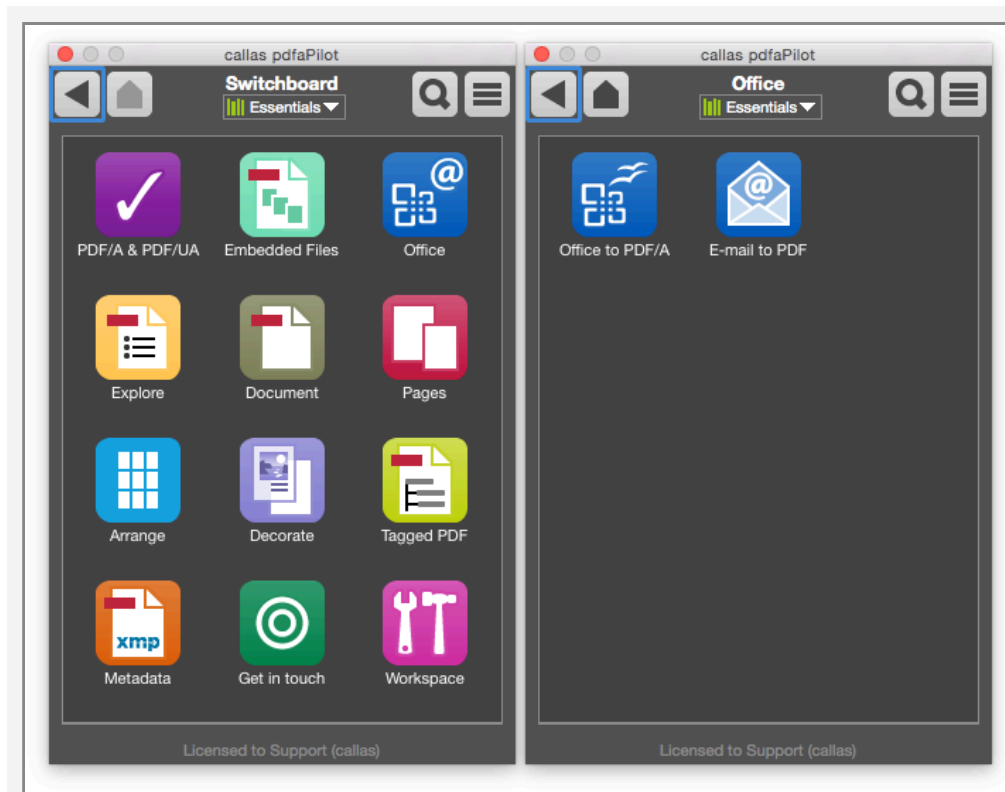


1. Two options are available under **Index page layout**: **Classic layout** and **User-defined layout**.
2. Under **PDF/A compatibility**, you can select the desired PDF or PDF/A version.
3. The **Relationship** can only be set if you have selected PDF/A-3-level compatibility above. The options here are: Source, Data, Alternative, Attachment or Unknown. This defines the embedded file's "relationship" with the primary document.
4. The output file can be set under **Select folder** by clicking **Browse**.
5. Click **Execute** to start the conversion process.

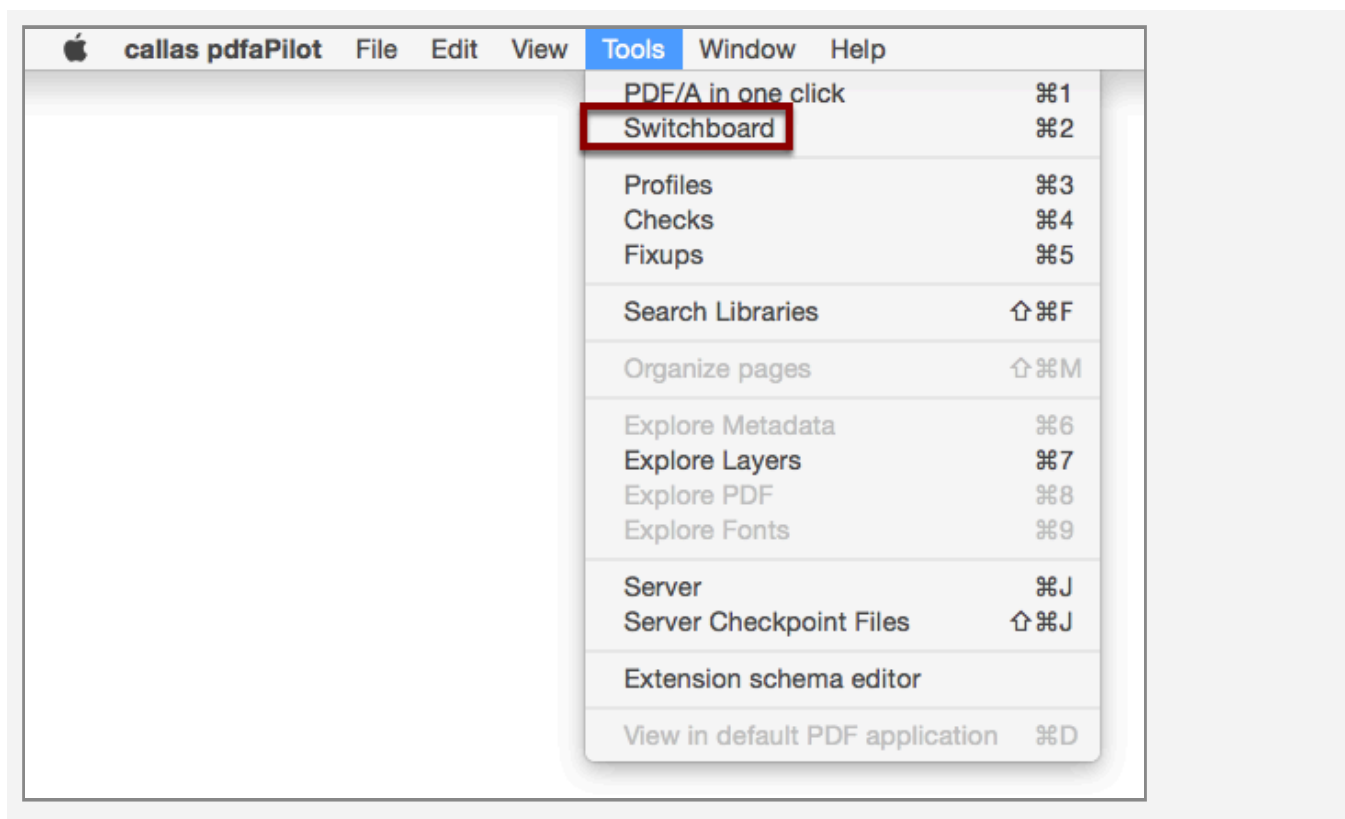
2.6 Switchboard: Office/E-mail to PDF

To allow you to save emails in PDF format, pdfaPilot offers a range of options for converting text and attachments into a “standard” PDF or the long-term archiving format PDF/A.

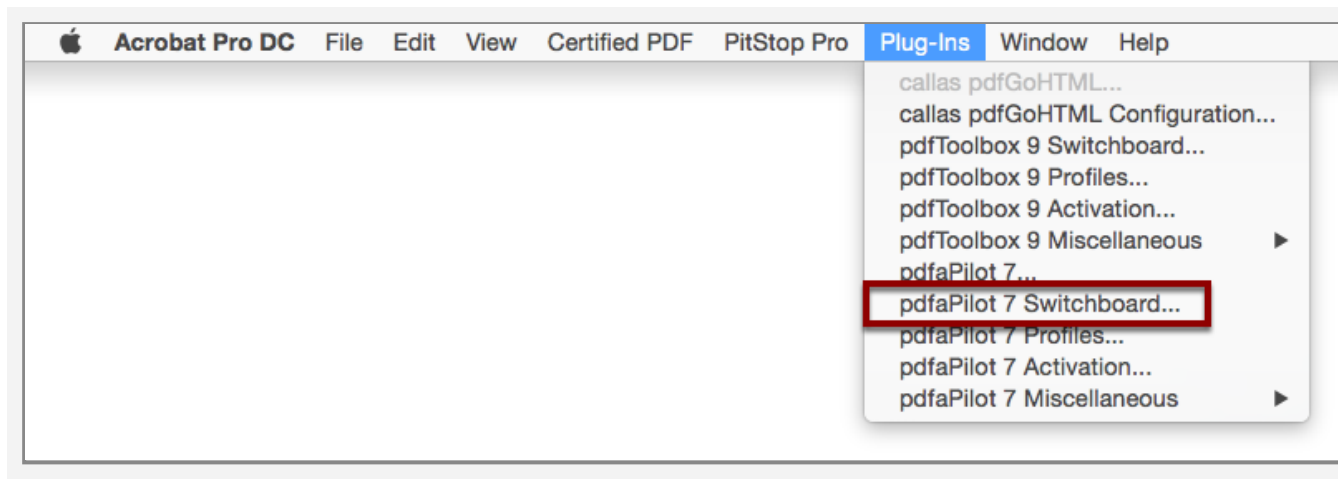
A wide range of settings are available for managing text and attachments.



Open the Switchboard



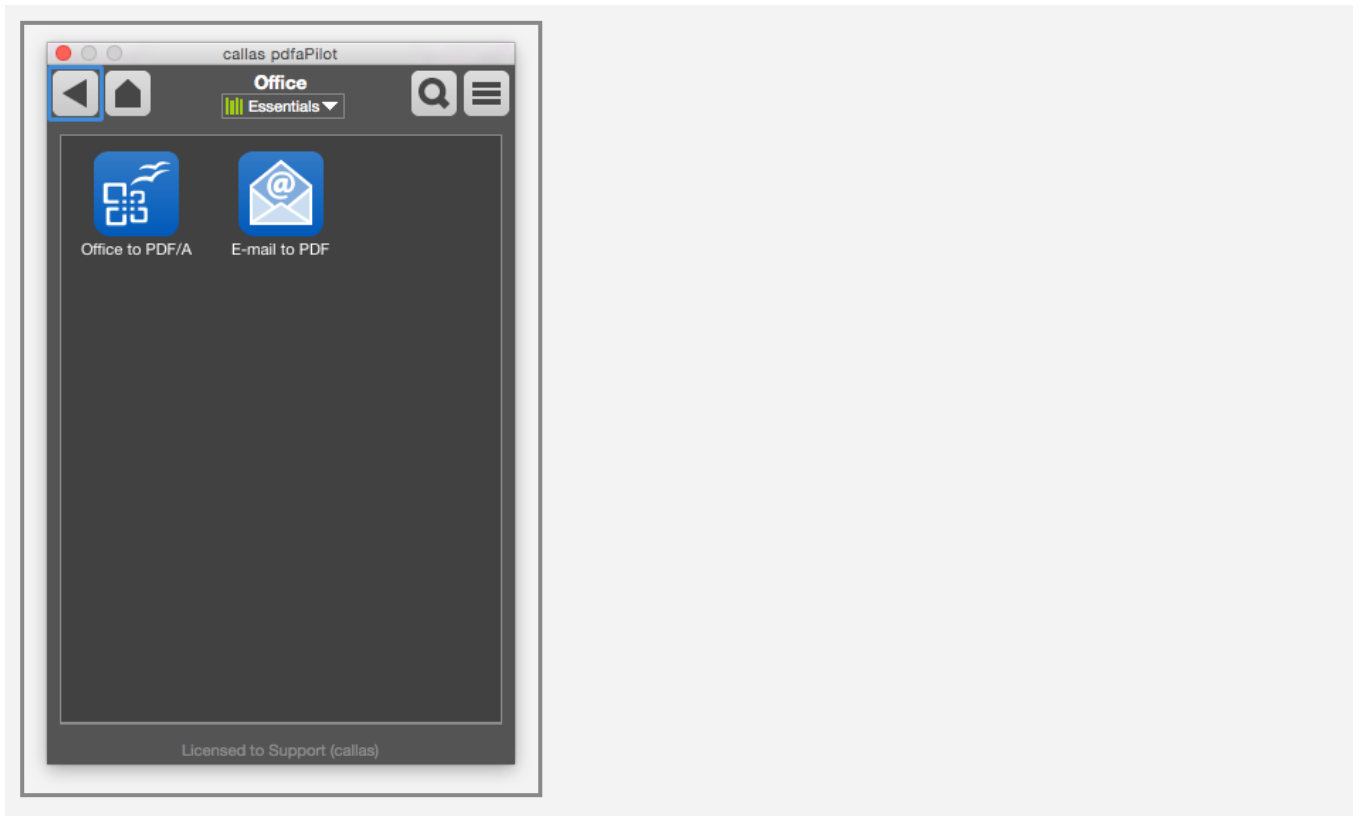
In pdfaPilot standalone, open the switchboard via **Tools > Switchboard** in the menu or use the keyboard shortcut **Cmd+2**.



If you are using pdfaPilot as an Acrobat plug-in, you can open the switchboard via **Plug-Ins > pdfaPilot <version number> Switchboard**.

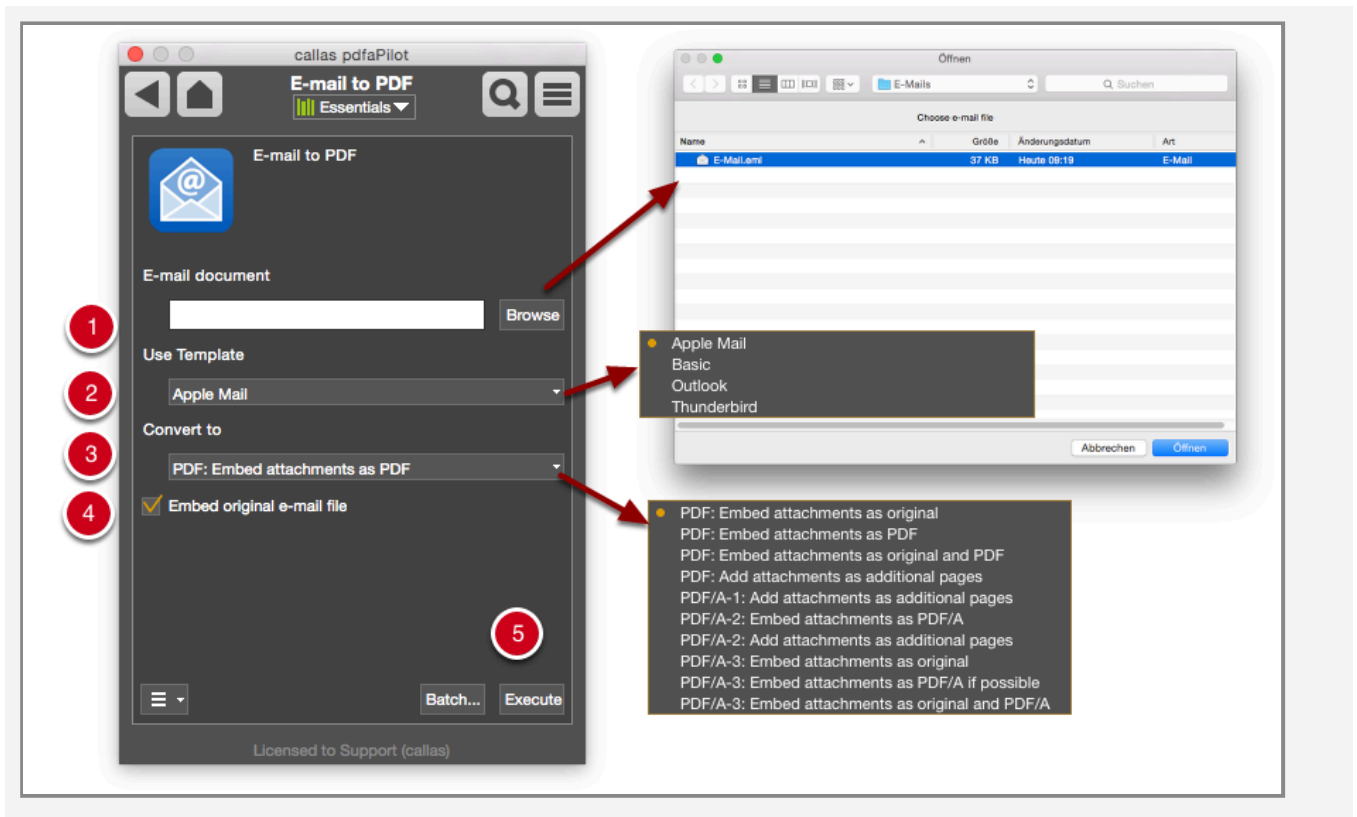
The “E-mail to PDF” Action

The “E-mail to PDF” action is part of the “Office” group.



Click on the E-mail to PDF symbol to open the settings window:

Settings



1. You can select an E-mail document to process by clicking “Browse” and selecting an email file from the window that opens.
2. The Use Template section lets you select the appropriate version for the email program used (Apple Mail, Basic, Outlook or Thunderbird)
3. pdfaPilot offers a range of options under Convert to. The basic choice is between converting to PDF or PDF/A. The list also offers a range of options for **handling attachments** (as original / PDF / PDF/A / additional PDF pages).
4. The check box lets you specify whether to **embed the original email file**.
5. Click Execute to start the conversion process.

Embed attachments

- PDF: Embed attachments as original
Generates a regular PDF document and embeds any at-

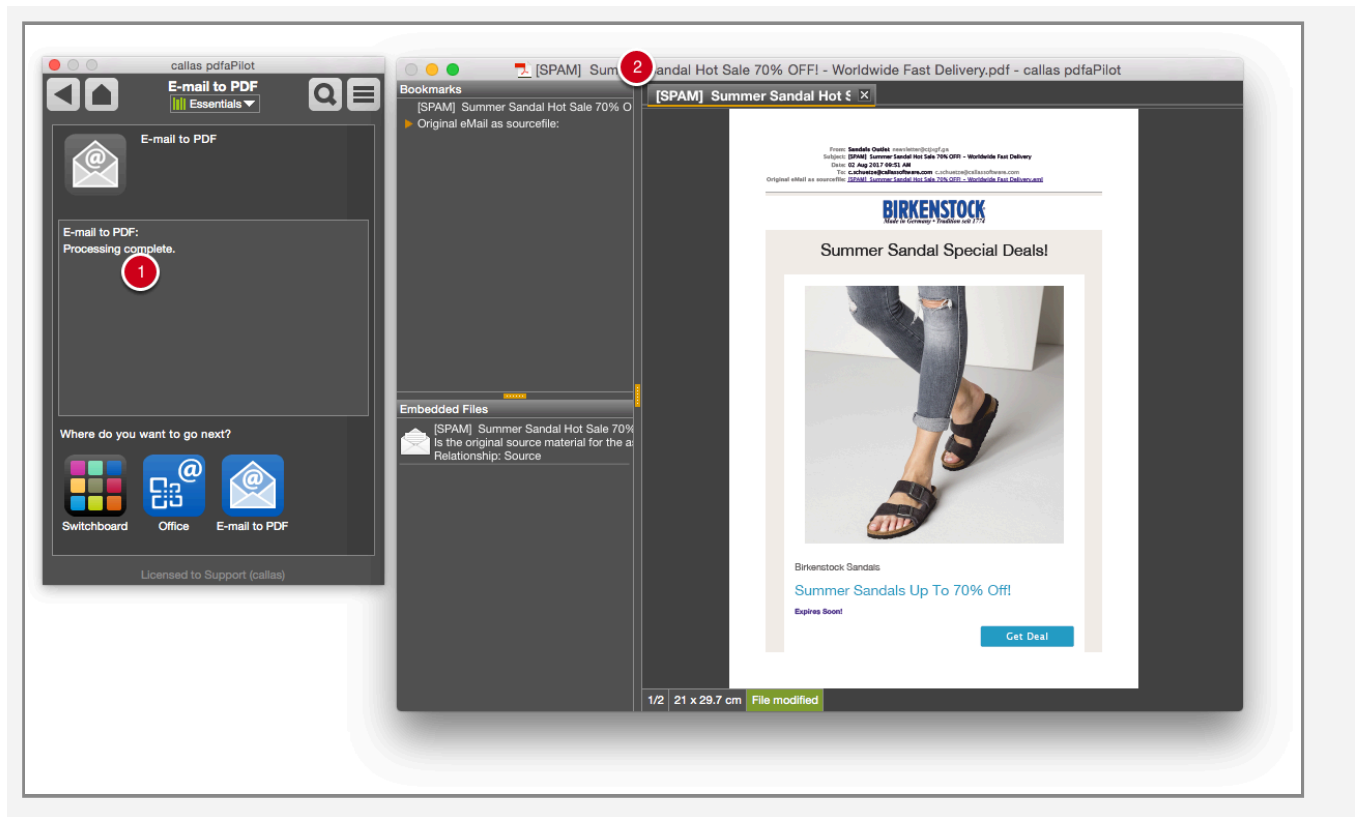
tachments from the email in the generated PDF document in their original format.

- **PDF: Embed attachments as PDF**
Generates a regular PDF document. If the email has attachments they are converted into PDF documents and then embedded in the generated PDF document.
- **PDF: Embed attachments as original and PDF**
Generates a regular PDF document. If the email has attachments they are converted into PDF documents. The result of the conversion to PDF and the original attachments are both embedded in the generated PDF document.
- **PDF: Add attachments as additional pages**
Generates a regular PDF document. If the email has attachments they are converted to PDF and appended at the end of the converted PDF document as additional pages.
- **PDF/A-1: Add attachments as additional pages**
Generates a PDF/A-1 document. If the email has attachments they are converted to PDF and appended at the end of the converted PDF/A-1 document as additional pages. This is the only option for PDF/A-1 because the PDF/A-1 standard does not allow embedded files (whatever they are).
- **PDF/A-2: Embed attachments as PDF/A**
Generates a PDF/A-2 document. If the email has attachments they are converted into PDF/A-2 documents and then embedded in the generated PDF/A-2 document.
- **PDF/A-2: Add attachments as additional pages**
Generates a PDF/A-2 document. If the email has attachments they are converted to PDF and appended at the end of the converted PDF/A-2 document as additional pages.
- **PDF/A-3: Embed attachments as original**
Generates a PDF/A-3 document and embeds any attachments from the email in the generated PDF/A-3 document in their original format.
- **PDF/A-3: Embed attachments as PDF/A if possible**
Generates a PDF/A-3 document. If the email has attachments they are converted into PDF/A-3 documents and then embedded in the generated PDF/A-3 document. If conversion to PDF/A is not possible for a particular attachment it is embedded in its original format instead.
- **PDF/A-3: Embed attachments as original and PDF/A**
Generates a PDF/A-3 document. If the email has attach-

ments they are converted into PDF/A documents. The result of the conversion to PDF/A and the original attachments are both embedded in the generated PDF/A-3 document.

- **PDF/A-4: Embed attachments as PDF/A**
Generates a PDF/A-4 document. If the email has attachments, they are converted into PDF/A-4 documents and then embedded in the generated PDF/A-2 document.
- **PDF/A-4: Embed attachments as additional pages**
Generates a PDF/A-4 document. If the email has attachments, they are converted to PDF and appended at the end of the converted PDF/A-4 document as additional pages.
- **PDF/A-4f: Embed attachments as original**
Generates a PDF/A-4f document and embeds any attachments from the email in the generated PDF/A-4f document in their original format.
- **PDF/A-4f: Embed attachments as PDF/A if possible**
Generates a PDF/A-4f document. If the email has attachments, they are converted into PDF/A-4 documents and then embedded in the generated PDF/A-4f document. If conversion to PDF/A is not possible for a particular attachment it is embedded in its original format instead.
- **PDF/A-4f: Embed attachments as original and PDF/A**
Generates a PDF/A-4f document. If the email has attachments, they are converted into PDF/A documents. The result of the conversion to PDF/A and the original attachments are both embedded in the generated PDF/A-4f document.

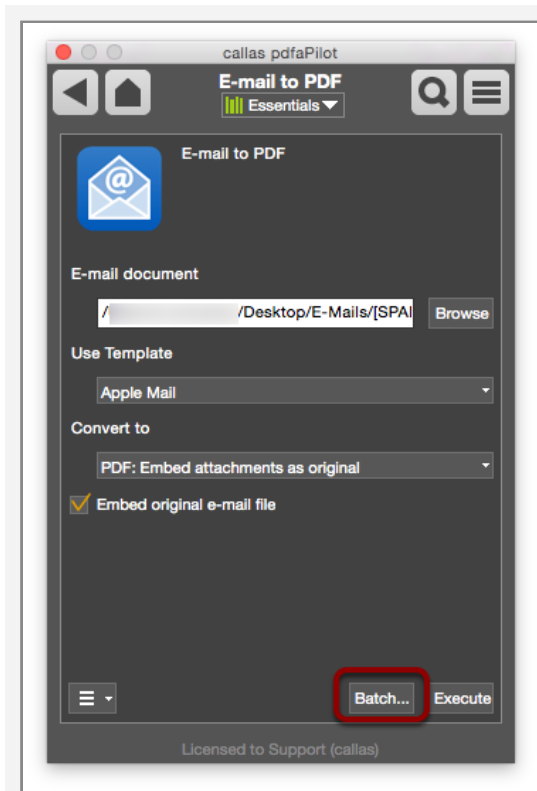
Result



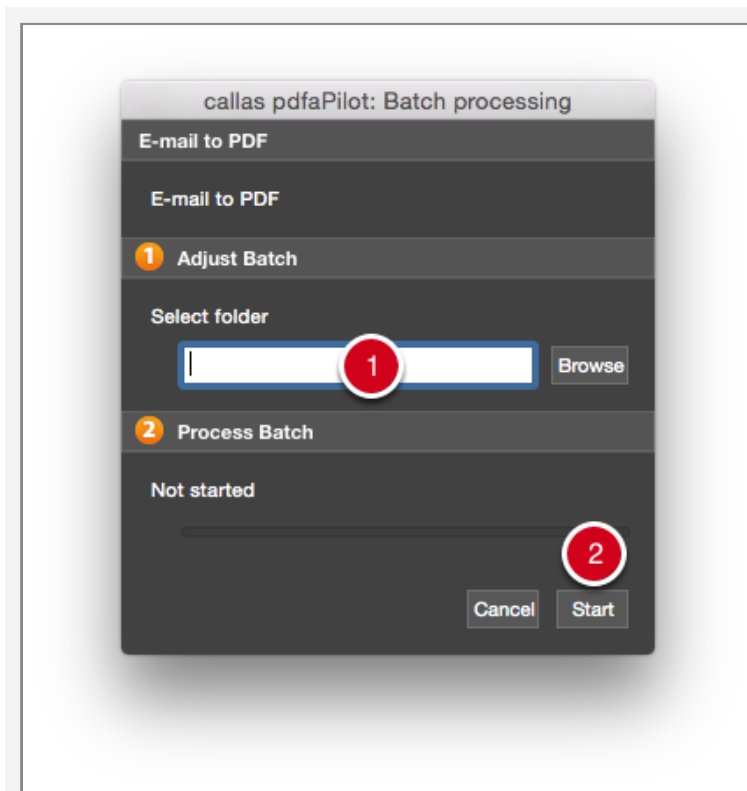
1. pdfaPilot will then report that processing is complete.
2. The resulting PDF or PDF/A file will then be shown in the file window.

The “E-mail to PDF” Action for batch processing

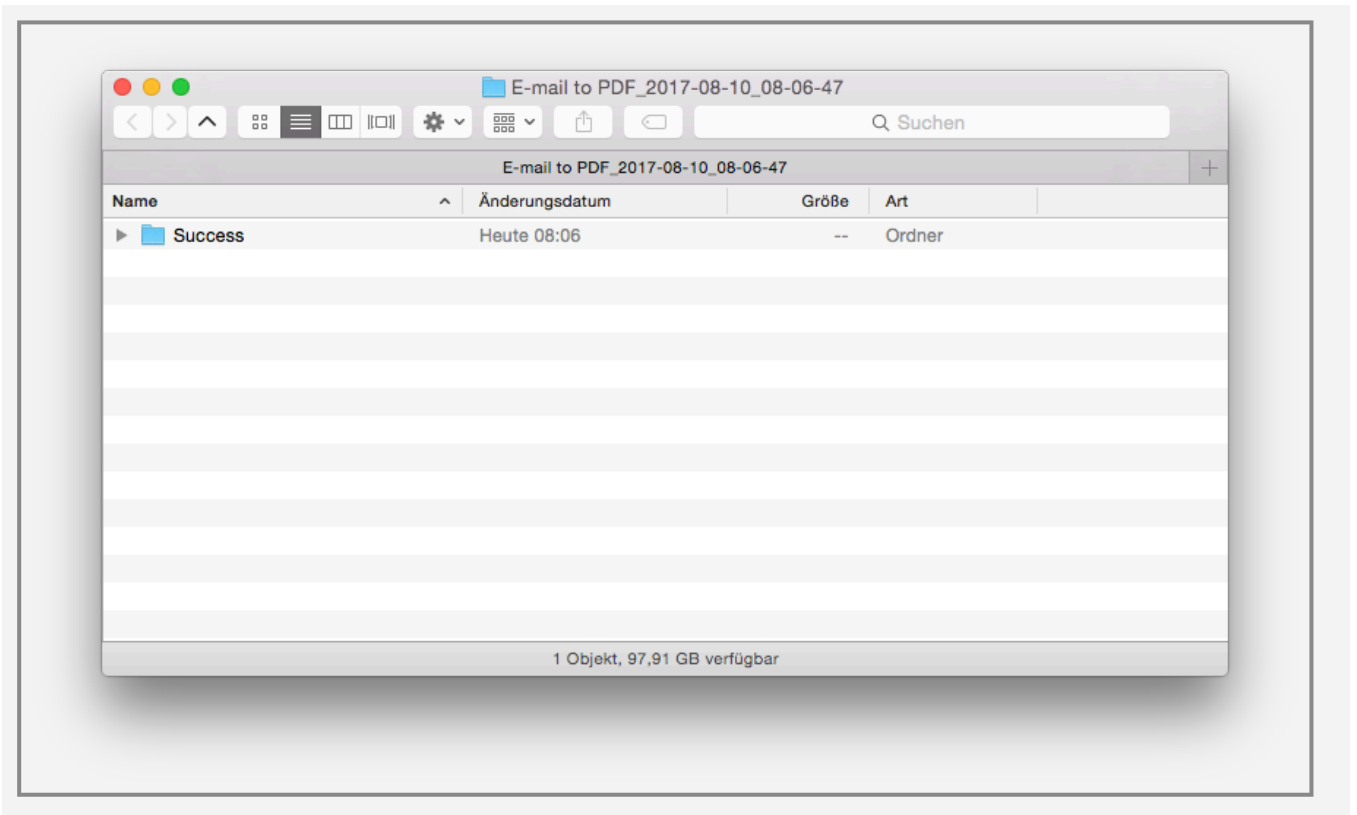
The E-mail to PDF Action can also be used for batch processing.



To do so, click on the **Batch** button after applying the settings in the Action window.



1. The batch processing dialog allows you to select the folder containing the files to be processed.
2. Once this is done, click **Start**.

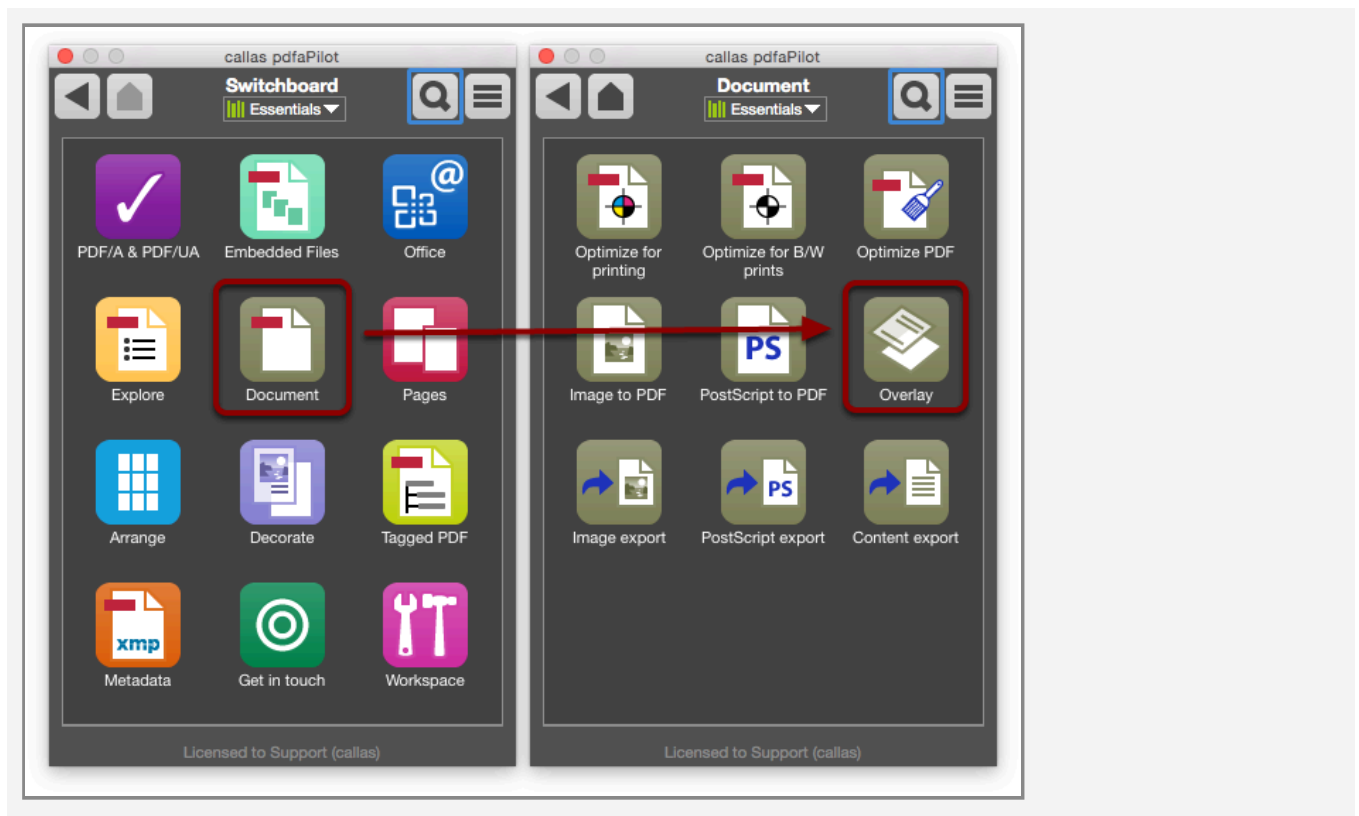


The resulting files will be stored in a folder named **Successful**, located in a *timestamped* sub-folder in the original files' directory.

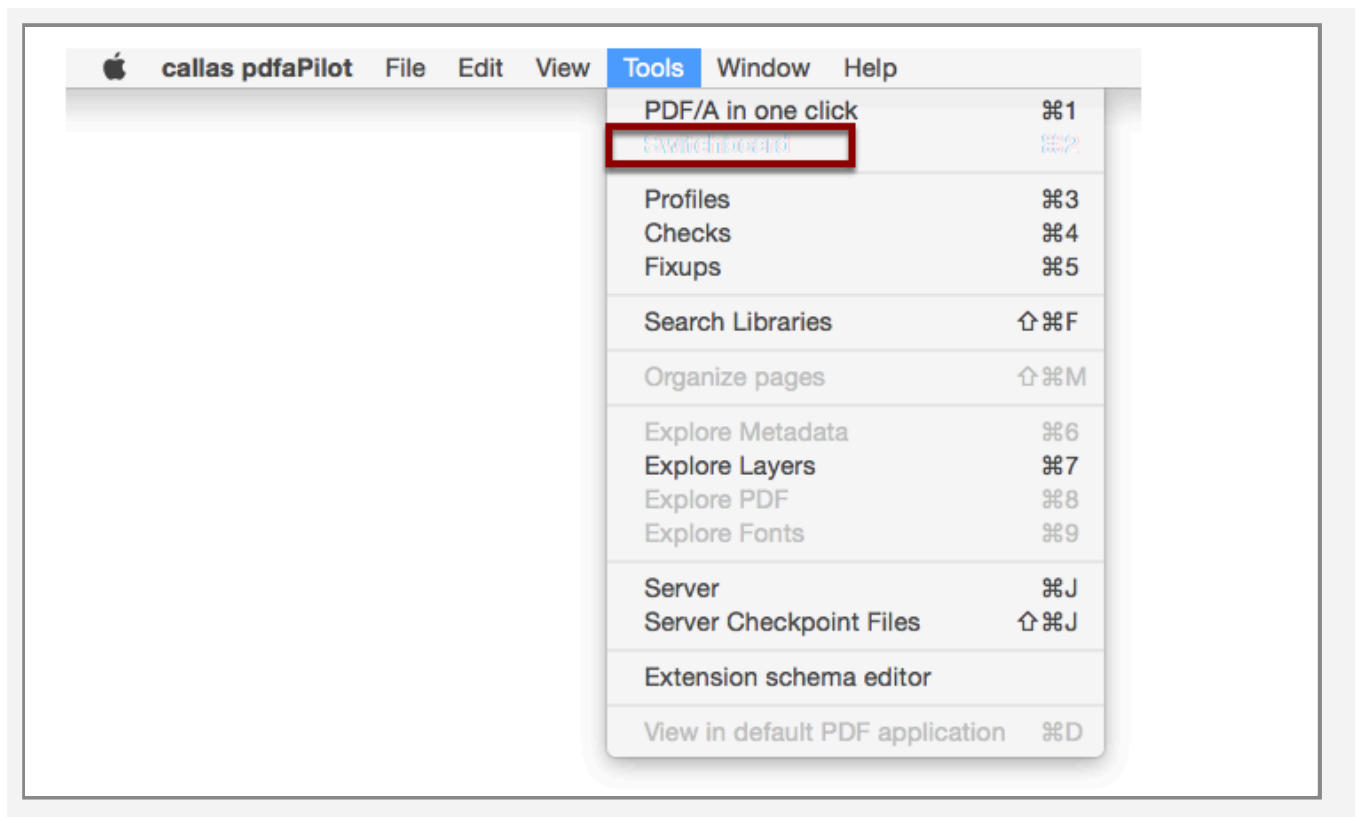
2.7 Switchboard: Document – Overlay

The **Overlay** action in the **Document** group lets you place content on a PDF document page similarly to a stamp.

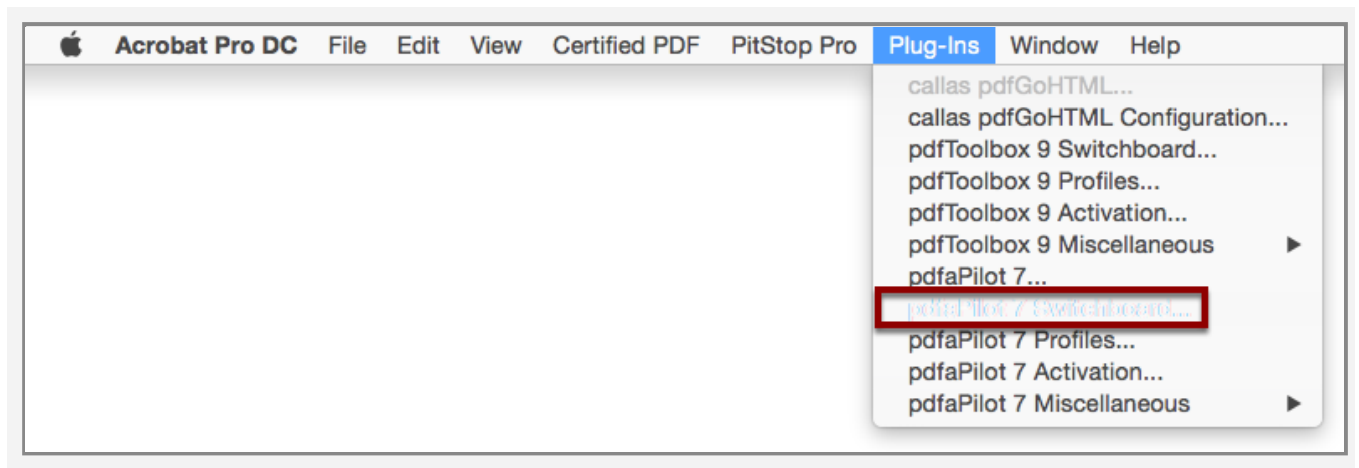
pdfaPilot provides a number of standard overlays, but the user can also apply their own overlays (as long as they are in PDF format).



Open the Switchboard



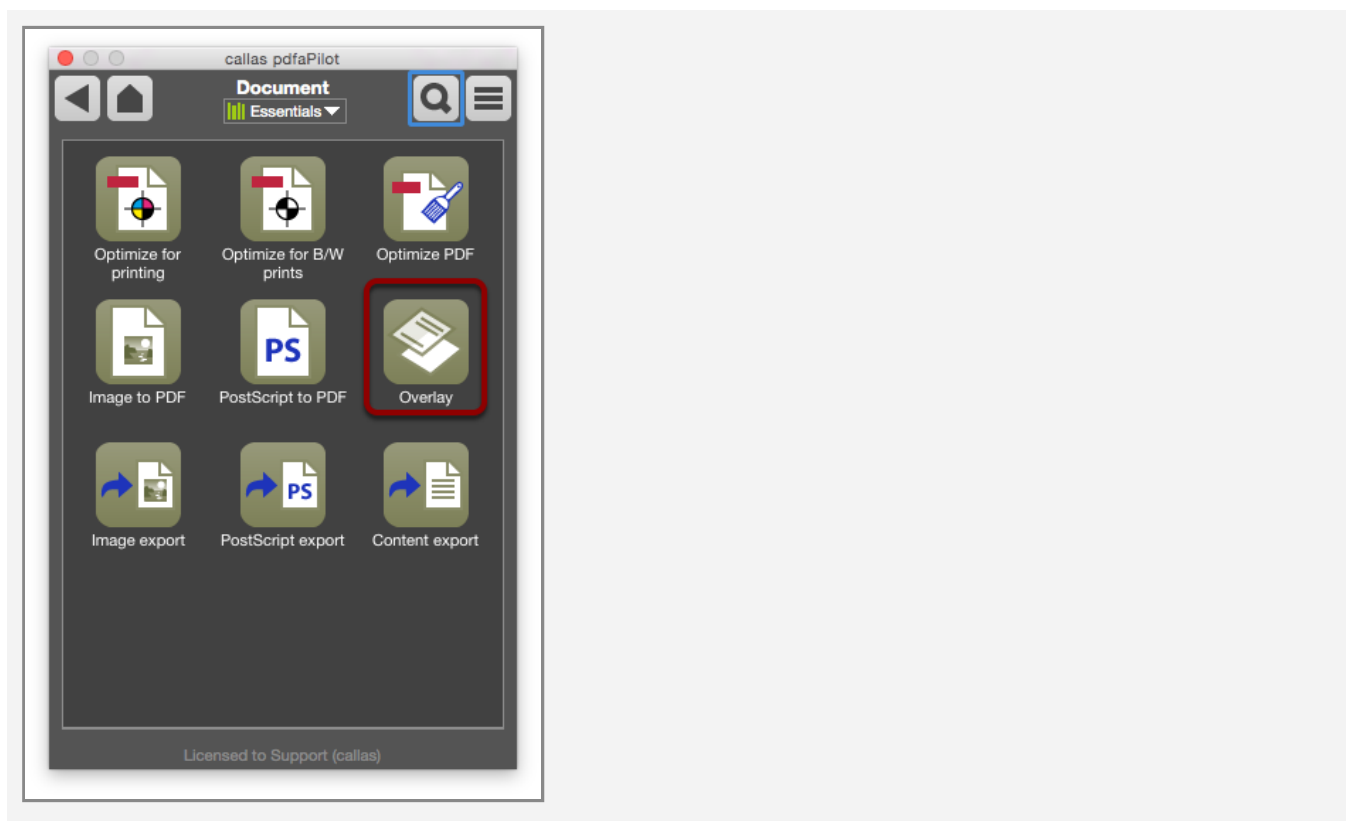
In pdfaPilot standalone, open the switchboard via **Tools > Switchboard** in the menu or use the keyboard shortcut **Cmd+2**.



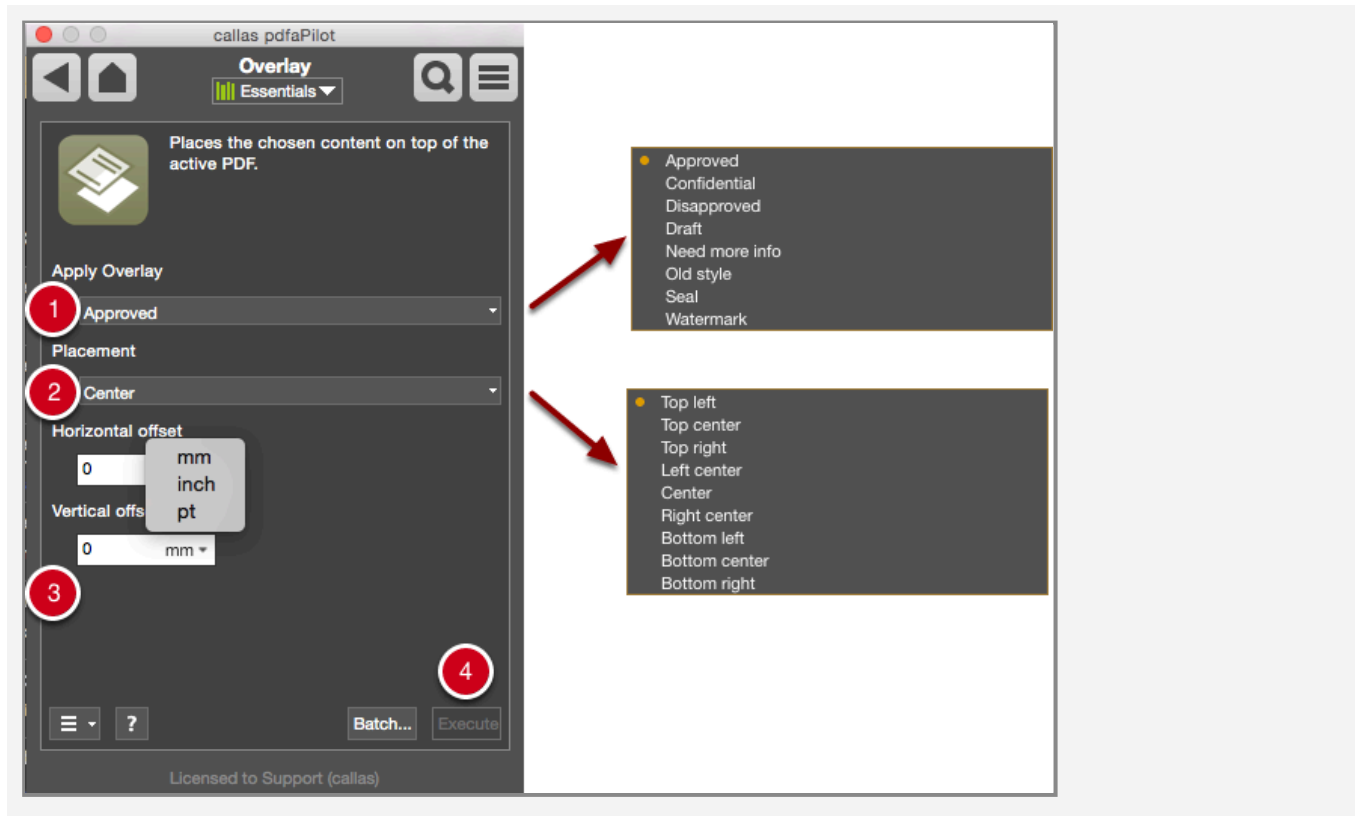
If you are using pdfaPilot as an Acrobat plug-in, you can open the switchboard via **Plug-Ins > pdfaPilot <version number> Switchboard**.

The “Overlay” Action

The Overlay Action is a part of the Document group.

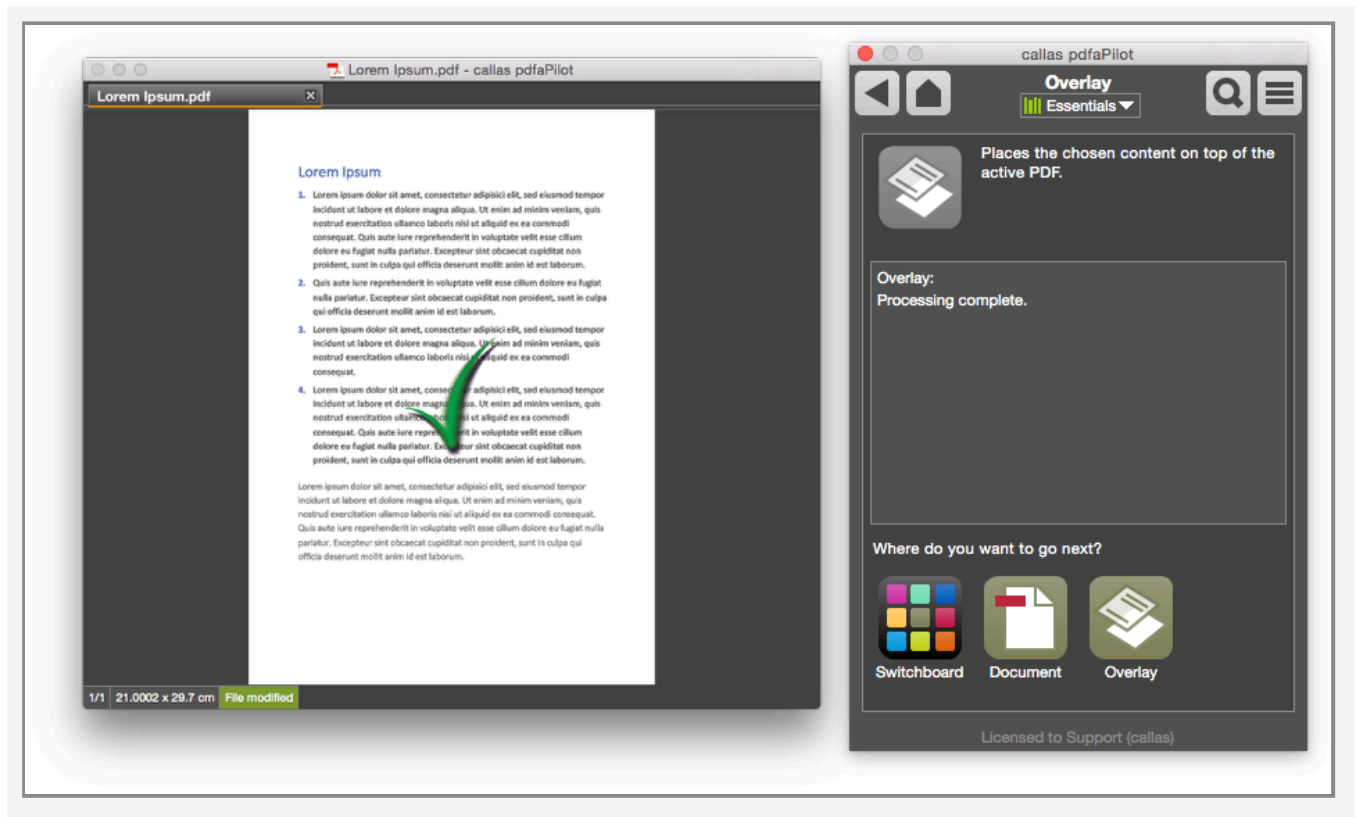


Click on the **Overlay** symbol to open the settings window:



1. The **Apply overlay** menu provides a range of templates to choose from, including **Declined**, **Accepted**, **Draft** and others.
2. The **Position** menu lets you specify where on the page to place the overlay (such as **Top left**, **Center** or **Bottom right**.)
3. To make fine adjustments to the positioning, use the **Horizontal offset** and **Vertical offset** settings. The units can be specified in **mm**, **inches** or **pt**.
4. Click **Execute** to start processing the current PDF document.

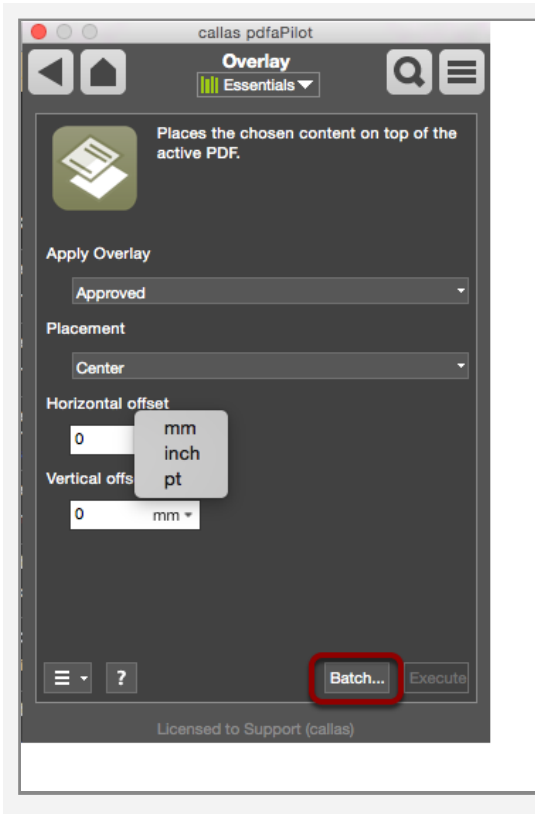
Result



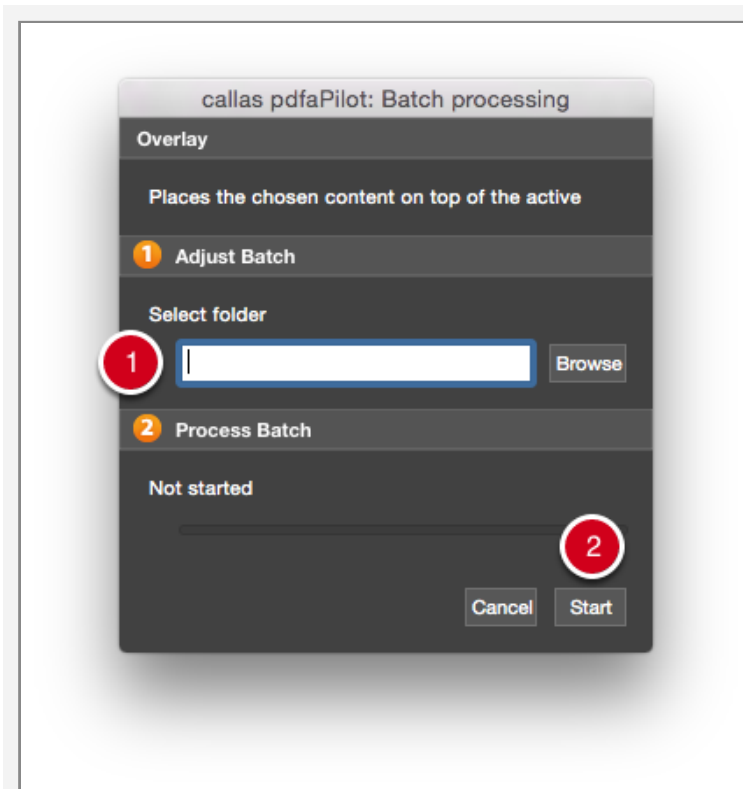
1. The overlay is placed on the PDF page.
2. pdfaPilot will report that processing is complete.

Running the “Overlay” Action in batch processing mode

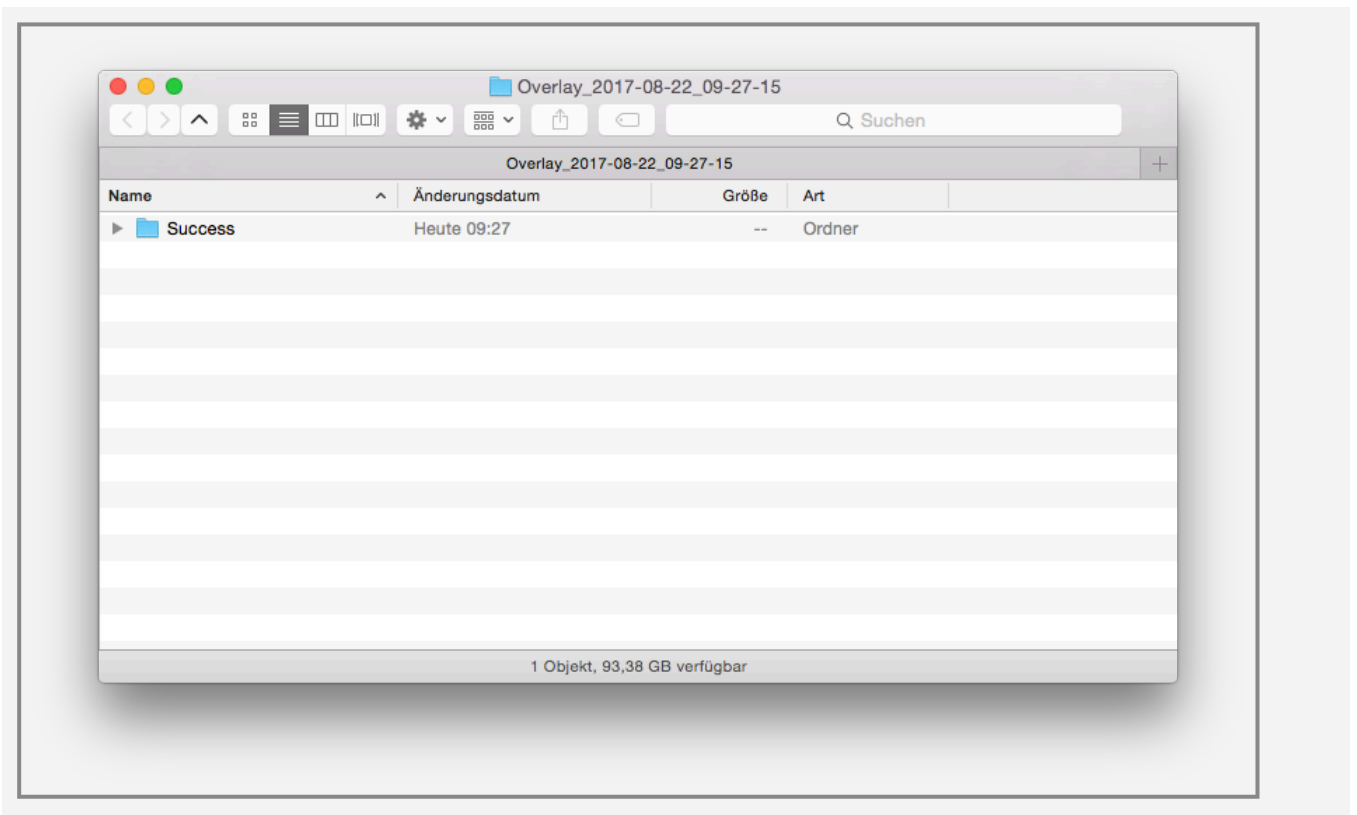
The Overlay Action can also be used for batch processing.



To do so, click on the **Batch** button after applying the settings in the Action window.

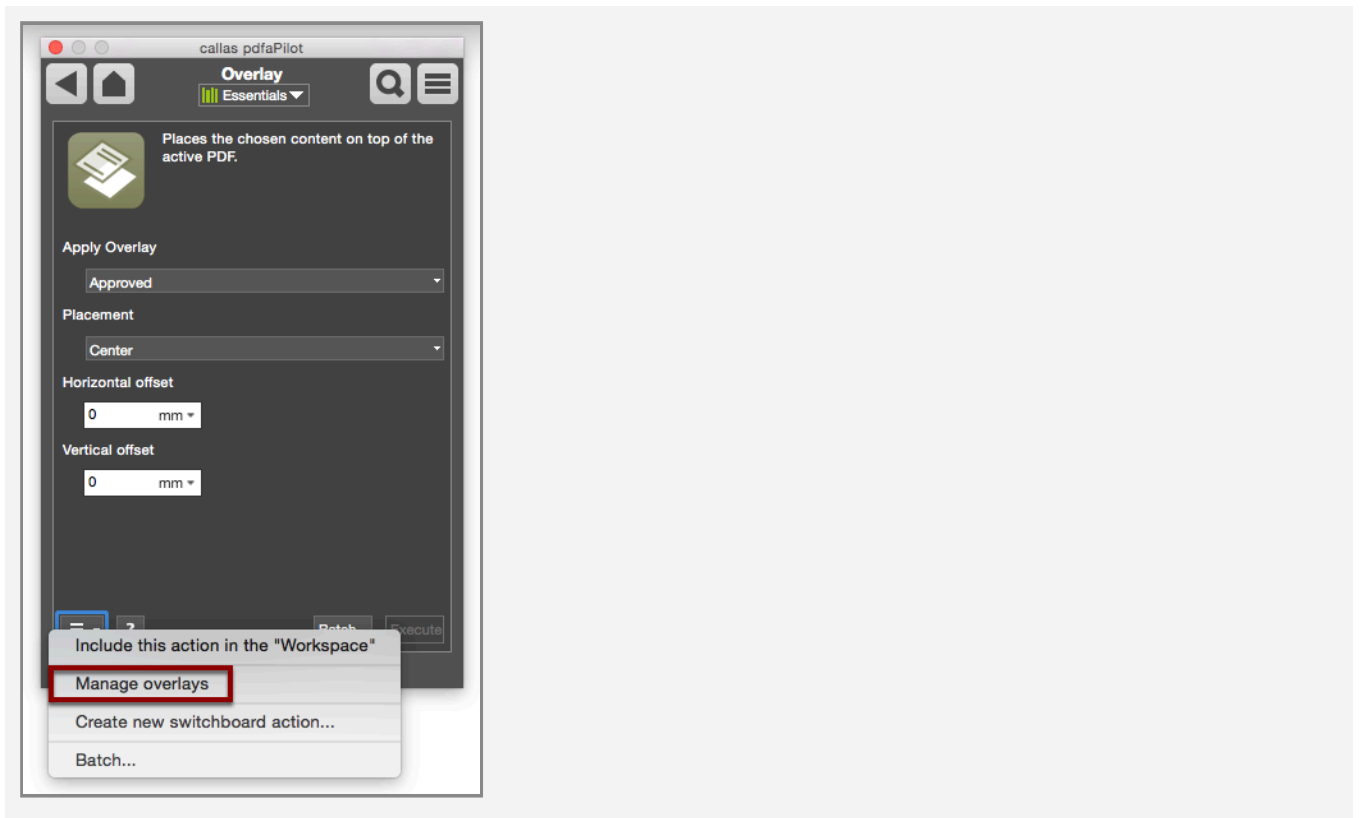


1. The batch processing dialog allows you to select the folder containing the files to be processed.
2. Once this is done, click **Start**.

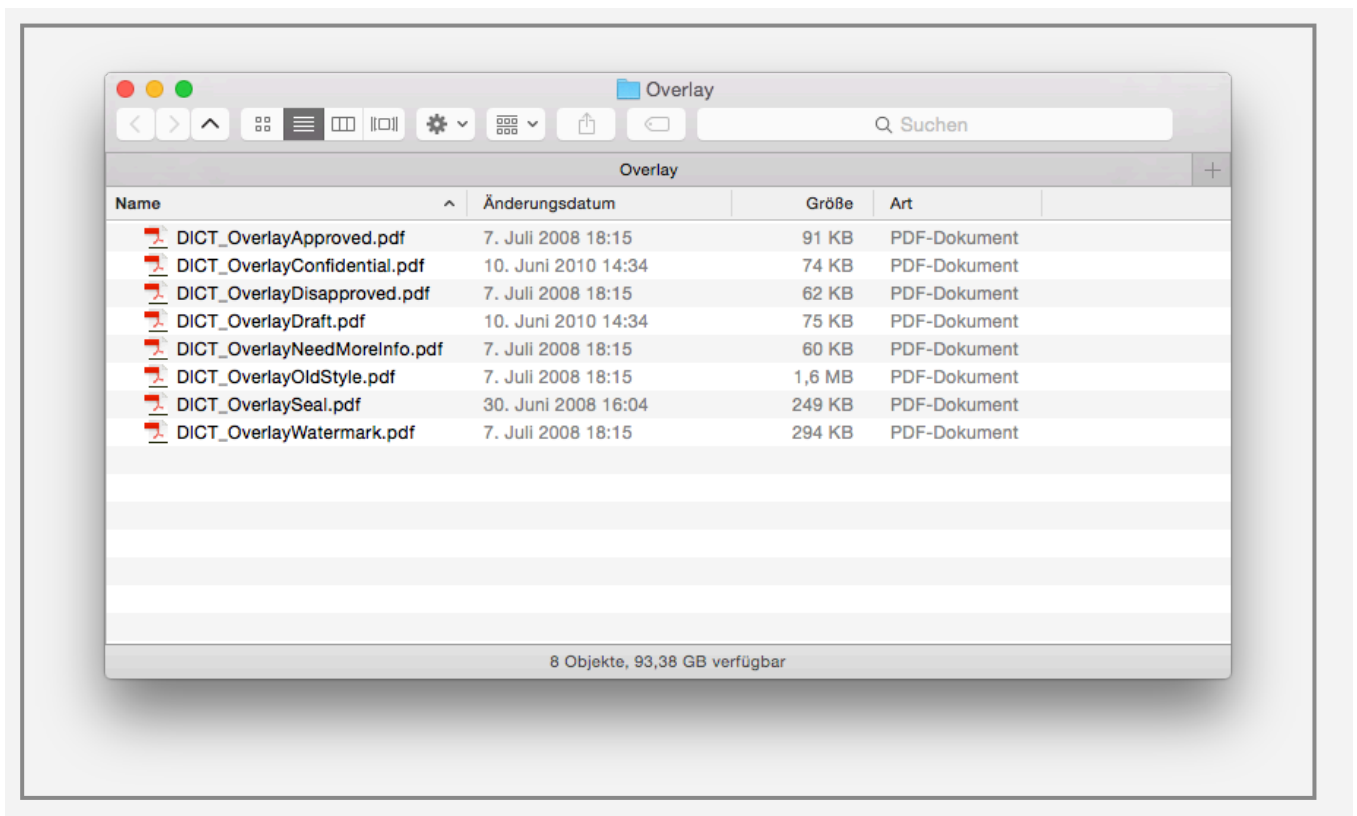


The resulting files will be stored in a folder named **Successful**, located in a *timestamped* sub-folder in the original files' directory.

Using your own overlays



The button at the bottom left lets you **Manage overlays**.



Here, you can add your own templates.

These must be in PDF format and have a transparent background.

The user-specific templates will then be shown in the **Apply overlay** list next time you launch the Switchboard Action.

2.8 Switchboard: Pages - Scale and enlarge

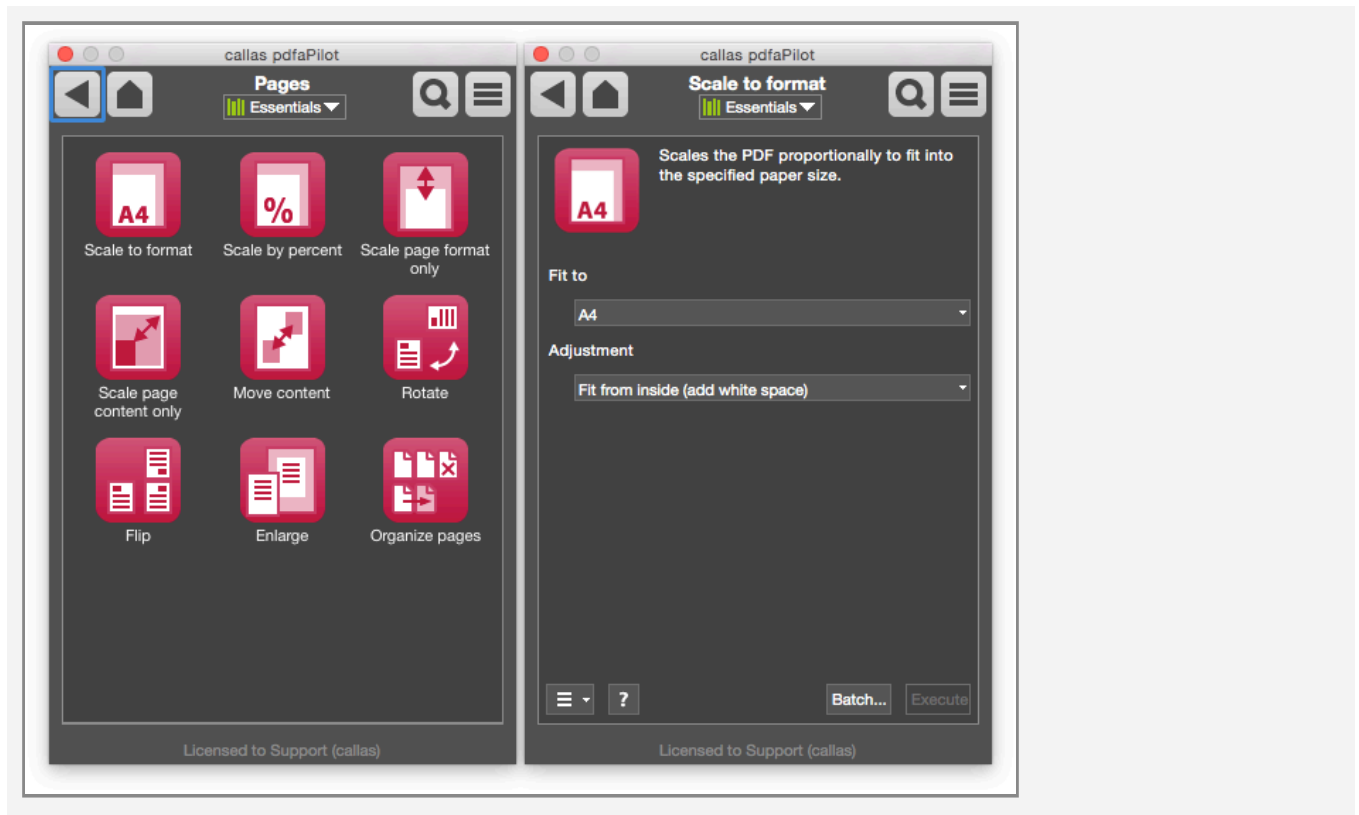
pdfaPilot's Pages Group offers a range of options for scaling page content and enlarging page regions.

This article will provide a brief overview of these Actions.



Numerous scaling Actions can be found under the Pages group.

“Scale to format” Action

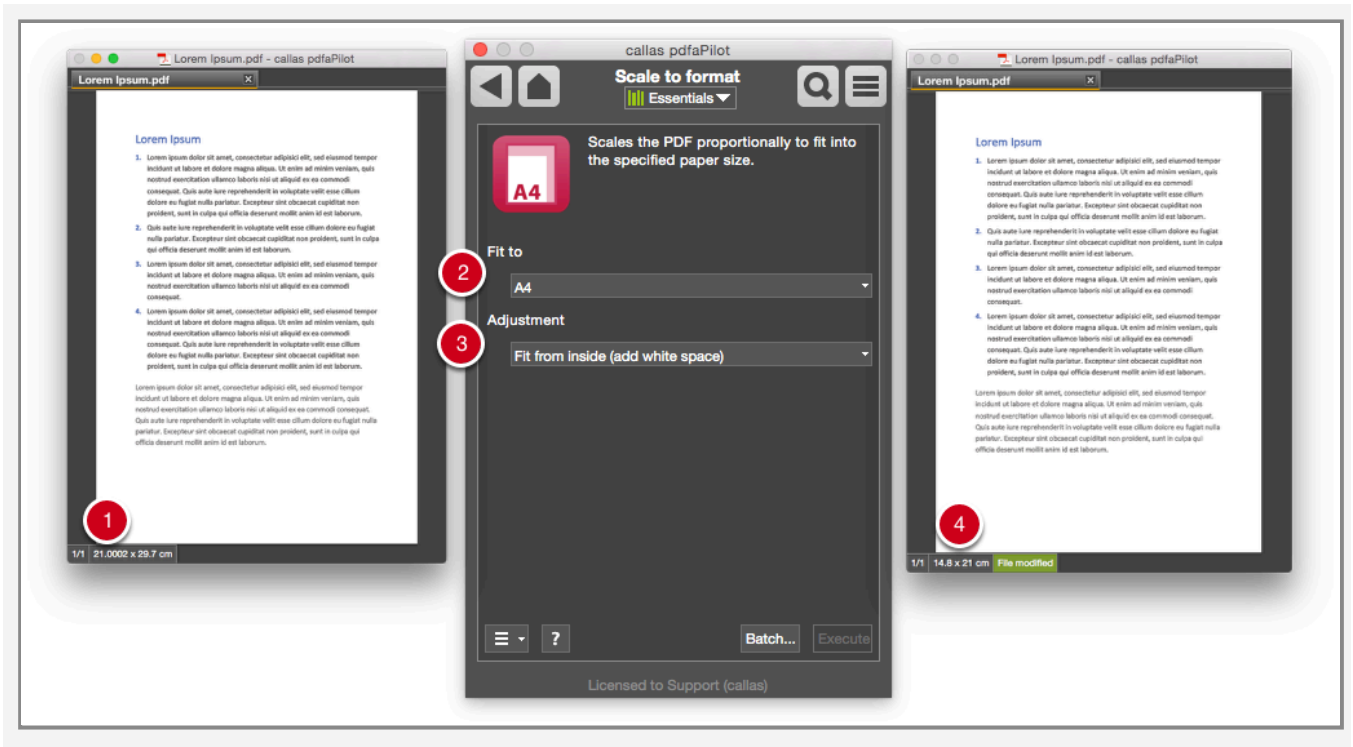


The **Scale to format** Action offers the following options:

1. The **Fit to** menu provides an extensive list of international standard formats to choose from.
2. The **Adjustment** menu provides a range of ways to compare the page content against the page, in case the original document's height-width ratio is different from that of the document to be created.

Click the **Execute** button to begin processing.

Effect of the “Scale to format” Action



Example: DIN A4 to DIN A5:

1. The input document measures 21 x 29.7 cm = DIN A4 format. The dimensions are shown at the bottom left of pdfaPilot's document window.
2. In this case, the target size under **Fit to** is *DIN A5*.
3. We have set the **Adjustment** to **Fit from inside** (which in this case is irrelevant for an A4 to A5 conversion, since the ratio remains the same.)
4. The converted PDF document measures 14.8 x 21 cm = DIN A5.

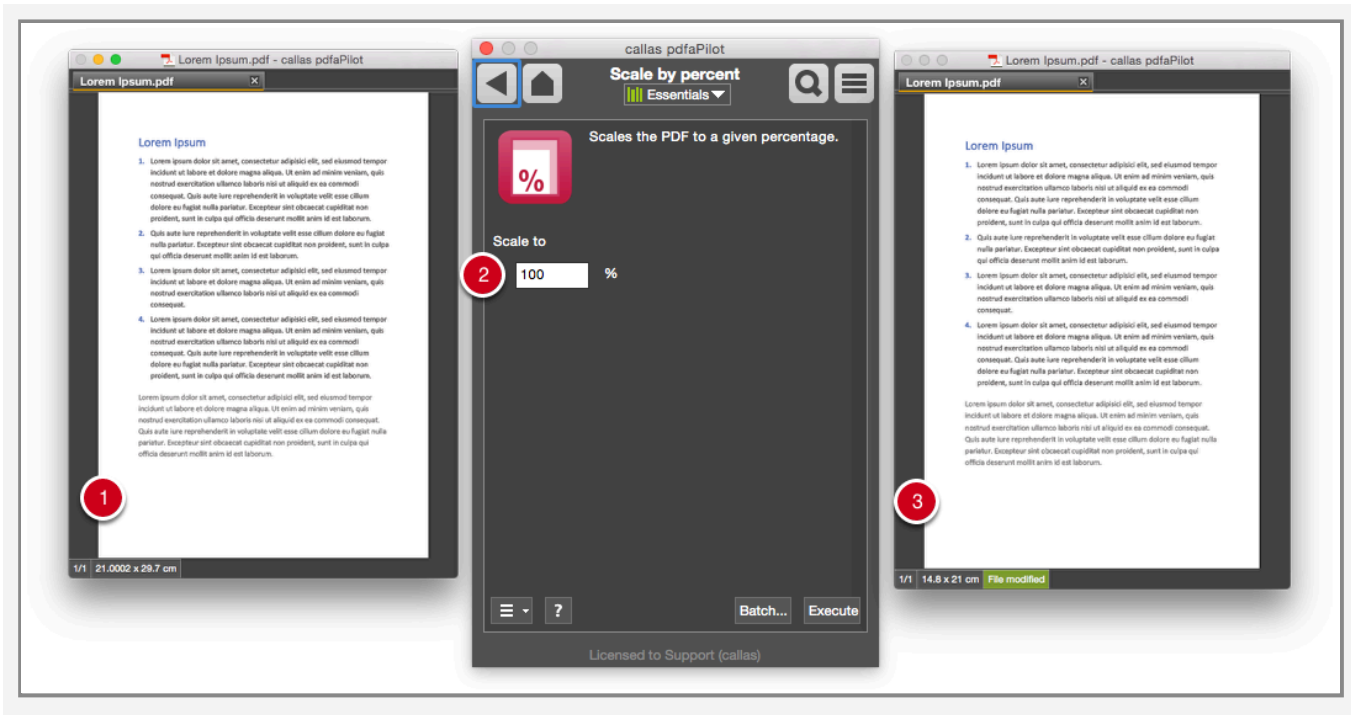
“Scale by percent” Action



The **Scale by percent** Action is intended for use when working with target page sizes which do not correspond to standard formats such as A4 or Letter. The following options are available:

1. The **Scale to** field lets you enter the desired percentage.
2. Click the **Execute** button to begin processing.

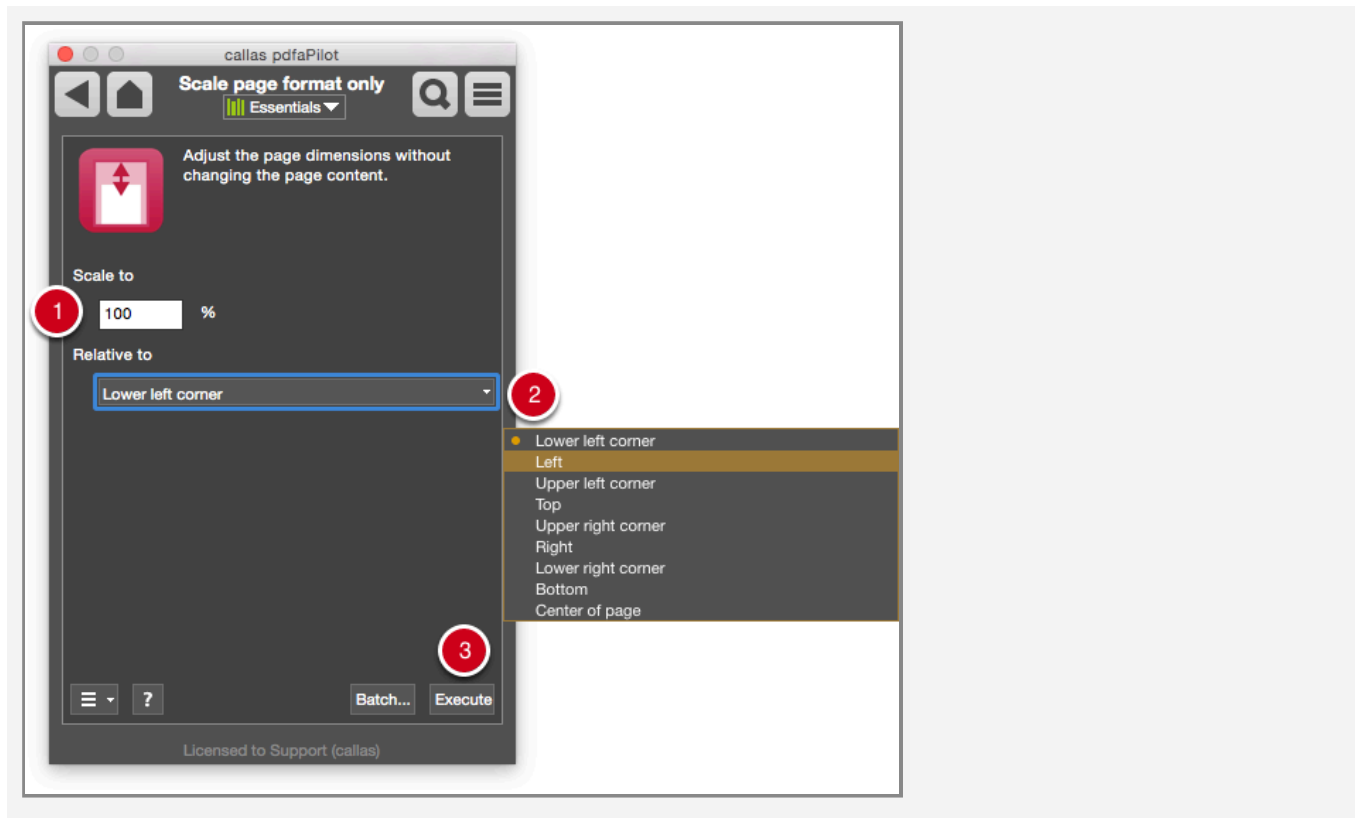
Effect of the “Scale by percent” Action



Example: Scaling by 75 percent:

1. The input document measures 21 x 29.7 cm = DIN A4 format. The dimensions are shown at the bottom left of pdfaPilot's document window.
2. Here, the Scale to value is set to 75%.
3. The converted PDF document measures 15.75 x 22.27 cm = 75 percent.

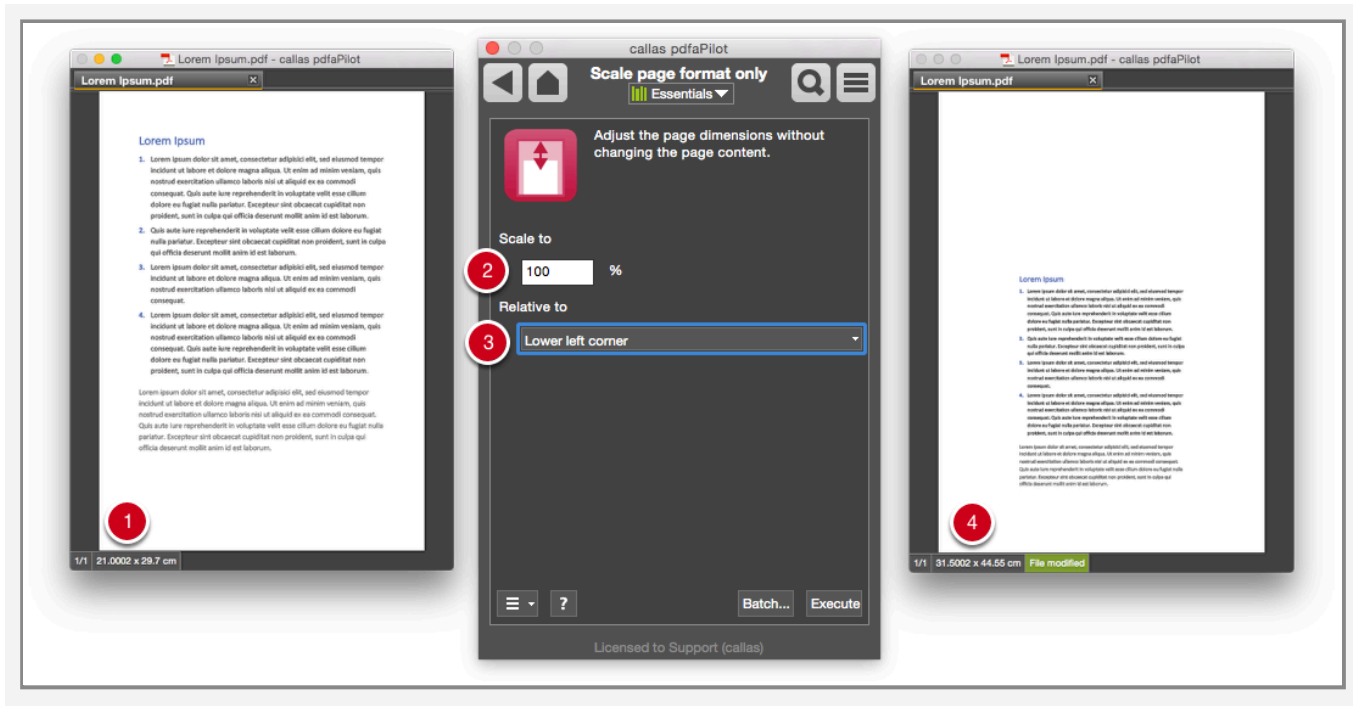
“Scale page format only” Action



The **Scale page format only** Action adjusts the page dimensions without changing the page’s content. The following options are available:

1. The **Scale to** field lets you enter the desired percentage.
2. Under **Relative to**, you can specify which point will be used as a reference for scaling (each corner, each edge or the center.)
3. Click the **Execute** button to begin processing.

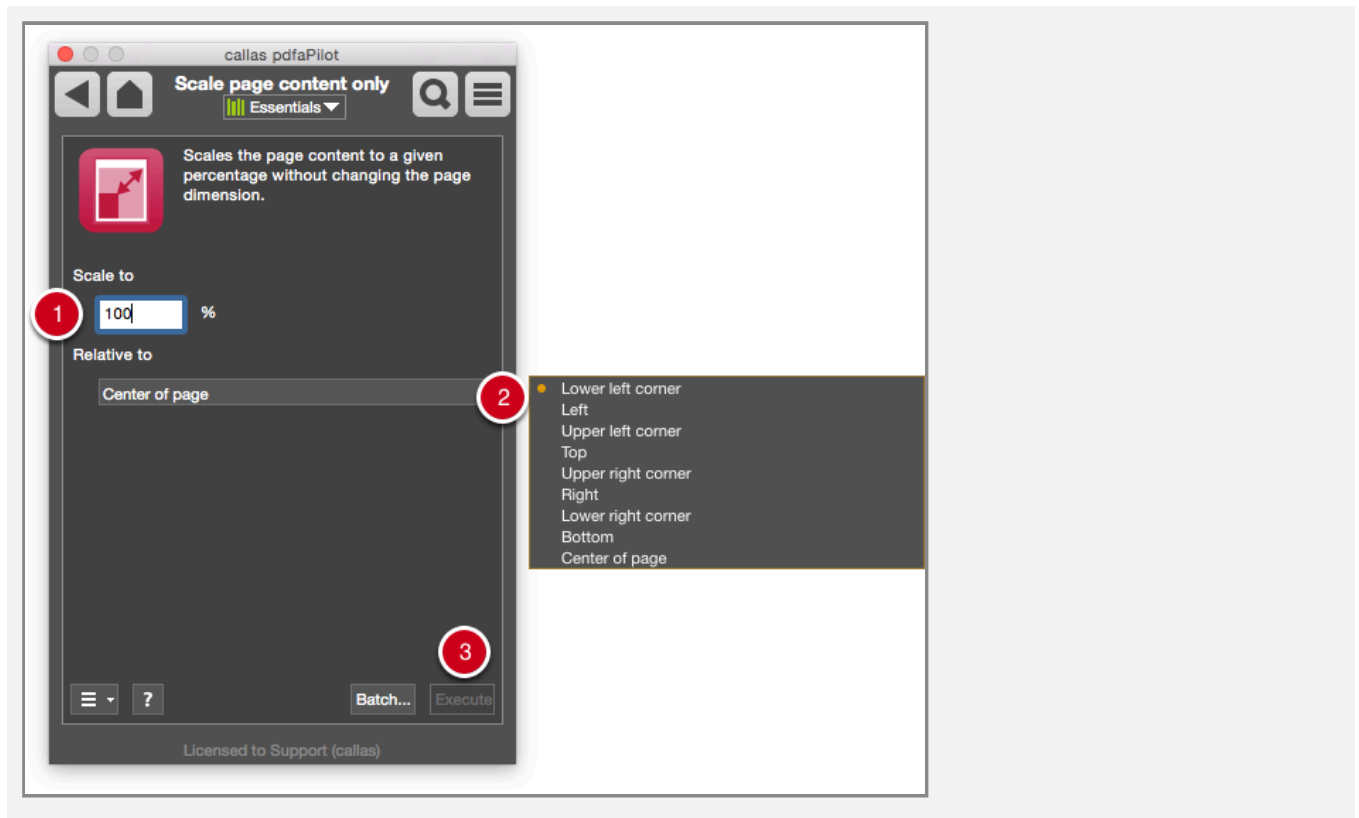
Effect of the “Scale page format only” Action



Example: Scaling by 150 percent:

1. The input document is in DIN A4 format.
2. Here, the Scale to value is set to **150%**.
3. Relative to has been set to **Bottom**.
4. The converted PDF document measures **31.50 x 44.55 cm** = 150 percent. The original page content has not been scaled; instead, white space has been added relative to the reference point.

“Scale page content only” Action



The **Scale page content only** Action adjusts the page content without changing the page’s dimensions. The following options are available:

1. The **Scale to** field lets you enter the desired percentage.
2. Under **Relative to**, you can specify which point will be used as a reference for scaling (each corner, each edge or the center.)
3. Click the **Execute** button to begin processing.

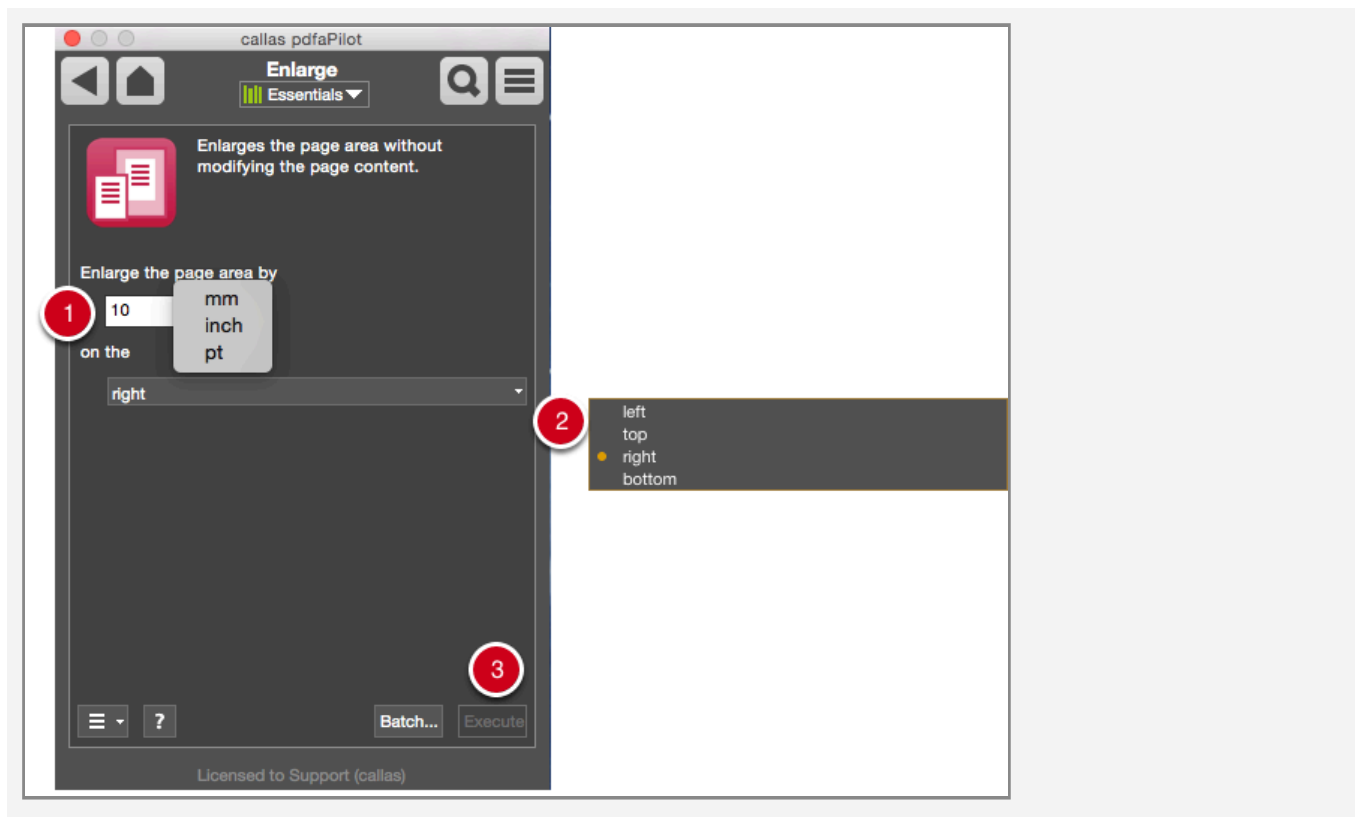
Effect of the “Scale page content only” Action



Example: Scaling by 75 percent:

1. The input document is in DIN A4 format.
2. Here, the **Scale to** value is set to **75%**.
3. **Relative to** is set to **Center of page**.
4. The converted PDF document remains in DIN A5 format.
The page content has been scaled by 75 percent relative to the *center of the page*.

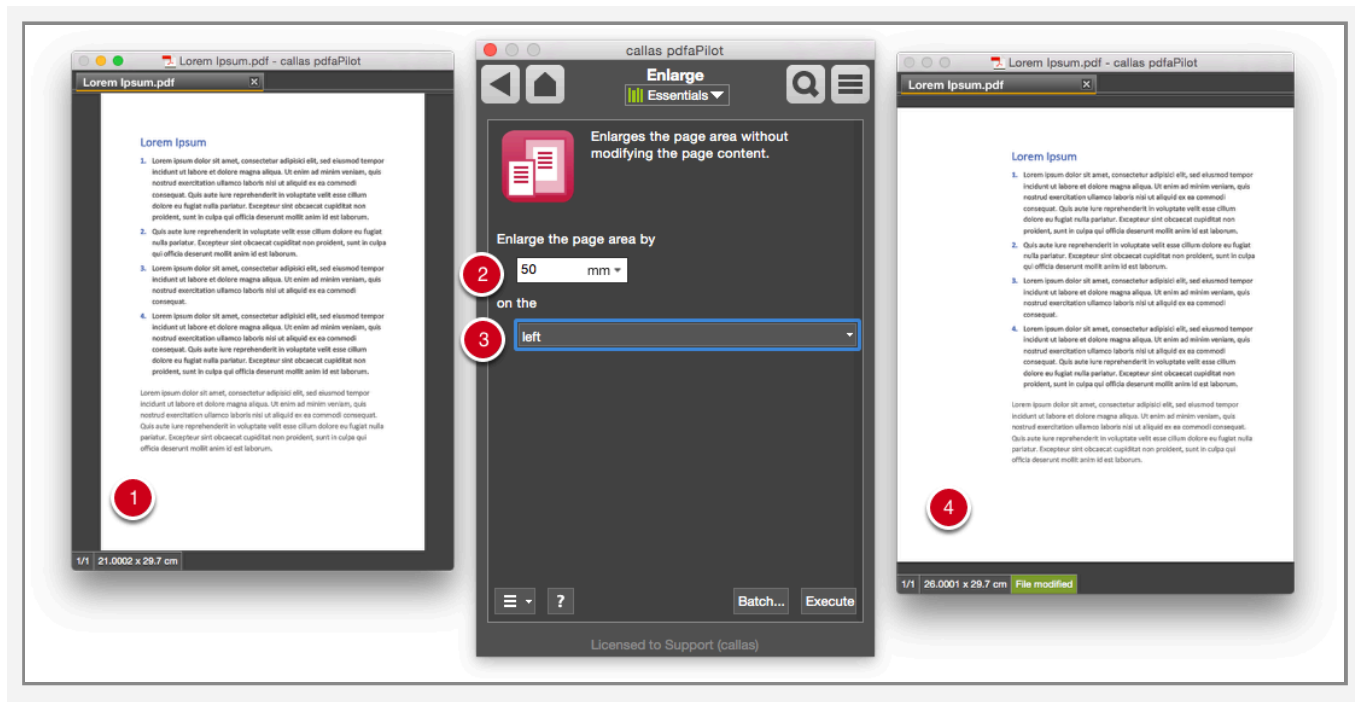
“Enlarge” Action



The **Enlarge** Action individually add white space to each edge of the page:

1. The **Enlarge the page area by** field lets you enter the desired value in **mm**, **inches** or **pt**.
2. Under **Relative to**, you can specify which point will be used as a reference for scaling (*left*, *top*, *right* or *bottom*.)
3. Click the **Execute** button to begin processing.

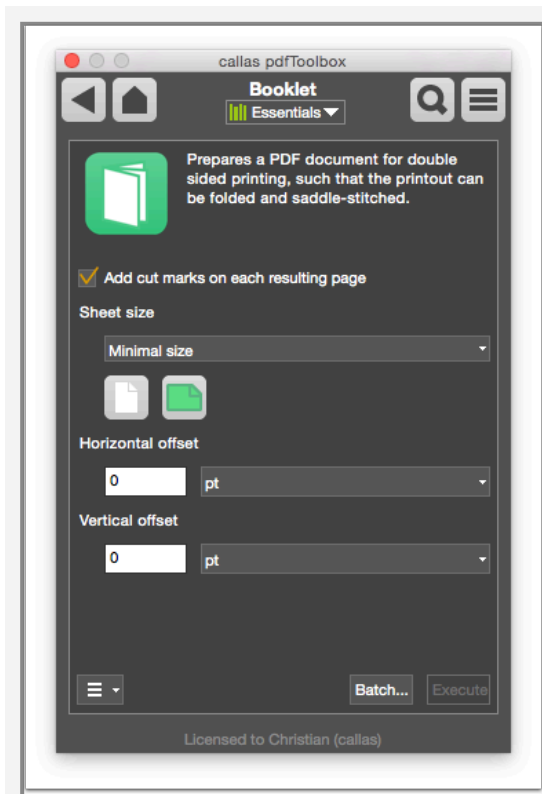
Effect of the “Enlarge” Action



Example: Adding 50 mm to the left edge:

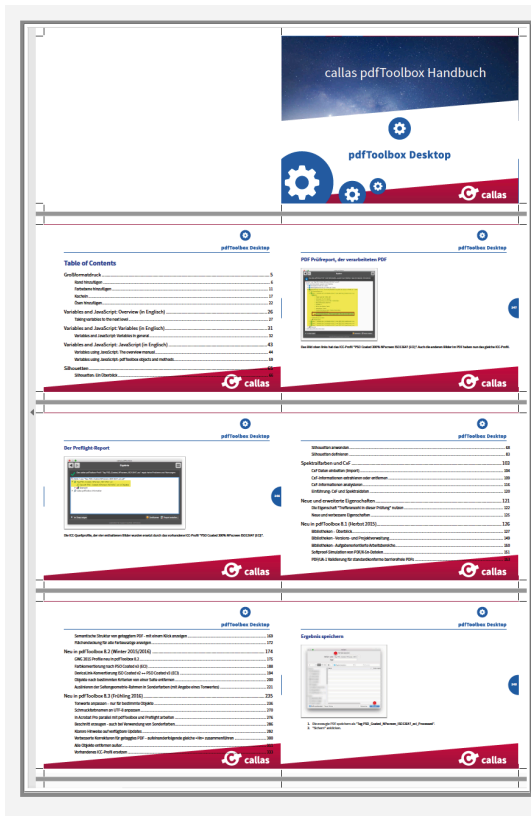
1. The input document is in DIN A4 format.
2. The value under Enlarge the page area by has been set to **50 mm**.
3. It is being added on the **left** side.
4. The converted PDF document now has an extra 50 mm of space on the left side.

Here, you must most importantly set the desired sheet size for the finished brochure, as well as defining its orientation.




Results with double-sided pages


The parameters shown in the screenshot above will produce the following result when executed. This is designed for double-sided printing with saddle stitching.

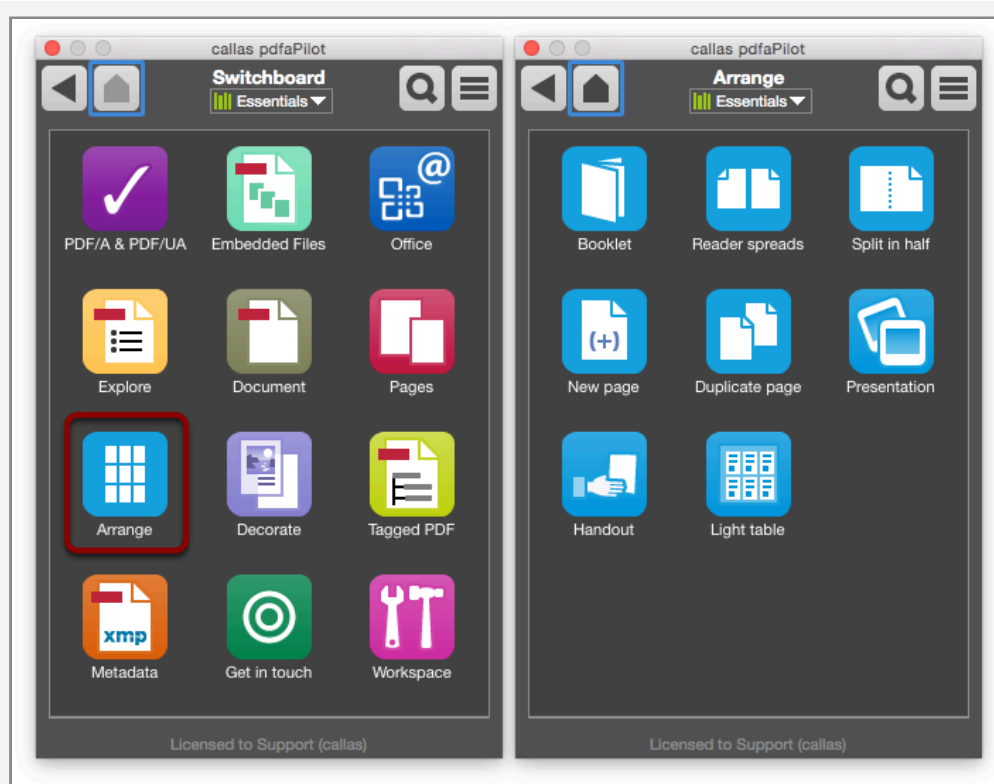


2.10 Switchboard: Arrange – Reader spreads/Split in half

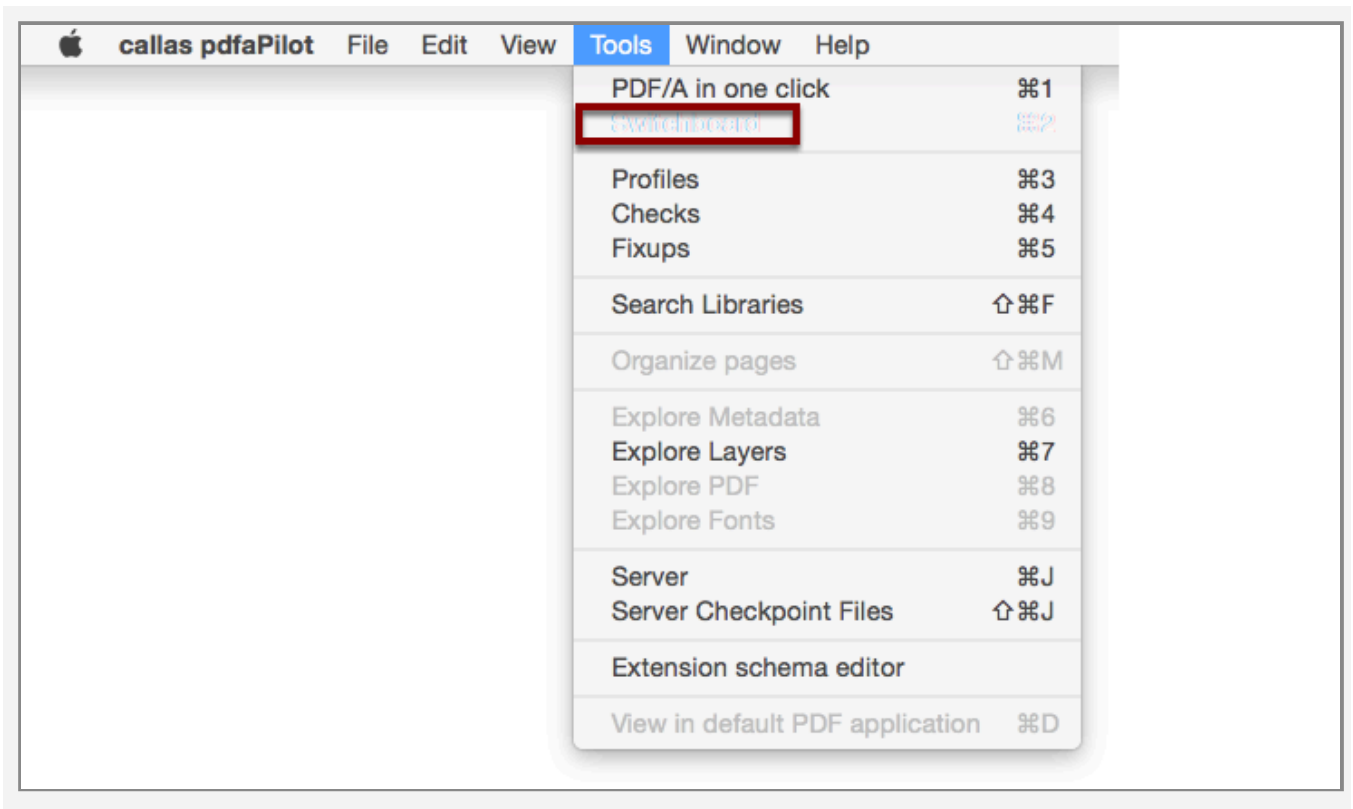
The Arrange group provides a range of Actions used to impose files.

 Lorem Ipsum_8p_spread.pdf

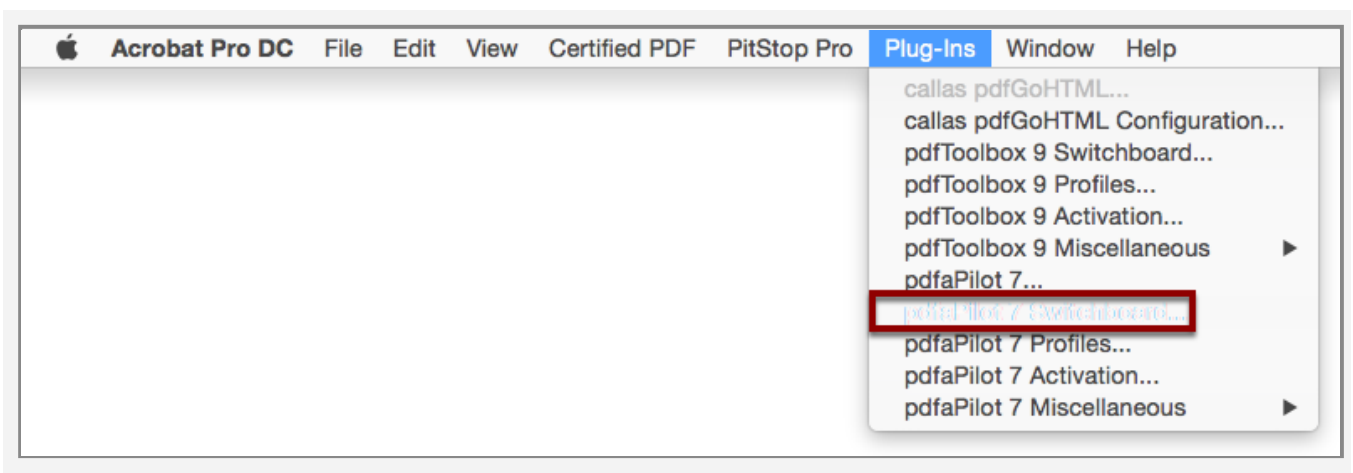
 Lorem Ipsum_8p.pdf



Open the Switchboard

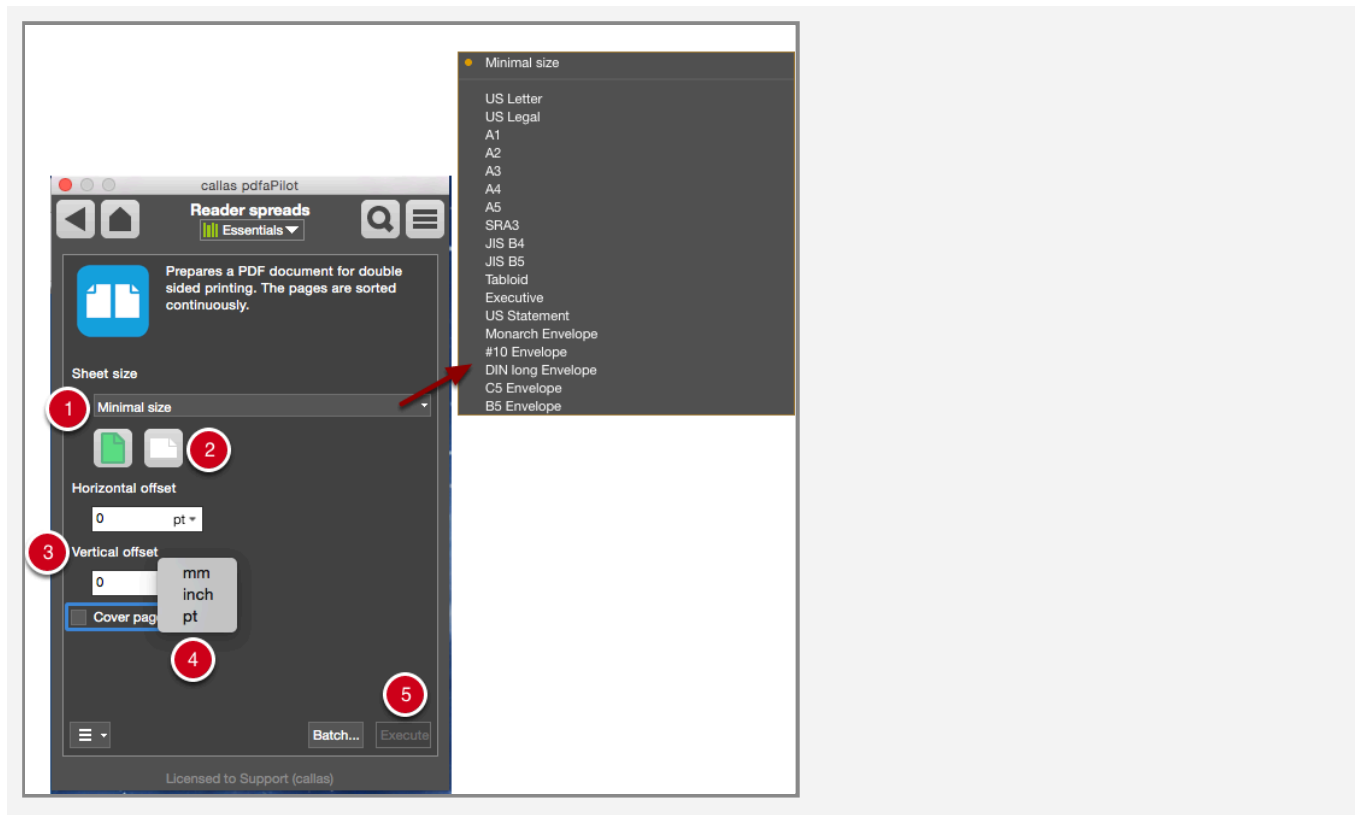


In pdfaPilot standalone, open the switchboard via Tools > Switchboard in the menu or use the keyboard shortcut Cmd+2.



If you are using pdfaPilot as an Acrobat plug-in, you can open the switchboard via Plug-Ins > pdfaPilot <version number> Switchboard.

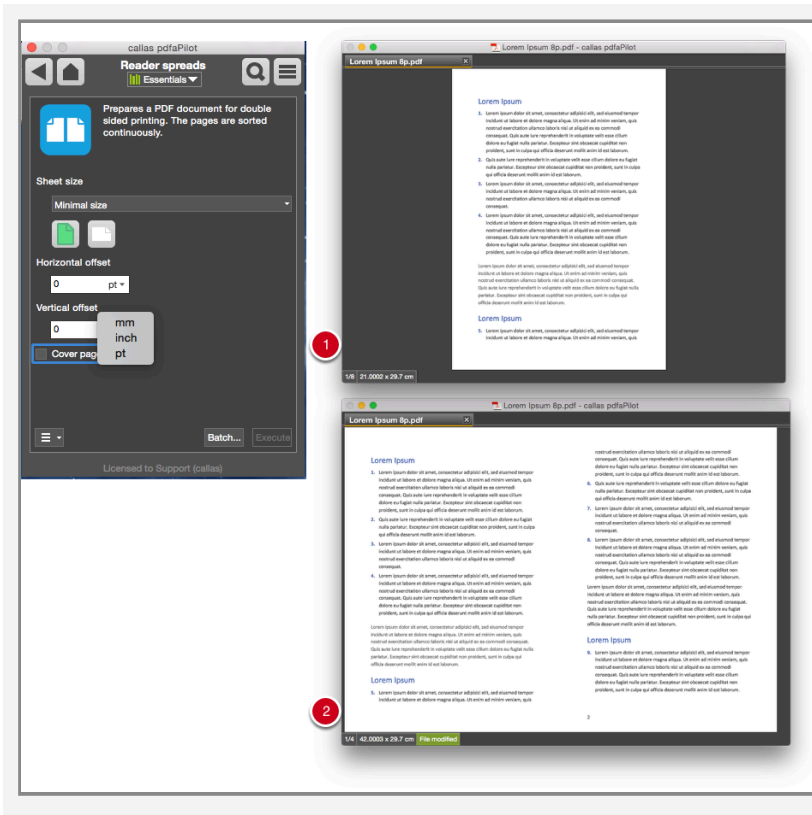
The “Reader spreads” Action



The “Reader spreads” Action turns two individual pages into one print sheet. The following settings are available:

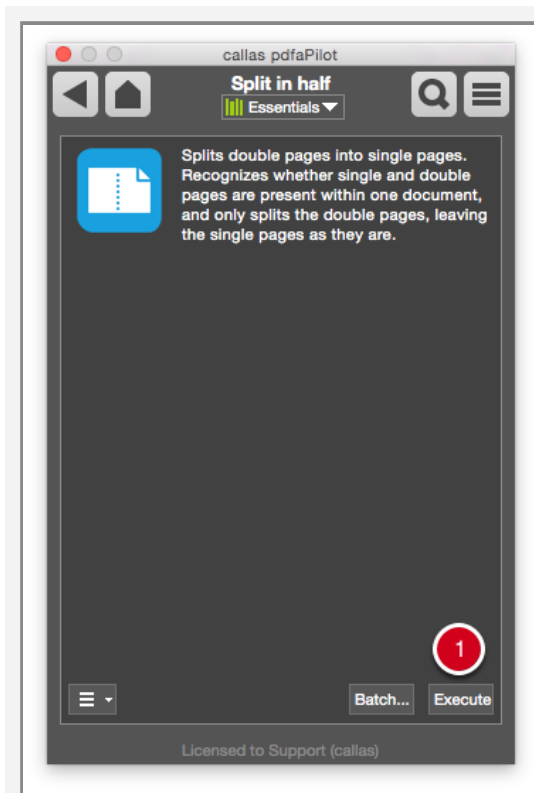
1. The **Sheet size** can be set to *Minimal size* or one of the many *standard page sizes*.
2. You can also choose between *horizontal* and *vertical* page alignment.
3. Values for **horizontal** and/or **vertical offset** can be given in **mm**, **inches** or **pt** in the fields provided.
4. If the original document has a **title page**, this can be accounted for by checking the corresponding box.
5. Click **Execute** to start processing the current PDF document.

Result



1. The input document is 8 pages in DIN A4 size.
2. The output document is 4 pages in DIN A3 size.

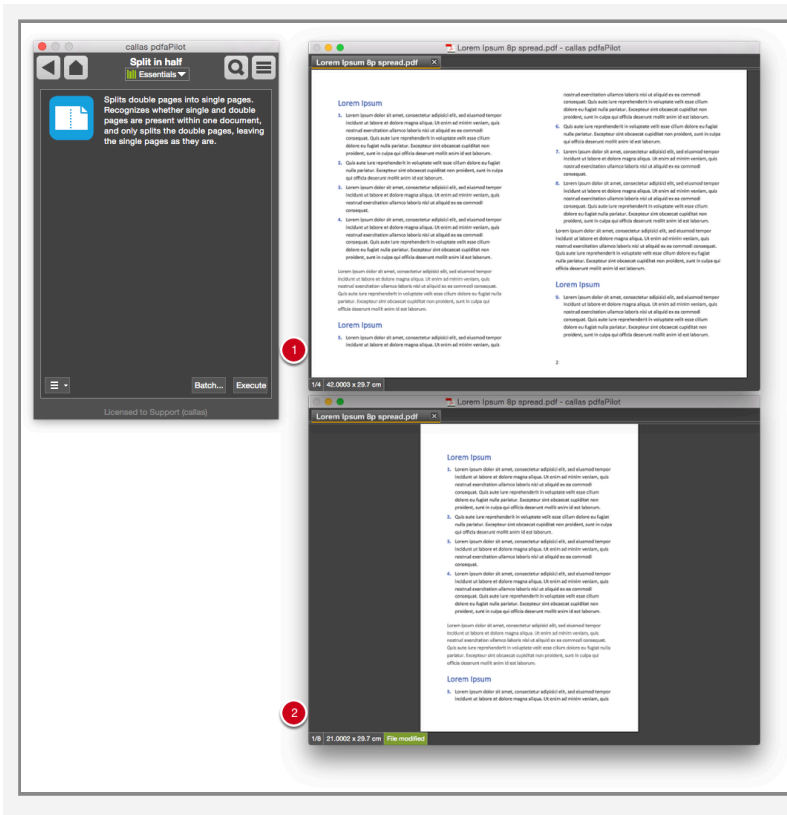
The “Split in half” Action



The **Split in half** action works in reverse, turning one print sheet into two individual pages. No further settings are required for this process.

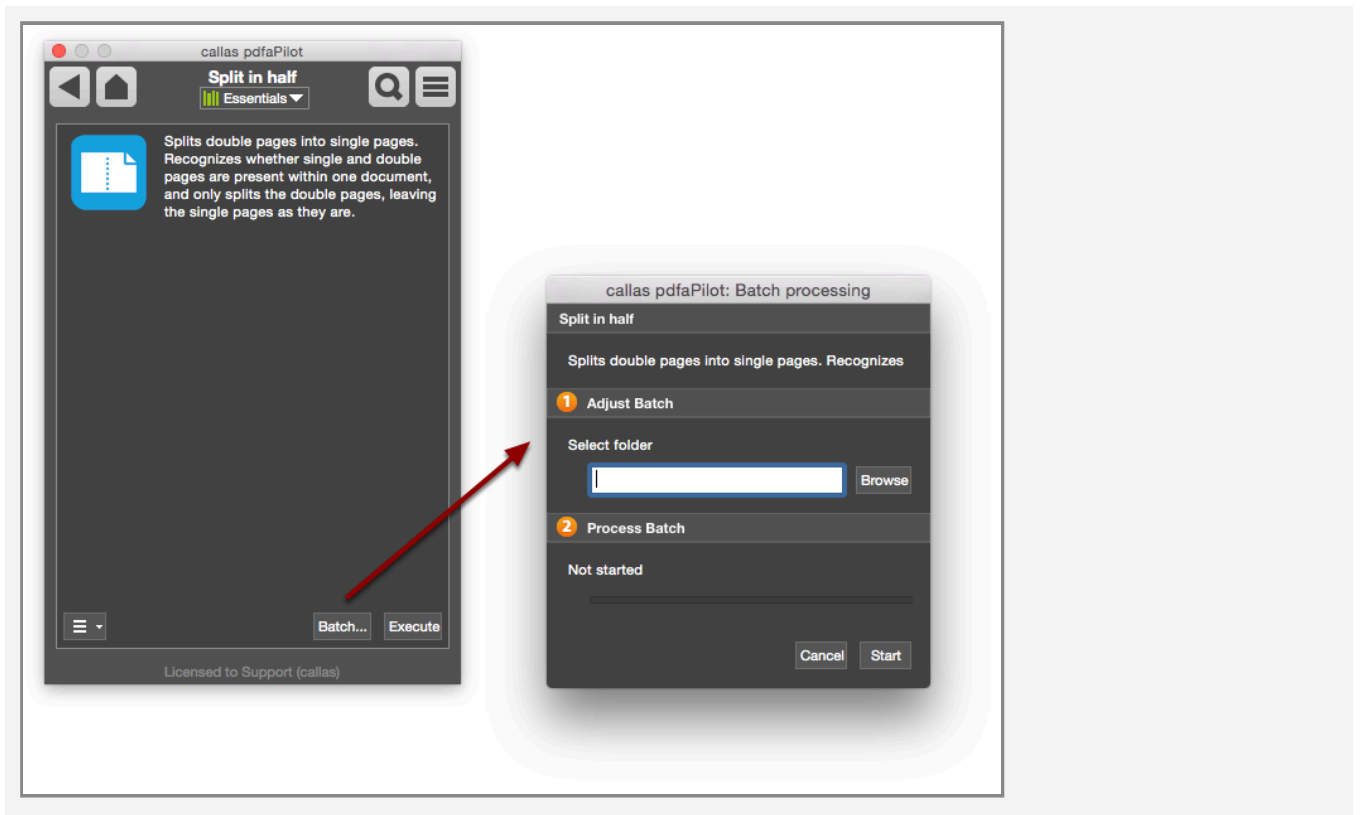
1. Click **Execute** to start processing the current PDF document.

Result



1. The input document is 4 pages in DIN A3 size.
2. The output document is 8 pages in DIN A4 size.

Batch processing:



- i** Both of these Actions can be processed in batches using the **Batch...** button, allowing you to process multiple PDF files within a selected folder.

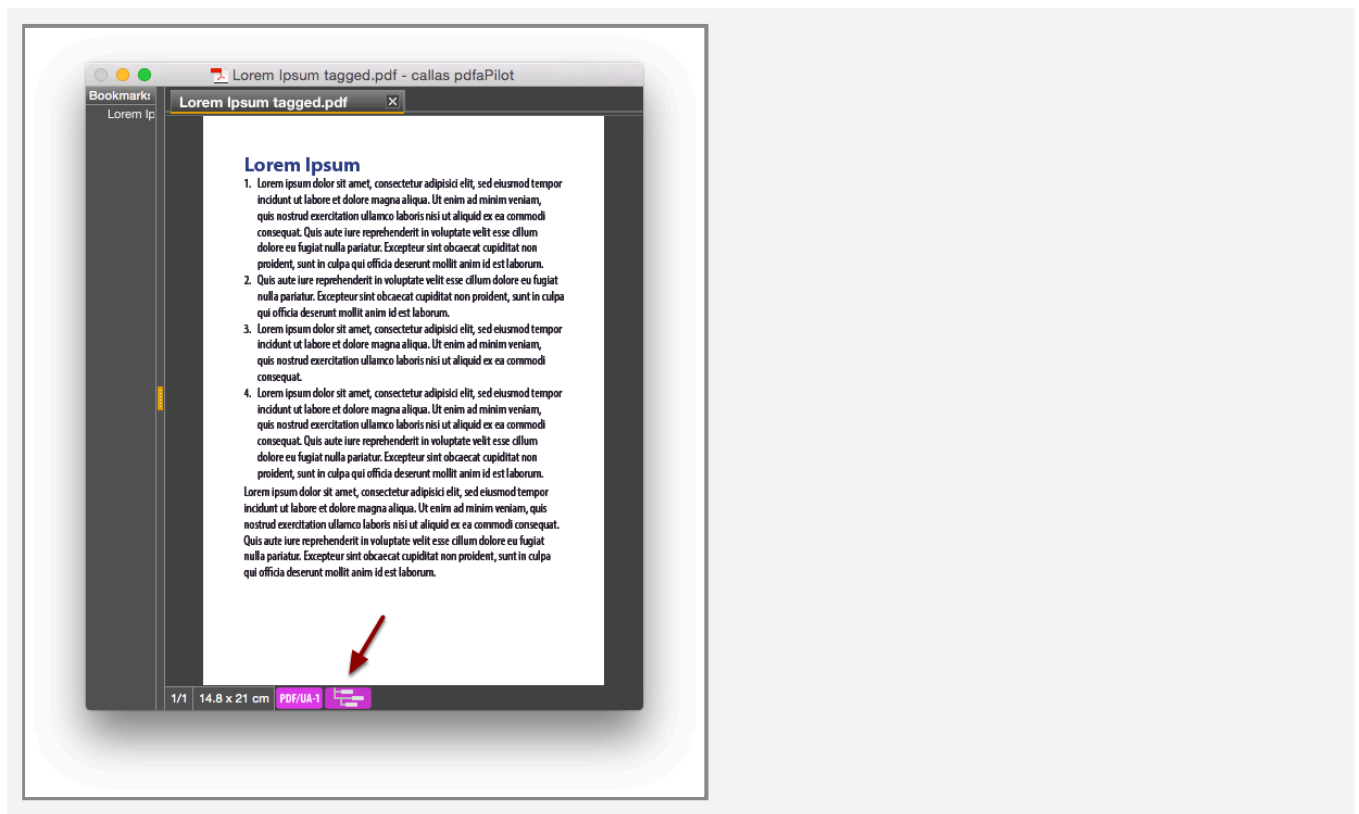
2.11 Switchboard: Tagged PDF – HTML export

PDF files structured using so-called tags can be exported to other structured file formats. These include HTML and EPUB.



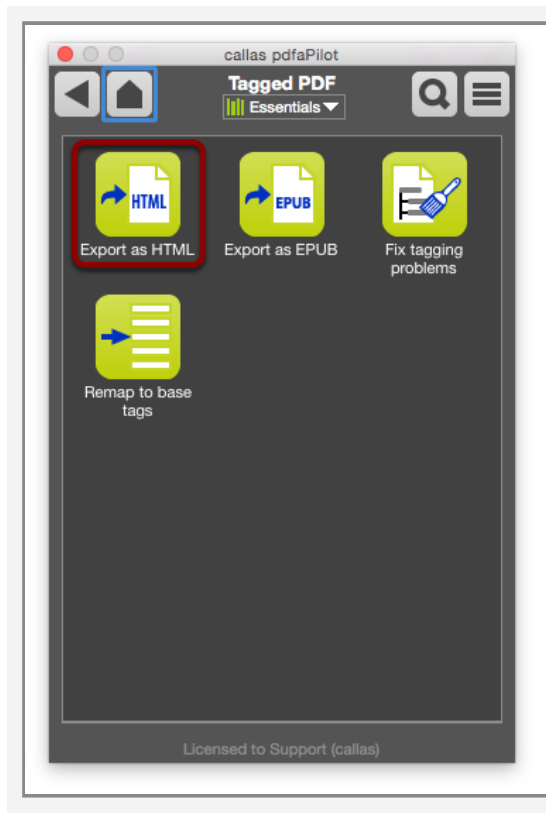
Lorem Ipsum_tagged.pdf

How can I tell whether a PDF document has a structure?



pdfaPilot will show a *symbol* in the file window to indicate whether a PDF file is tagged (i.e. whether it has a structure.)

The “Tagged PDF” Group



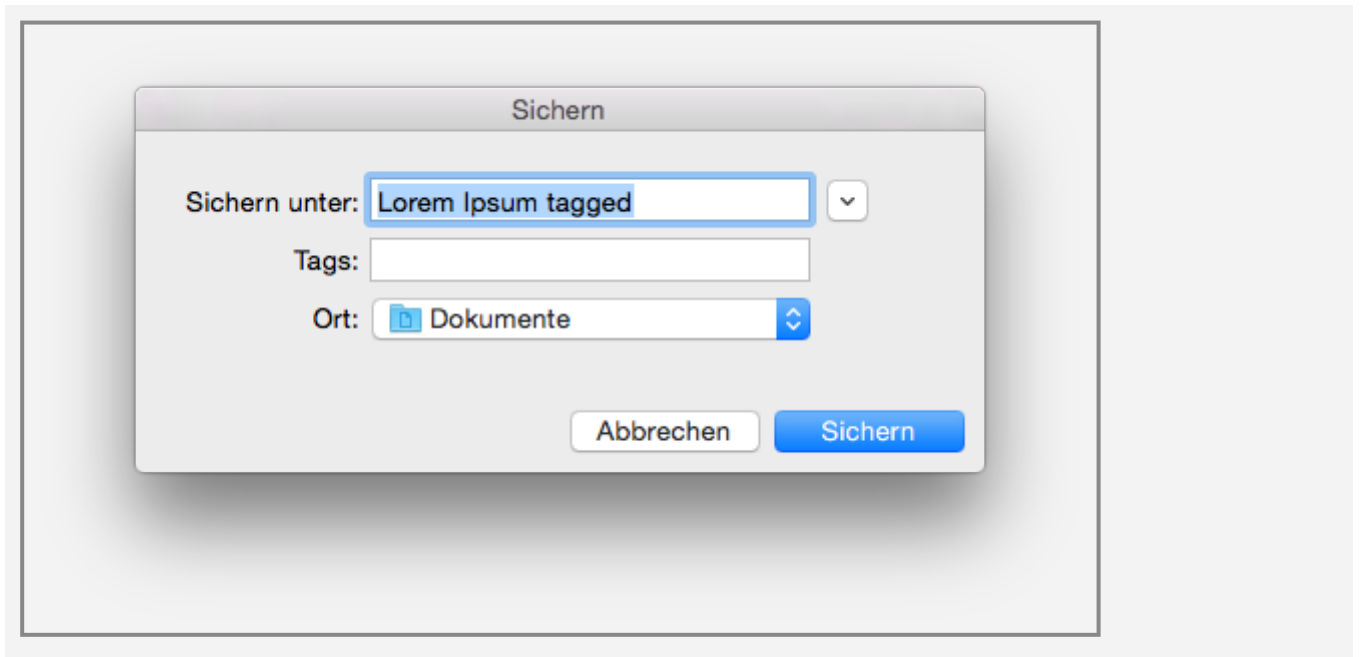
The **Tagged PDF** Switchboard Group contains a number of Actions relating to tagged PDF files and structures.

One action in this Group is named **HTML Export**.

The HTML Export Action

“HTML Export” converts a tagged file to HTML and opens it in your default browser.

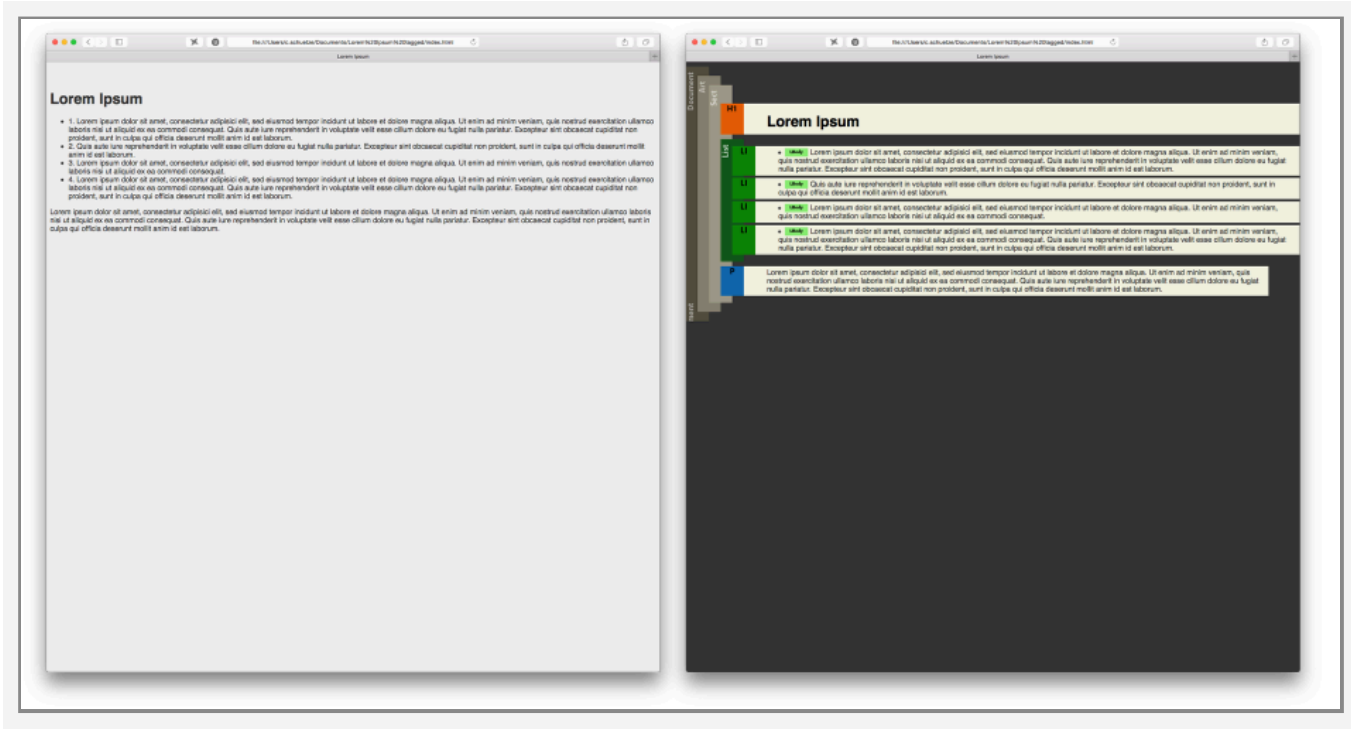
1. The **HTML template** option lets you choose a template from a list (*Simple, Easy to read, Structure tags* and others.)
2. Among other things, the **Action button** allows you to **manage HTML templates**. This also lets you use your own templates.
3. The **Batch** button lets you apply the HTML export process to entire folders if required.
4. Click the **Execute** button to start HTML export.



After the HTML export process starts, a Save dialog will open where you can provide a name and a location for the exported file.

Result

Following HTML export, the output file will be automatically opened in your default browser.



The example above shows the results using *Easy to read* and *Structure tags* settings.

Customizing templates

You can customize export templates according to your own requirements. The folder containing the current library's current templates can be found via the menu button at the bottom-left corner of the Switchboard Action: click "Manage HTML templates."

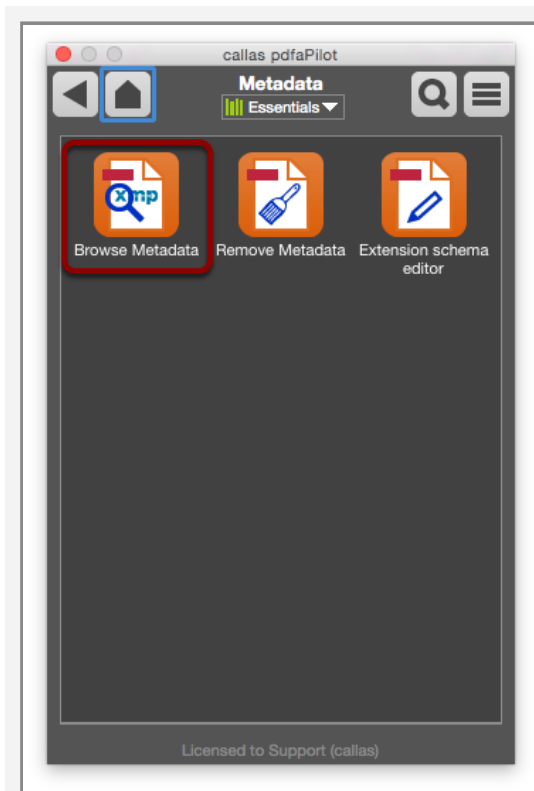
If you have any questions about how to use this feature, our Support team will be happy to help.

2.12 Switchboard: Metadata – Browse metadata

PDF documents can contain a variety of metadata. pdfaPilot allows you to view this information.

The “Browse metadata” Action

Under the Metadata group, you will find the Browse metadata Action.



Click on the symbol to open the Action.



Click **Execute** to open a new **Metadata Explorer** dialog providing a complete overview of the metadata in the current document.

- i** The **Metadata Explorer** can also be opened either using the “Metadata...” menu item in the Acrobat “File” menu, or using the “Explore Metadata” menu item in the plugin or the standalone version of the software.

The Metadata Explorer



1. As well as the document's XMP metadata, the Metadata Explorer also shows metadata for individual page objects. It lists all objects, shows a small preview and indicates their position on the page.
2. Metadata can be exported in the form of a configurable XML report.

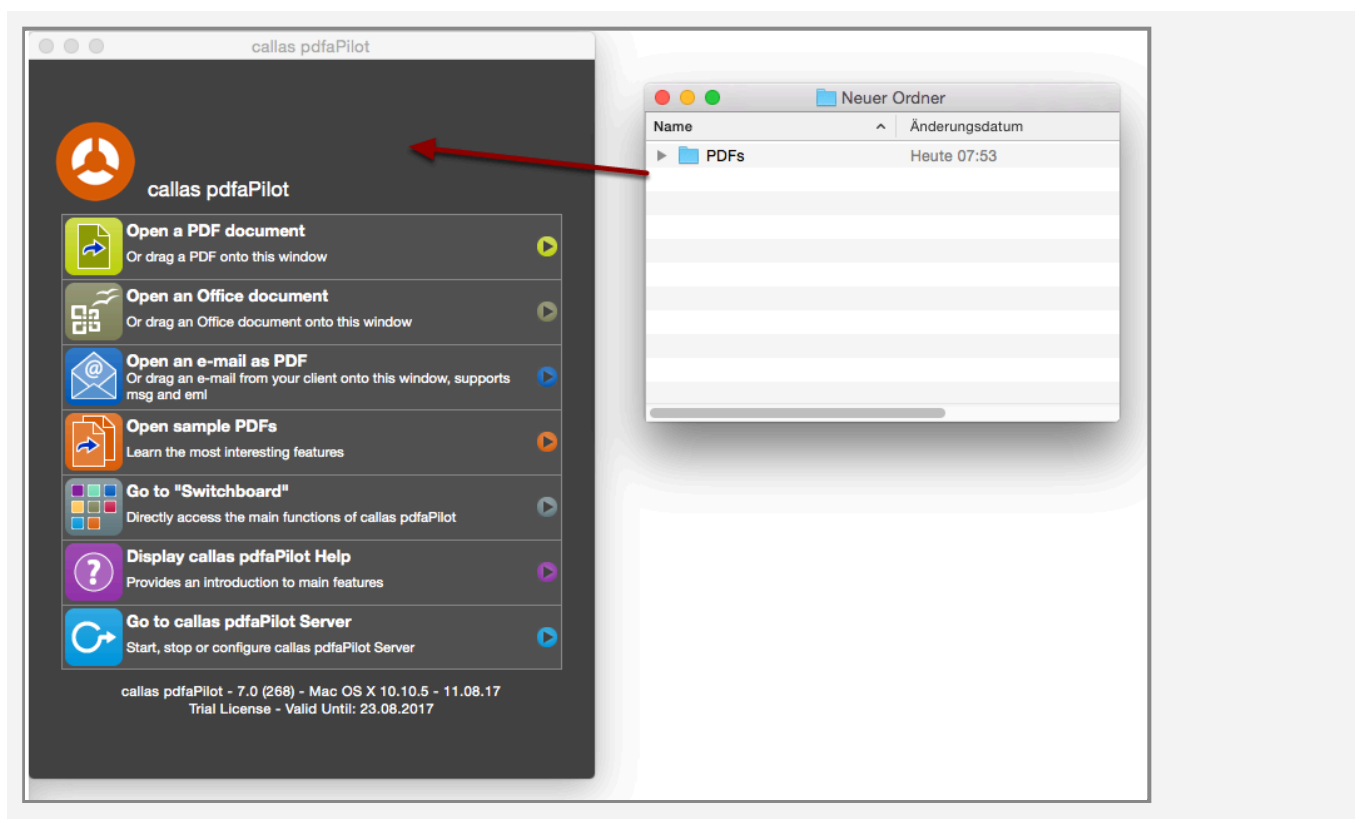
A detailed explanation of how to specify XML export settings can be found here: [XMP Metadata Reports](#).

2.13 Convert entire folder into an archive PDF

pdfaPilot lets you convert folders containing multiple PDF files (or other file formats) into an archive PDF using a simple drag-and-drop process.

This archive PDF will then contain (depending on the selected conversion type) the files from the folder, including all sub-folders.

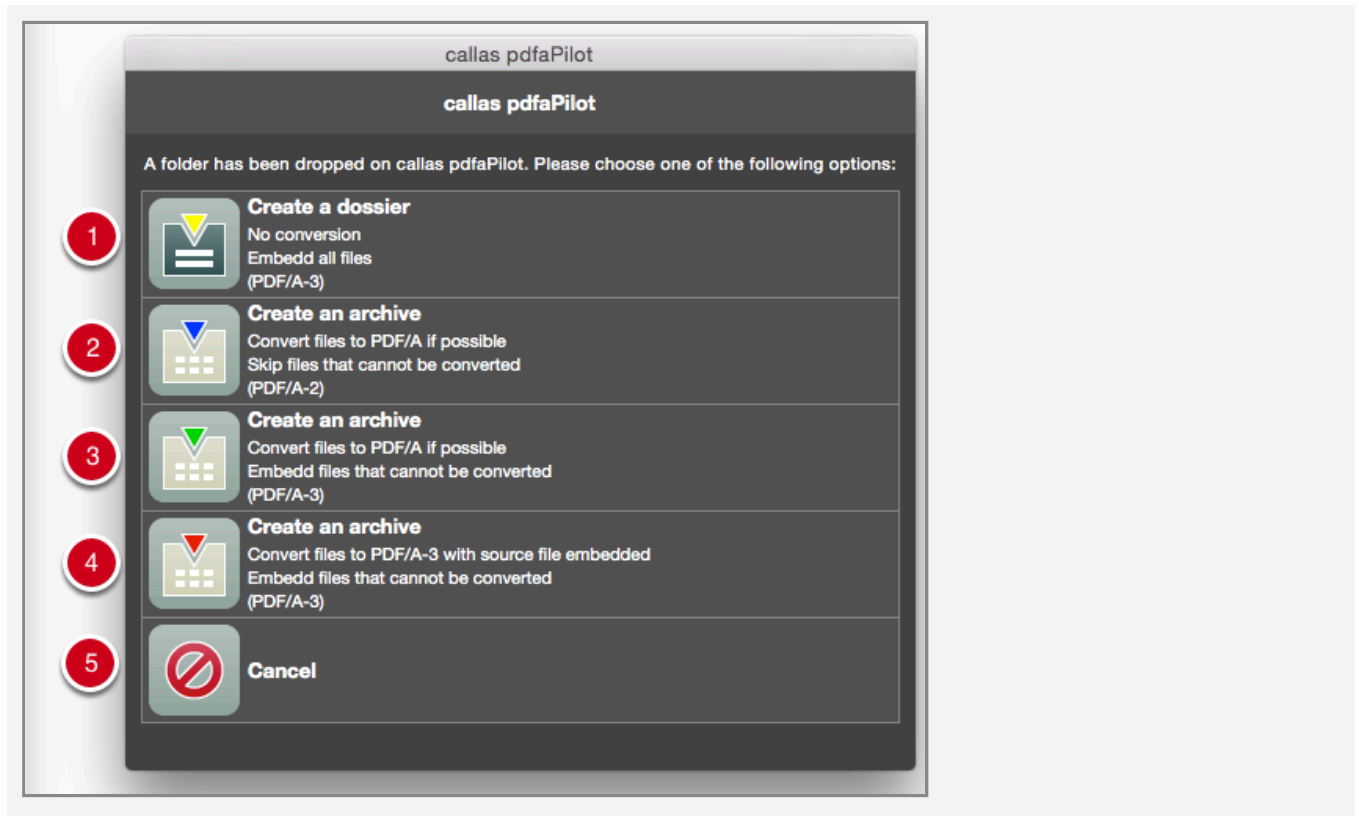
Select folder containing PDF files



Select the folder containing the desired files and drag it into the pdfaPilot window.

Select processing options

A dialog box will open.



A message will appear at the top of the window: **A folder has been dropped on callas pdfaPilot. Please choose one of the following options:**

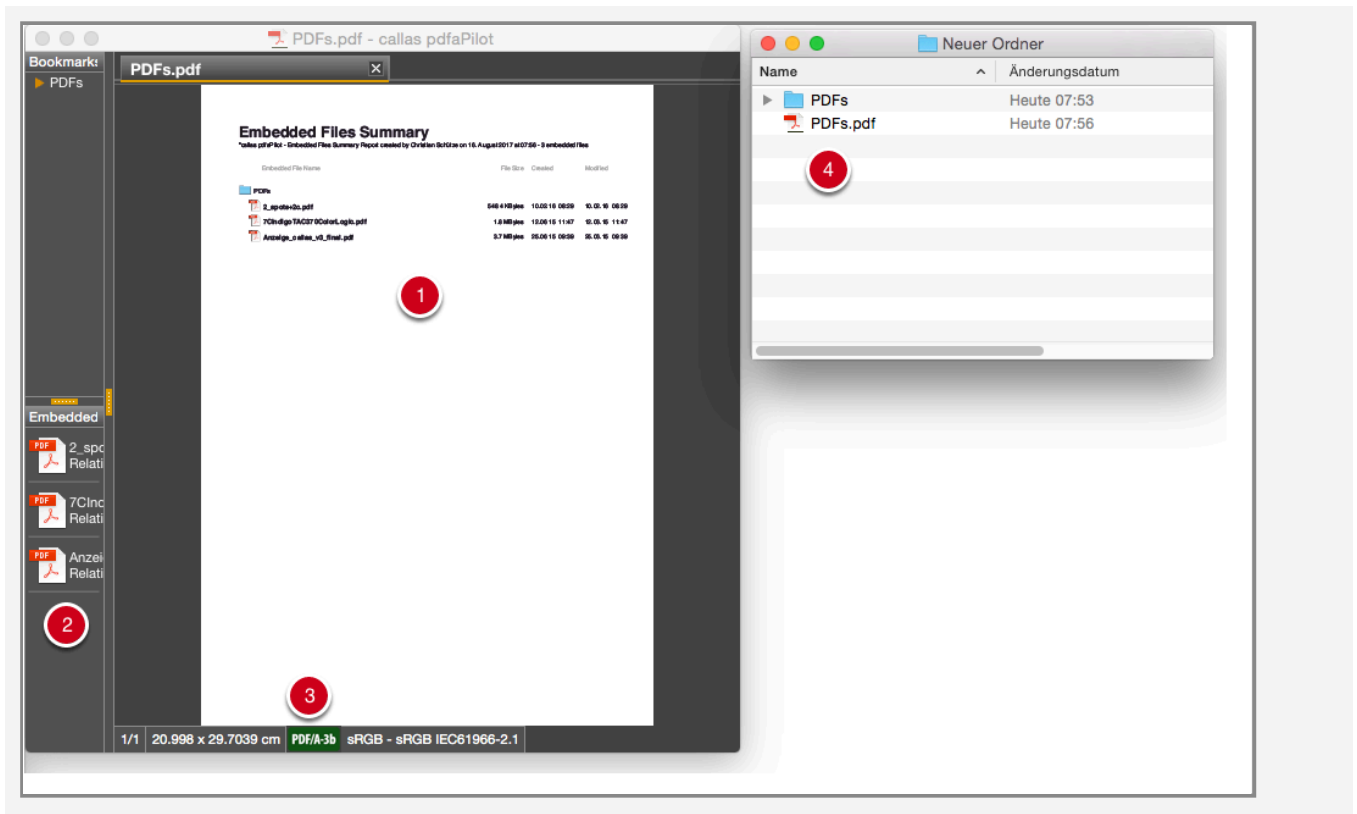
1. **Create a dossier**: In this case, instead of converting files, they will all be embedded in their original format. The resulting PDF will be a PDF/A-3 file.
2. **Create an archive, PDF/A-2**: Files (and file formats) will be converted if possible. Files which cannot be converted will be excluded. The resulting PDF will be a PDF/A-2 file.
3. **Create an archive, PDF/A-3**: Files will be converted if possible. Files which cannot be converted will be embedded in their original format. The resulting PDF will be a PDF/A-3 file.
4. **Create archive, PDF/A-3, plus source file**: Files will be converted to PDF/A-3. Files which cannot be converted will be embedded. The resulting PDF will be a PDF/A-3 file.

Files which can be converted to PDF from other formats will be embedded in addition to the original file.

5. **Cancel:** Cancel and close the window.

pdfaPilot begins processing. A **progress bar** reports on the process.

Result

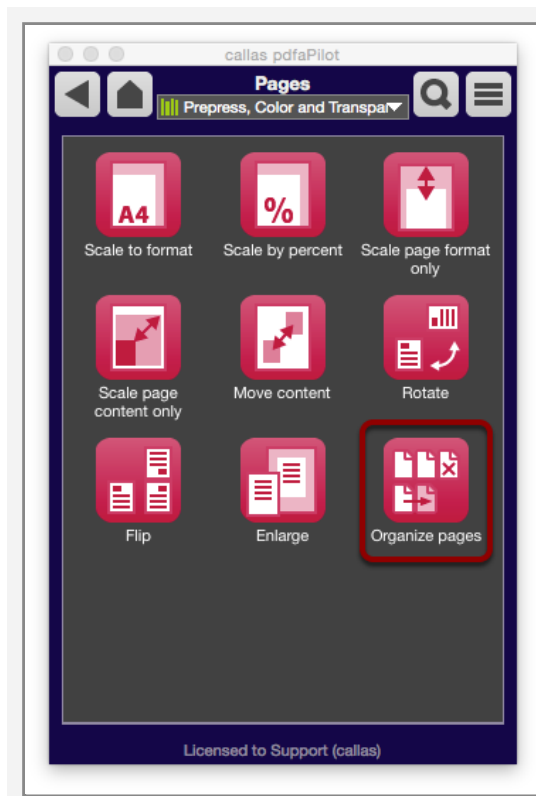


1. A cover page will show the user a **summary of embedded files**.
2. The **Embedded files** will also be shown in a list on the left.
3. At the bottom of the Document window, you will also see the name of the PDF standard used.
4. The newly created PDF file will be named after **the folder that was processed**. It will be stored one level up from the processed folder.

2.14 Organize pages

The *Organize Pages* action is a flexible and easy-to-use tool for rearranging (moving, deleting, duplicating or selecting just a few) pages within a multi-page PDF file. It is even possible to move or copy pages from one PDF into another. We'll show you how you can use this tool.

Switchboard: Pages



The *Organize Pages* action can be found in the Switchboard under the *Pages* group.

Click on the symbol to open the following window:

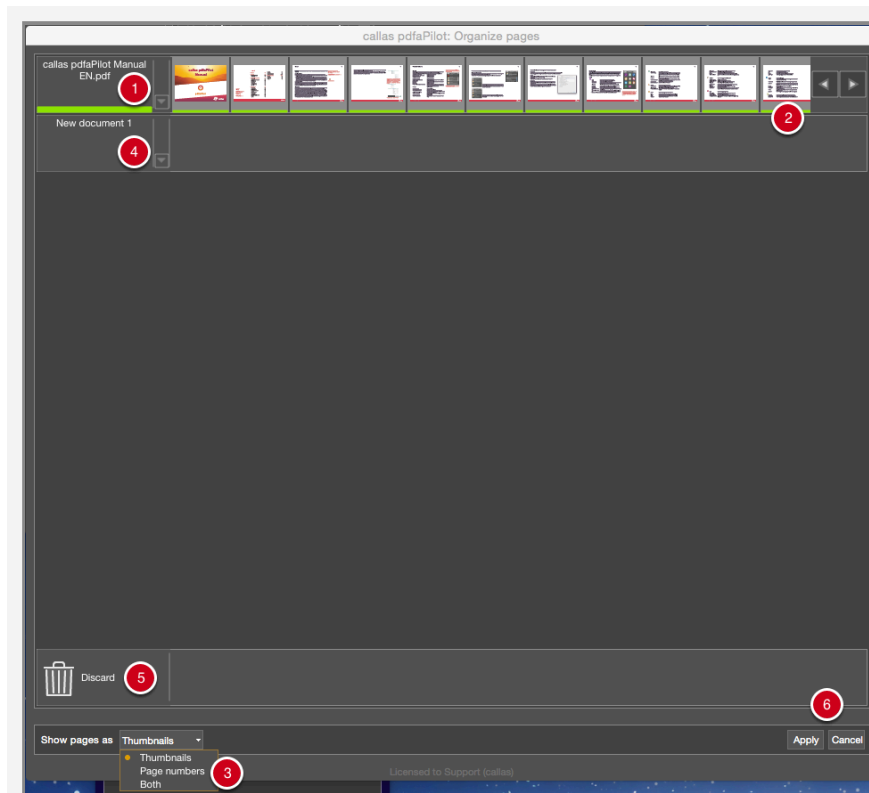


1. The **Organize Pages** window provides a brief description of the action.
2. Click on the **Run** button to open a large overview window where you can rearrange the pages in the currently open PDF document. (The Run window will only be active if a PDF document is open.)

In pdfToolbox Standalone you will find an Organize Pages entry as well in the Tools menu or you use the keyboard shortcut (Win: CMD+Shift+M, Mac: Strg+Shift+M).

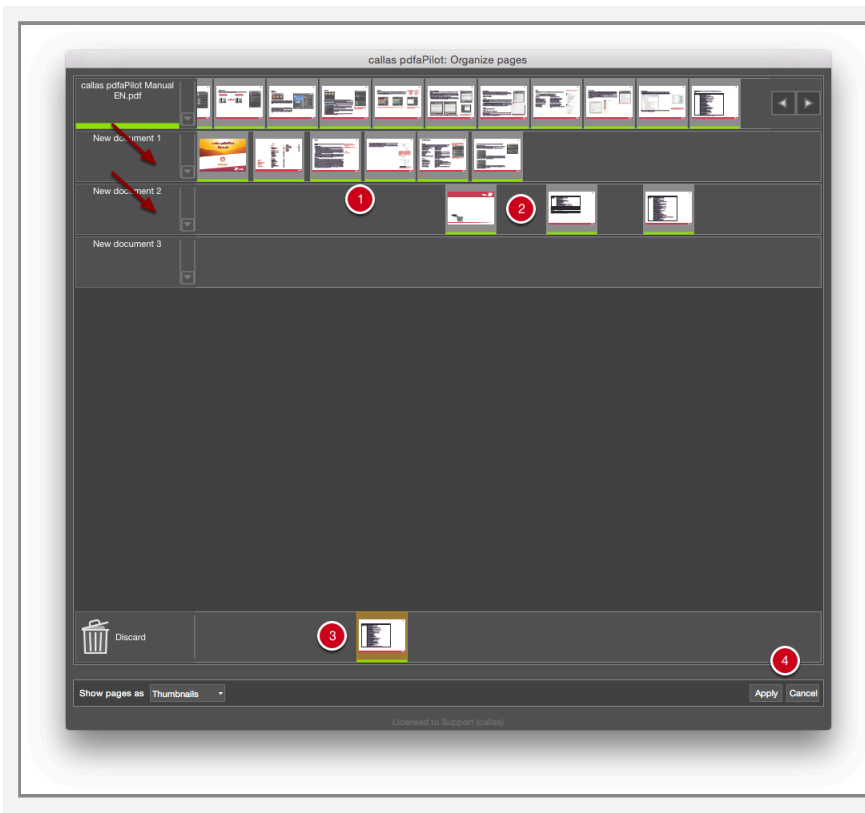
Tip: As reorganizing pages may also alter the original document in some cases, it may be necessary to work with a duplicate if it is essential to leave the source file in its original state. Otherwise, when saving the new document, you will need to make sure that the original is not saved.

Window: Organize pages



1. The top part of the **Organize Pages** window shows all pages within the open document in a row.
2. For documents with large numbers of pages, you can use the arrows on the right to adjust the previewed pages shown.
3. The type of page view can be configured using a pull-down menu in the lower-right corner: **Miniatures**, **Page Counts** or **Both**.
4. Below the row of pages in the original document, you will see an initially empty row to which you can drag selected pages. This area will initially be given the name **New Document 1** (this name can be changed when you save the document).
5. Below, there is a row labeled **Remove** where you can place pages to be removed.
6. At the very bottom right, you will find buttons to **Apply** the changes or **Cancel** the action.

Organizing pages within the application

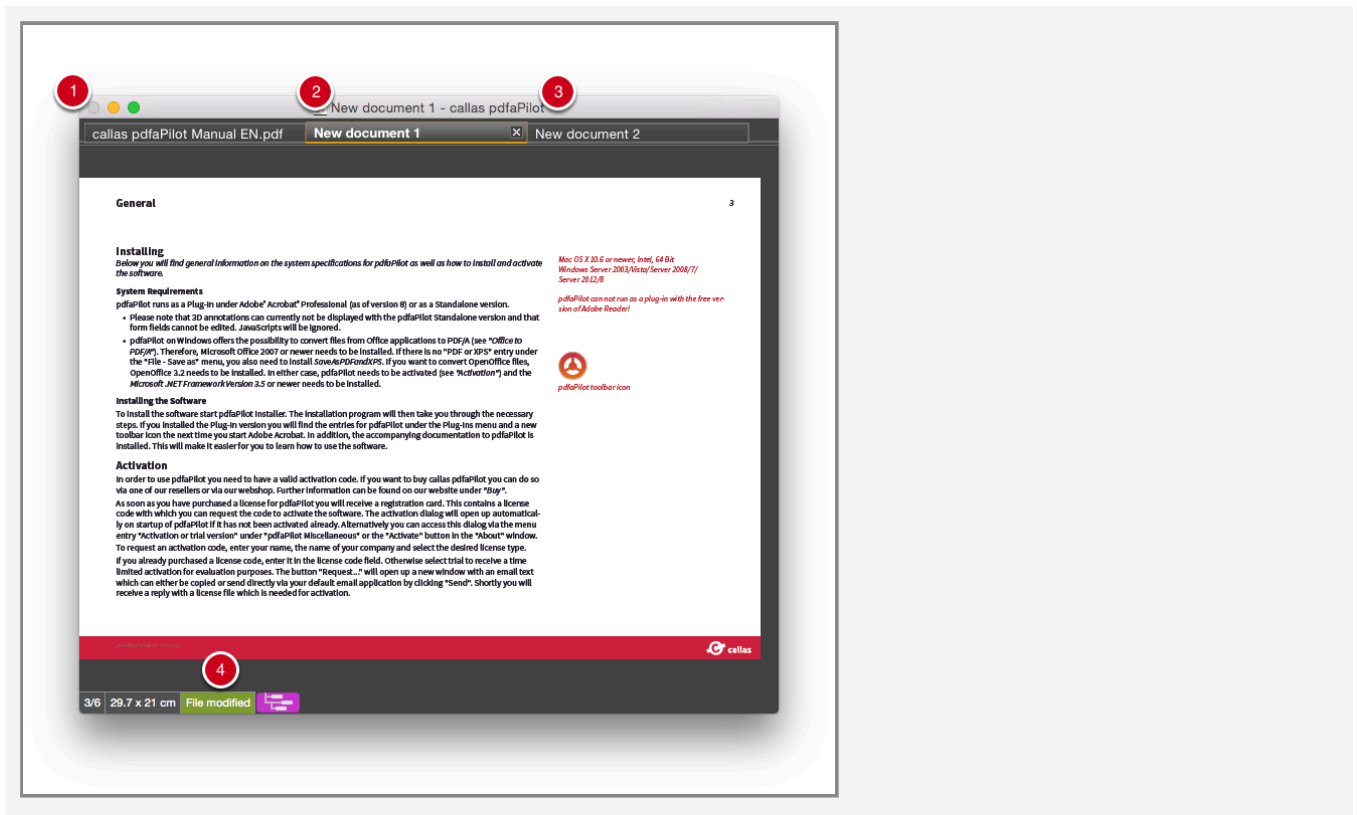


User-defined **page arrangement** for a new PDF document (or, as seen here, for multiple new documents) is a very intuitive process.

We have chosen the **Both** view here for a clearer picture of the process.

1. When the desired pages are moved to the **New Document 1** row, a new row is automatically created (**New Document 2**, and so on.) A new empty row will always be provided at the end of the list; if all pages are removed from a row, this row will be hidden.
Tip: You can **duplicate** pages by holding the **Alt** key and dragging them.
2. Pages can be **freely moved** within a row (here, the former Page 10 has been moved in front of Pages 8 and 9.)
3. Pages to be deleted can be moved to the **Remove** row.
4. Click **Apply** to apply the changes.

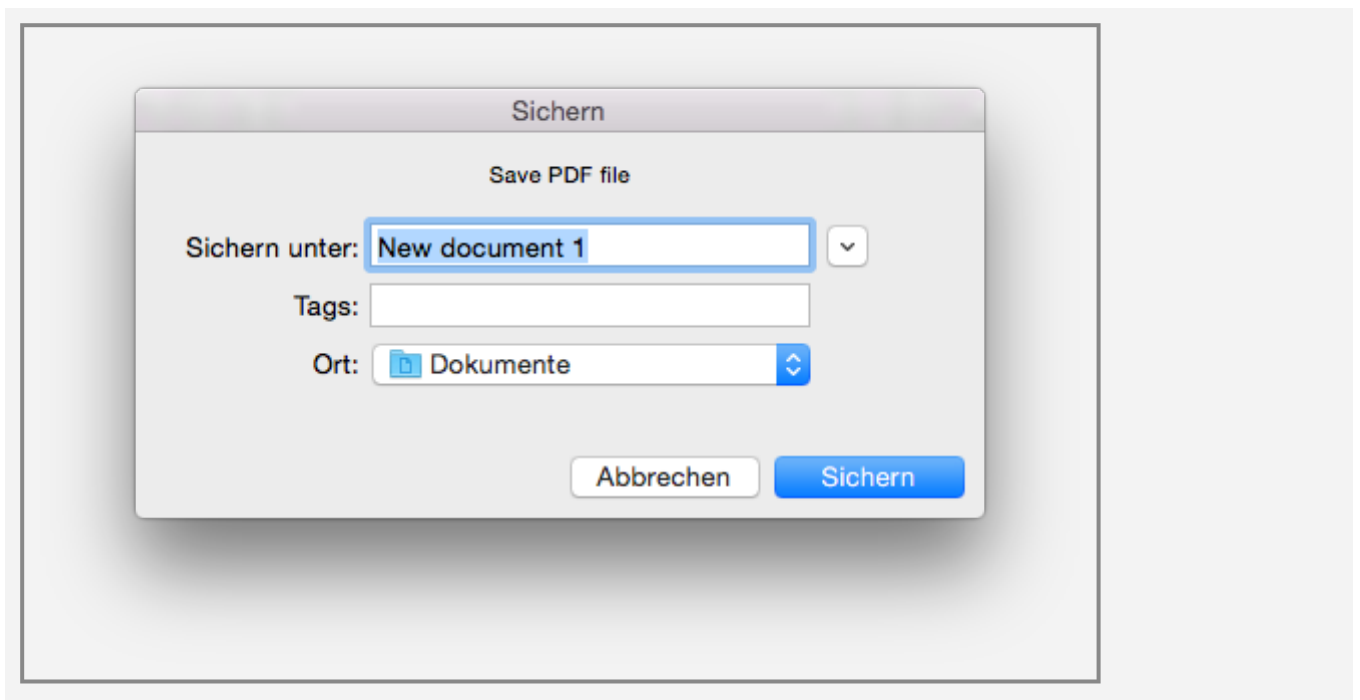
The altered files



pdfToolbox shows all altered files:

1. The original document (if applicable, minus the pages moved to the new documents or those that were removed.)
2. New Document 1 with the selected pages in the desired order.
3. New Document 2 with the selected pages in the desired order.
4. At the bottom of the window, you can see that the file has been altered.

Saving the new documents



You can then save the new document(s) under a new name.

Note: Do not save the original document when closing the window if you want to keep it in its original form.

2.15 PDF/UA-1 validation for standards-compliant universally accessible PDFs



PDFUA-Reference-04_(Danish_Blind_Association).pdf

PDF/UA, the PDF standard for universal accessibility, must meet certain conditions in order to be usable with screen readers, braille output devices, or other systems.

PDF/UA uses tags to create structured PDF files and alternative text for images, as well as for compliance with linguistic and font-related requirements.

As of version 8.1 of pdfToolbox, PDF files can be tested for compliance with PDF/UA-1.

The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

Notification of an existing PDF/UA document in the document window



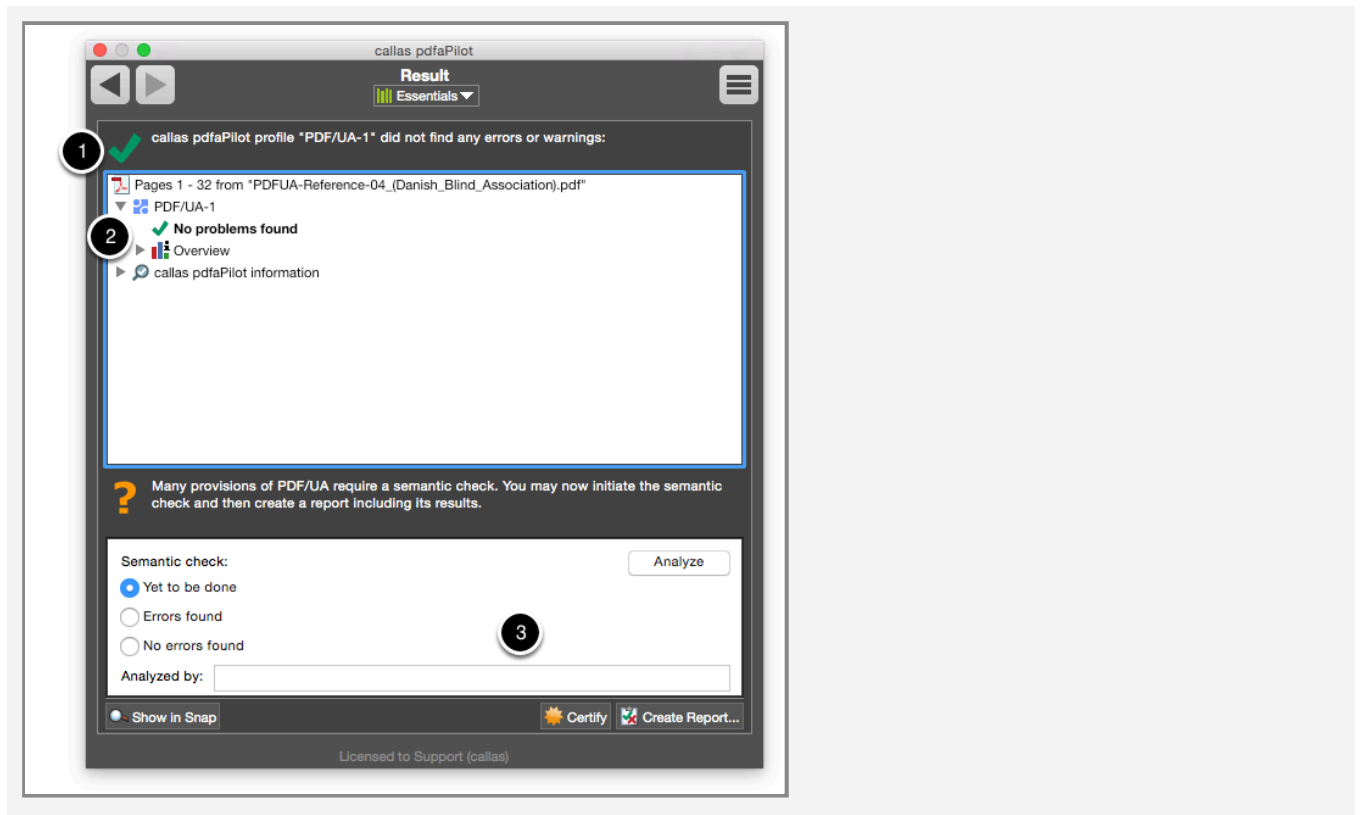
If you open a PDF/UA document, it will be identified with a PDF/UA-1 symbol in the document window (1).

As PDF/UA requires tags for structuring purposes, compliant PDF/UA files will also be marked with the corresponding structure symbol (2).

PDF/UA files generally also count as PDF/A documents with Level A (Accessible) compliance; in this case, the file is PDF/A-2a-compliant. This will also be identified with another symbol (3).

The user can validate PDF/UA compliance by clicking on the PDF/UA-1 symbol (1).

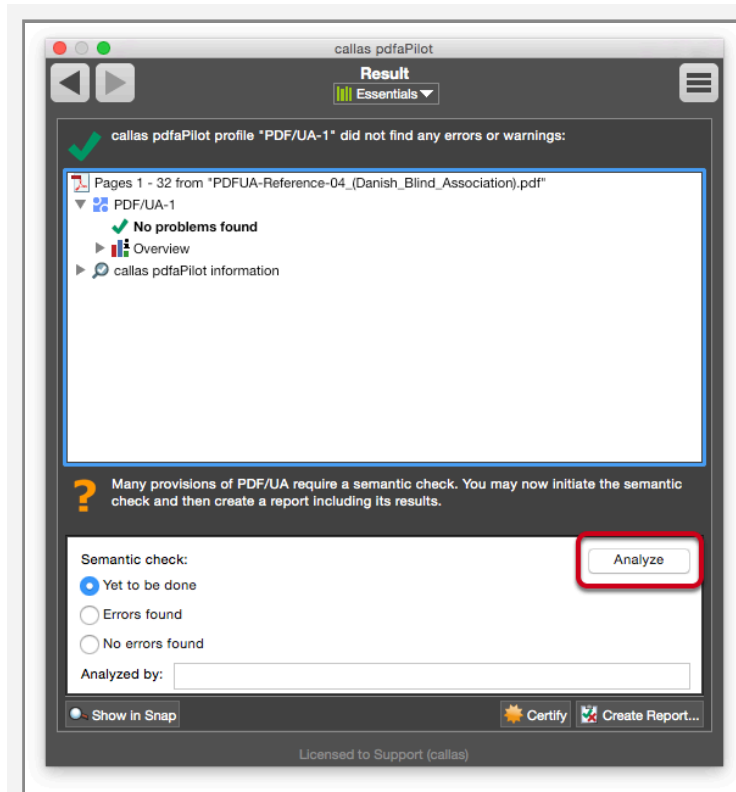
PDF/UA-1 validation - No problems found



If the existing PDF document really is PDF/UA compliant, this will be indicated with a green check mark (1) and the message **No problems found** (2) in the Result window.

The bottom of the Result window displays additional steps for a **Semantic check** (3), as not all PDF/UA validation steps can be fully automated.

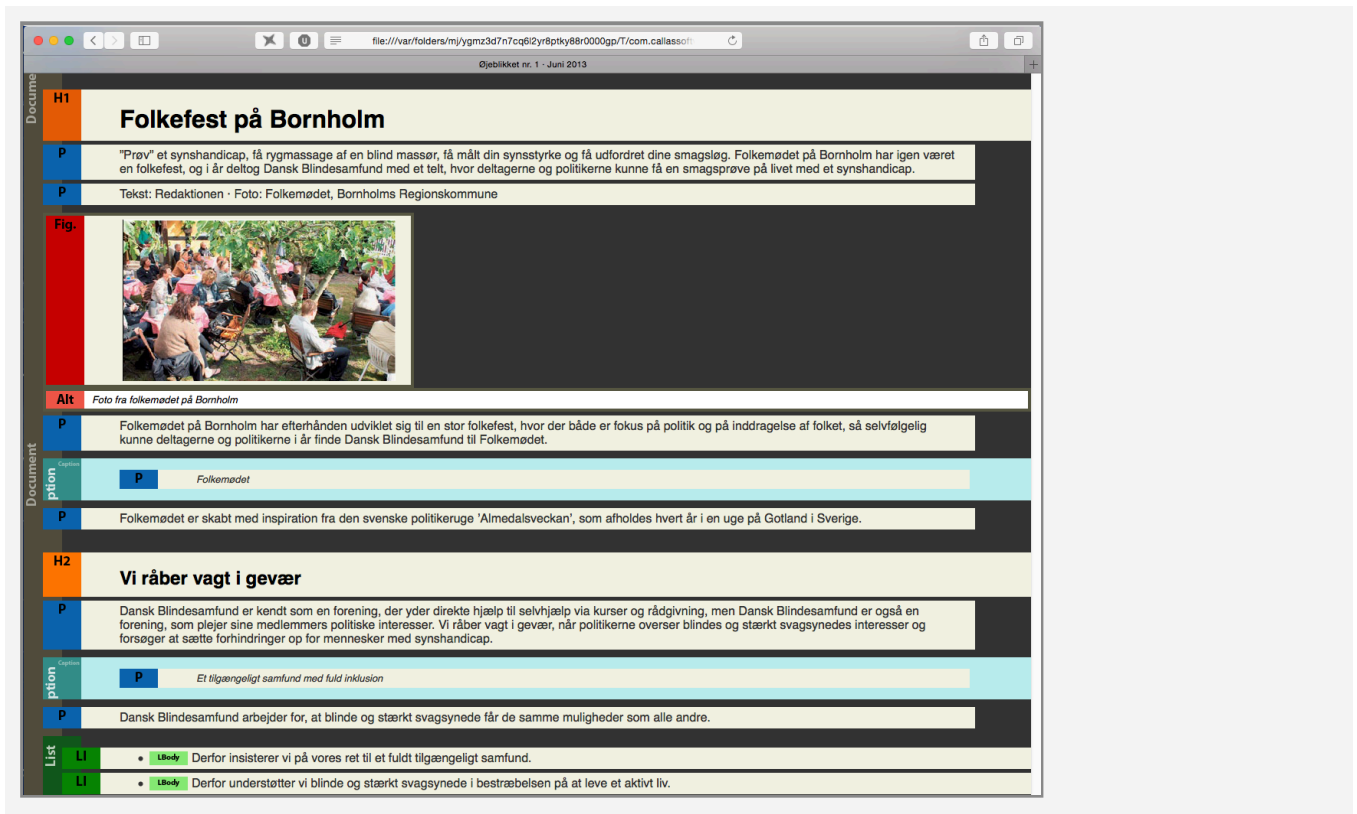
Options for semantic checks



Testing a PDF/UA file's accessibility cannot be done fully automatically. For example, a computer system cannot determine whether or not a document's structure makes sense. pdfToolbox provides a specific view to make it easier for users to complete the testing process.

First, click **Analyze**. This will open a window in your default browser showing the structure of the PDF document.

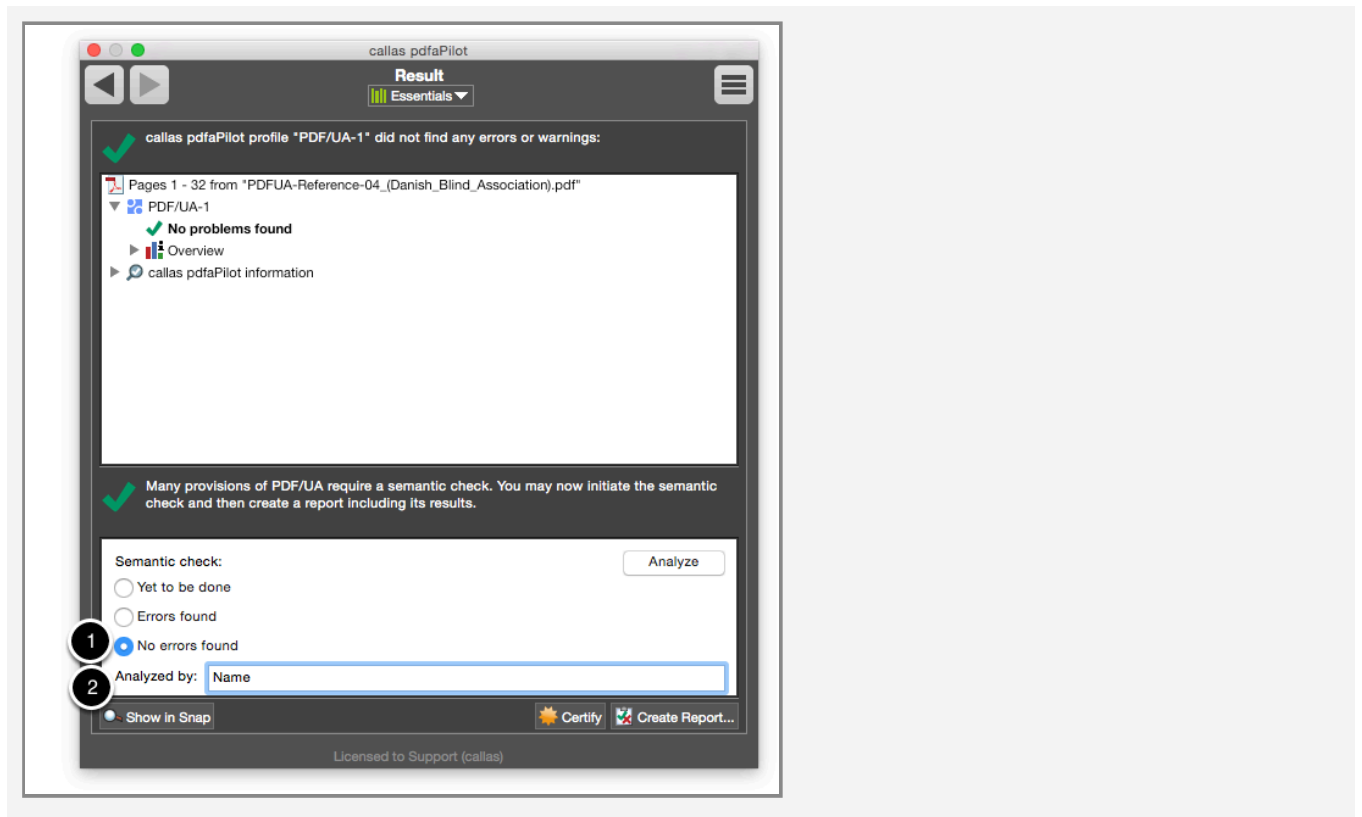
Semantic check using the PDF document's structural overview



The structure of the PDF file is visible in the browser and is indicated using colored highlights.

This makes it easy to check whether the order of chapters, headings and text is correct and to make sure that alternative text is in place for images as expected.

Semantic check - No errors

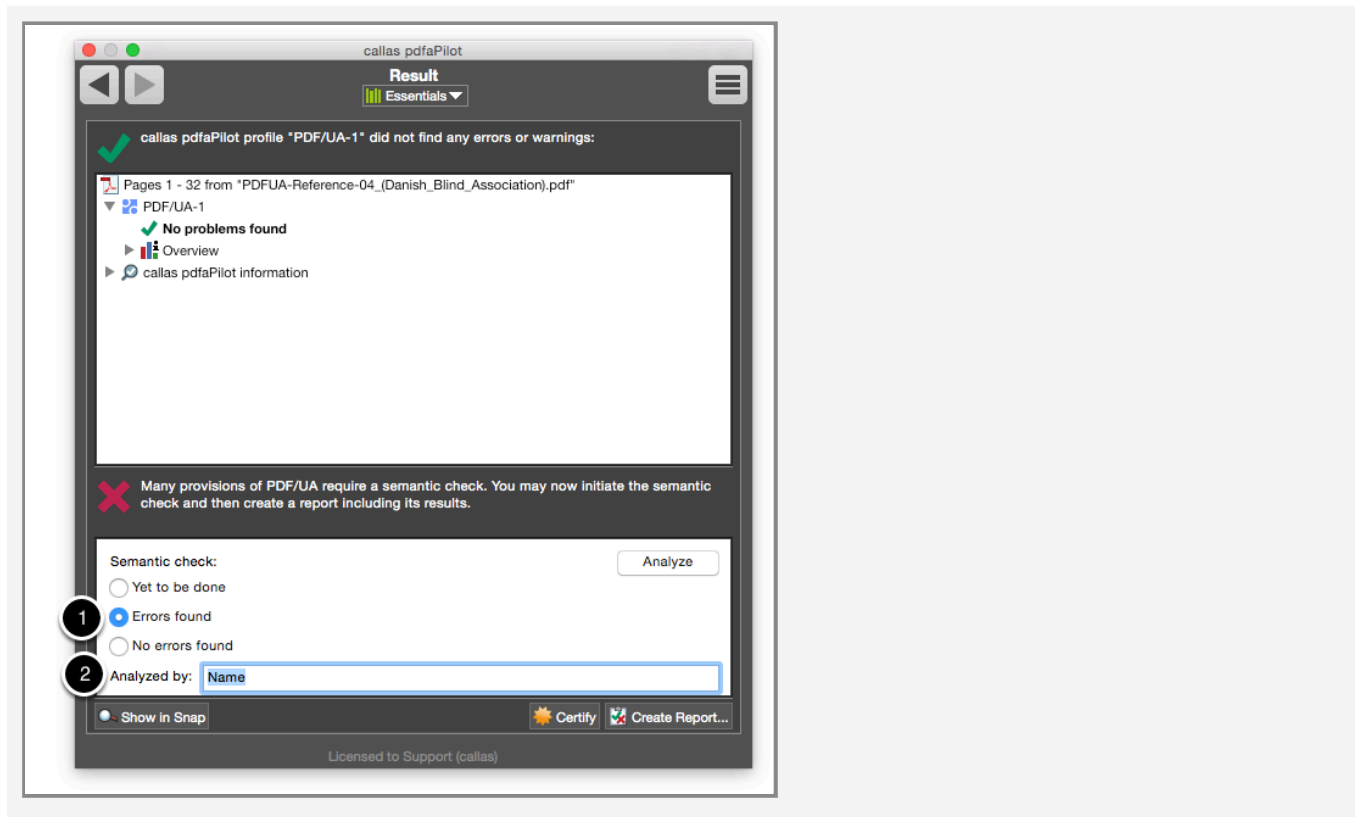


If the user identifies no errors after a semantic check, they can acknowledge this:

1. By selecting the “No errors found” radio field,
2. And by providing their own name in the “Analyzed by” field.

Users can also create a report which lists the results of the check.

Semantic check - Errors found

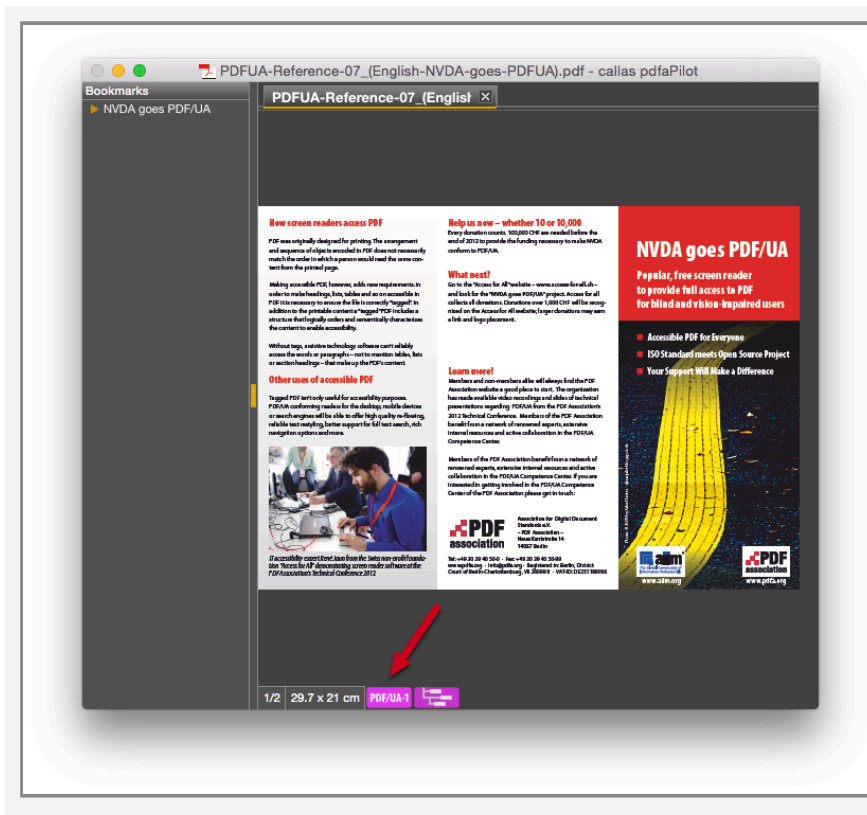


If problems are identified in the course of the visual inspection, the user can acknowledge this with the “Errors found” radio field (1).

In this case, the user can enter their name in the “Analyzed by” field (2).

They can then create a corresponding report containing this information.

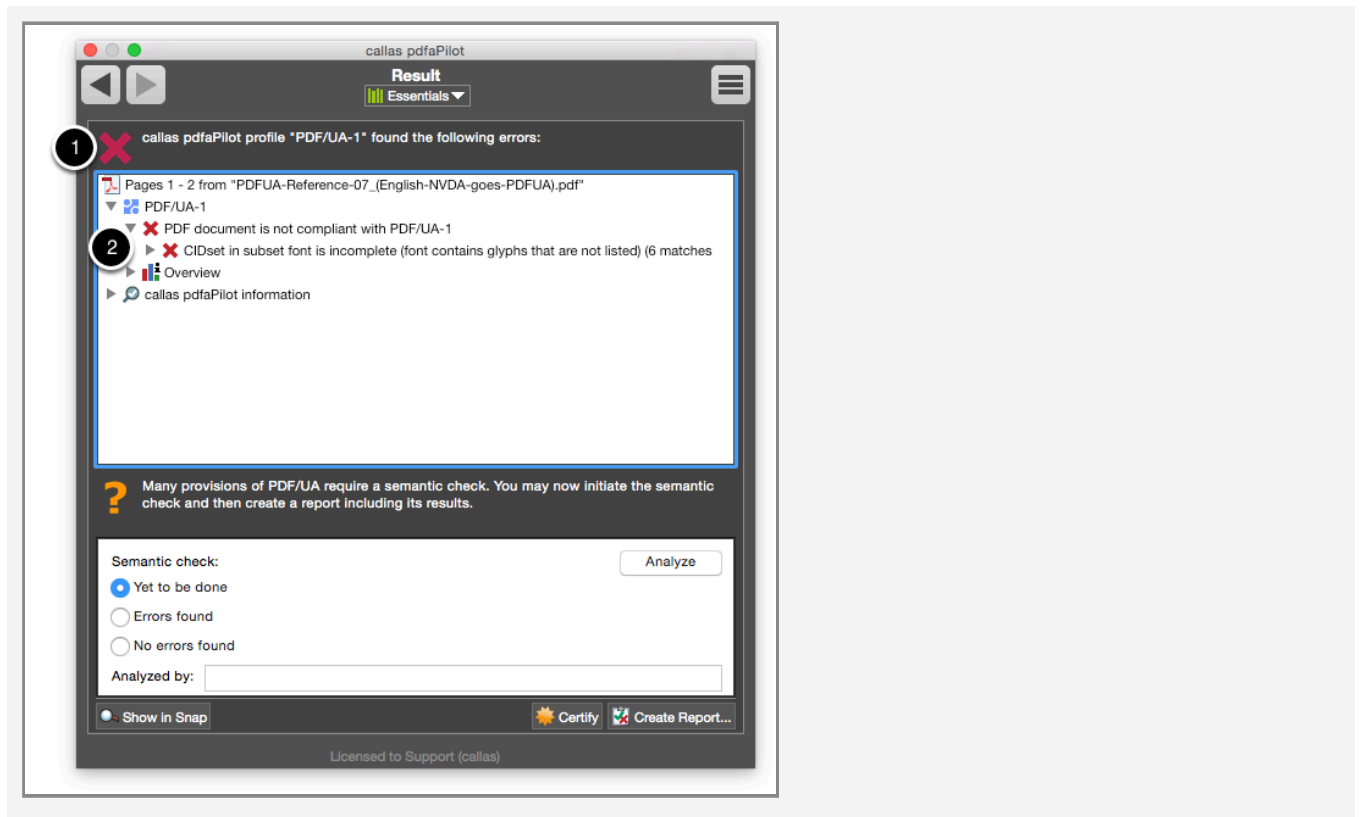
PDF/UA validation identifies problems



Not all PDF documents identified as PDF/UA-1 will successfully pass an accessibility test.

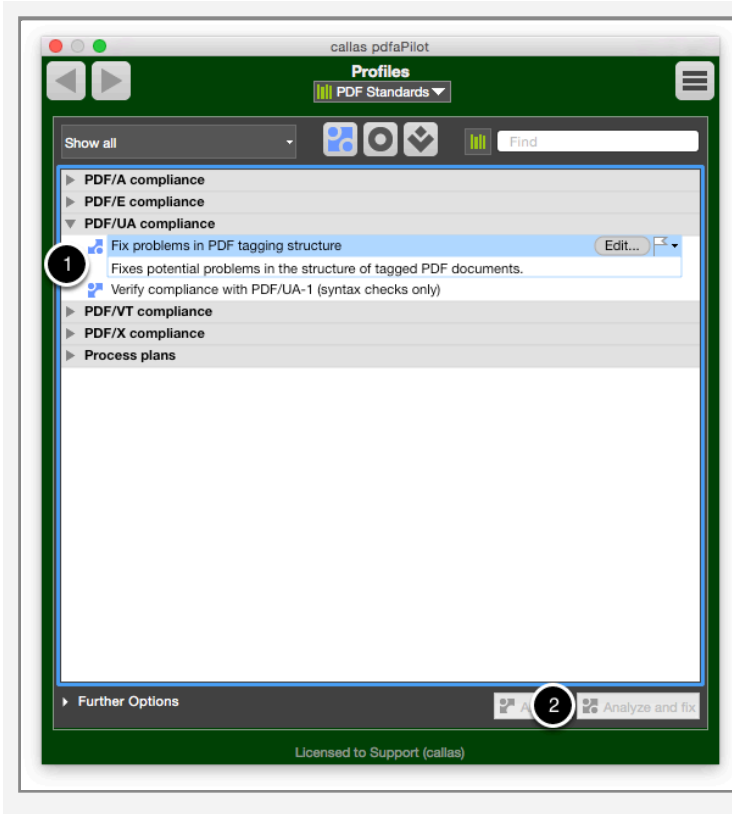
Click on the PDF/UA-1 symbol to begin the validation process.

Result window displays problems



1. The Result window will display a red X symbol to indicate that problems were encountered during validation.
2. In the example shown here, the Detail view indicates that there is a font-related problem.

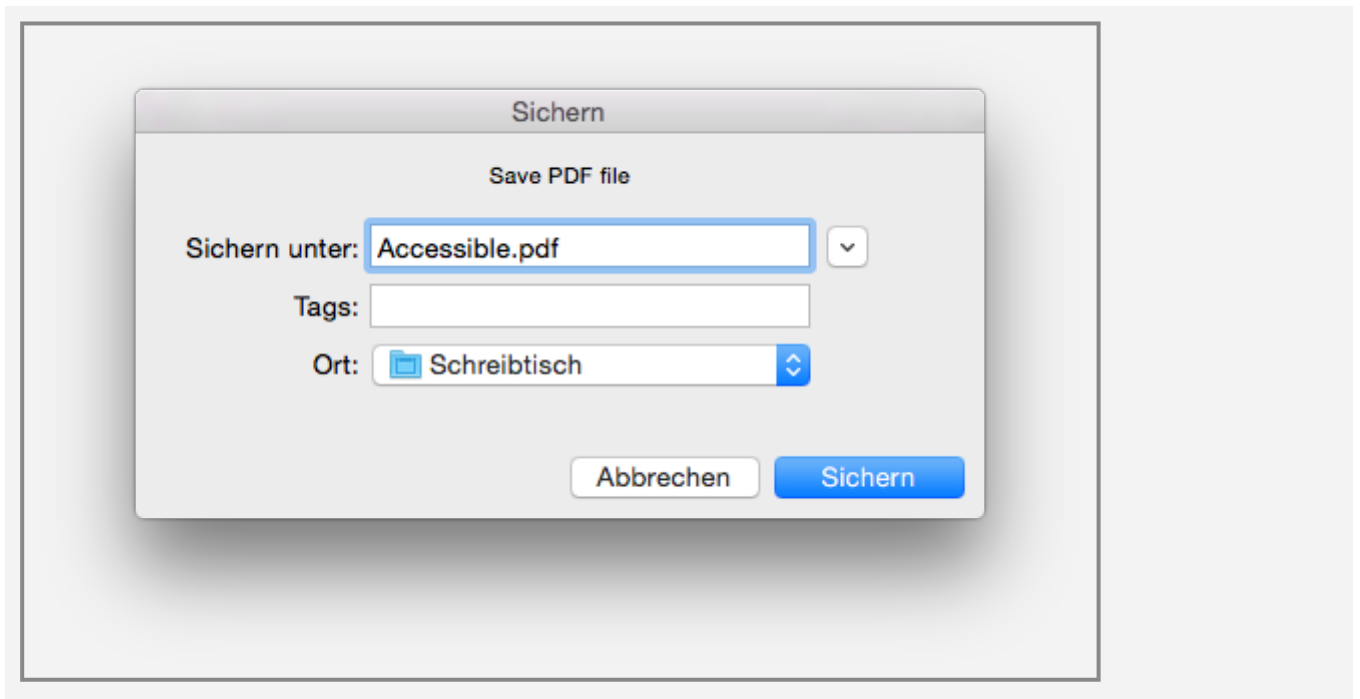
Resolving problems with PDF/UA-1 files (as of pdfToolbox 8.3)



The software can resolve some common problems related to PDF/UA files.

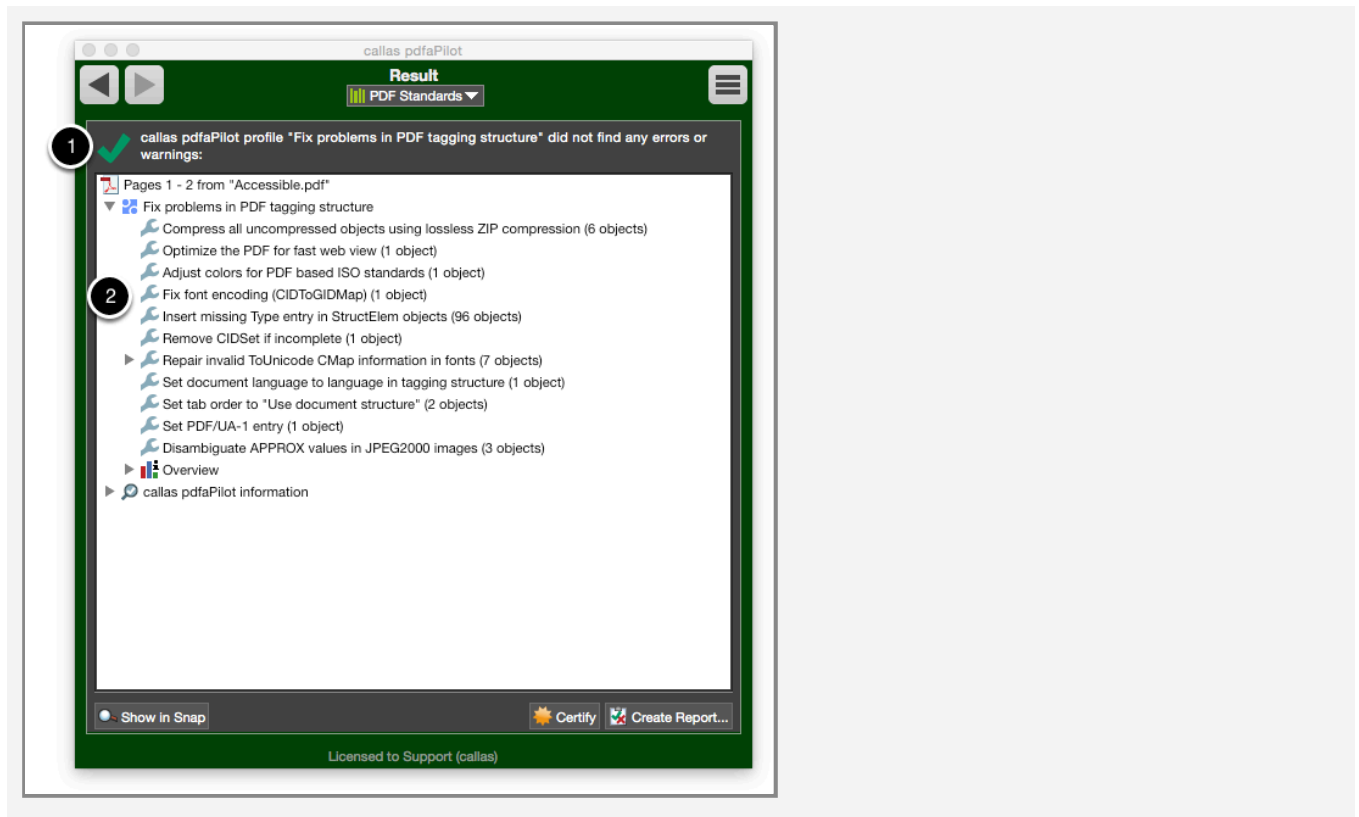
1. To do so, switch to the “Profile” tool (Cmd+3) and open “Resolve structural problems with tagged PDF documents” under the PDF/UA section (to do this, you may first need to switch to the “PDF Standards” library.)
2. Click “Check and fixup” to begin the Fixup process.

Copy document



The user has the option of first saving the PDF document under a new name by clicking **Save as copy**.

Results after correcting a PDF/UA document



1. If the Fixup has been successful, this will be indicated with a green check mark in the Result window.
2. The Detail view will indicate which Fixups were applied.

Users can then carry out the [Semantic check](#) as described previously.

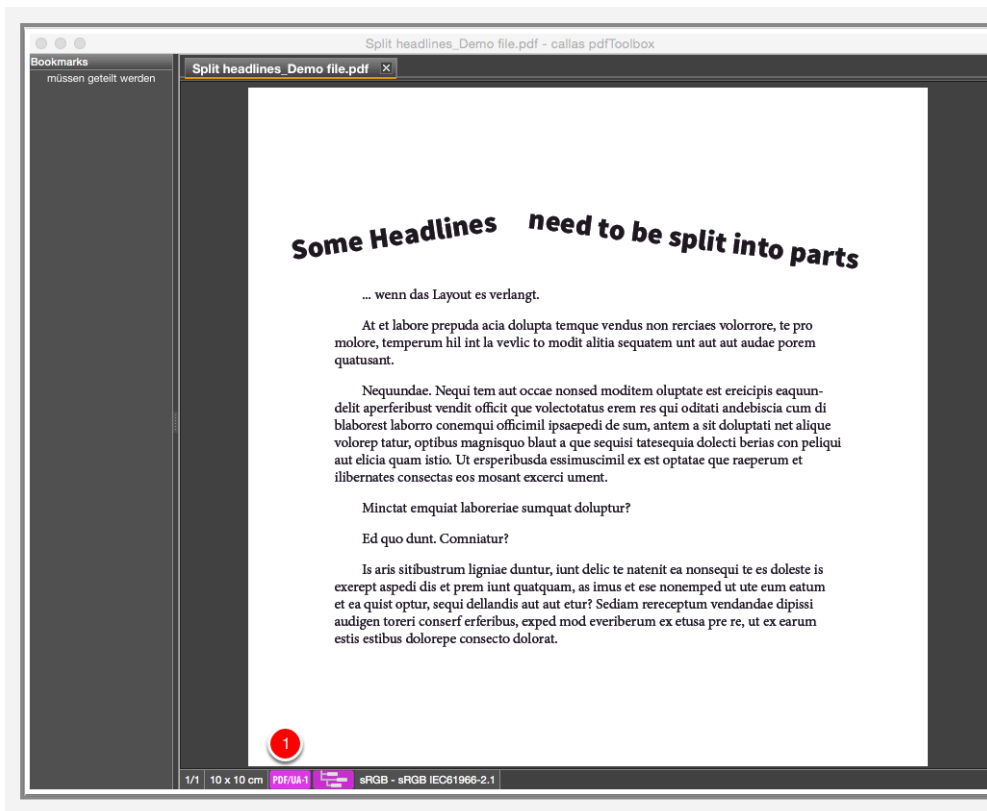
2.16 Improved merging of adjacent heading (H1..H6) tags

For accessible PDF (PDF/UA) (clean structured PDF) and for PDF files with "invisible" structure information, it is important that coherent texts are also displayed contiguously in the PDF structure. This does not always happen in the creation of the PDF.



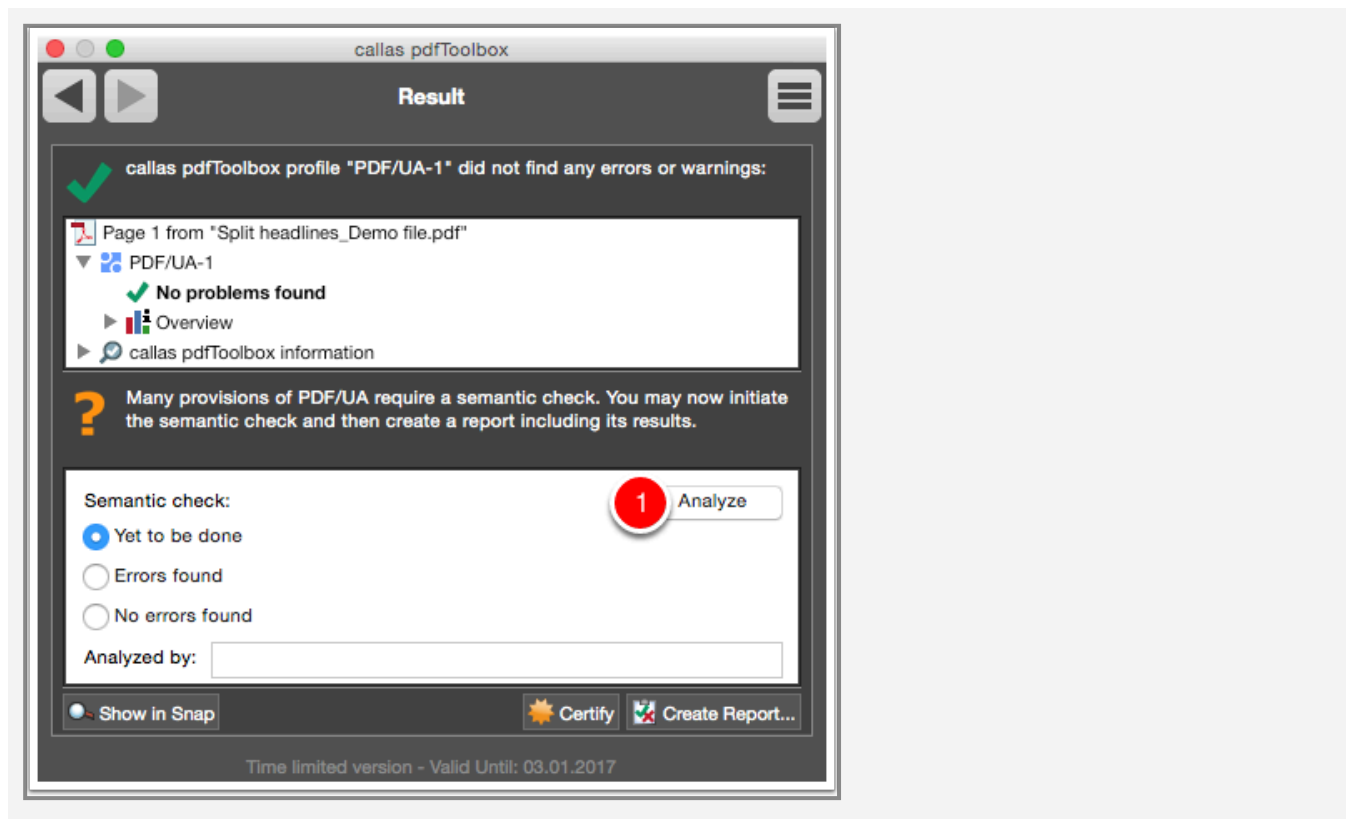
Split_headlines_Demo_file.pdf

The following PDF file contains a heading, which consists of two components.

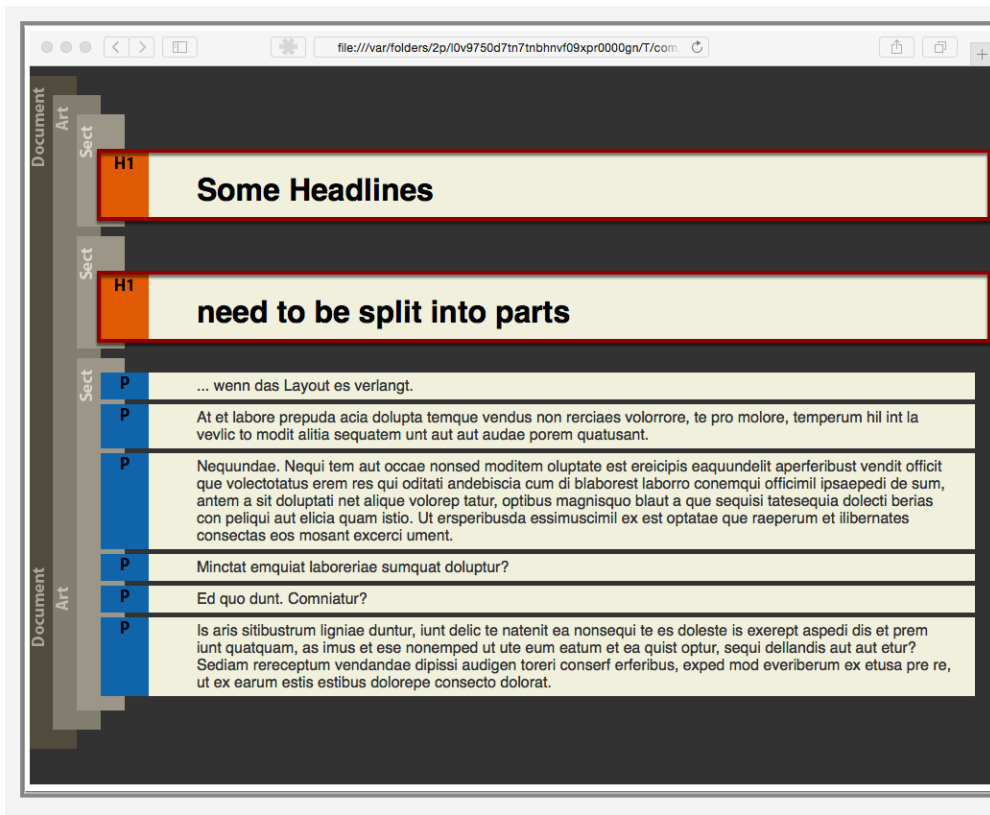


By clicking on the PDF/UA-1 button below, the result window will open to start the validation of the structure. No problems are found during the validation. However, this is not yet sufficient for PDF/UA, which always requires an inter-

active analysis as well. For this, the "Semantic check" must be started. To do this, click on "Analyze".



Analyze the PDF/UA-1 file structure



During the analysis, the structure information of the PDF file is displayed. As you can see from the screenshot, the headline was split to two H1 "tags". To fix this problem, a predefined Fixup can be applied.

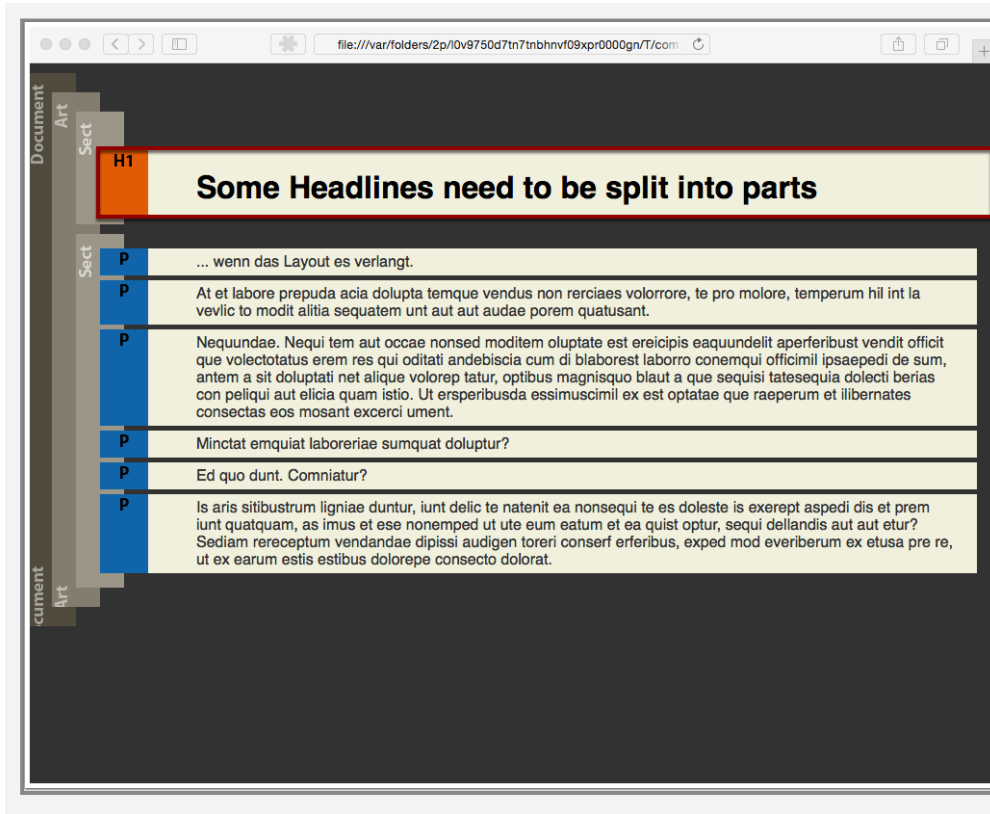
Fixup "Merge adjacent headings if heading level is the same"



After you run this Fixup and saved the PDF file, click the PDF/UA button on the bottom menu bar again. Then execute PDF/UA validation again and click "Analyze" afterwards to view the structure.

Result: Analyze the PDF/UA-1 file structure

The previously separate parts of the header are now connected in one structure tag.

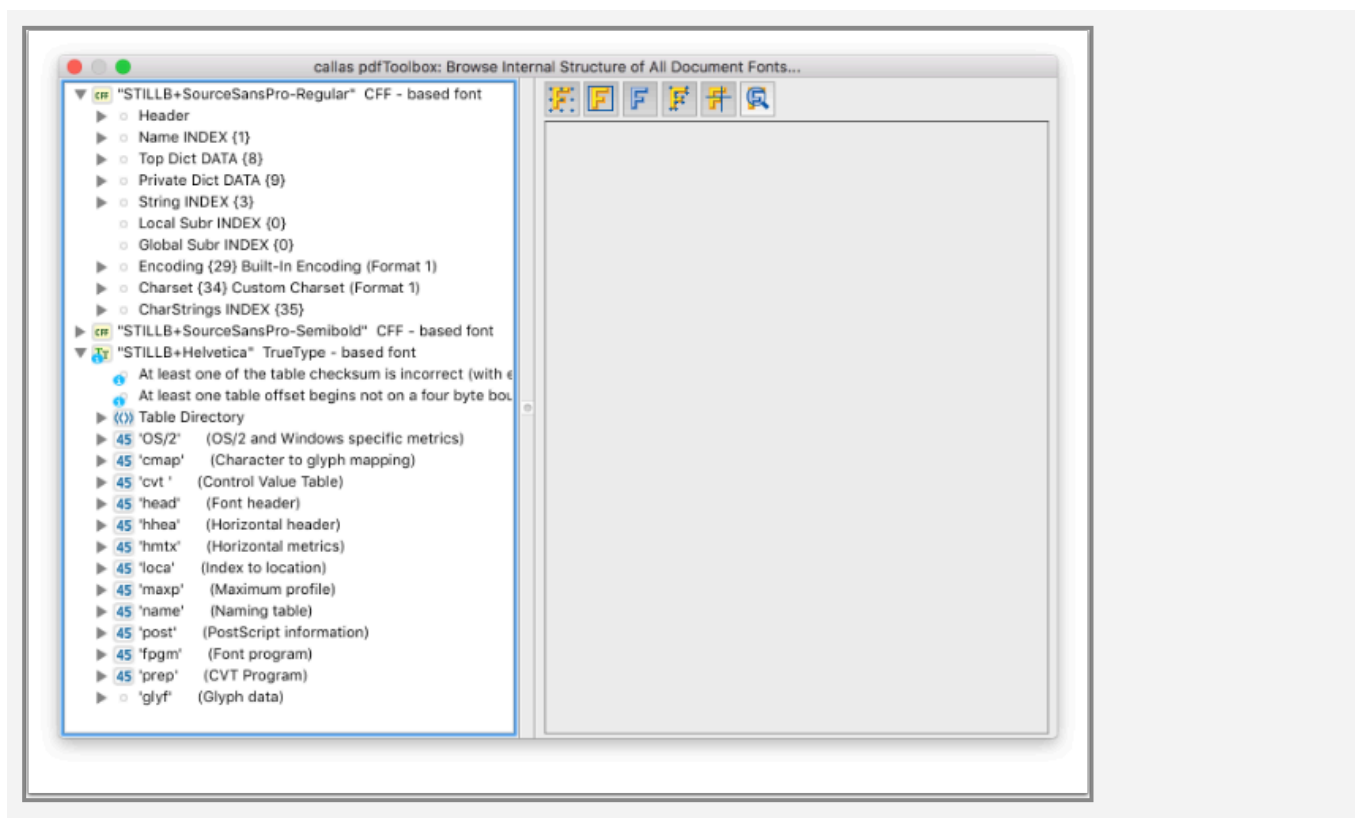


2.17 Explore Fonts

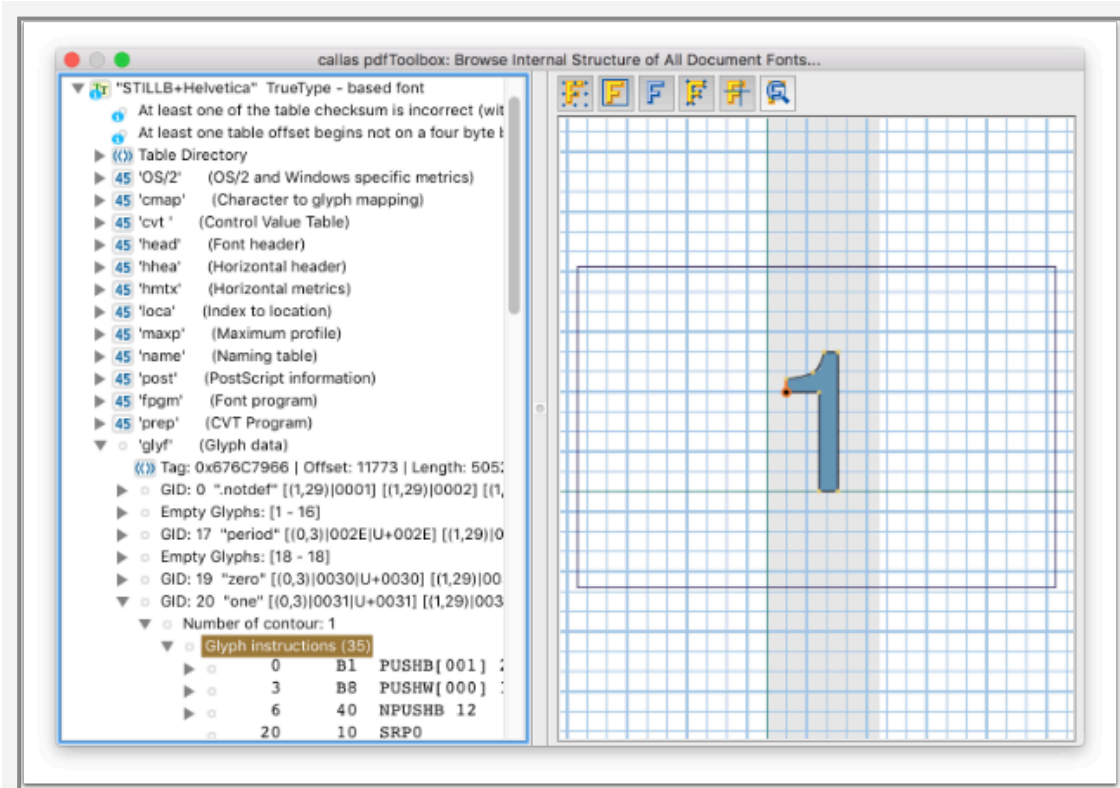
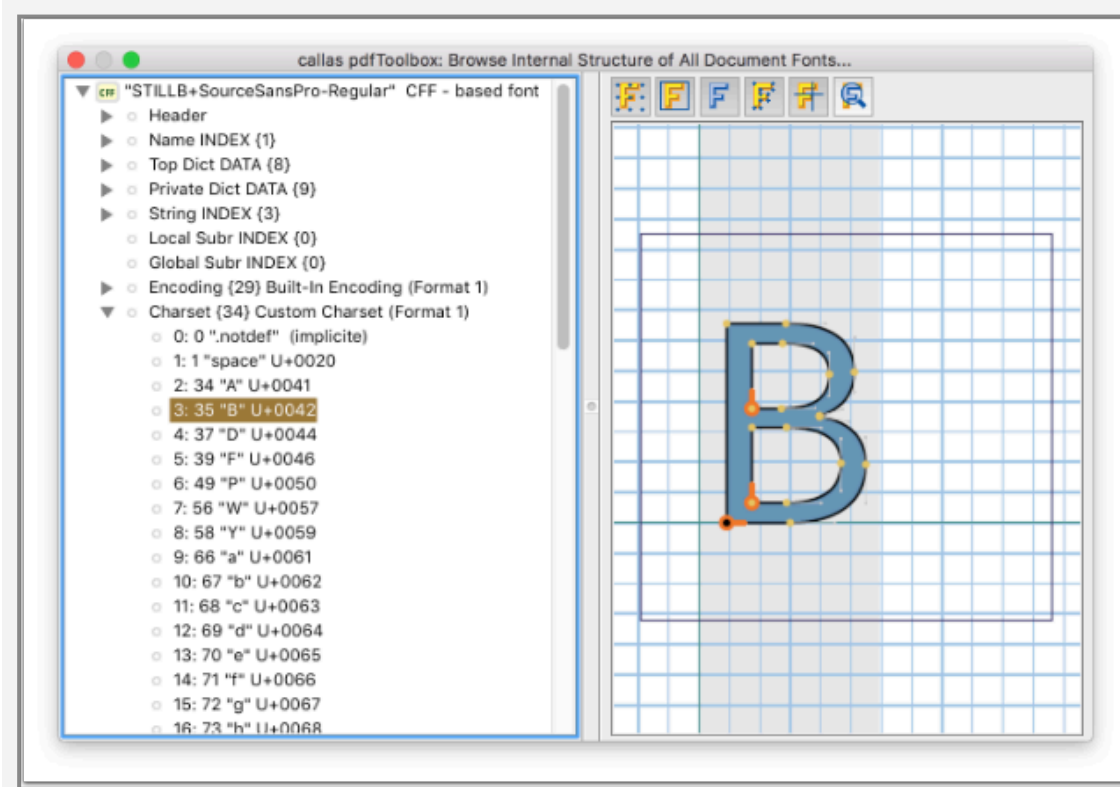
The "Explore Fonts" function gives you an insight into the internal font structure of PDF files. This is usually not needed for common use, but might be helpful if you encounter a damaged file or if you simply are interested in learning more about the internal font structure. The entry "Explore Fonts..." can be found inside the "Plug-Ins" menu of Acrobat ("Miscellaneous") or the "Tools" menu in the Standalone version.

In the "Explore Fonts" dialog you will find information about e.g. font type and embedding state of all fonts present in the current document.

Depending on the type of font (e.g. TrueType, Type1, ...), the way how information of the font is shown is different:



Also, painting information about the contained glyphs of embedded fonts will be displayed:



2.18 Explore PDF

The "Explore PDF" functionality gives you an insight into the internal structure of PDF files.

Usually, this is not needed for common use, but might be helpful if you encounter a damaged file or if you are simply interested in learning more about the internal structure of a PDF file. The entry "Explore PDF..." can be found in the "Plug-Ins" menu of Acrobat ("Miscellaneous") or the "Tools" menu in the Standalone version.

With the "Explore PDF" tool you can have a look into the data structure of a PDF file, exposing the several commands and details that are forming the page objects.

Document Structure view

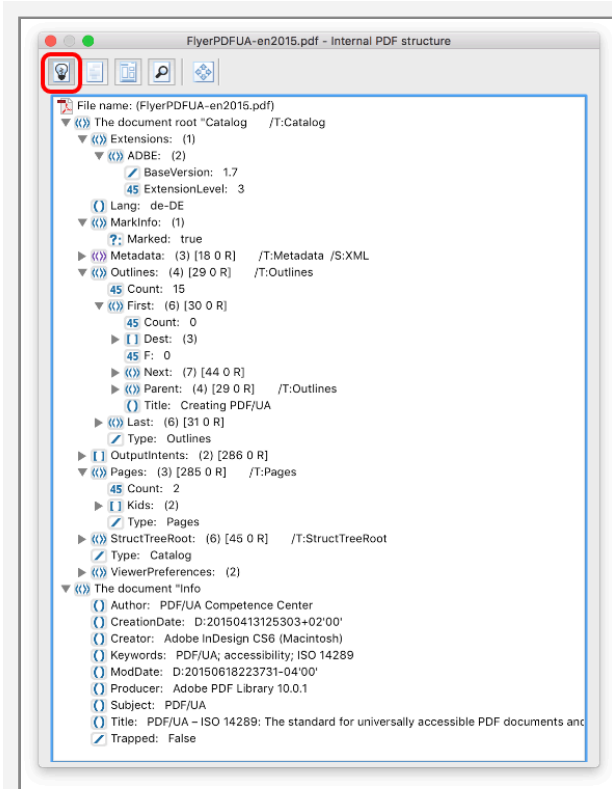
The first button opens the "Document Structure" view, which lists 2 items:

- The document root "Catalog"

The root of a document's object hierarchy is the "Catalog" dictionary. The catalog contains references to other objects, defining the document's contents, outlines, article threads, named destinations and other attributes. In addition, it contains information about how the document shall be displayed on screen, such as its outline and thumbnail page images shall be displayed automatically and whether some location other than the first page shall be shown when the document is opened.

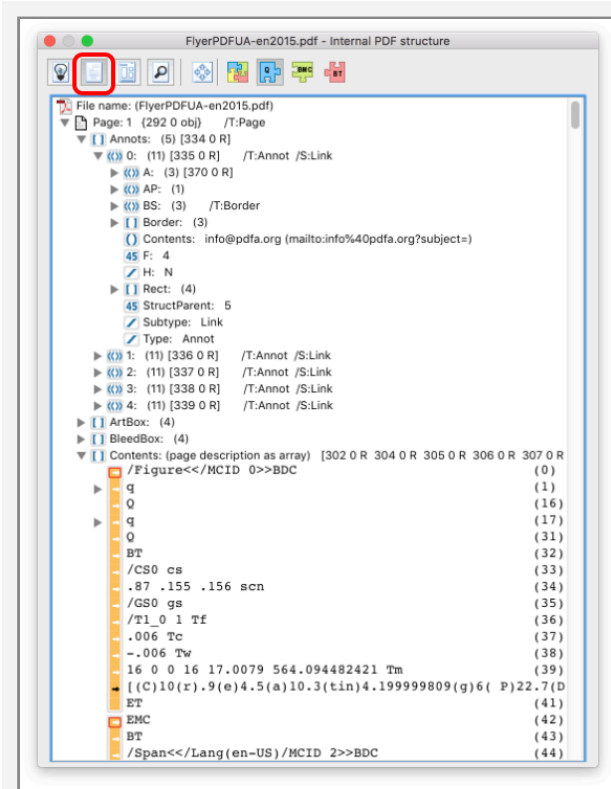
- The document info

The document info area lists some basic information about the file, like title, author, creation date, producer, creator, keywords and so on



Logical Structure view

While the "Document Structure" view contains the complete view of the documents content, the "Logical Structure" view offers a page-by-page view of the different properties of a page like page geometry boxes, used resources, content stream and more as well as other optional attributes like annotations or thumbnails.

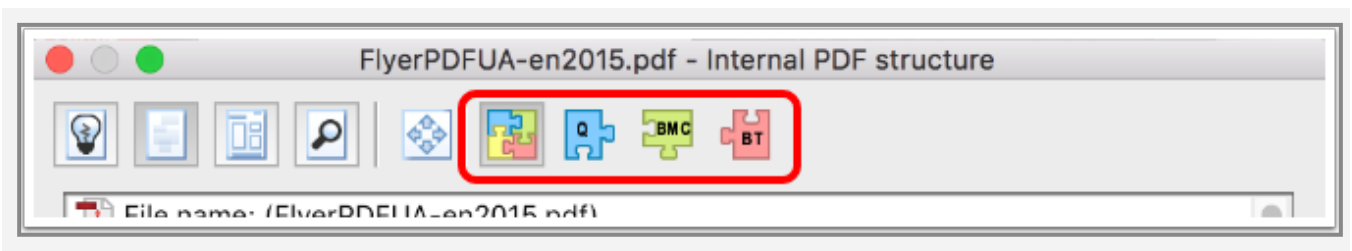


Different views of the content stream

The "Logical Structure" view offers 4 different representations of the content stream:

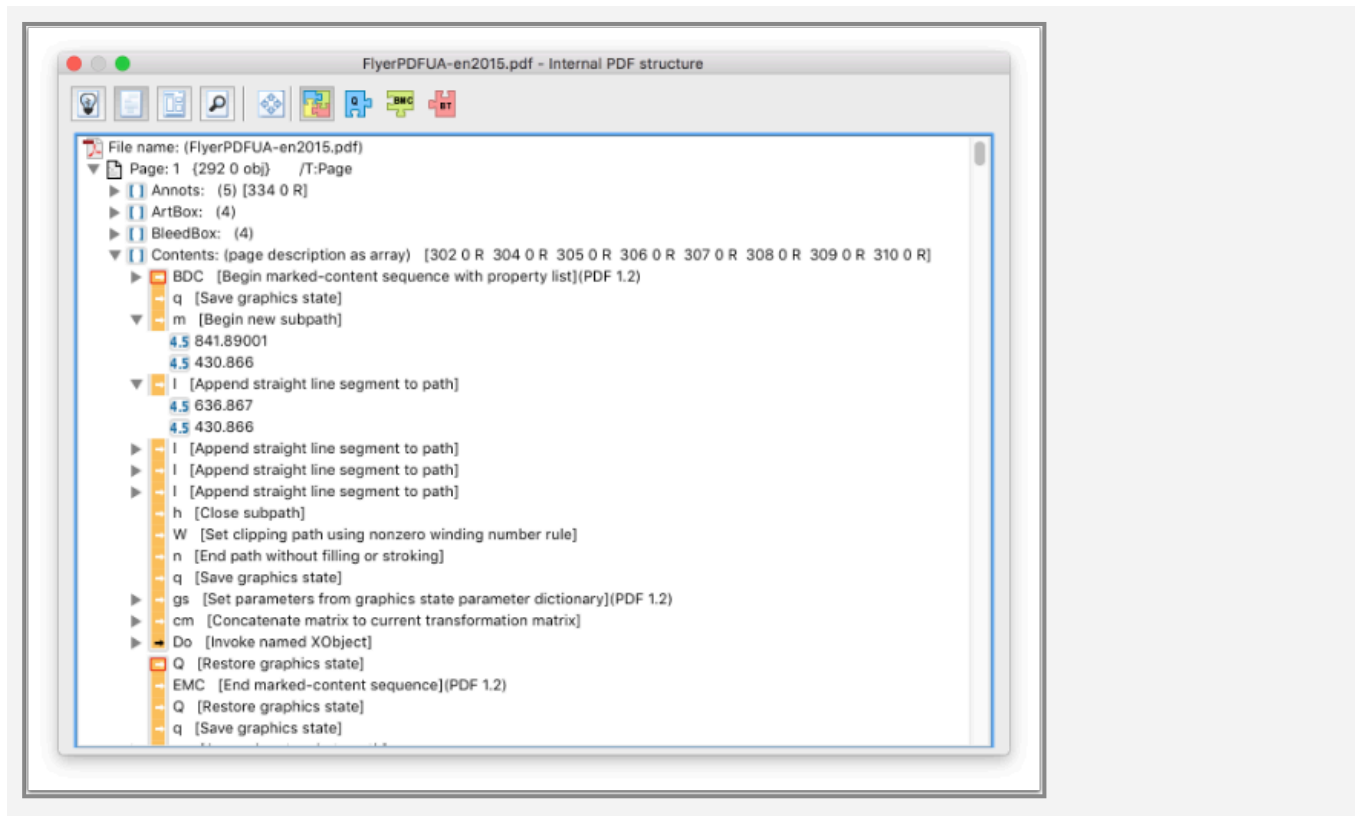
- Content Stream snippets: explained
- Content Stream snippets: q/Q pairs
- Content Stream snippets: Marked content
- Content Stream snippets: text

These views can be selected using the 4 colored buttons on the right of the selection bar:



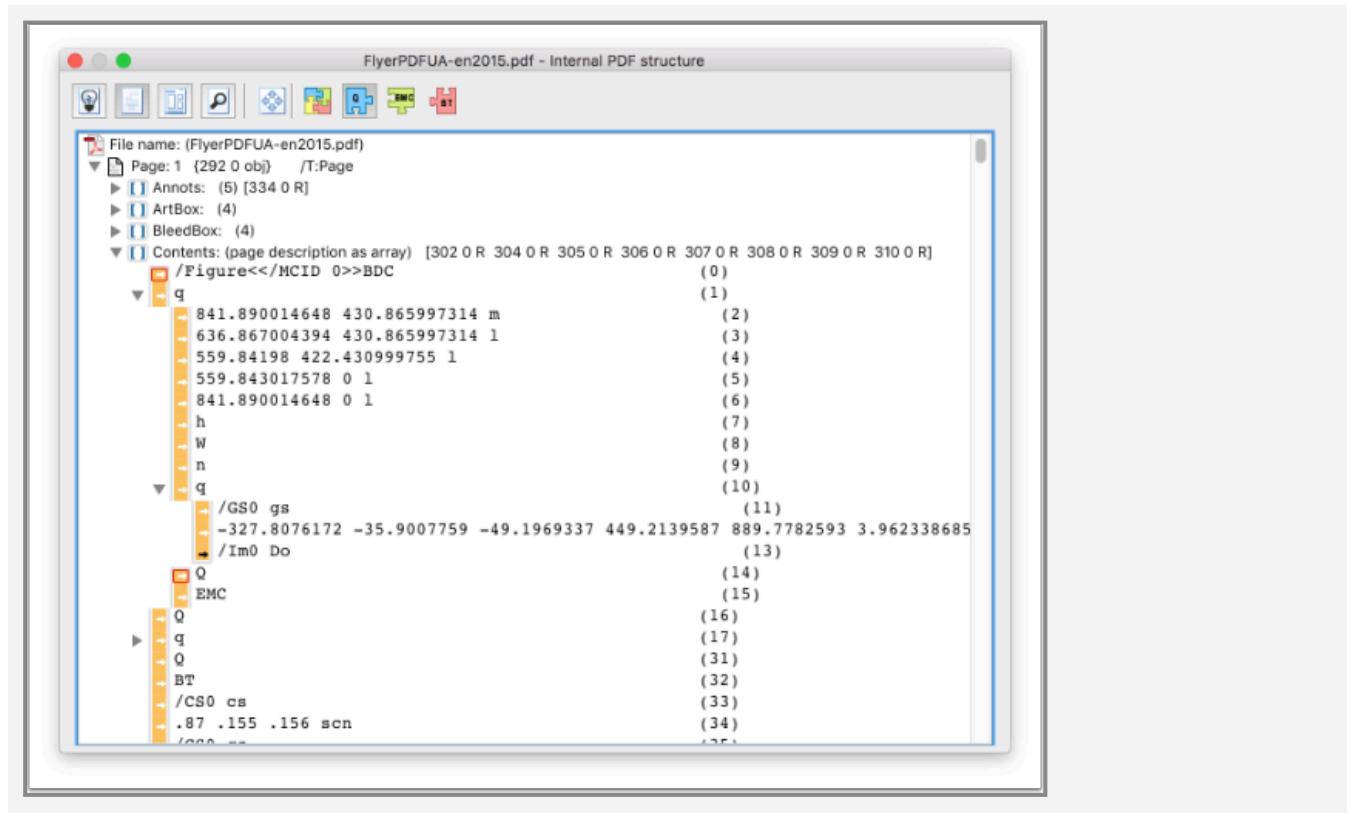
Content Stream snippets: explained

This mode shows minute explanations for all painting sequences of the content stream and makes it easy to understand the way the content of a page is composed step by step.



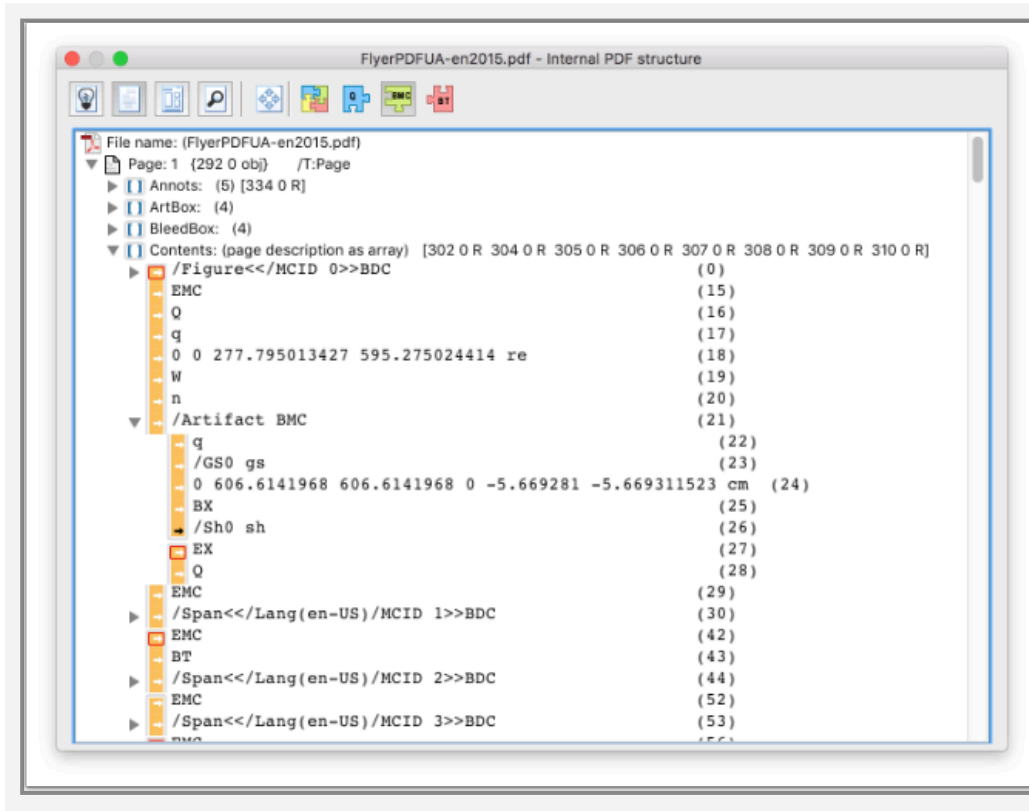
Content Stream snippets: q/Q pairs

The "q/Q pairs" view shows the content stream in a more condensed way, as the painting sequences are sorted in their respective graphic state nesting, also known as q/Q pairs.



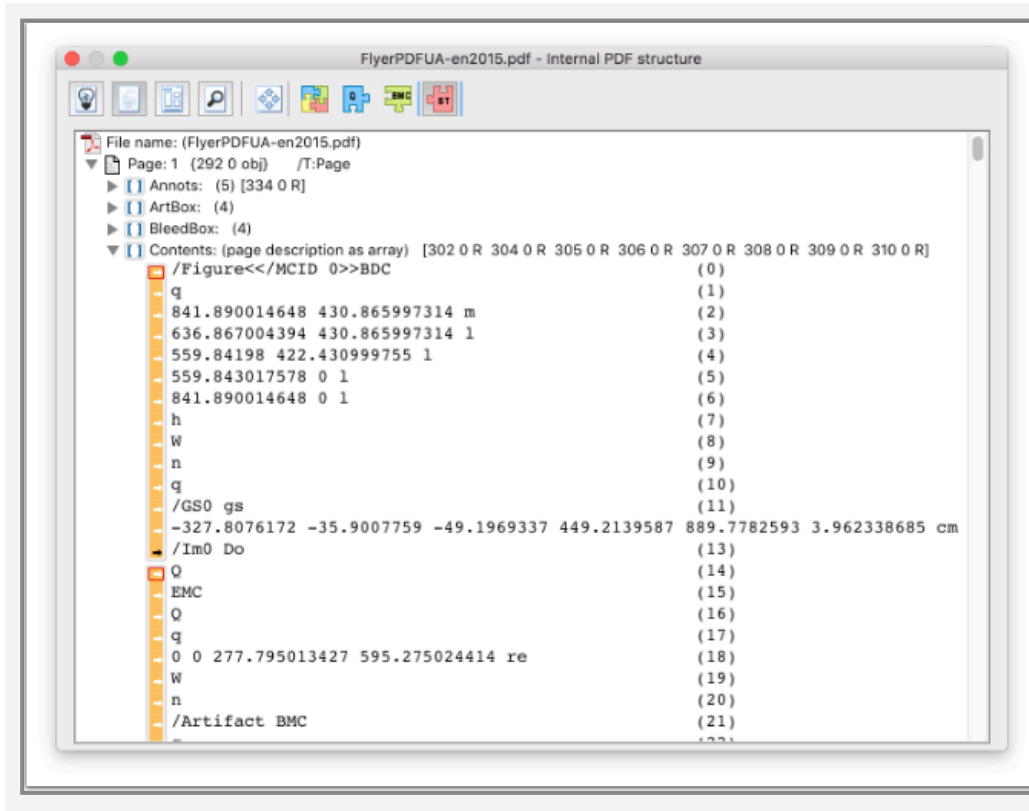
Content Stream snippets: Marked content

The "Marked content" view shows the content stream from the tagging or marked content perspective, as the content stream is grouped by the respective BMC or BDC properties.



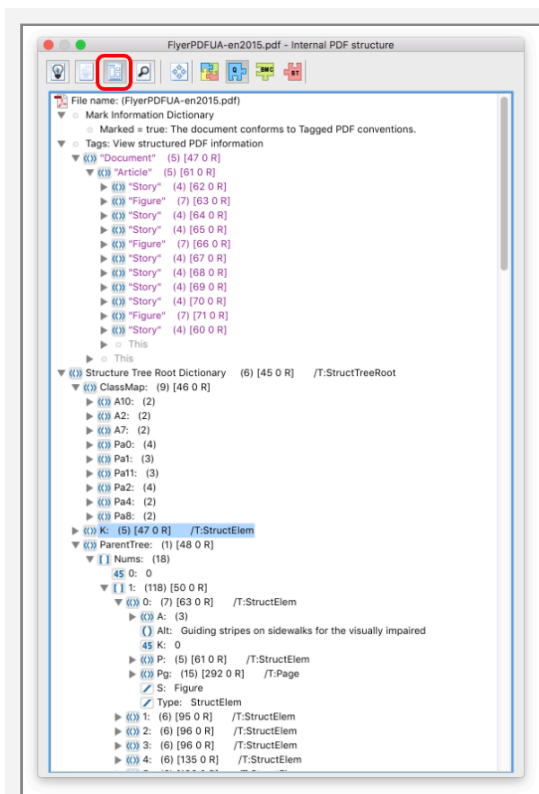
Content Stream snippets: text

The fourth view offers a plain text view of the content stream in a readable fashion.



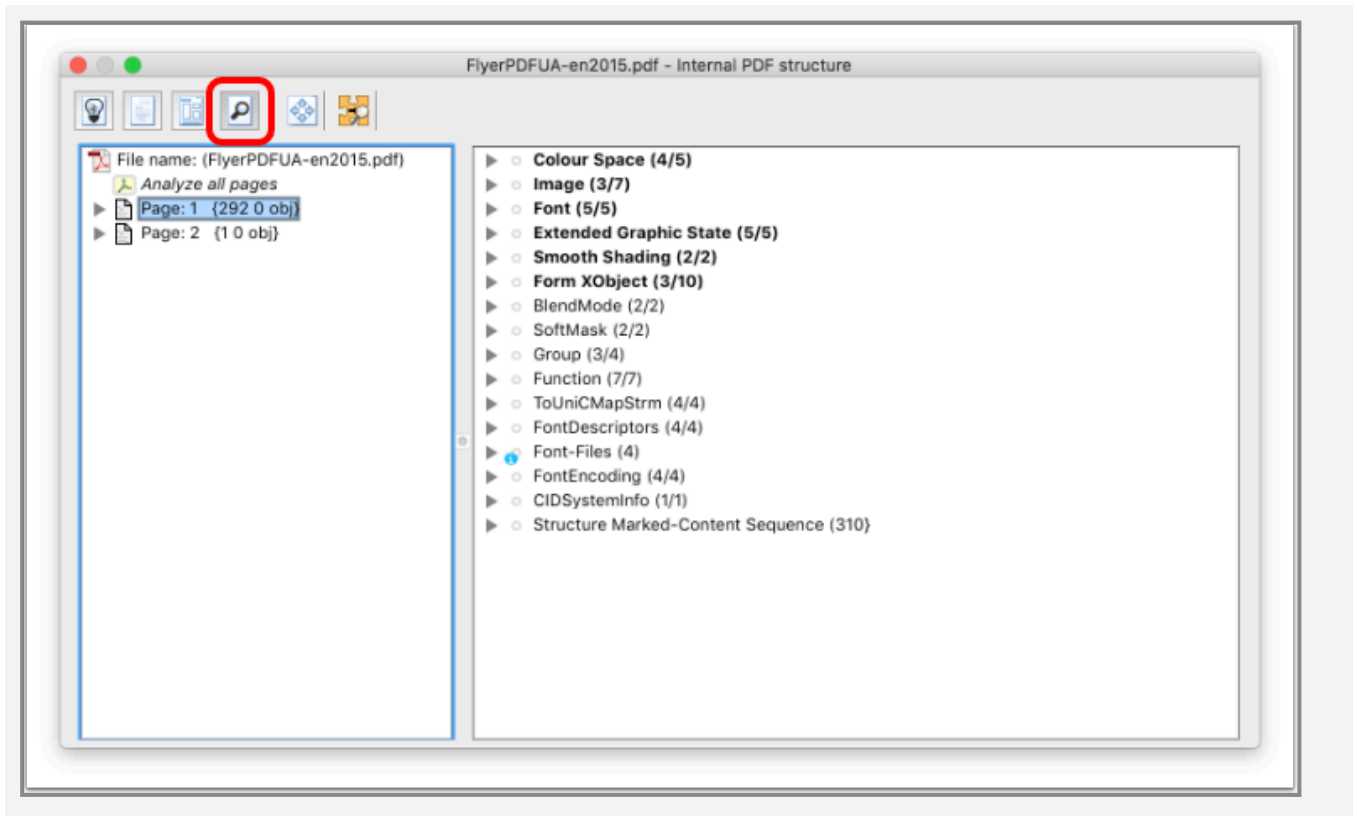
Tagging Structure view

When a PDF file contains tagging information, this view gives a detailed overview about all the details like the ClassMap, a structural view of the various tags as well as other interesting details.



Resource view

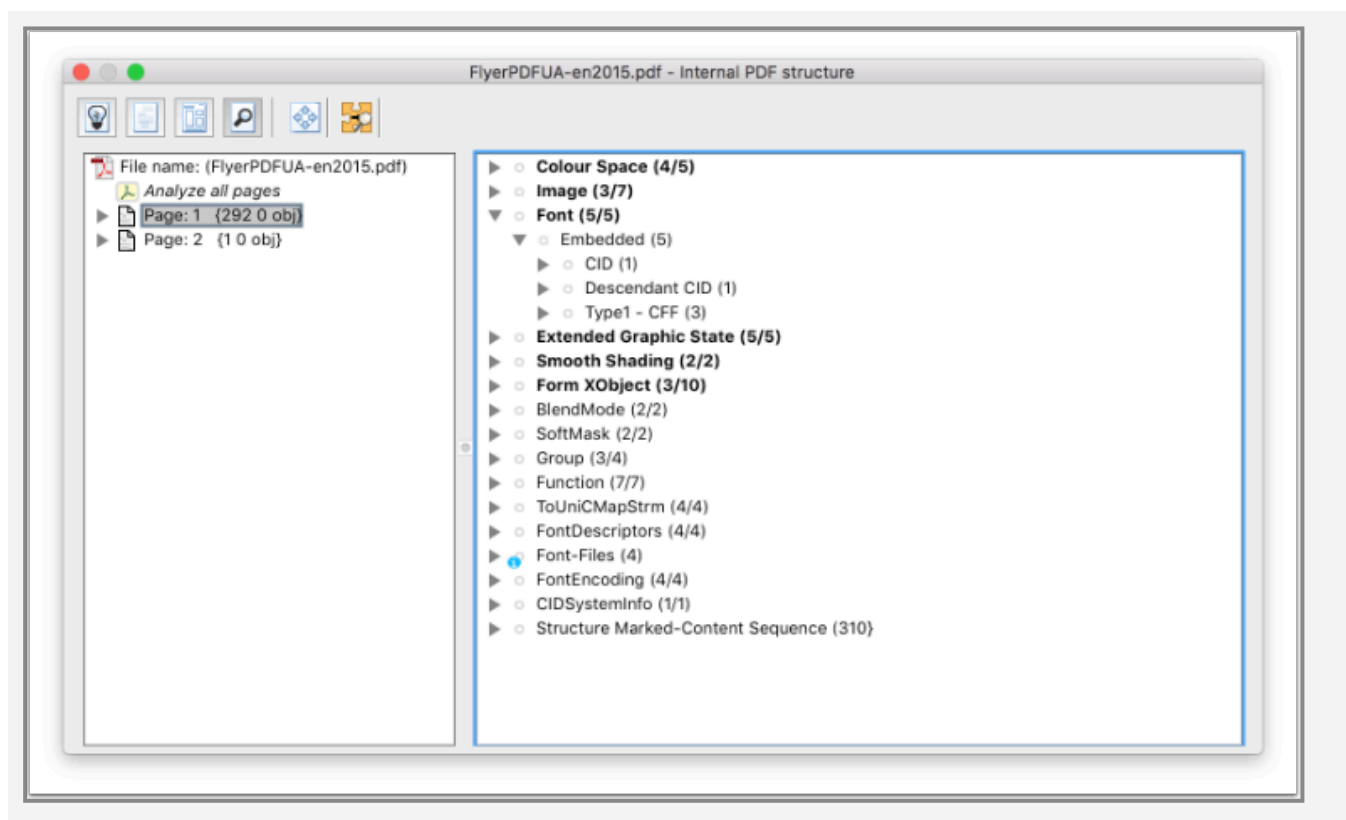
While the Logical view gives you (amongst others) a view into the content stream, the "Resource" view offers a page-by-page listing of all painting objects (like images, shadings, text, vector objects, ...).



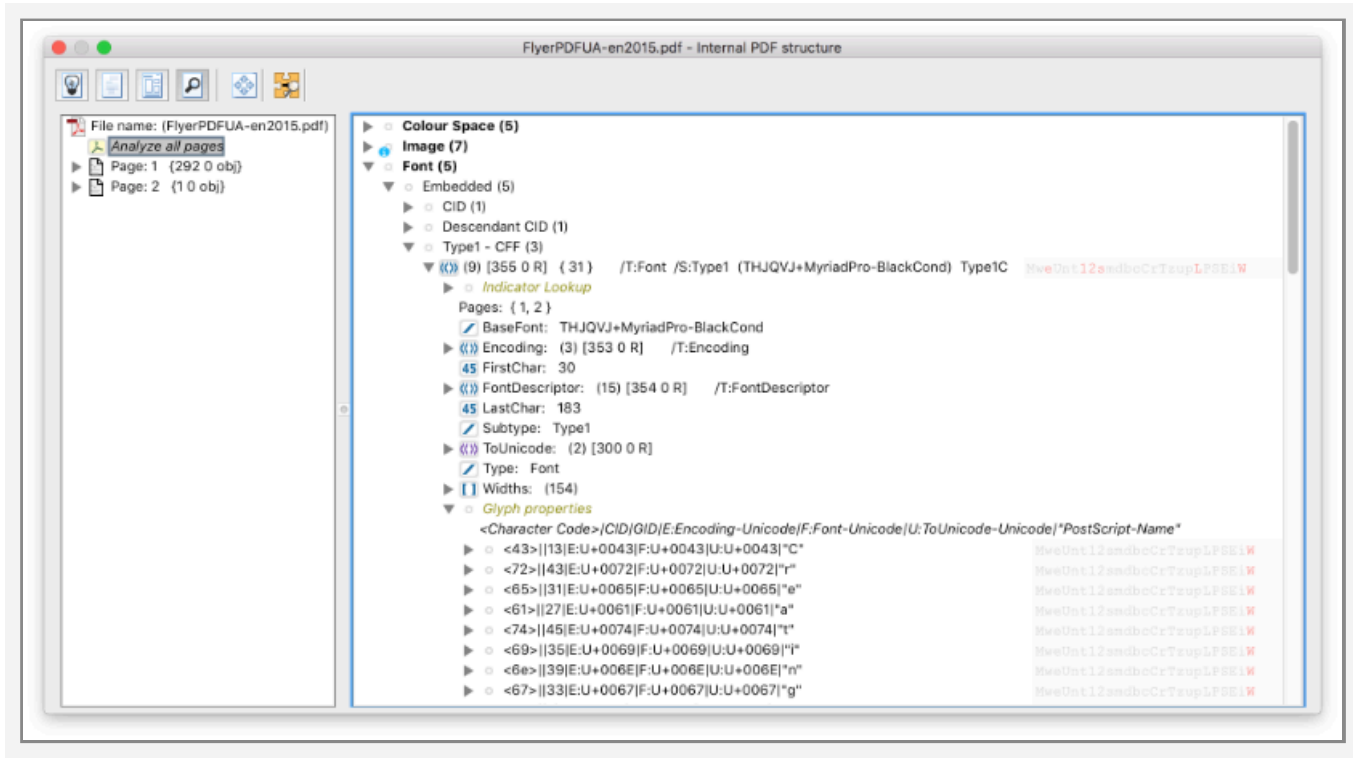
Each of the resources has a specific substructure with further information. Also all resources are listed that are used on the selected page, independent from whether they are specified in the page resource or in Form XObject resources.

The "Font" section is specifically rich with many results from pdfToolbox' font engine.

Detailed glyph information for embedded fonts



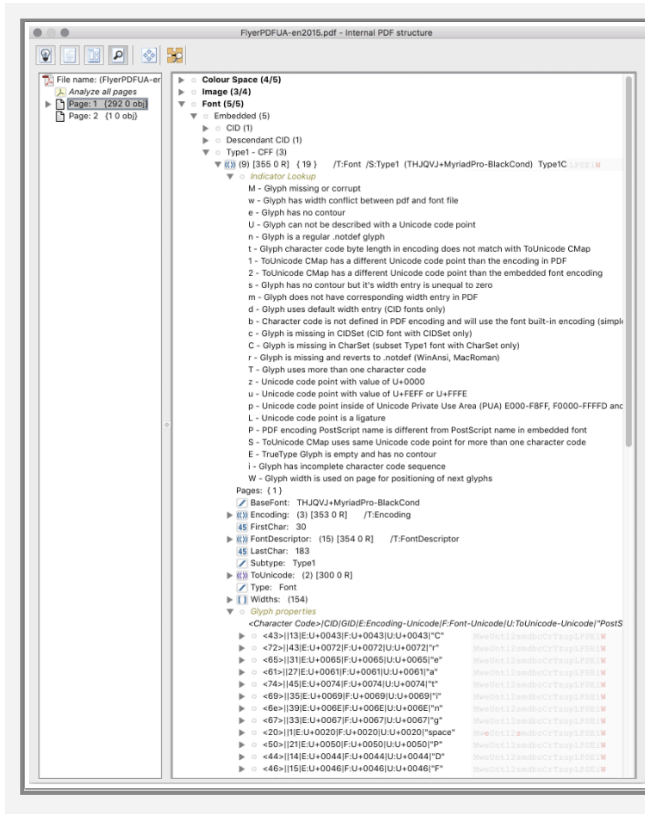
The font section shows embedded and not embedded fonts and then font types. The results of the font engine are available for the whole font and for each glyph by a list of indicators behind the respective entries.



When selecting a font, a lot of detailed information about the font file itself and the contained glyphs is available. Depending on if the selection on the left pane is on a specific page or on "Analyze all pages", the fonts used on that page or in the whole document are listed.

For all glyphs of an embedded font, there are several indicators behind each glyph. If such an indicator is red, this means that the corresponding property of the indicator applies to the glyph. This does not have to be a problem right away, it can help to make the different properties of the glyphs quickly accessible.

In this example for almost all glyphs a capital "W" and for some glyphs, also "e" and "s" are active - as indicated by the indicator being red. The section "Indicator lookup" explains the indicators.

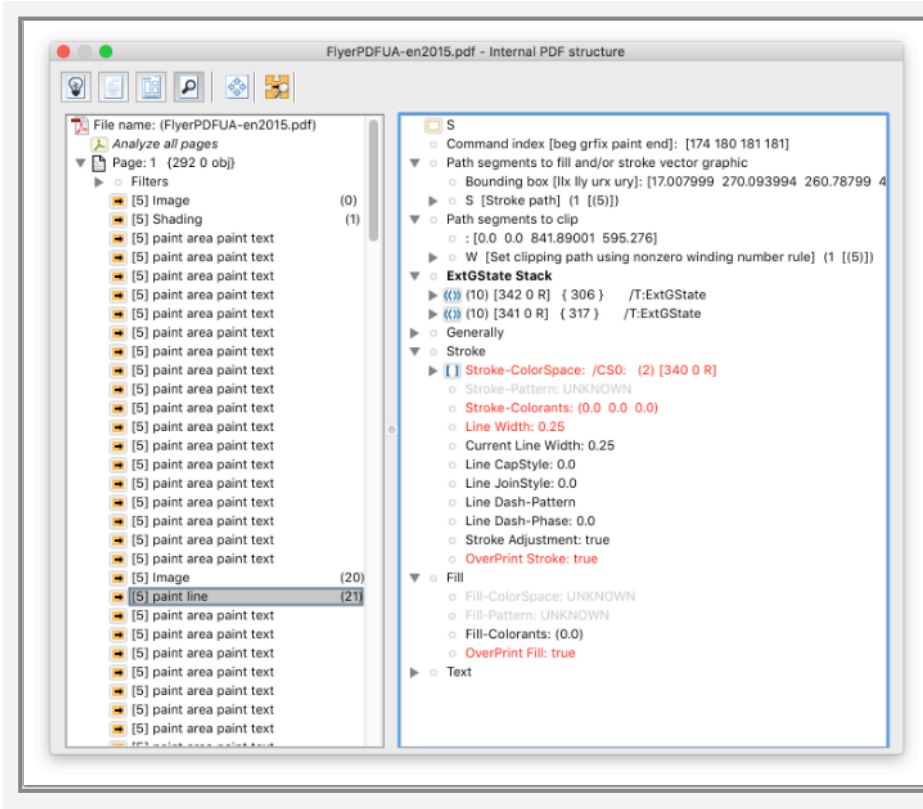


A list of the available indicators can be found at "Indicator lookup" entry. Please note, that the order of the indicators has to be considered.

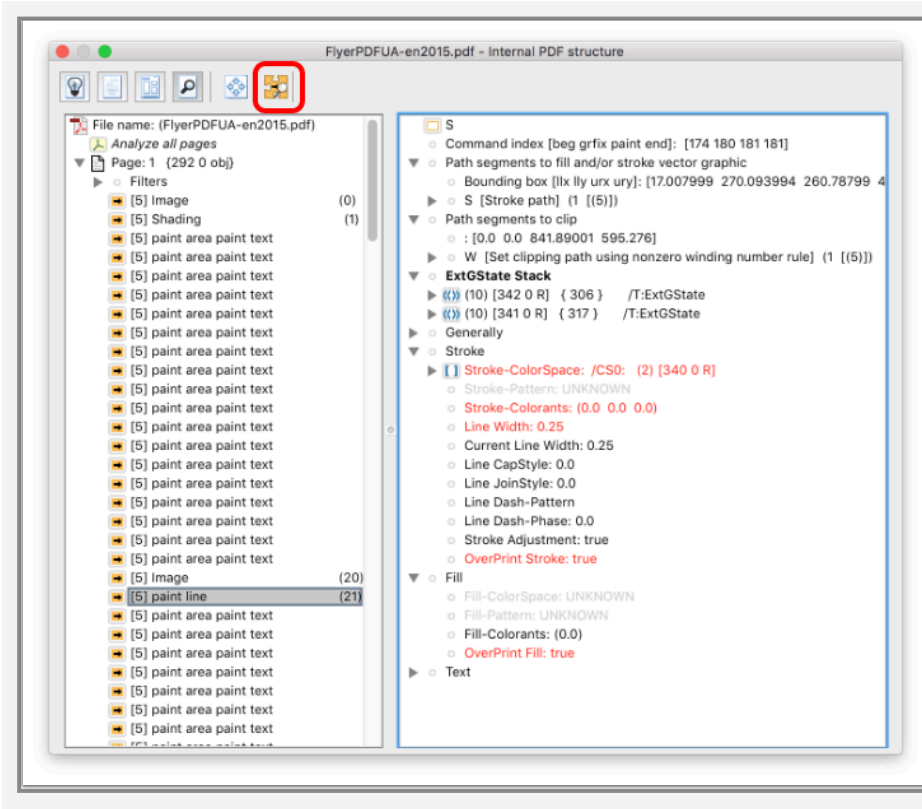
The indicator lookup informs us that the capital "W" means that the glyph width is used for positioning (the glyph is not positioned using coordinates but the width of a previous glyph). "e" stands for glyphs without contour and "s" is for such empty glyphs with a width, so in fact the respective glyph is a whitespace.

Analyze snippets and export selected snippets to a new PDF

For each object, a detailed view shows the painting and clipping area, the used color space as well as a lot of other information of the current Extend GraphicState, a used transformation matrix, for text the used font and font size, blend spaces, overprint modes and much much more.



The "open snippet as PDF" icon at the top of the Resource view allows for creating a PDF from the current selection in the left pane. The selection might also include several objects and Filters are provided to select all object of a certain type. Such PDF parts can be used for analysis to simplify a PDF.



The new PDF will be opened separately and can be used for further investigation of the PDF.

You'll see the functionality indicated by the red rectangle in the screen shot above.

PDF sample file used

The attached file has been used to show the various views of "Explore PDF".

This file has been created by the PDF Association.



FlyerPDFUA-en2015.pdf

2.19 Keyboard shortcuts for pdfaPilot

For some functionality callas pdfaPilot offers the possibility to use keyboard shortcuts.

callas pdfaPilot

Function	Shortcut	
	Mac	Windows
callas pdfaPilot Preferences	Cmd+,	
Hide callas pdfaPilot	Cmd+H	
Hide Others	Alt+Cmd+H	
Quit callas pdfaPilot	Cmd+Q	

File

Function	Shortcut	
	Mac	Windows
Open	Cmd+O	Ctrl+O
Save	Cmd+S	Ctrl+S
Save as	Cmd+Shift+S	Ctrl+Shift+S
Revert	Cmd+R	Ctrl+R
Close	Cmd+W	Ctrl+W
Exit	Cmd+Q	Alt+F4

Edit

Function	Shortcut	
	Mac	Windows
Cut	Cmd+X	Ctrl+X
Copy	Cmd+C	Ctrl+C
Paste	Cmd+V	Ctrl+V
Select All	Cmd+A	Ctrl+A

View

Function	Shortcut	
	Mac	Windows
First Page	<-	Home
Previous Page	⇧	Page up
Next Page	⇩	Page down
Last Page	->	End
Visualize ink coverage	Shift+Cmd+I	Ctrl+Shift+I
Output preview	Shift+Cmd+O	Ctrl+Shift+O
Compare documents	Shift+Cmd+C	Ctrl+Shift+C

Tools

Function	Shortcut	
	Mac	Windows
PDF/A in one click	Cmd+1	Ctrl+1

Function	Shortcut	
Switchboard	Cmd+2	Ctrl+2
Profiles	Cmd+3	Ctrl+3
Checks	Cmd+4	Ctrl+4
Fixups	Cmd+5	Ctrl+5
Organize pages	Shift+Cmd+M	Ctrl+Shift+M
Explore Metadata	Cmd+6	Ctrl+6
Explore Layers	Cmd+7	Ctrl+7
Explore PDF	Cmd+8	Ctrl+8
Explore Fonts	Cmd+9	Ctrl+9
Server	Cmd+0	Ctrl+0
Server Checkpoint Files	Shift+Cmd+0	Ctrl+Shift+0
View in default PDF application	Cmd+D	Ctrl+D

Window

Function	Shortcut	
	Mac	Windows
Loupe	Cmd+L	Ctrl+L
Show next document	^→	Ctrl+TAB
Show previous document	^Shift→	Ctrl+Shift+TAB

Help

Function	Shortcut	
	Mac	Windows

Function	Shortcut	
callas pdfaPilot Help	F1	F1

3. callas pdfaPilot Basics

3.1 Fixups, Checks, Profiles, Process Plans and Libraries

callas pdfaPilot uses Profiles, Checks, Fixups, Process Plans and Libraries. These can be briefly summarized as follows:

- Checks can be used to search PDF files for specific factors, and Fixups allow you to alter files in line with specific criteria.
- Profiles are designed to link Checks and Fixups together.
- Process plans can be used to ensure that multiple Profiles, Checks or Fixups run one after another in a controlled fashion.
- Finally, Libraries enable users to bring all pre-specified categories together into personalized collections for a better overview.

The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

Checks



Checks can be used to test PDF files for specific parameters. Checks inspect PDF documents, but do not alter them.

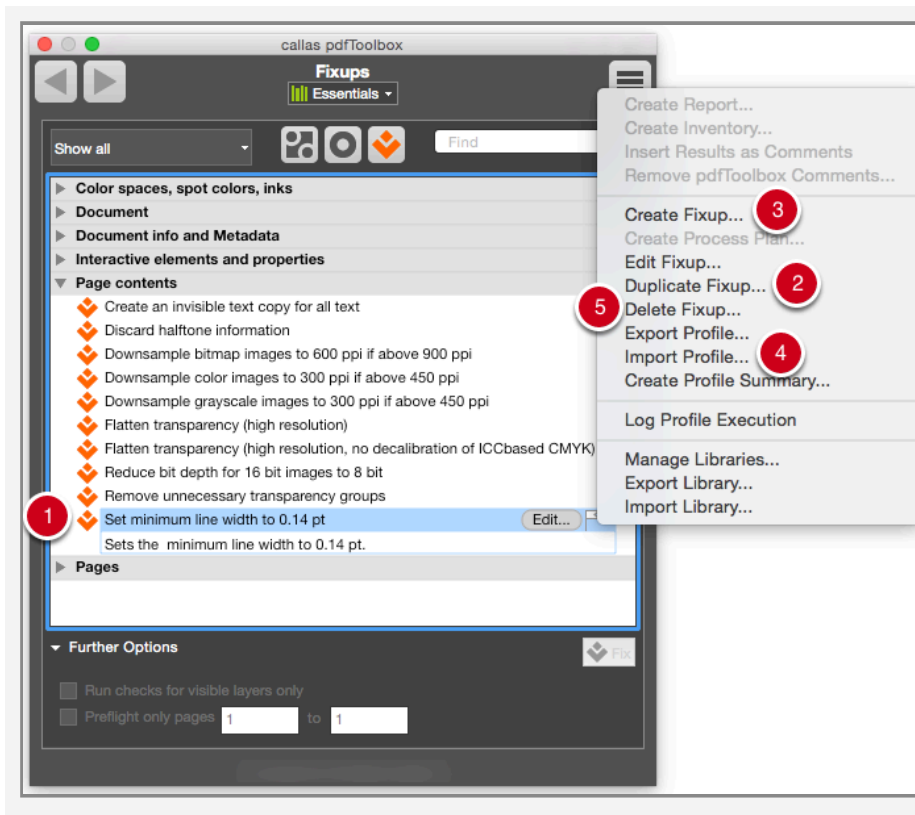
A number of different Checks are supplied as standard. Clicking on an entry in the list containing the Checks will provide a short description of the relevant check parameters (1).

If custom Checks are needed, users can create their own check Profiles. These can be altered versions of existing Checks - in which case you can work with duplicates (2) - or new Checks can be created from scratch (3).

Checks can also be exported and imported (4). This may be of interest if you need to forward or distribute them within your organization, or if you work with external partners.

Users can also delete Checks (5).

Fixups



Fixups can be used to **alter PDF files**. From a technical perspective, a fixup is always preceded by a check which tests the PDF file for specific parameters which must exist in order to trigger the check in the first place.

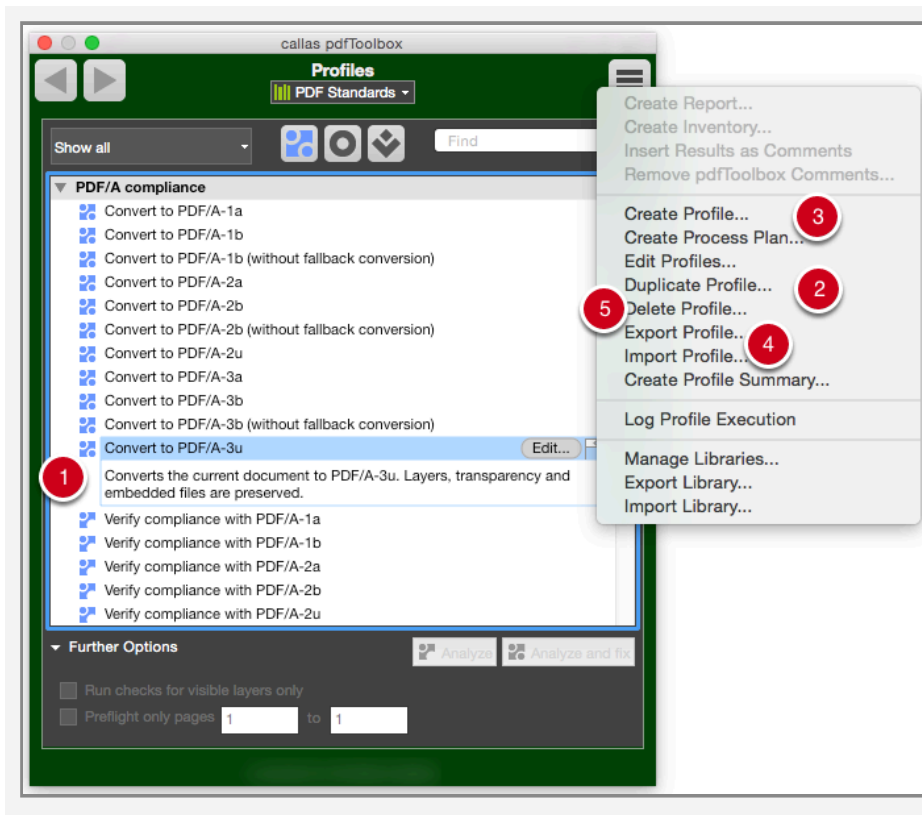
A number of different Fixups are supplied as standard. Clicking on an entry in the list containing the Fixups will provide a short **description** of the parameters that will be corrected in each case (1).

If custom Fixups are needed, users can create **their own fixup Profiles**. These can be altered versions of existing Fixups - in which case you can work with duplicates (2) - or new Fixups can be created from scratch (3).

Fixups can also be **exported and imported** for forwarding or distribution within your organization or with external partners (4).

Users can also **delete** Fixups (5).

Profiles



Profiles can be used to **check and/or fix** PDF files. For example, Profiles are used to test the various PDF standards and fix files as necessary.

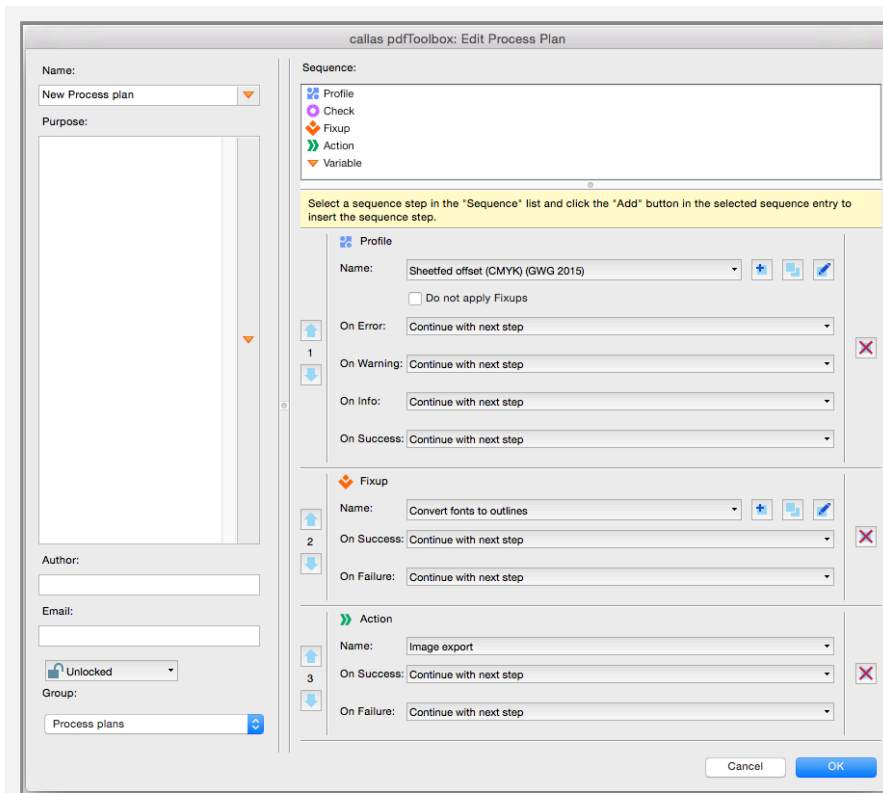
A number of different Profiles are supplied as standard. Clicking on an entry in the list containing the Profiles will provide a short **description** of the relevant details (1).

If custom tasks are needed, users can create **their own Profiles**. These can be altered versions of existing Profiles - in which case you can work with duplicates (2) - or new Profiles can be created from scratch (3).

Profiles can also be **exported and imported** for forwarding or distribution within your organization or with external partners (4).

Users can also **delete** Profiles (5).

Process plans



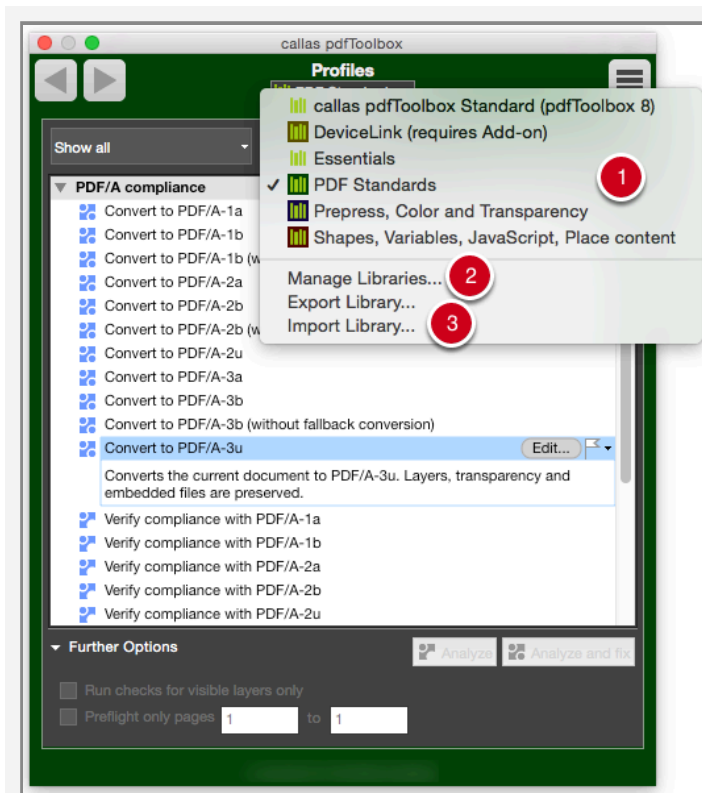
Process plans allow you to organize steps in interrelated sequences.

Parameters can be drawn from five different fields for the sequence steps.

- Profiles
- Checks
- Fixups
- Actions
- Variables

The steps can process conditional instructions (if..., then...). Examples of results that occur when specified parameters result in a given report include **Success**, **Warning**, **Error** and **Info**. This allows you to assemble processes which make use of conditional switches at specified points in the process.

Libraries



Libraries let you organize sets of Profiles, Checks, Fixups, switchboard actions, output conditions and imposition configurations.

Libraries offer a broader overview for users who, for example, wish to create pdfaPilot environments for a range of jobs or for multiple customers.

1. pdfaPilot is supplied with a range of preconfigured Libraries (1). In the Switchboard, and in the overview of Profiles, Checks and Fixups, users can open the desired library from the pop-up menu.
2. Libraries can be edited (by clicking Manage (2)), exported and imported (3).

All other information relating to Libraries can be found in the chapter [Libraries - Overview](#).

3.2 Protect Fixups, Checks, Profiles, Process Plans and Libraries against changes

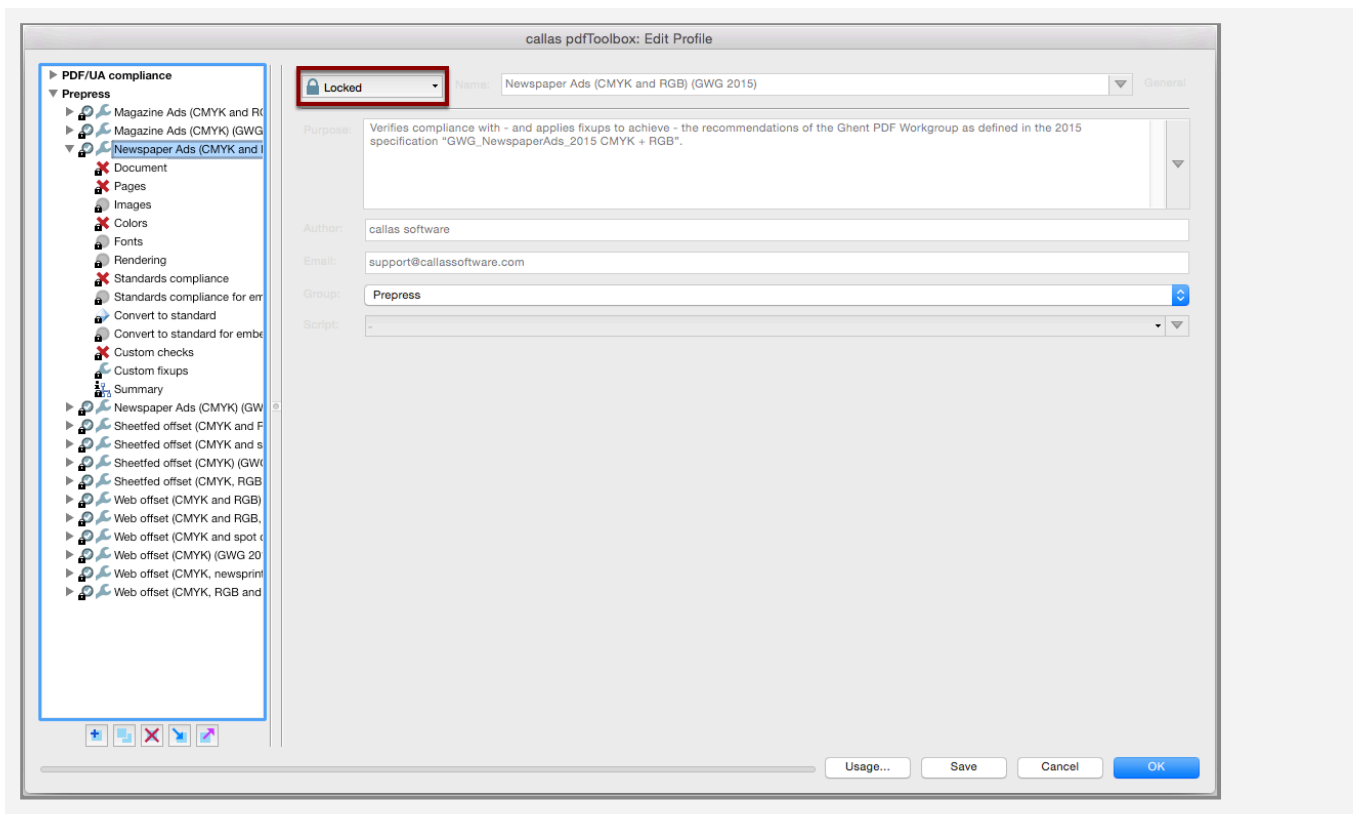
You can prevent unplanned changes to Fixups, Checks, Profiles, Process Plans and Libraries by locking them - i.e. by making them write-protected.

Checks and Fixups cannot be individually locked; instead, they are automatically locked whenever a Profile which uses them is itself locked.

Some of the predefined Fixups, Checks and Profiles are protected by default. They can be identified by looking for the lock icon.

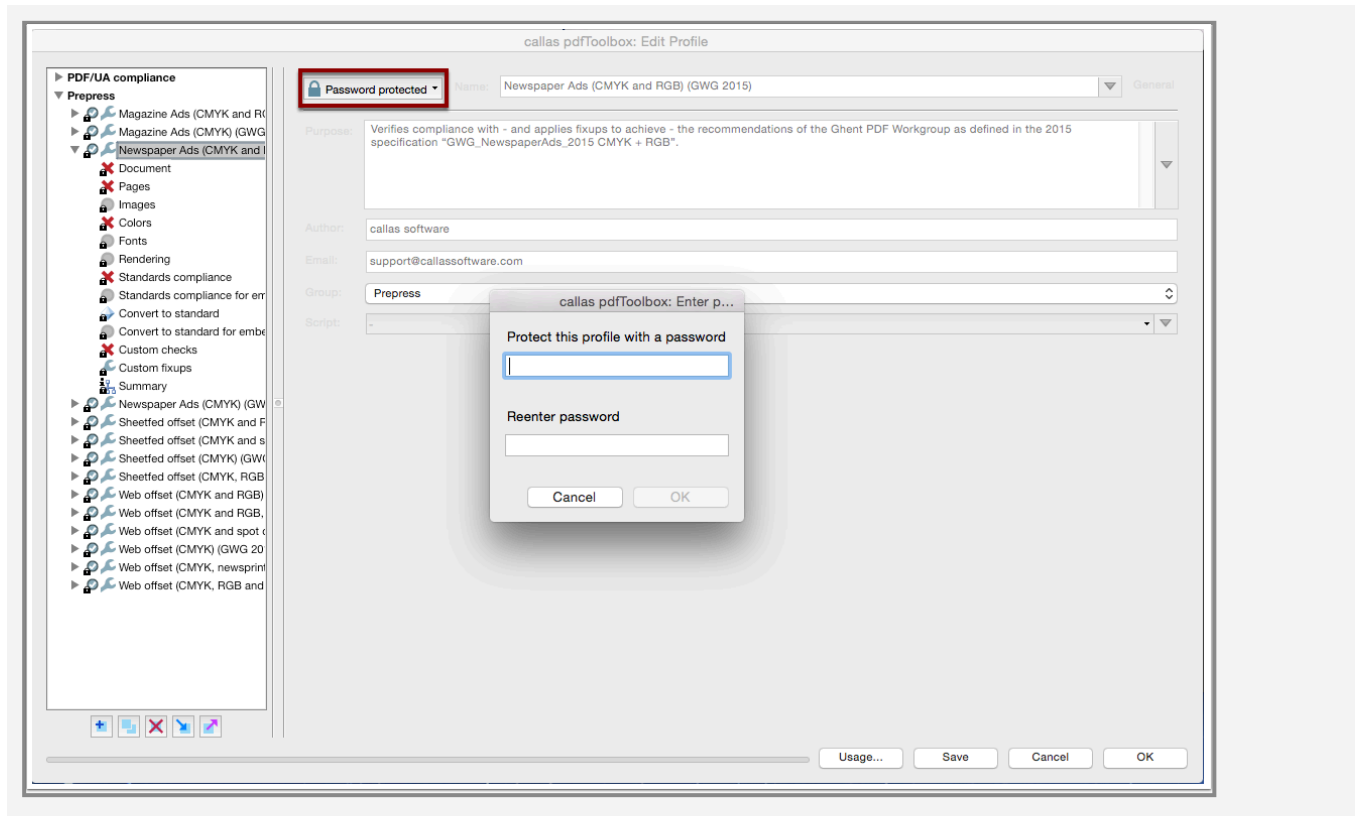
The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

Locked Profile



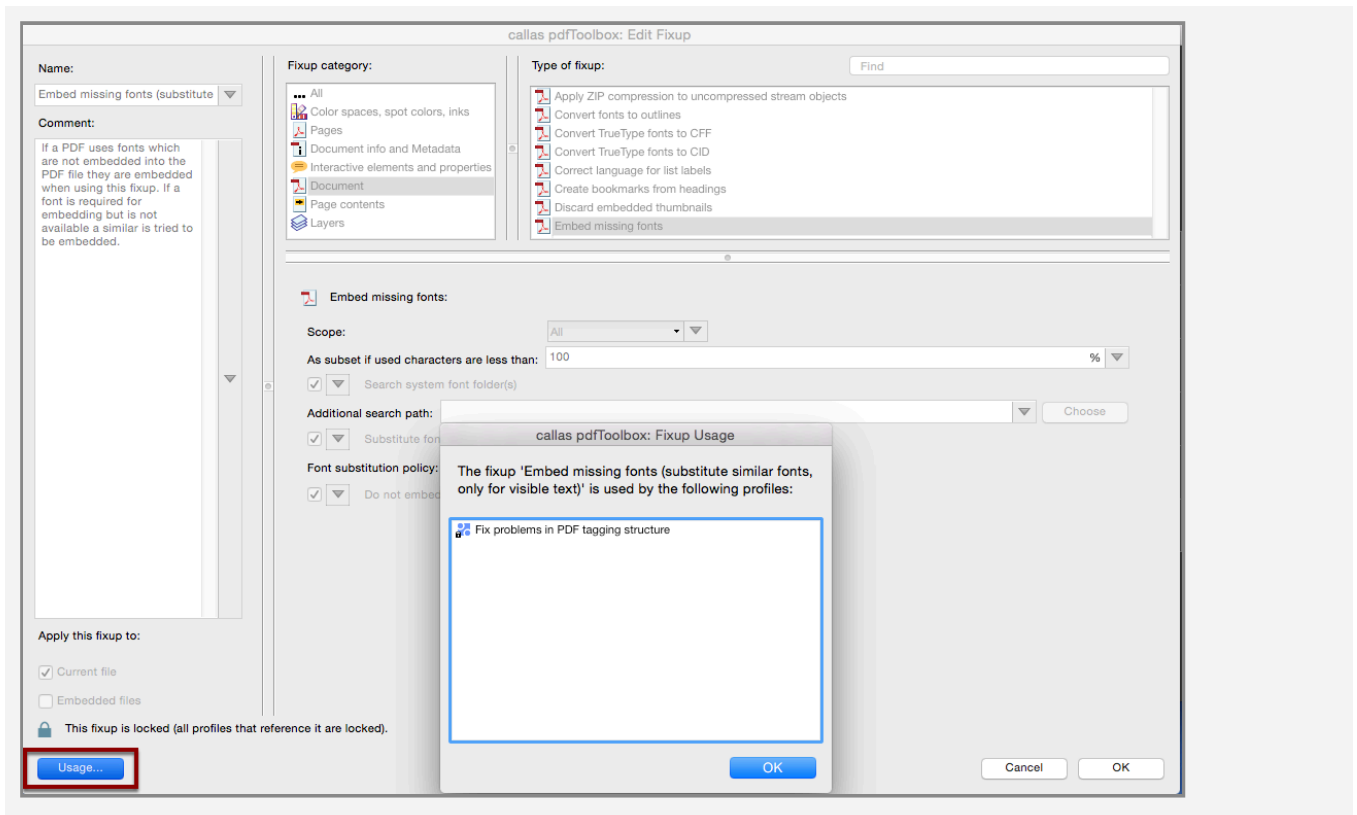
A locked profile is protected against changes.

Password-protected Profile



Alternately, a Profile can be assigned a password.

Fixup used in other Profiles or Process Plans



A Fixup (or a Check) can only be changed if it is not being used in a locked Profile or Process Plan.

You can see how it is being used in the corresponding interface marked “Usage...” within the same dialog.

Call up the listed Profiles as required, click “Edit...” and change the lock setting to “Open”.

You can then edit the Fixup (or the Check).

3.3 Duplicate and edit Profiles, Checks and Fixups

In some cases, it may make sense to create a new Profile based on an existing one, as an existing Profile may be quite well suited to a new task but require changes to certain settings within its own workflow (such as different resolutions, color spaces and many other options). The same principle also applies to Checks and Fixups.

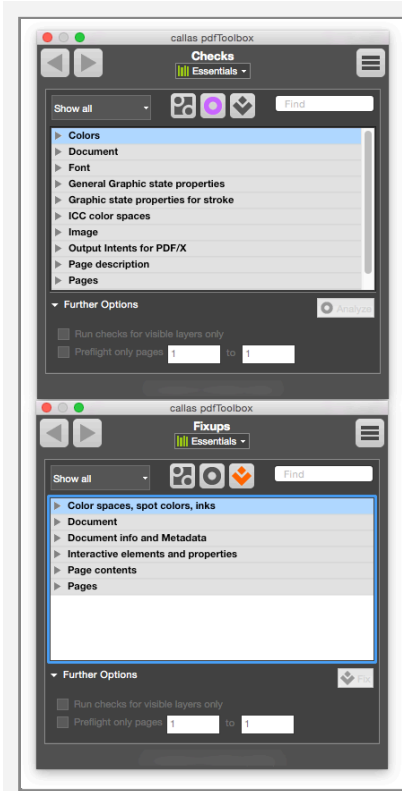
The starting point can be one of the Profiles supplied as standard or a custom one with specific details that need to be altered. The user can duplicate the existing Profile and alter specific elements as necessary. We'll show you how this works step by step.

pdfaPilot distinguishes between Profiles, Checks and Fixups.

- Checks and Fixups are edited in a broadly similar fashion to one another.
- Profiles are edited and set up in a different way, as from a technical perspective they (may) combine both Checks and Fixups.
- We will therefore first show you how duplicates of Checks and Fixups can be used. In a later section, we will then demonstrate how the process applies to Profiles.

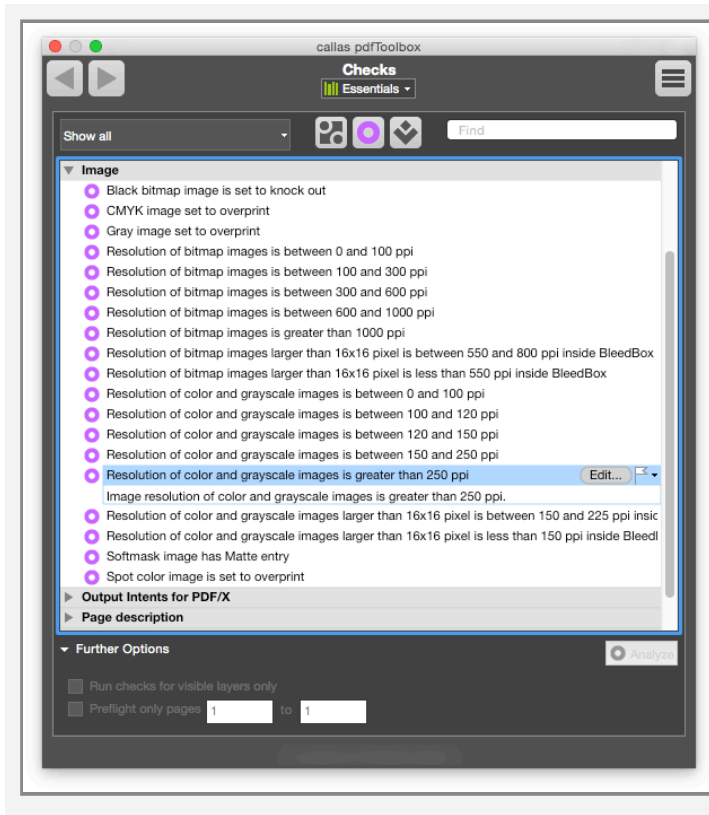
The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

Creating and setting up a duplicate of a Check (or Fixup)



Checks and Fixups are set up in a similar way in pdfaPilot.

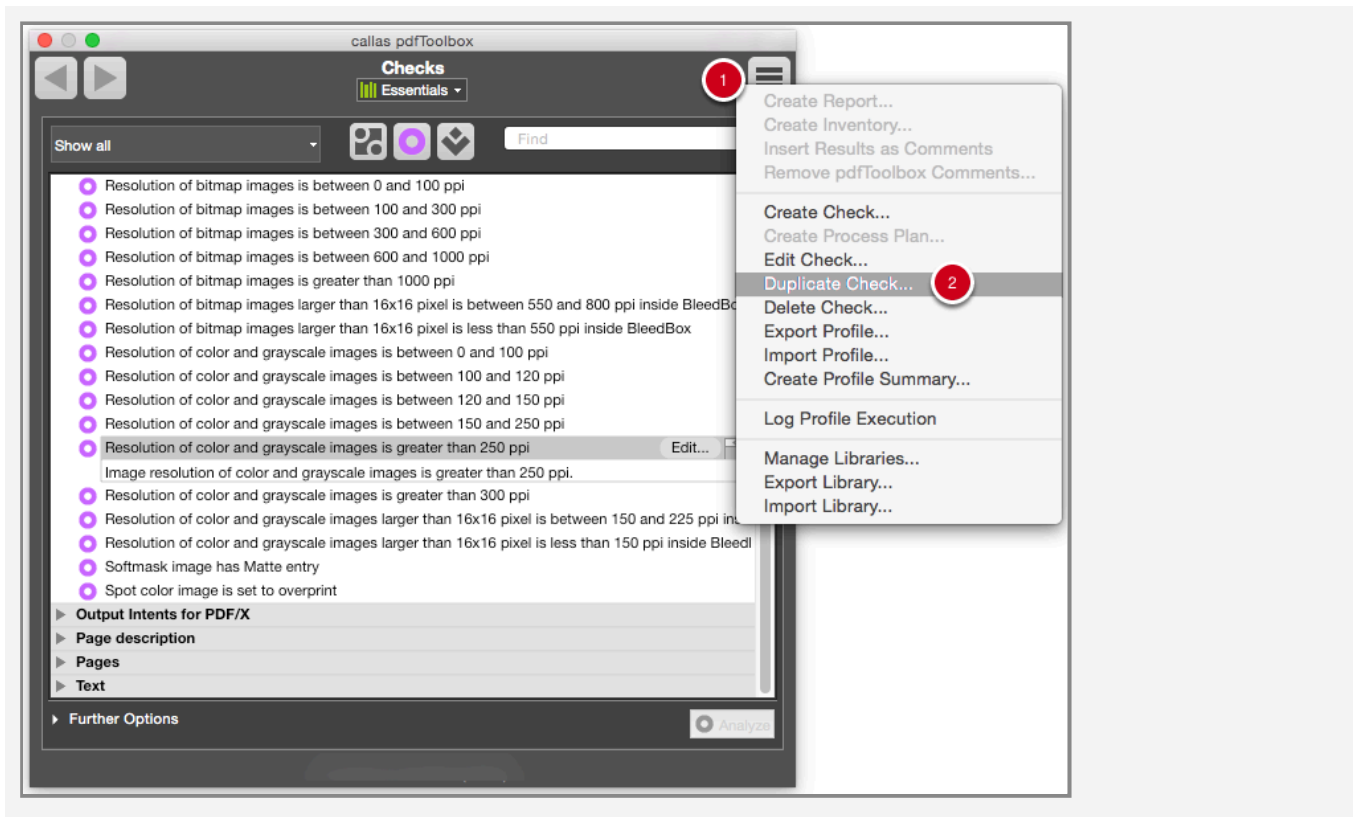
Selecting a Check for the duplicate



In the example shown, the Check “Resolution of color and grayscale images is greater than *250 ppi*” needs to be changed, raising the threshold to *300 ppi*.

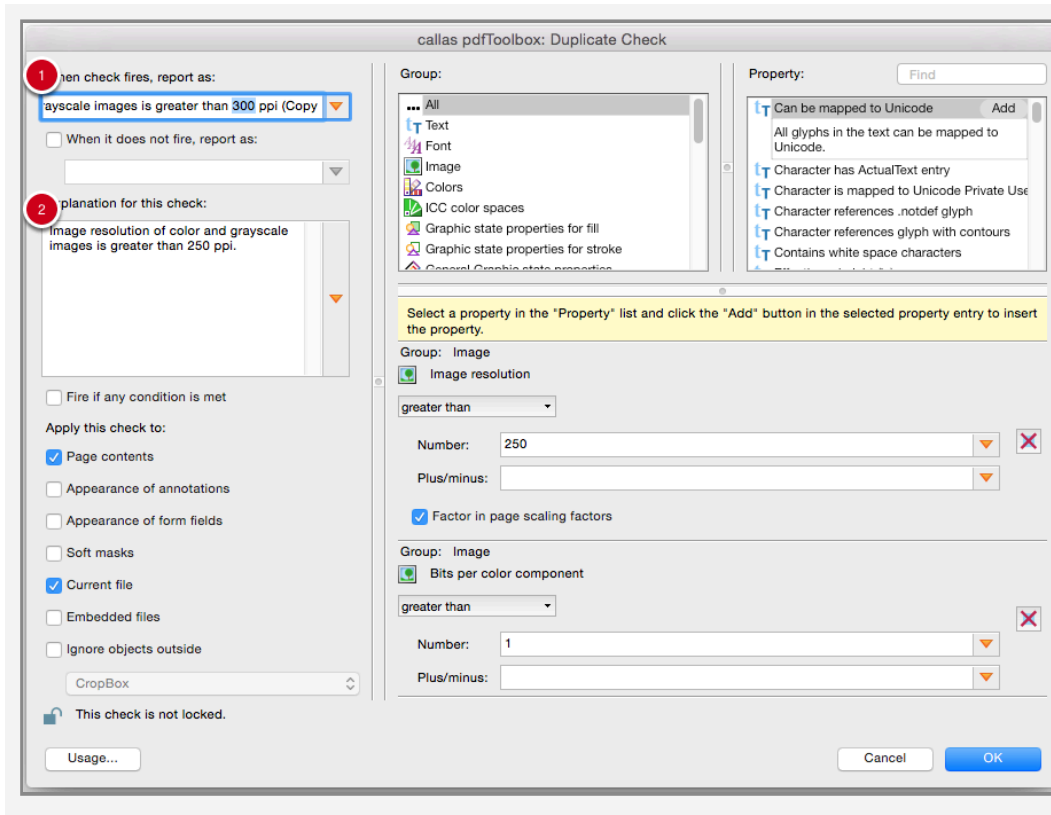
The Check can be found under the “Image” category and is a part of the “Essentials” library.

Duplicating the Check



To duplicate a Check, Fixup or Profile, first select it. In the fly-out menu to the upper right (1), select “Duplicate Check (or Fixup)” (2).

Changing settings for the duplicated Check

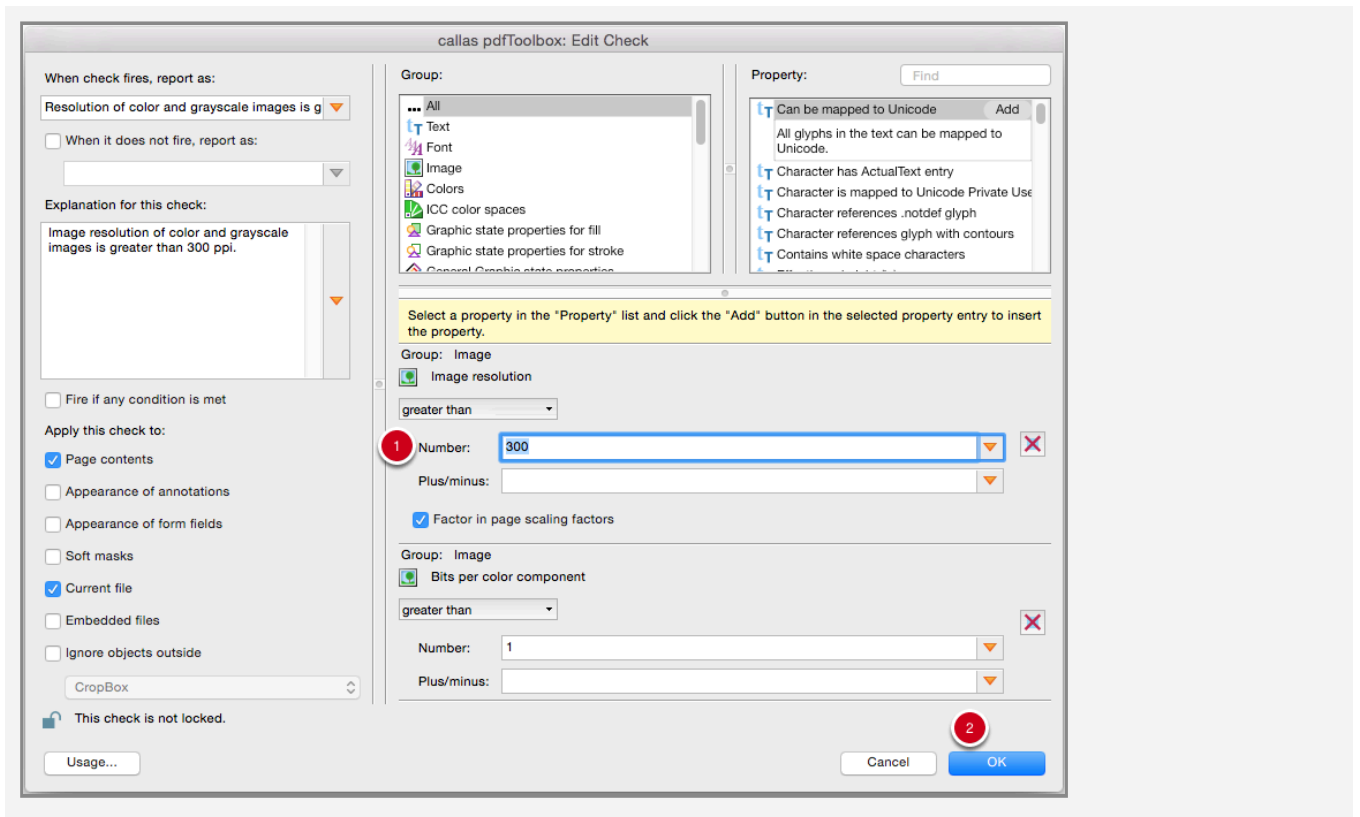


First, we should change the description of the Check so that its new function is clear.

In the example shown, the automatically generated text “Resolution of color and grayscale images is greater than 250 ppi (Copy 1)” can be changed to “Resolution of color and grayscale images is greater than 300 ppi” in the corresponding field (1).

We should also update the text in the Explanation field (2).

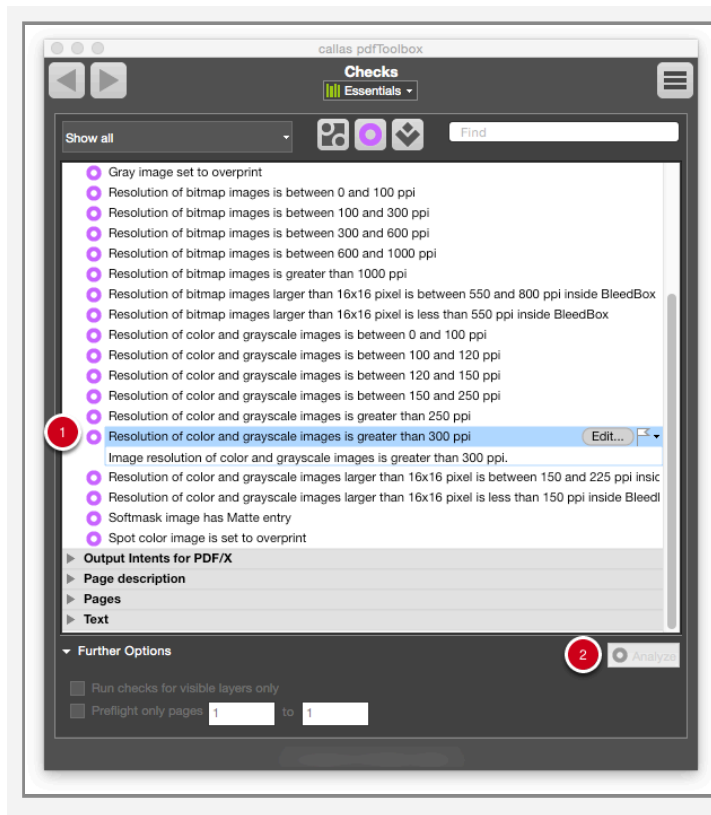
Editing properties



The property that needs to be changed in this example can be found under the “Image” group. The image resolution for the duplicate is now raised from 250 to 300 (1).

Click on the OK button (2) to save your changes.

The new Check in the Overview window



The new Profile is now shown in the list of Checks (1).

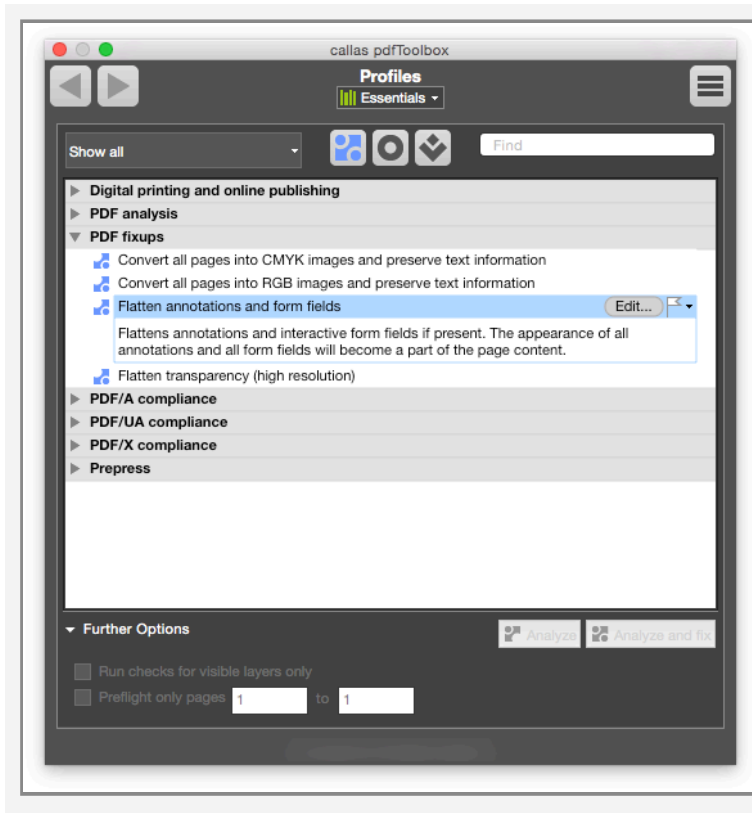
Click on **Analyze** (2) to find and report any applicable images in a PDF document with a resolution greater than 300 ppi.

Creating and setting up a duplicate of a Profile



In this section, we'll show you how to set up Profiles. Profiles consist of a combination of Checks and Fixups.

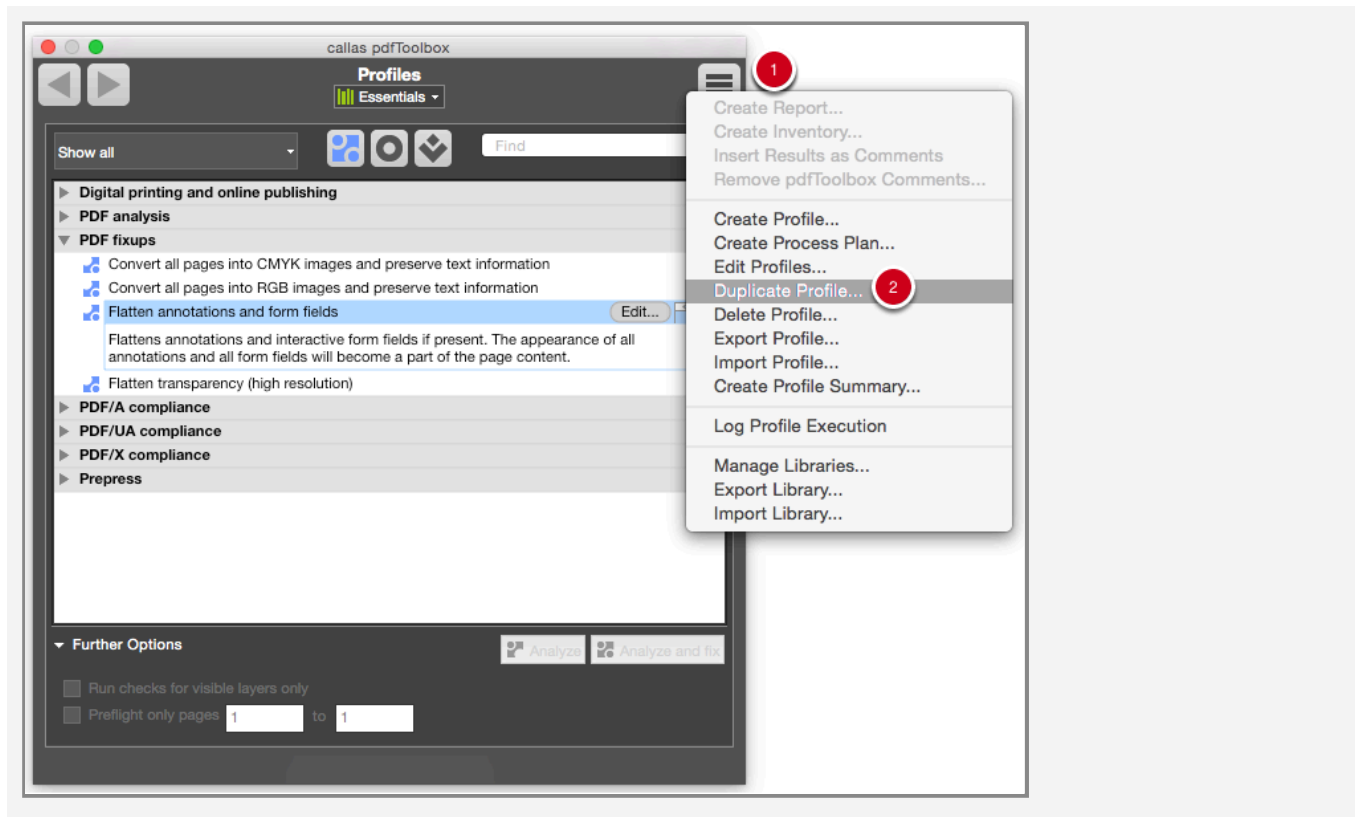
Selecting a source Profile for the duplicate



In the example shown, the Profile “Flatten annotations and form fields” needs to be altered. The new Profile should only flatten form fields and move annotations out of the Bleed-Box.

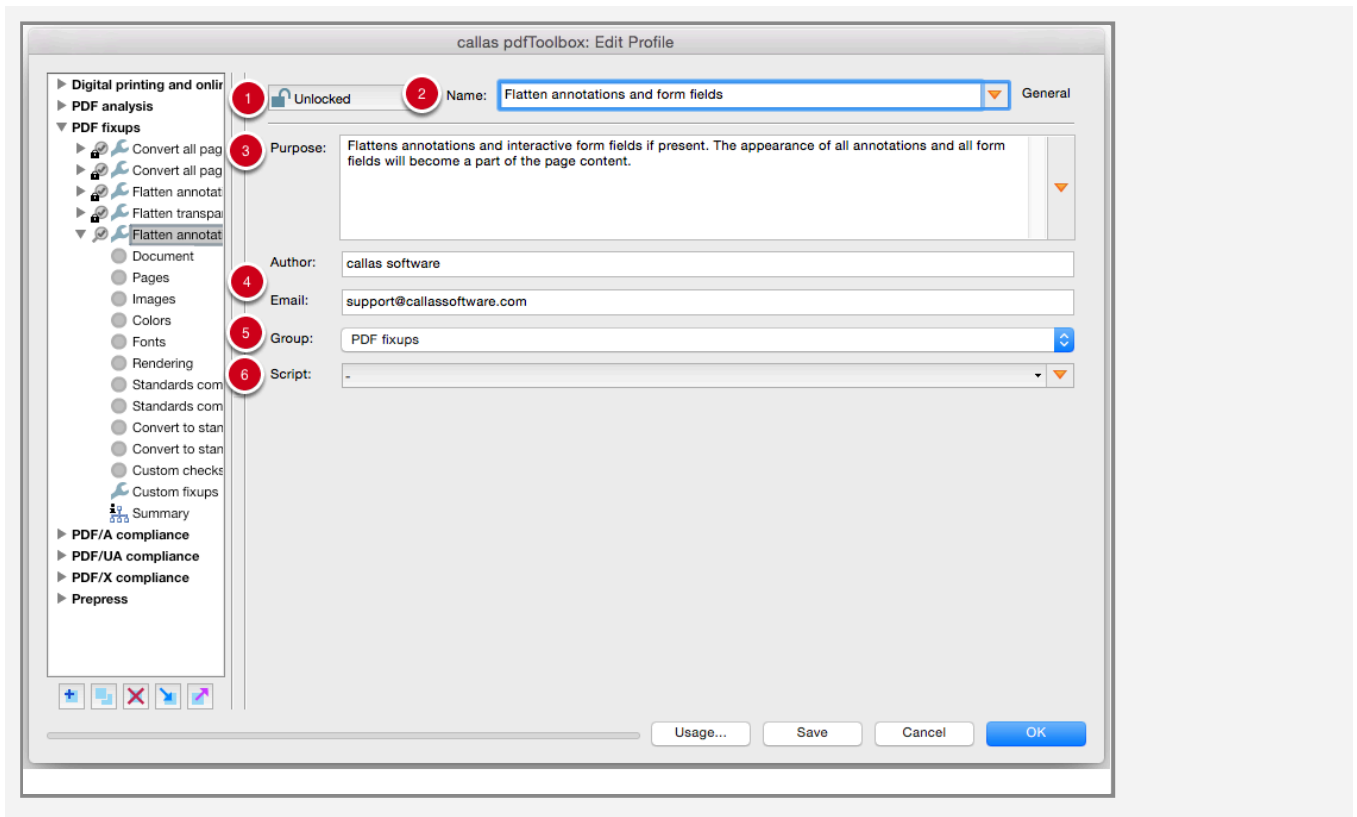
The Profile can be found in the “PDF Fixups” category and is a part of the “Essentials” library.

Duplicating the Profile



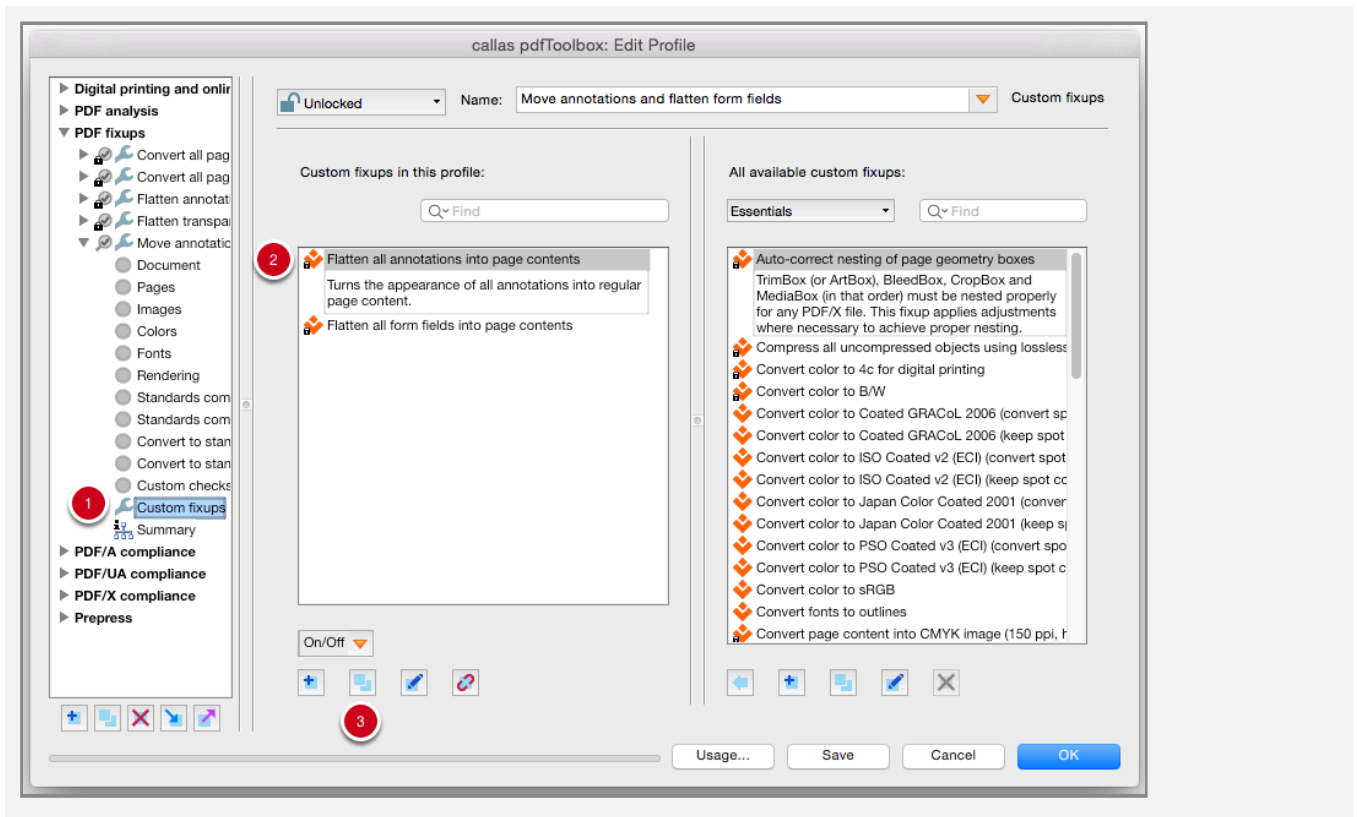
To duplicate a Profile, first select it. In the flyout menu to the upper right (1), select “Duplicate Profile” (2).

Changing settings for the duplicated Profile



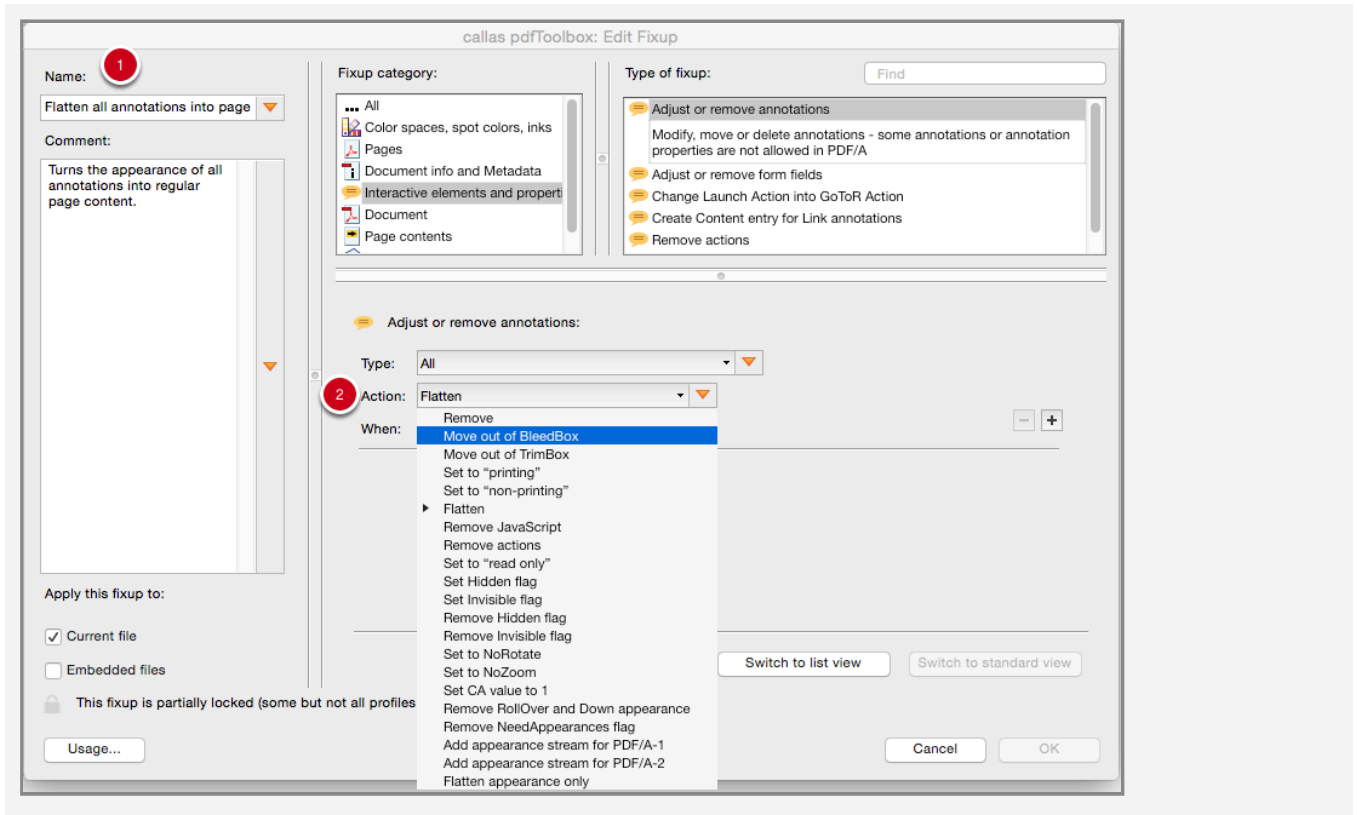
1. The user can **protect** the Profile from being changed by clicking on the lock symbol.
2. We should change the **name** of the Profile so that its new function is clear. In the example shown, the automatically generated text “Flatten annotations and form fields (Copy)” can be changed to “Move annotations and flatten form fields” in the relevant box.
3. We should also update the text in the **Comment** field.
4. Provide your own details in the **Author** and **Email** fields.
5. As this Profile is a duplicate, the “PDF Fixups” **group** is still suitable.
6. No **script** is used for this Profile.

Edit Profile



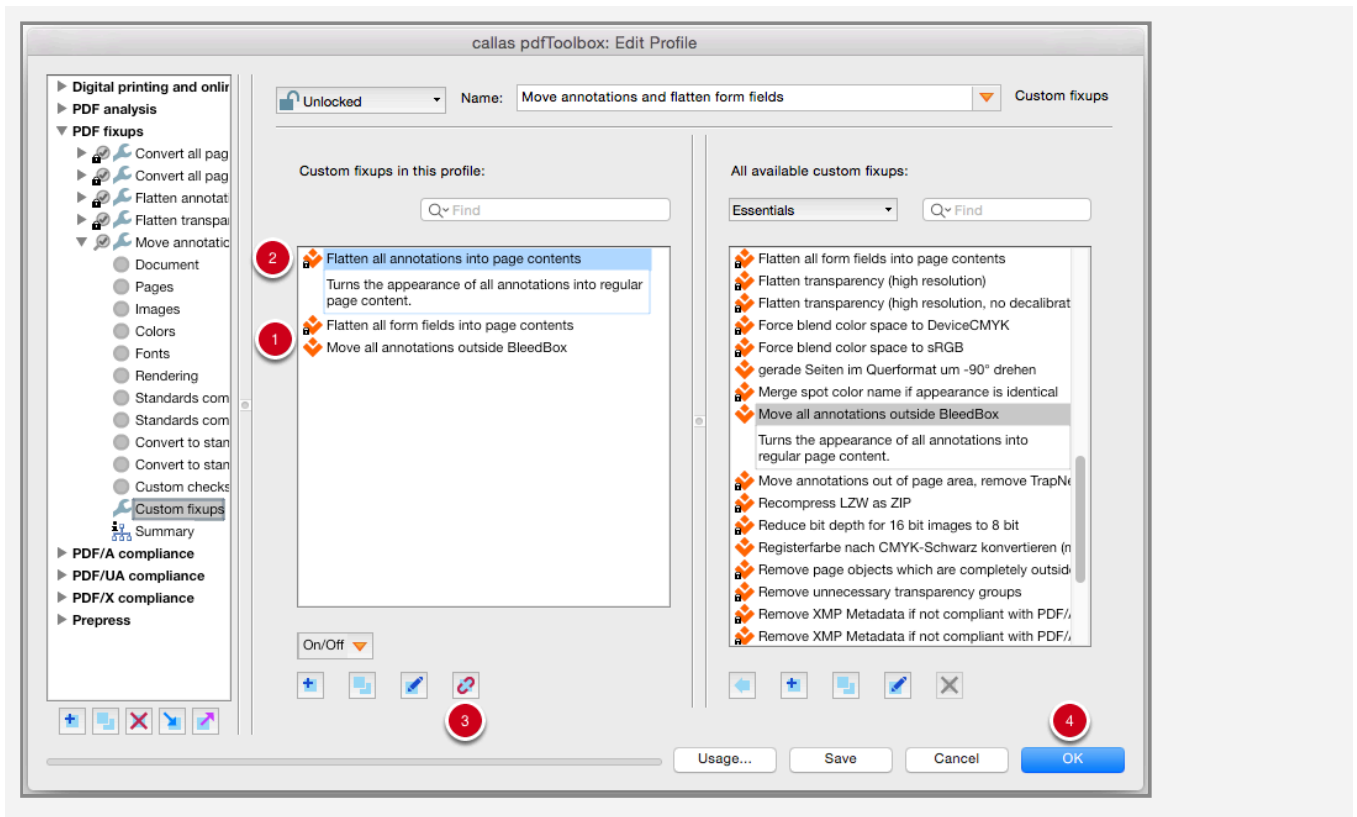
1. The settings to be changed in this example can be found under “Custom Fixups.”
2. For the original Fixup, we will select “Flatten all annotations into page contents.”
3. We will then duplicate the Fixup by clicking on the duplicate symbol.

Settings under “Duplicate this Fixup”



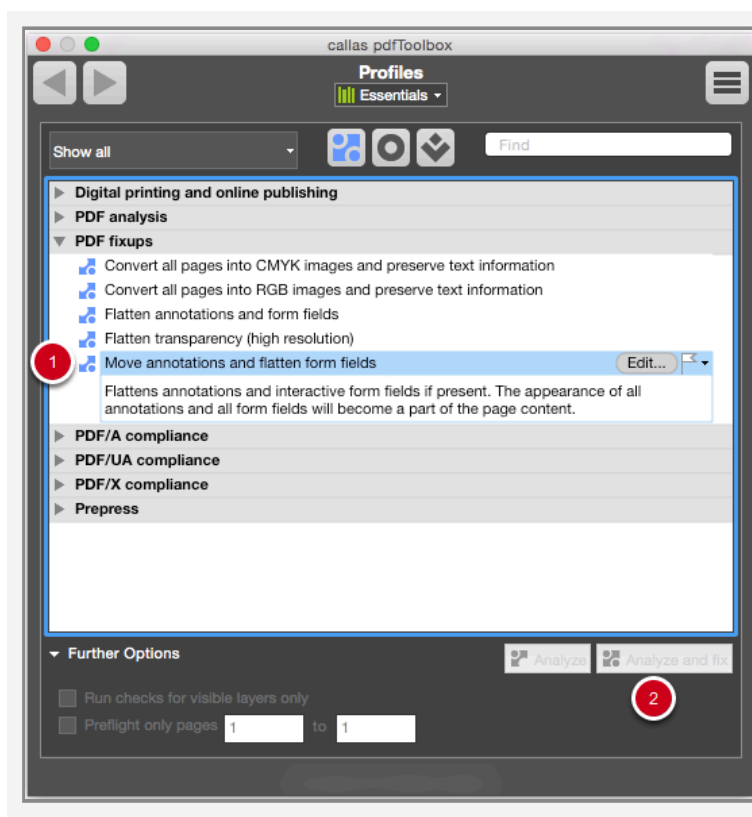
1. For this new duplicated Fixup, we should again edit the name and comment in the left column.
2. The new Profile should move annotations instead of flattening them. We therefore select the appropriate entry from the pull-down menu under “Action”.
3. Click **OK** to accept changes.

Results and advanced settings under “Edit Profile”



1. The new Fixup is now listed in the middle column.
2. The previous Fixup must now be removed. To do so, select the corresponding item in the list.
3. To remove it, click on the broken link symbol to break the link between the selected Fixup and the Profile.
4. Click OK to save and close the altered Profile.

The new Profile in the Overview window



The new Profile is now shown in the list of Profiles (1).

Click **Analyze and fix** (2) to execute the Profile.

3.4 Creating Profiles



Prefer a video of this content? Watch it [here...](#)

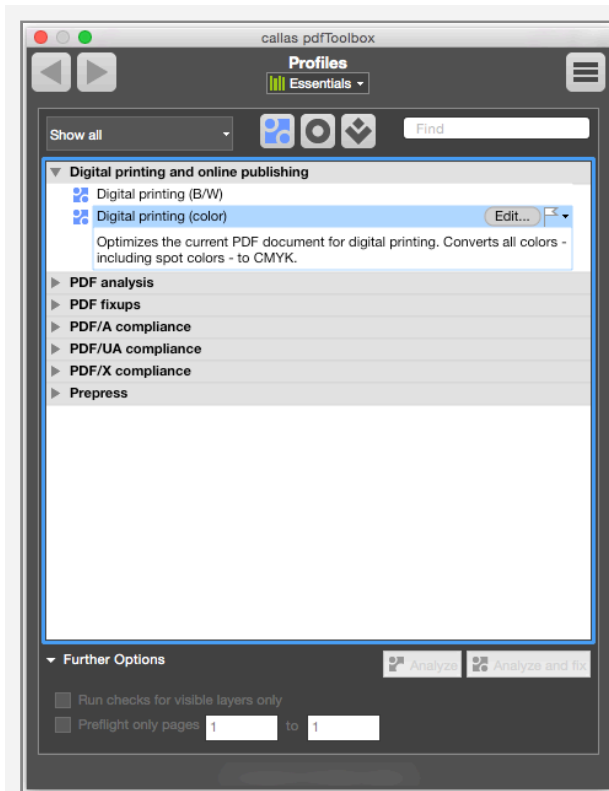
A Profile represents a collection of checking and/or fix-up points for a PDF document and (optionally) the relevant problems.

pdfaPilot provides a wide range of Profiles from a number of different categories, including digital printing, prepress, PDF analysis, PDF fixups and PDF standards.

Users can also set up custom Profiles for use in their own workflows.

We will now show you the options available in this context.

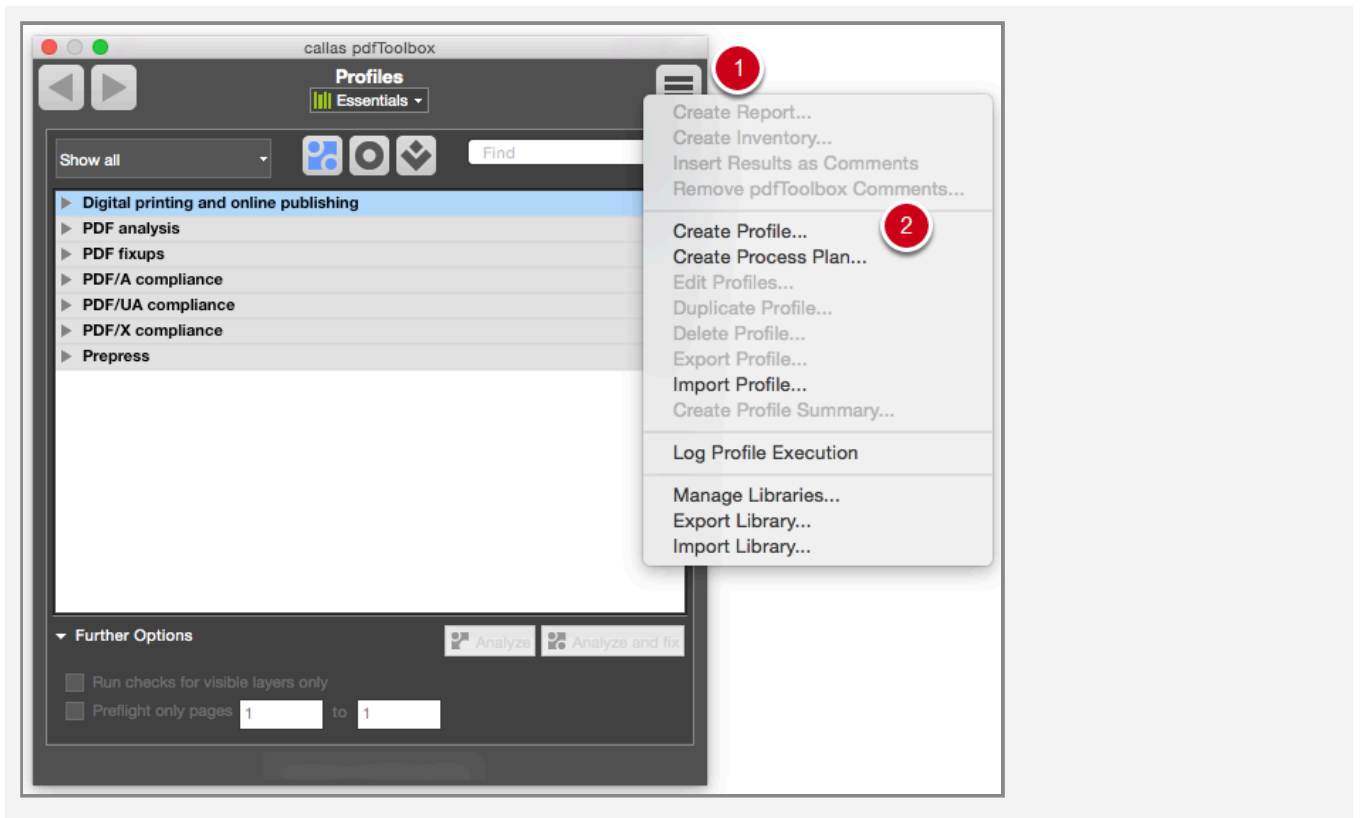
The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.



A wide range of Profiles are available in the standard release.

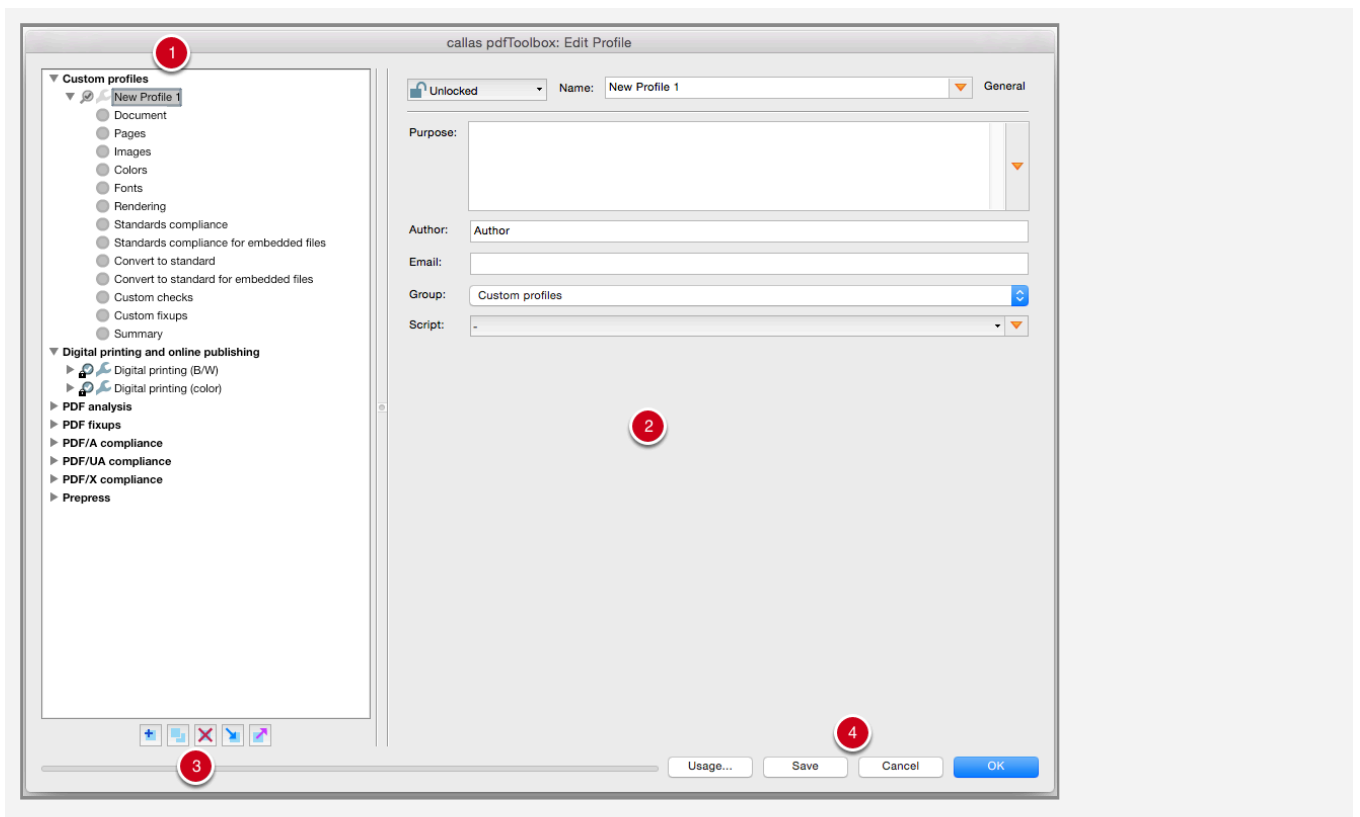
Custom Profiles can be set up in just a few steps.

Call the “Create Profile...” command



1. The button in the upper right of the Profiles window allows you to select from a number of options.
2. The pull-down menu also includes the command “Create Profile...”.

The “Edit Profile” window: an overview



A wide range of settings are available across a number of categories (1) for a (new) Profile, shown in a list to the left:

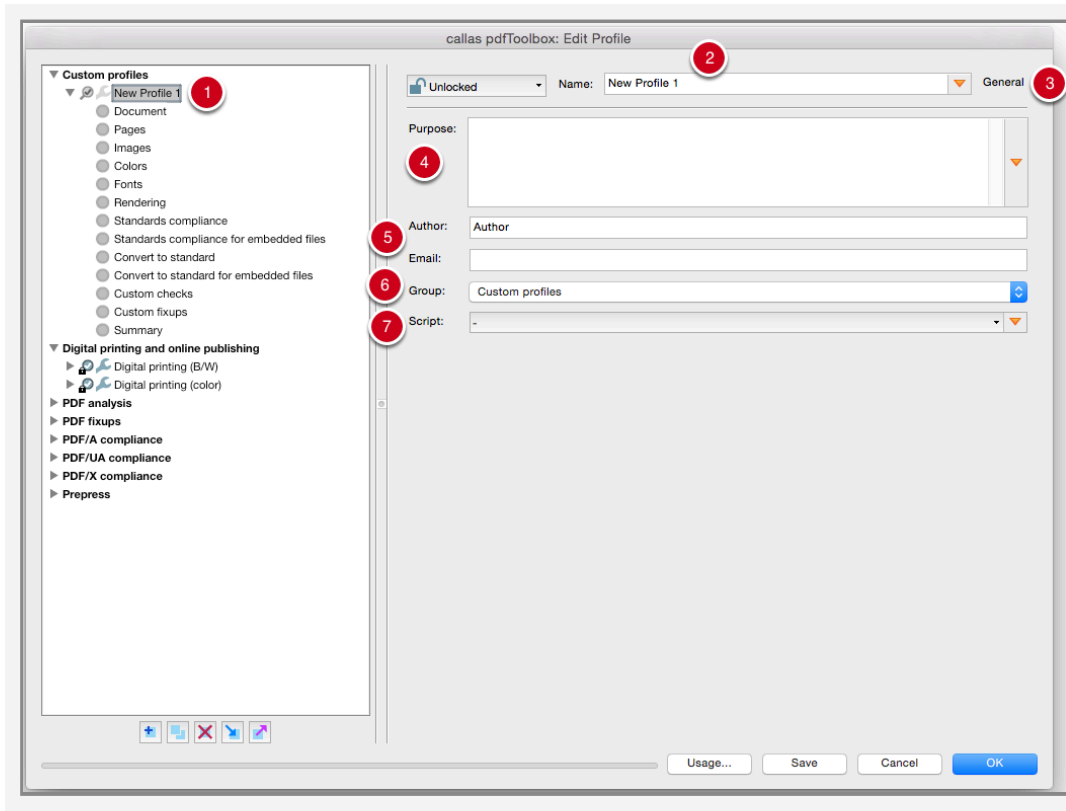
- General information
- Document
- Pages
- Images
- Colors
- Fonts
- Rendering
- Standards compliance
- Standards compliance for embedded files
- Convert to standard
- Convert to standard for embedded files
- Custom checks
- Custom fixups
- Summary

The right hand side of the window displays a number of settings depending on the selected category (2).

The buttons to the lower left allow you to add, duplicate, delete, import and export Profiles (3).

At the lower right of the window, you will find the Usage... button (determining which Profiles use a selected criterion), as well as Save, Cancel and OK (4).

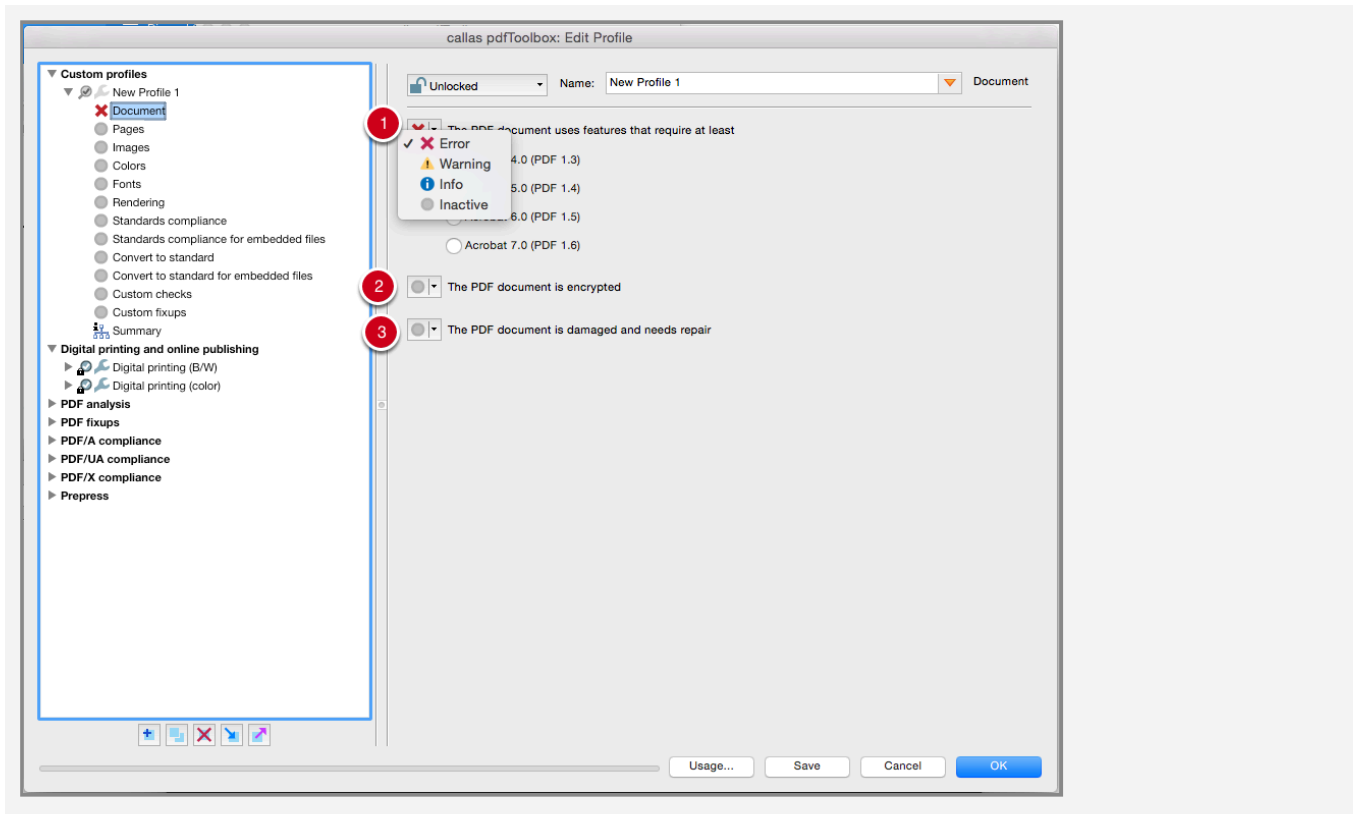
Edit Profile: General information



1. The new profile will be stored in the “Custom Profiles” group. It will initially be given the name “New Profile 1”.
2. The user can edit a number of general details on the right hand side of the window:
3. A profile can be **unlocked**, **locked** or **password-protected**; the name should be specified to match the profile’s intended purpose. The **orange triangle** (here and elsewhere) allows you to include **variables** and **scripts**.
4. The **Purpose** field allows you to add explanatory text which will later be shown in the profile list when selecting a profile.
5. **Author** and **Email** are intended to provide information about the author of the profile.

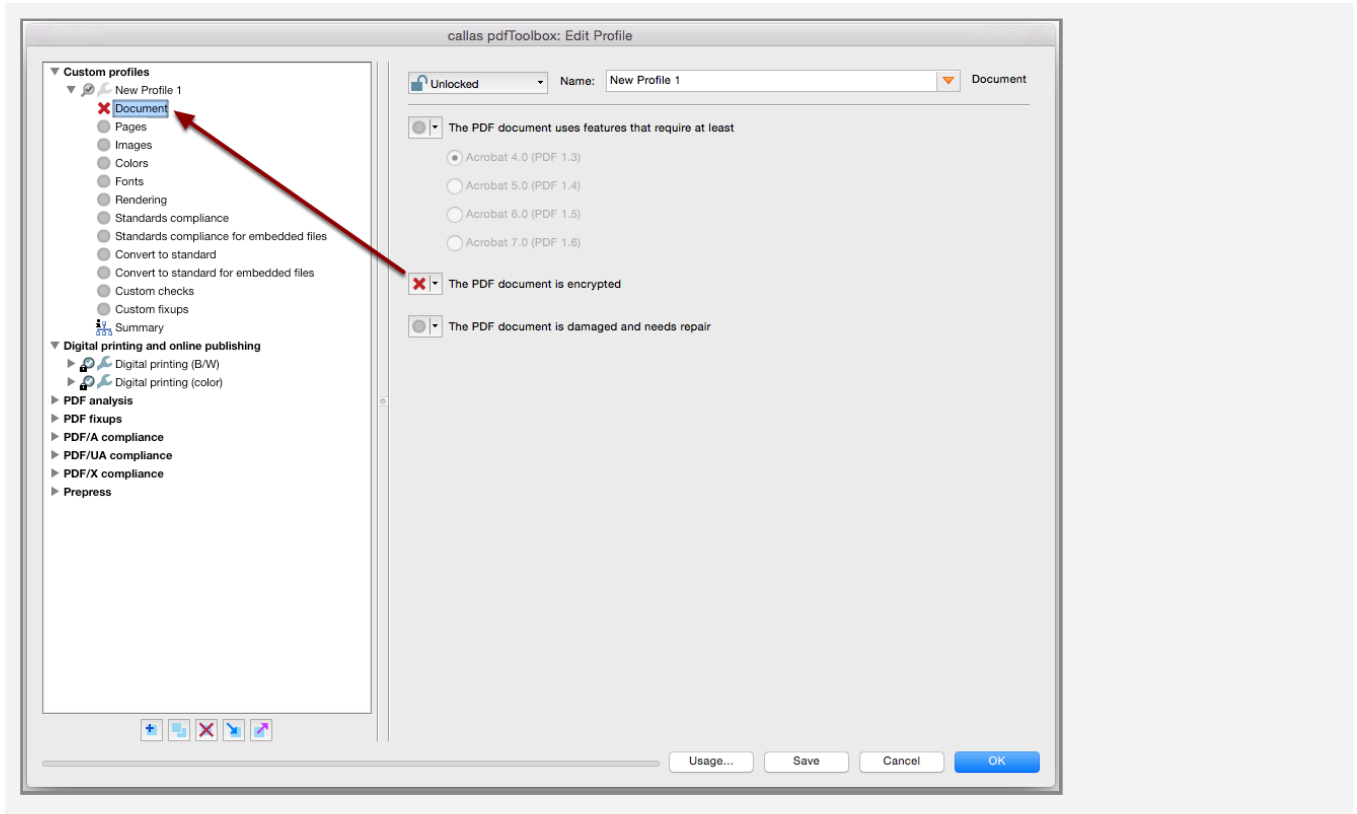
6. Under **Group**, you can specify whether to store the profile under **Custom Profiles** or other groups.
7. The profile can also be supplied with a **script**.

Edit Profile: Document



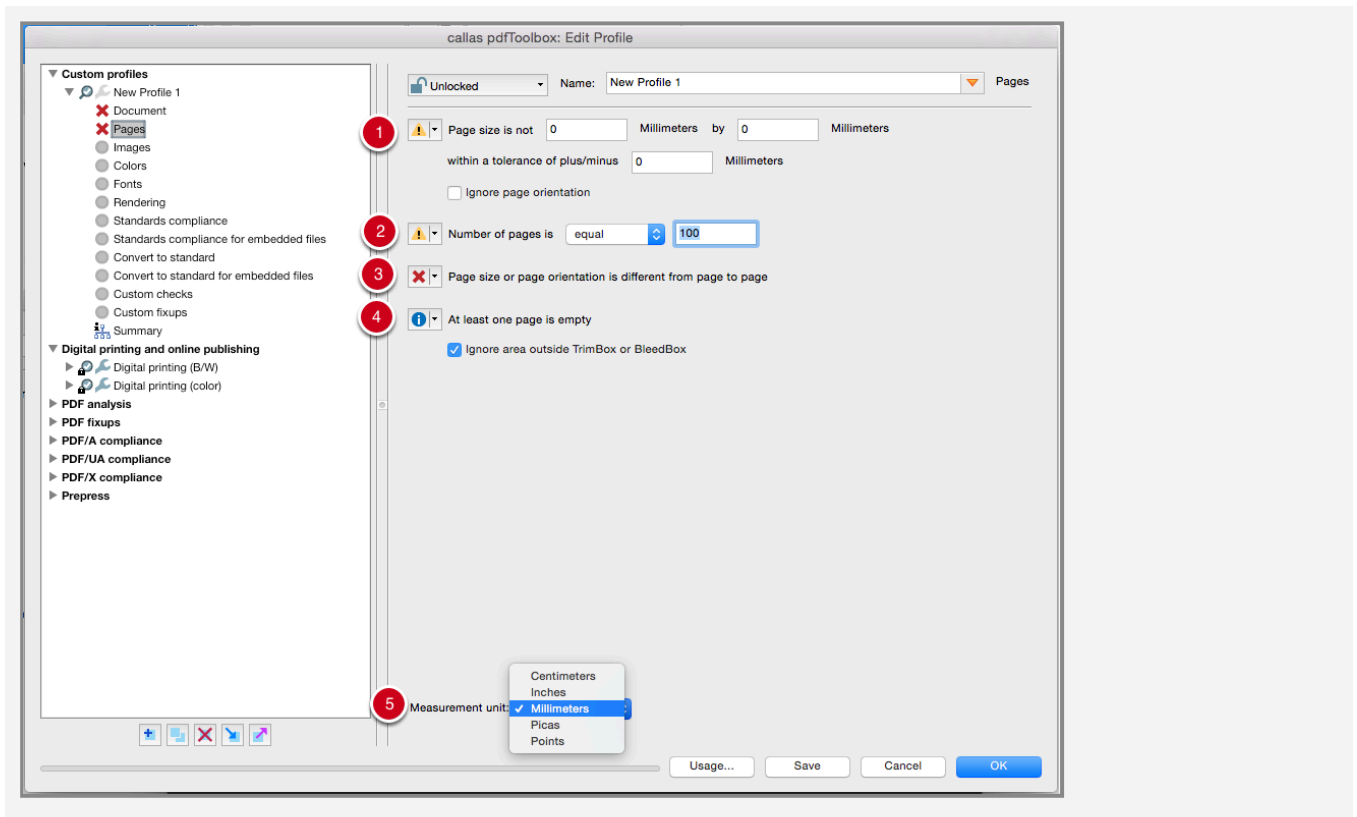
The **Document** section provides a number of document-related settings:

1. The first item affects **PDF functions** depending on the **PDF version**: “The PDF document uses features that require at least Acrobat 4.0 (PDF 1.3) and others.” Using the pull-down menu here (as with the other entries) you can specify how to report anomalies - as an **error**, as a **warning**, or as **info**. By default, this setting is **inactive**.
2. “Document is encrypted” indicates any **password protection** applied.
3. “Document is damaged and needs repair” tests whether the document is **intact**.



Provided that the user has set the “Document is encrypted” test to be reported as an error, the error warning will also be visible in the profile overview to the left. Here, the highest tier applicable will be shown (Error before Warning, before Info, before Inactive).

Edit Profile: Pages

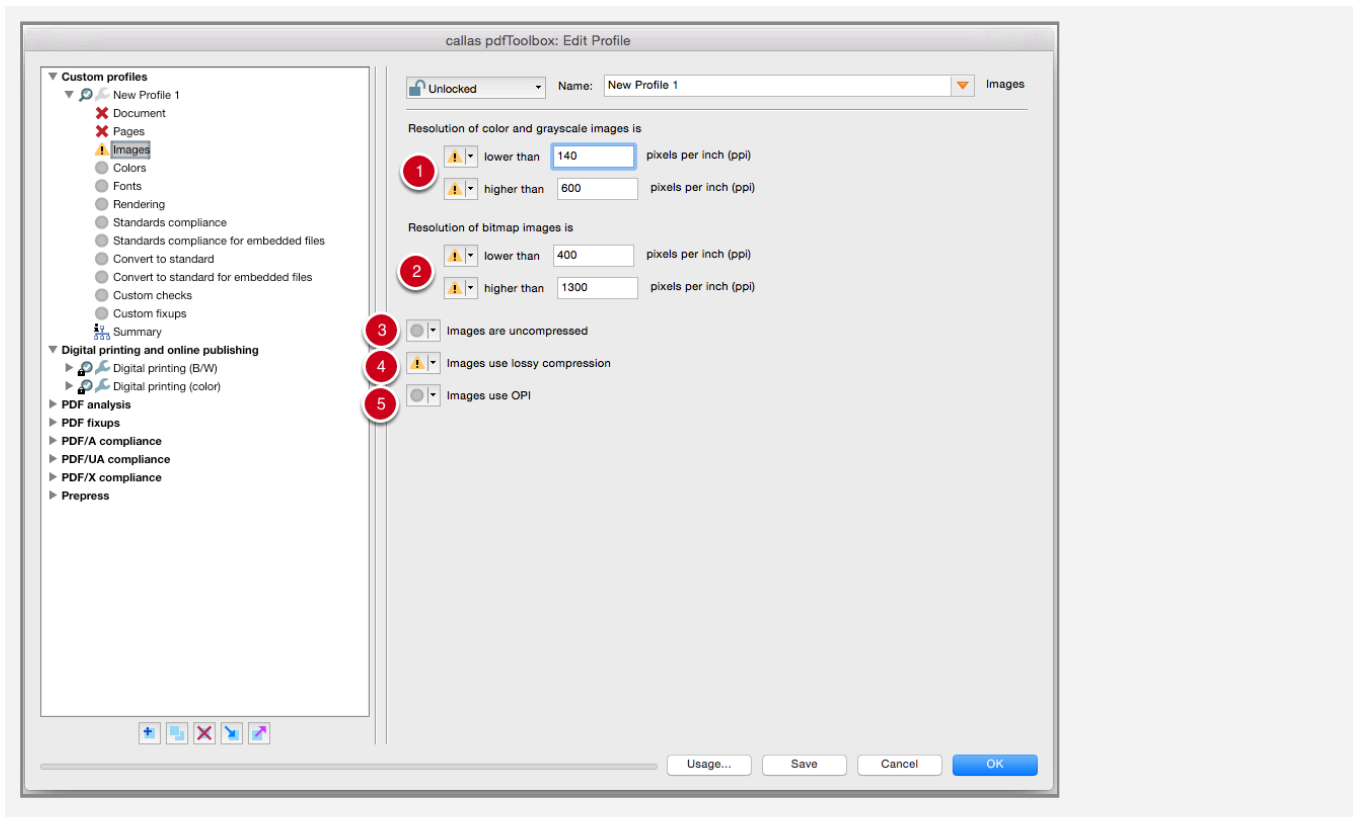


The **Pages** heading primarily allows you to set each individual parameter to one of the four settings **Inactive**, **Info**, **Warning** and **Error**.

Checks allow the following parameters:

1. The **page size** can be tested precisely or within a tolerance range.
2. When checking the **number of pages**, you can use the operators **equal**, **not equal**, **greater than** and **less than**.
3. The **Page size or page orientation is different from page to page** option allows you to check whether page sizes within a document are standardized.
4. The **At least one page is empty** option allows you to identify blank pages.
5. Finally, the user can also specify the **measurement unit**. The available options are **millimeters**, **picas**, **points**, **centimeters** and **inches**.

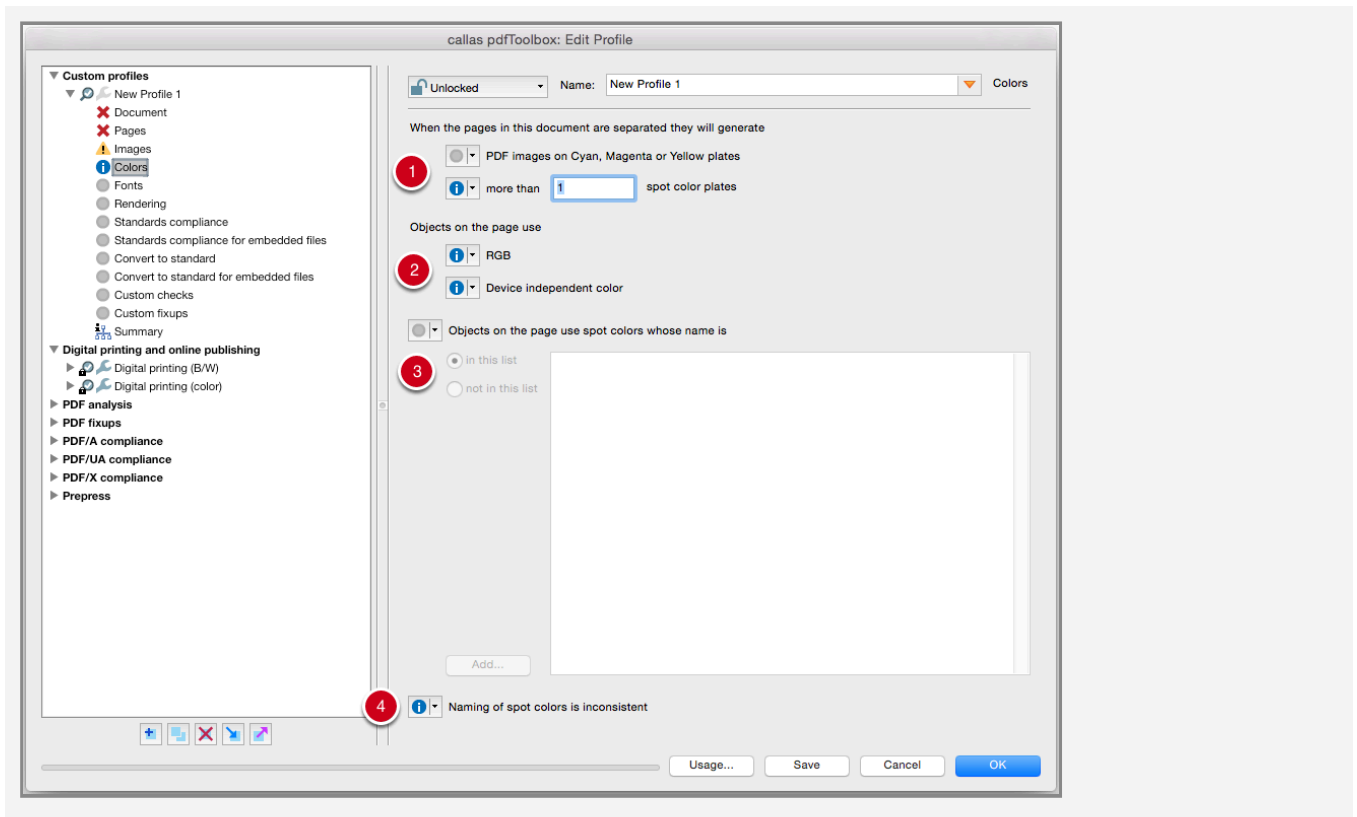
Edit Profile: Images



Images can primarily be tested in terms of their resolution and compression type:

1. Under **Resolution of color and grayscale images is**, you can specify minimum and maximum values for the resolution in pixels per inch.
2. The **Resolution of bitmap images** can also be checked in terms of upper and lower tolerances in pixels per inch.
3. **Images are uncompressed** detects images that are not compressed.
4. **Images use lossy compression** allows you to identify images that use compression methods such as JPEG. ZIP, on the other hand, is an example of a lossless option for reducing image file sizes.
5. **Images use OPI** relates to a currently rarely used process in which a file contains only low resolution placeholder images which are replaced with high resolution files on output.

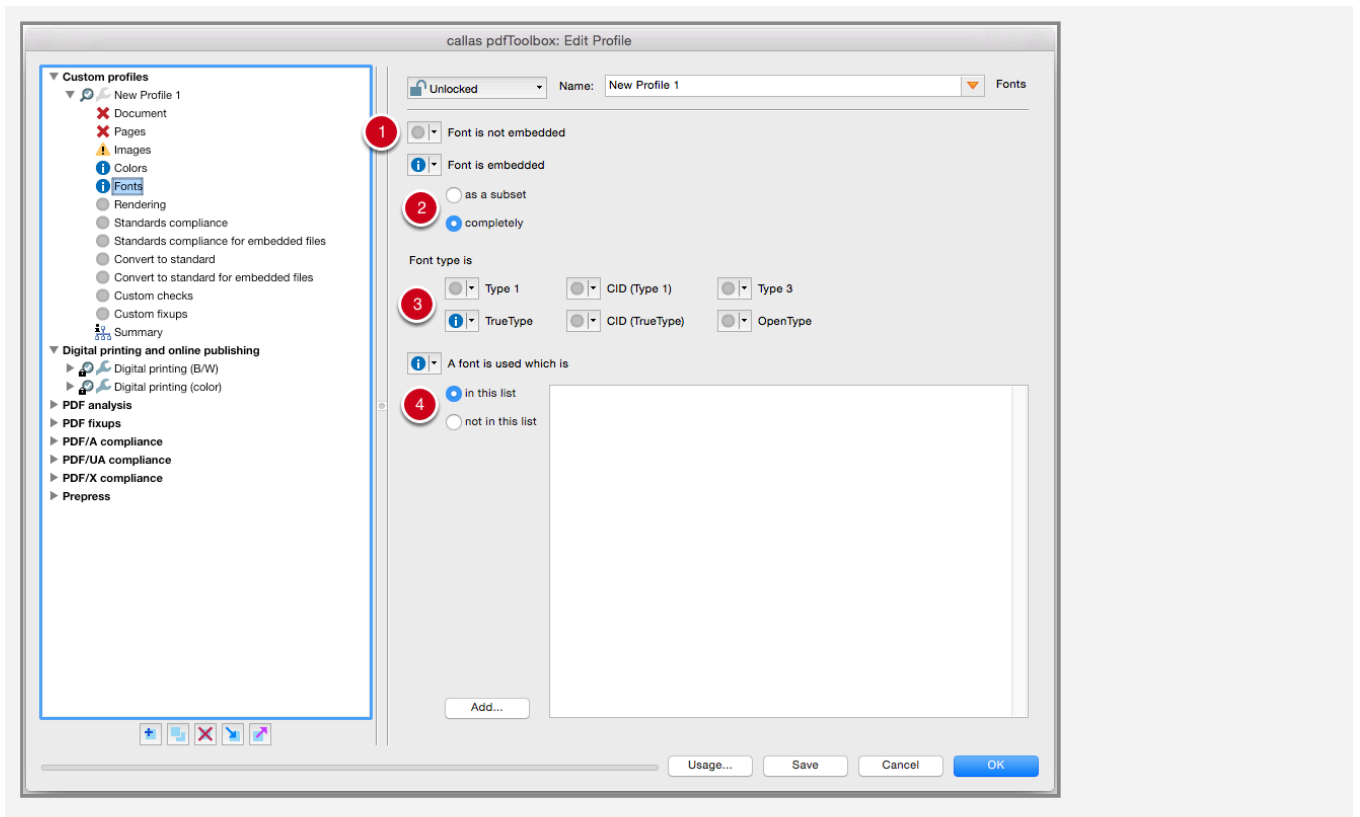
Edit Profile: Colors



The Color section offers the following checks:

1. **When the pages in this document are separated they will generate** lets you determine whether **Cyan, Magenta or Yellow plates** will be produced (which may, for example, be undesirable for a document which needs to be output in black/white plus spot color). You can also specify an upper limit for the **number of spot colors**.
2. The category **Objects on the page use** deals with color spaces. This allows you to query the following two items: **RGB** and **Device independent color (ICC, Lab)**.
3. Under **Objects on the page use spot colors**, you can make use of a **custom spot color list**. Messages can either be generated when the spot colors used agree with those on the list (the **in this list** option) or in case of a disagreement (when the **not in this list** option is selected).
4. The **Naming of spot colors is inconsistent** option allows you to identify issues such as incorrect or duplicated spot color names.

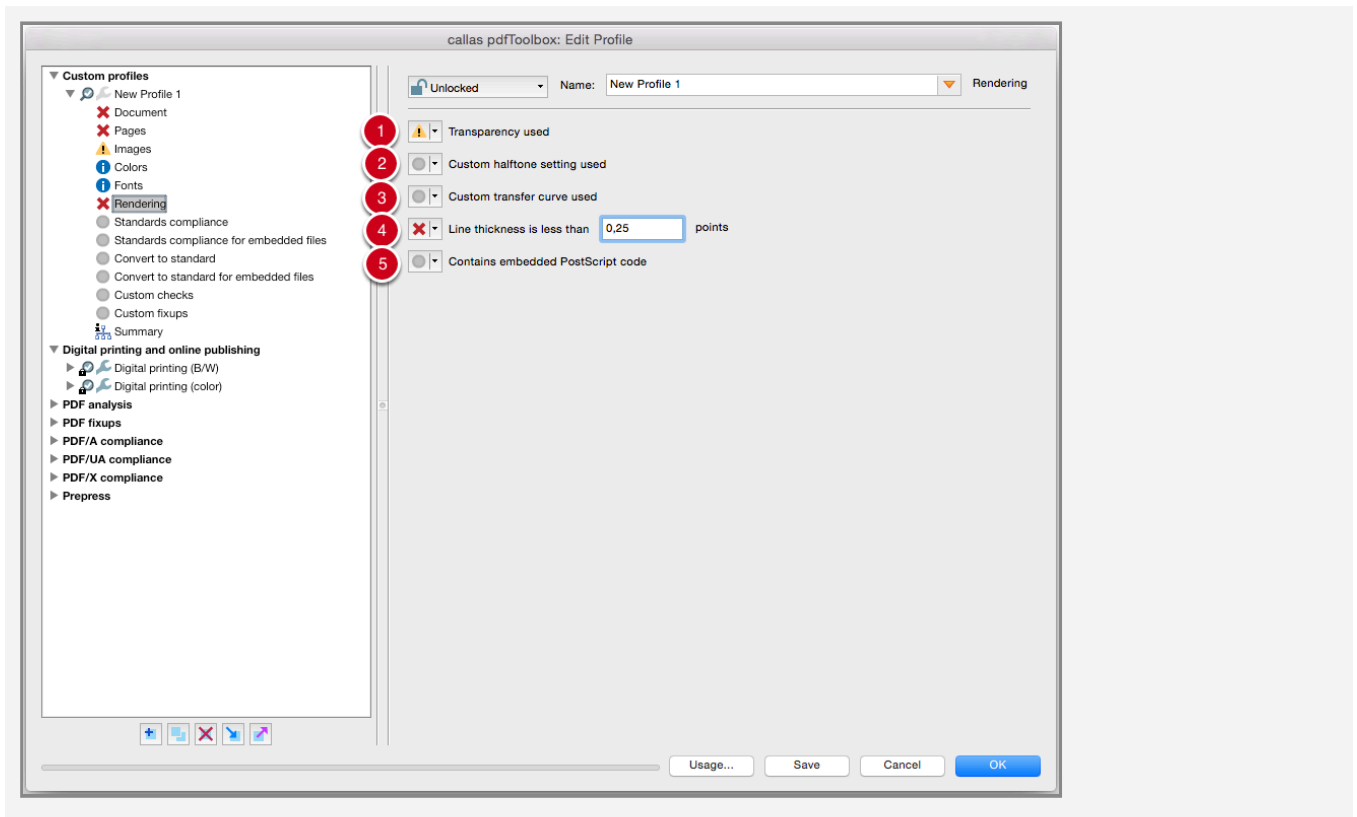
Edit Profile: Fonts



The Fonts section deals primarily with font embedding and the font type.

1. The first category allows you to determine whether any fonts are **not embedded**.
2. If fonts are **embedded**, you can test two conditions: as a **subset** or **completely**.
3. The **Font type** is option differentiates between six possible types which can be tested separately.
4. Finally, you can also use **custom lists** to check fonts, either to include or exclude them.

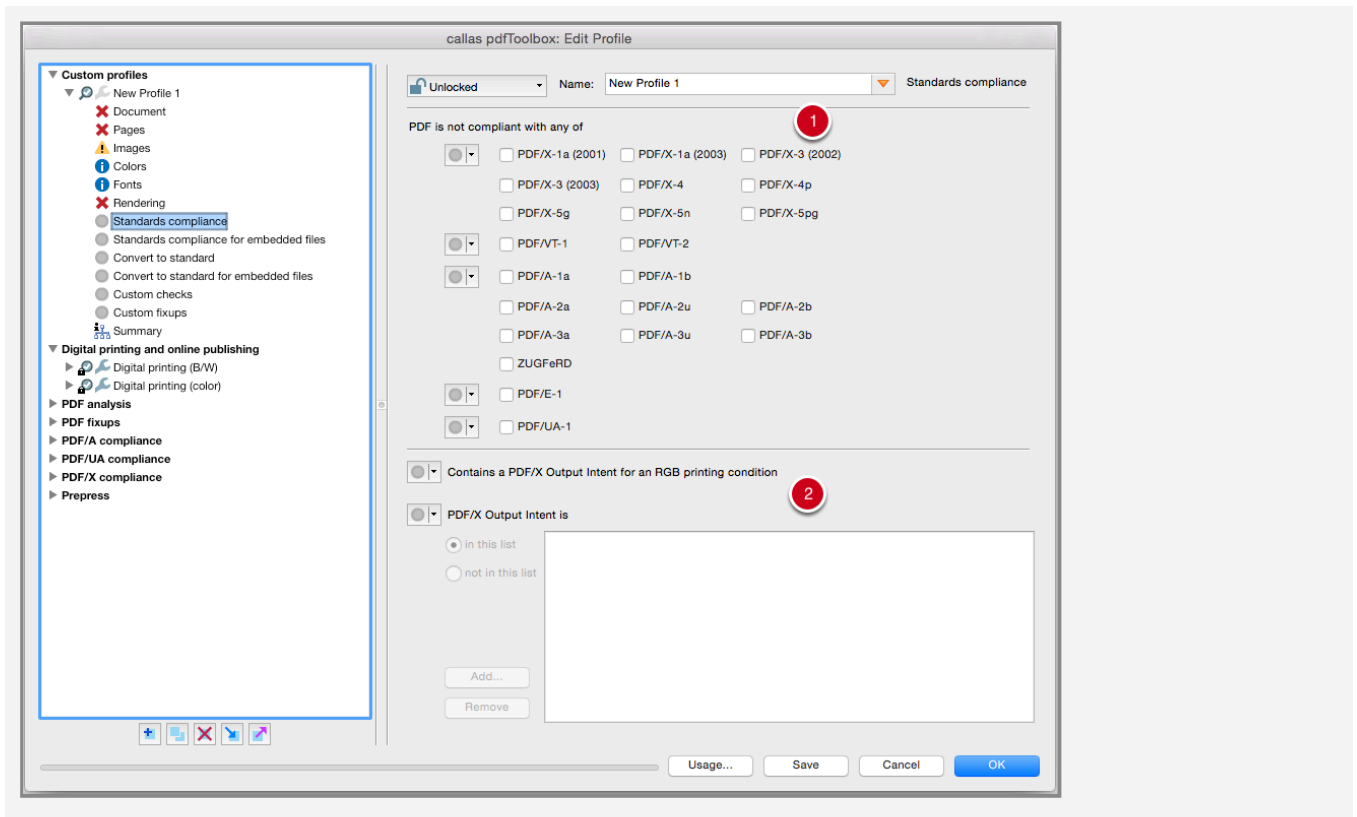
Edit Profile: Rendering



The **Rendering** section includes the following tests:

1. **Transparency used** detects elements that use transparency. This includes shadowing.
2. **Custom halftone setting used** finds unconventional halftone settings.
3. **Custom transfer curve used** finds unconventional transfer curve settings.
4. Under **Line thickness is less than ... points**, you can specify a threshold. Lines that are too thin can sometimes “break down” when printed.
5. The **Contains embedded PostScript code** option can be used to avoid potential printing issues and other problems.

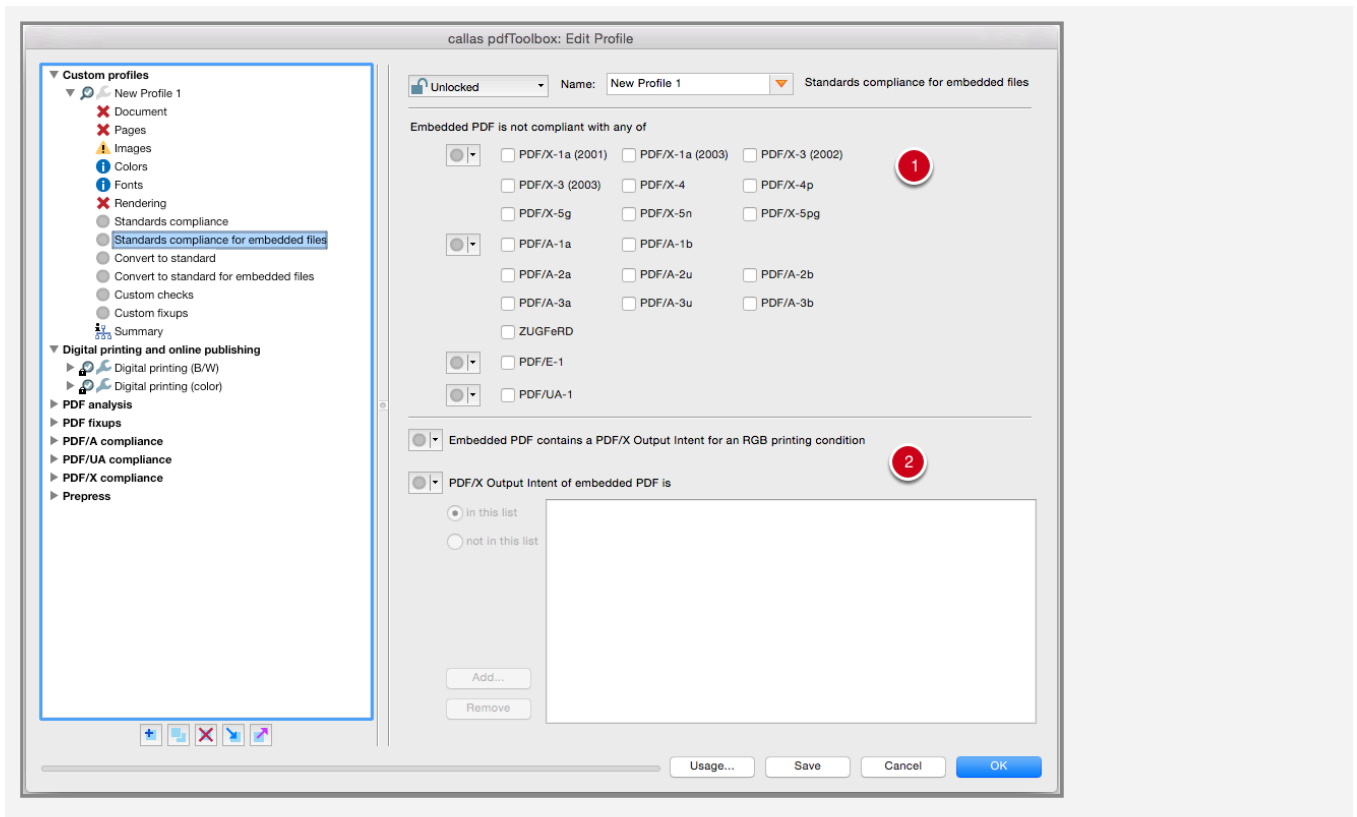
Edit Profile: Standards compliance



The **Standards compliance** section relates to the various ISO standards available for the Portable Document Format.

1. The category **PDF is not compliant with any of** allows you to integrate all currently available PDF standards into the query.
2. The second section is reserved for PDF/X, the PDF standard for prepress. Here, you can determine whether a PDF contains a **PDF/X Output Intent for an RGB printing condition**. An embedded PDF/X Output Intent can also be compared against a custom list of Output Intents.

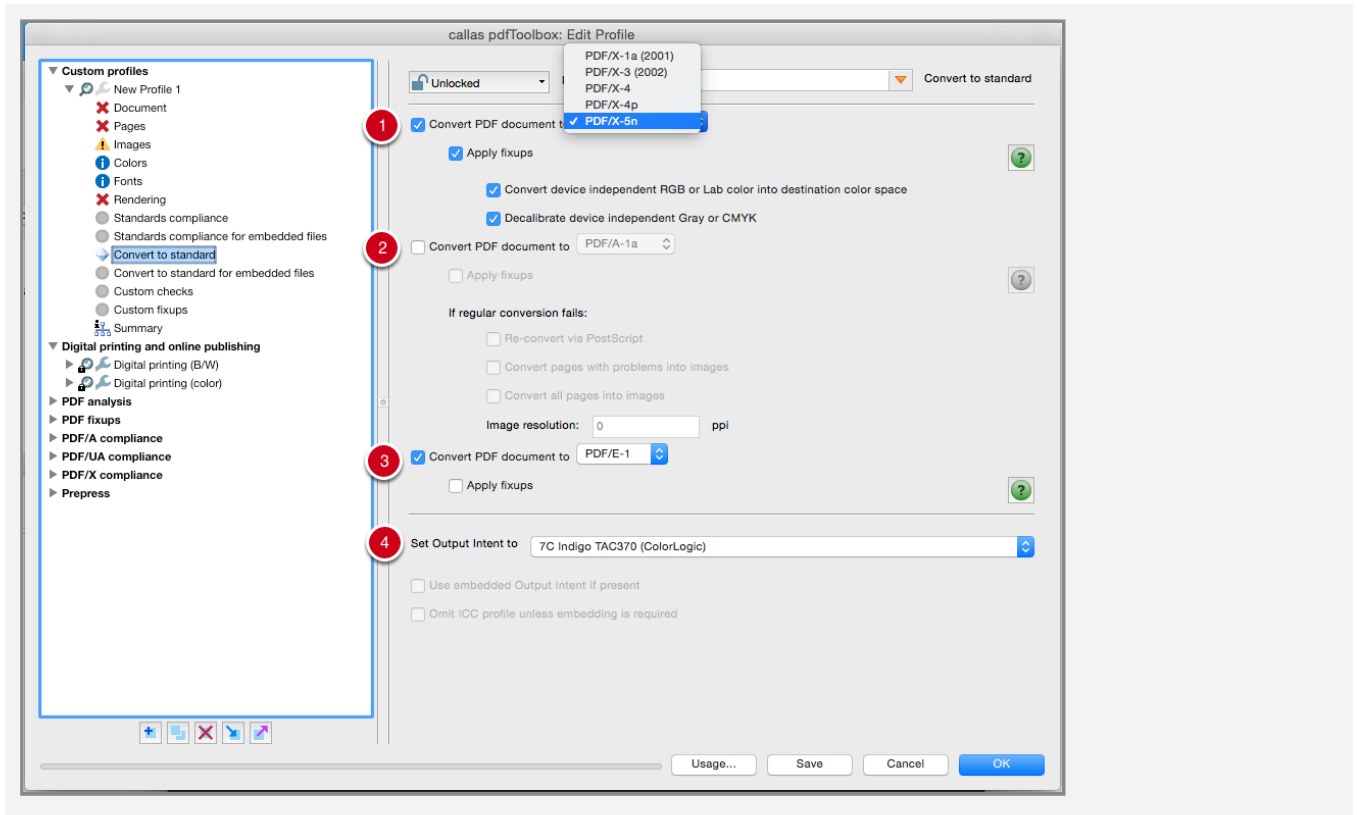
Edit Profile: Standards compliance for embedded files



The **Standards compliance for embedded files** section offers the same parameters as in the previous section. In this case, however, they relate to embedded PDF files:

1. The **PDF is not compliant with any of** option again relates to all currently available PDF standards.
2. The second section again turns to the **PDF/X** standard for PDF printing.

Edit Profile: Convert to standard



The **Convert to standard** section is particularly extensive.

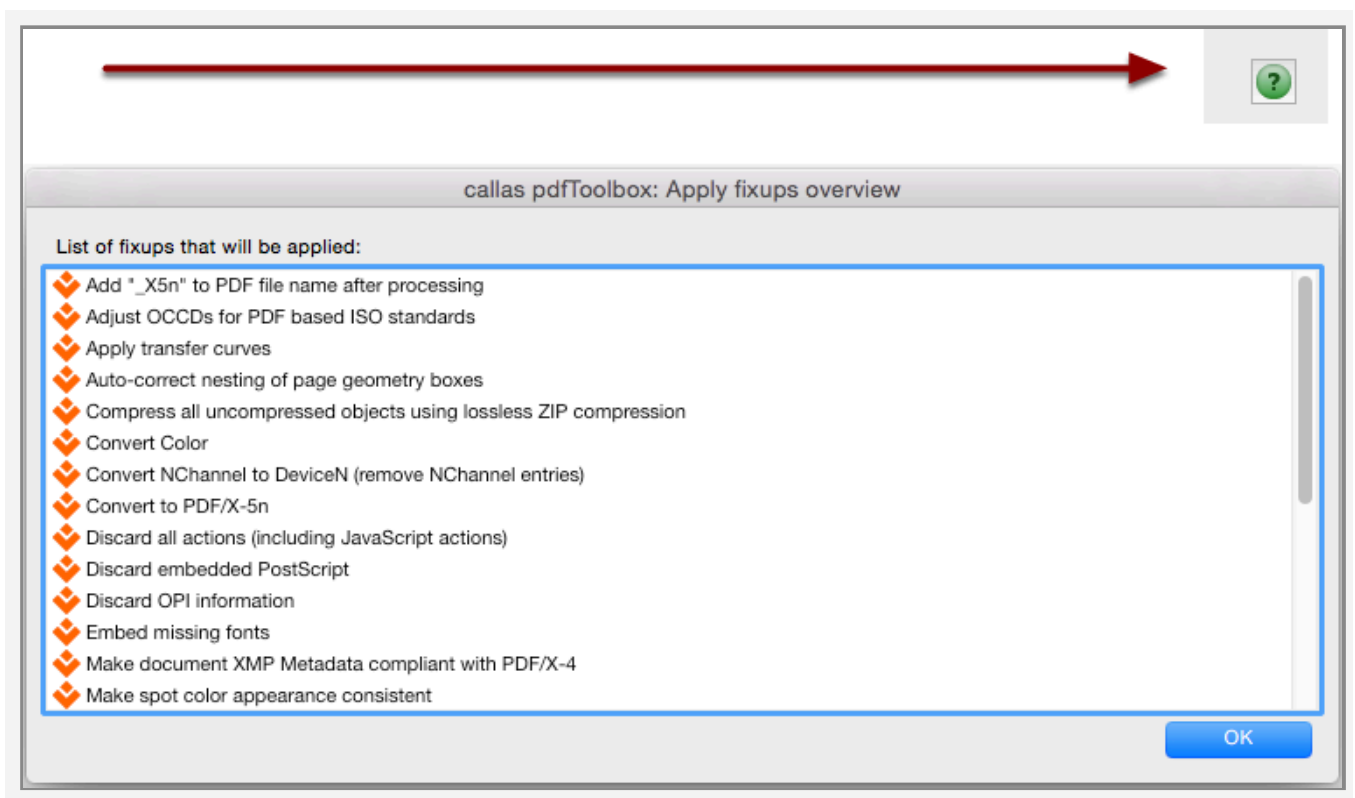
It is divided into conversion parameters for PDF/X (prepress), PDF/A (long-term archiving) and PDF/E (digital construction drawings).

This section converts and fixes PDF files, rather than simply checking them.

1. Under **Convert PDF document to PDF/X**, you can choose from a range of PDF/X versions from a pull-down menu. You can optionally **Apply fixups** by checking the box. Depending on the standard selected, two other options will be available: **Convert device independent RGB or Lab color into destination color space** and **Decalibrate device independent Gray or CMYK**.
2. Under **Convert PDF document to PDF/A**, you can choose from a range of PDF/A versions from a pull-down menu. The **Apply fixups** box can be checked. **If regular conversion fails**, a number of additional options are available. **Re-convert via PostScript**, **Convert pages with problems**

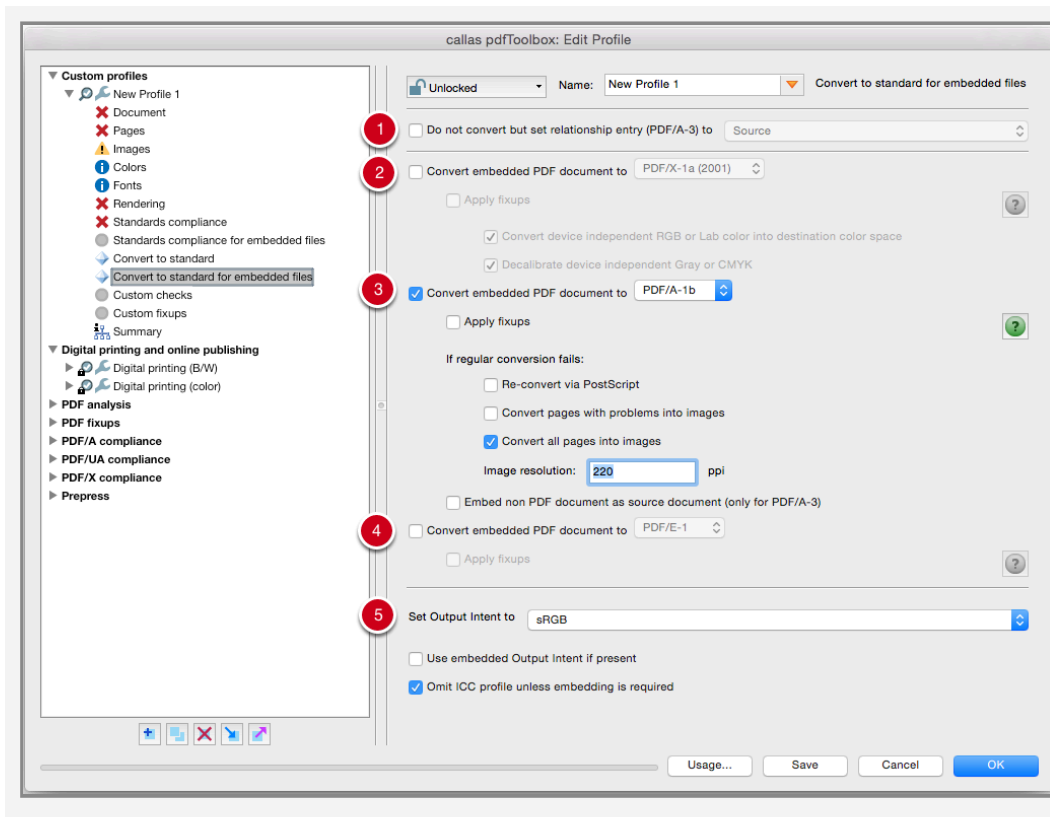
into images, or Convert all pages into images. The last two of these options allow you to specify an Image resolution.

3. The third standard is PDF/E. You can again Apply fixups here.
4. Finally, you can also select the Output Intent from a pull-down menu. Two options are available here: Use embedded Output Intent if present and Omit ICC profile unless embedding is required (e.g. with modern Output Intents; this allows you to reduce the size of the resulting file to the size of the Output Intent).



To learn more about the Fixups that will be applied during each conversion type, click on the green question mark button to open the corresponding list. (The button is gray, meaning inactive, when a conversion type is also inactive.)

Edit Profile: Convert to standard for embedded files

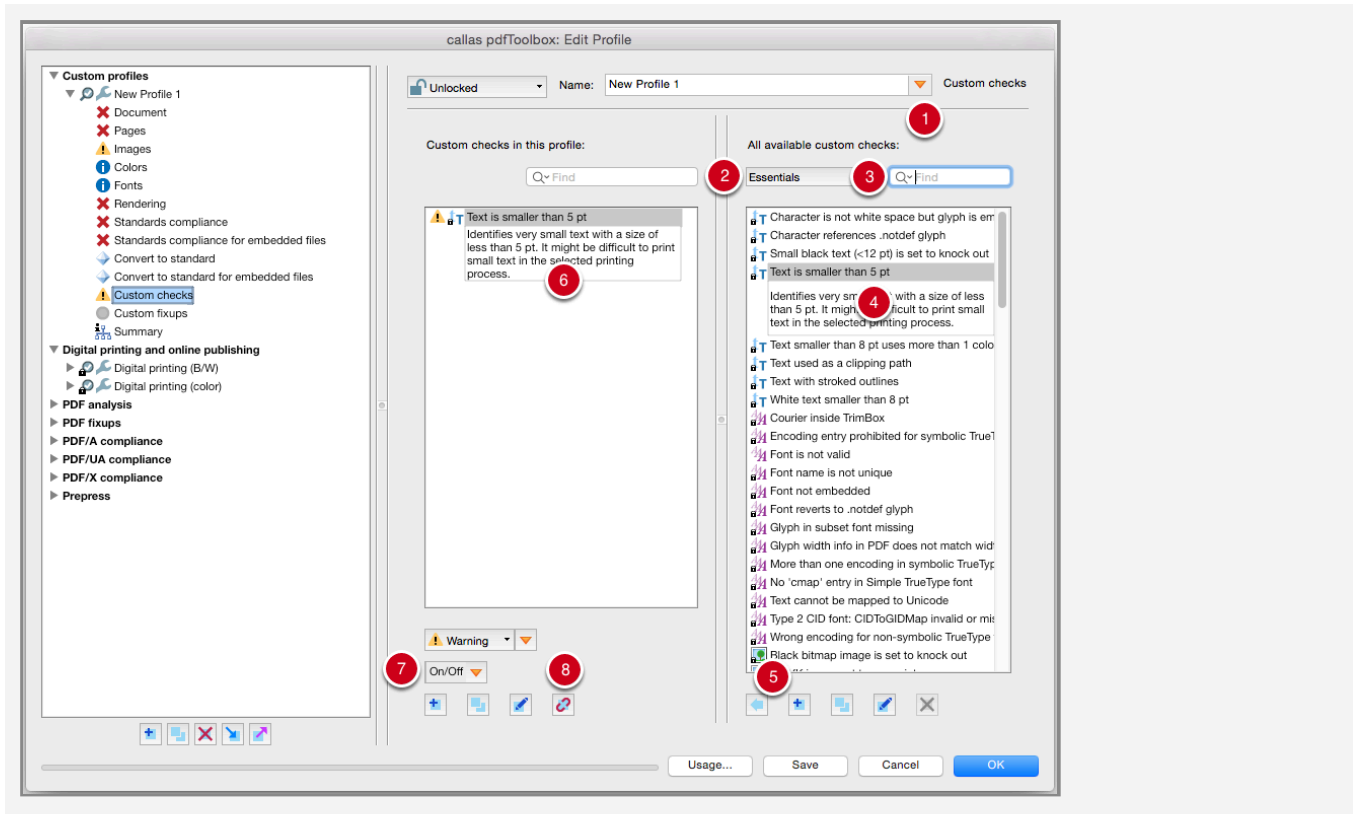


Convert to standard for embedded files offers very similar parameters to those shown in the previous step, but in this case they relate to embedded PDF files.

1. The first section deals with the special case of PDF/A-3. This PDF standard also allows you to embed files left in their original state (not just PDF, but also other formats). **Do not convert but set relationship entry (PDF/A-3) to ...** allows you to define the relationship in one of a number of different ways, including **Source**, **Data**, **Supplement** and others.
2. **Convert embedded PDF document to PDF/X** enables a number of different options as well as **optional fixups**.
3. **Convert embedded PDF document to PDF/A** enables a number of different options as well as **optional fixups**. In addition, there are also the same backup options as before to determine how to proceed if regular conversion fails.
4. **Convert embedded PDF document to PDF/E** plus **optional Fixups**.

5. Output Intent settings.

Edit Profile: Custom checks



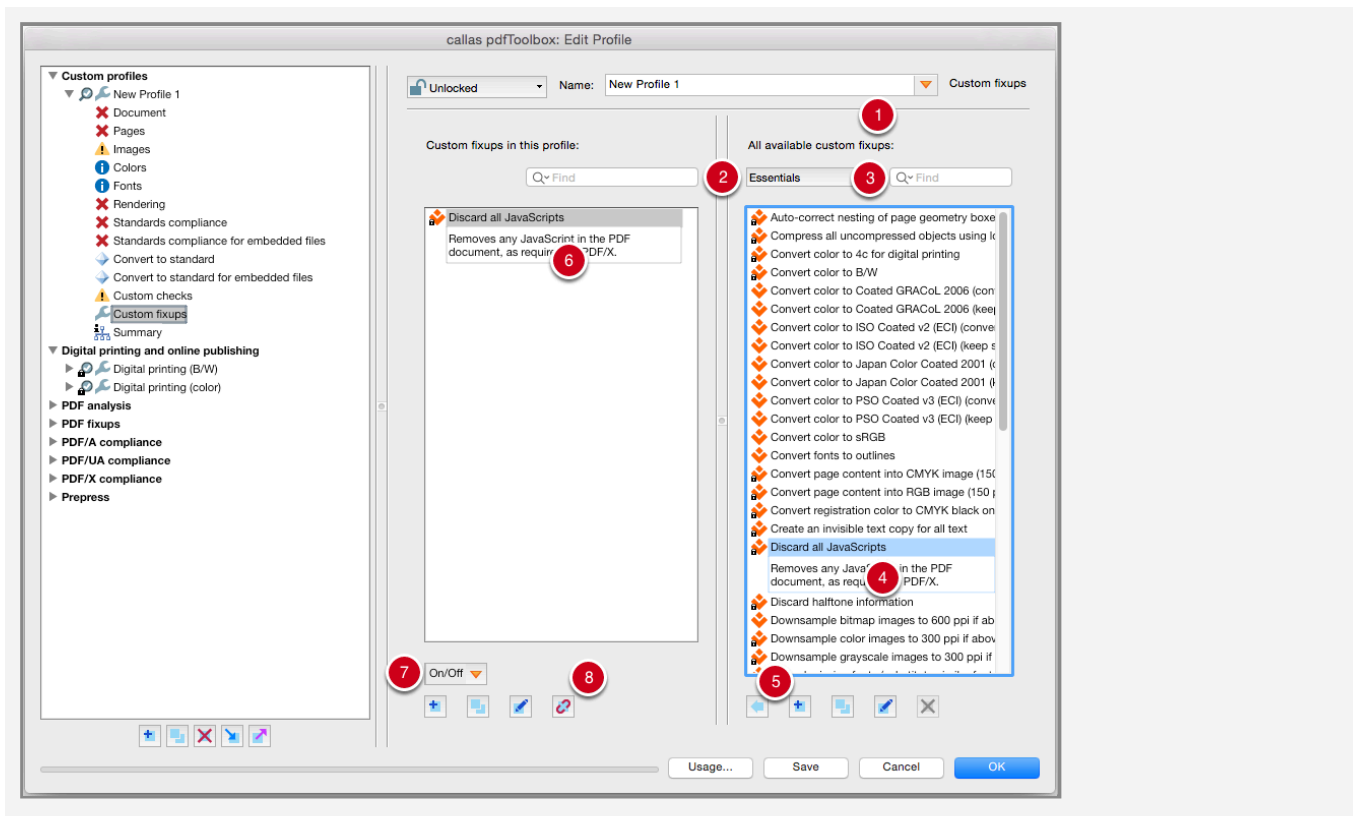
Under **Custom checks**, you can select entries from the list of all available pdfaPilot checks and integrate them into the new profile.

The window is structured as follows:

1. The right hand side shows the **list of Checks available** in pdfaPilot. It can be considered the “inventory” available to you.
2. The checks shown will be those available in the **currently selected library** (in this example, Essentials).
3. If you know the name of a given function, you can use the **Search tool** to make it easier to find.
4. Click on a check in the list to see more **detailed information**.
5. Click on the **blue arrow symbol to the left** to move the selected check to the middle column. This will add it to the profile you are currently creating or editing.
6. The middle column shows **all Checks used in the current Profile**.

- You can use the pull-down menu to specify whether the check should return an **Error**, a **Warning** or an **Info** message. You can also use the orange triangle to add and manage **variables and scripts**.
- The buttons allow you to **add, duplicate, edit and delete** checks.

Edit Profile: Custom fixups



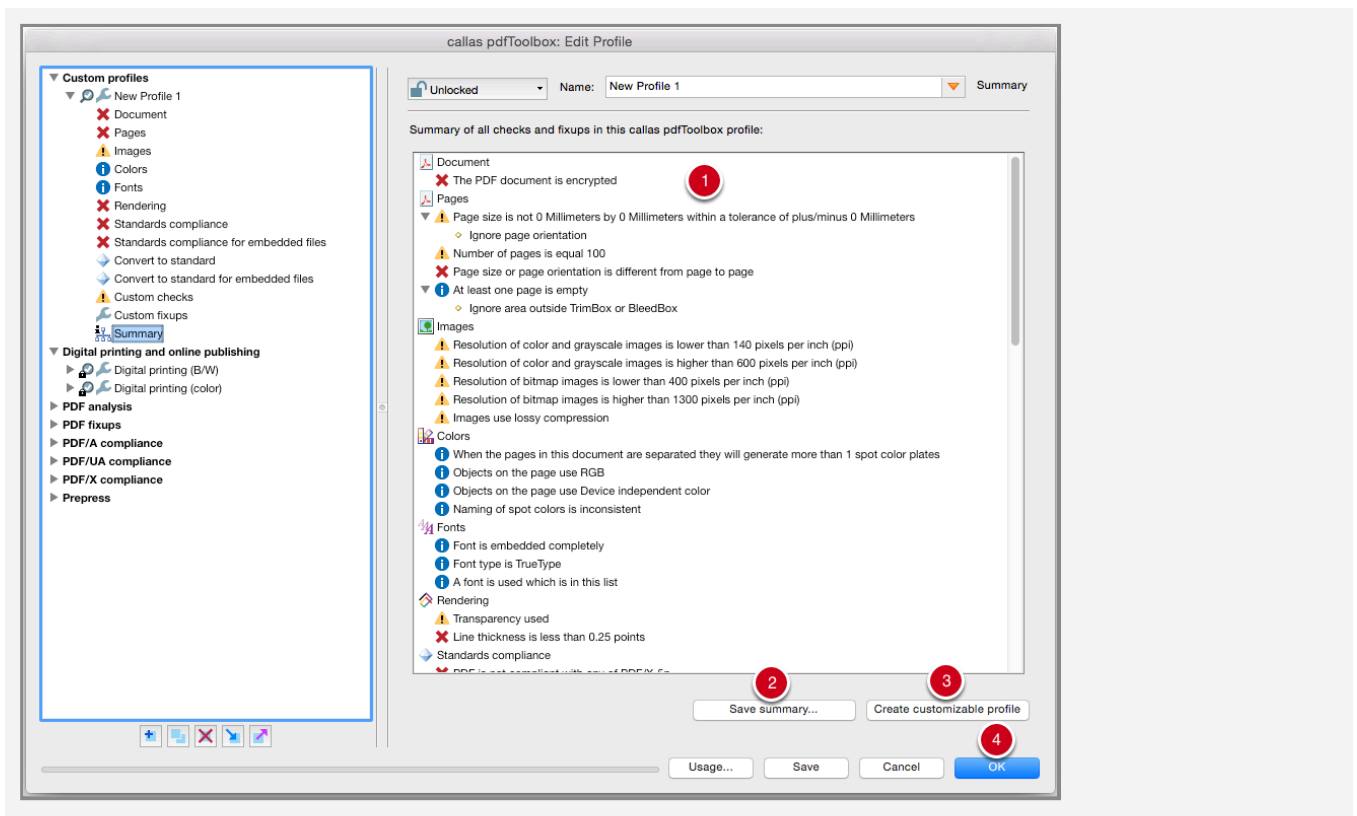
Under **Custom fixups**, you can select entries from the list of all available pdfaPilot fixups and integrate them into the new profile.

The window is structured as follows:

- The right hand side shows the **list of Fixups available** in pdfaPilot. It can be considered the “inventory” available to you.
- The Fixups shown will be those available in the **currently selected Library** (in this example, Essentials).
- If you know the name of a given Fixup, you can use the **Search tool** to make it easier to find.
- Click on a Fixup in the list to see more **detailed information**.

5. Click on the **blue arrow symbol to the left** to move the selected Fixup to the middle column. This will add it to the Profile you are currently creating or editing.
6. The middle column shows **all Fixups used in the current Profile**.
7. You can use the pull-down menu to specify whether the check should return an **Error**, a **Warning** or an **Info** message. You can also use the orange triangle to add and manage **variables and scripts**.
8. The buttons allow you to **add, duplicate, edit and delete** Fixups.

Edit Profile: Summary



In the **Summary**, you can see a detailed overview of all Checks and Fixups included in the newly created (or edited) Profile.

1. The entries are grouped by section, i.e. **Document, Pages, Images, Colors, Fonts** and so on.

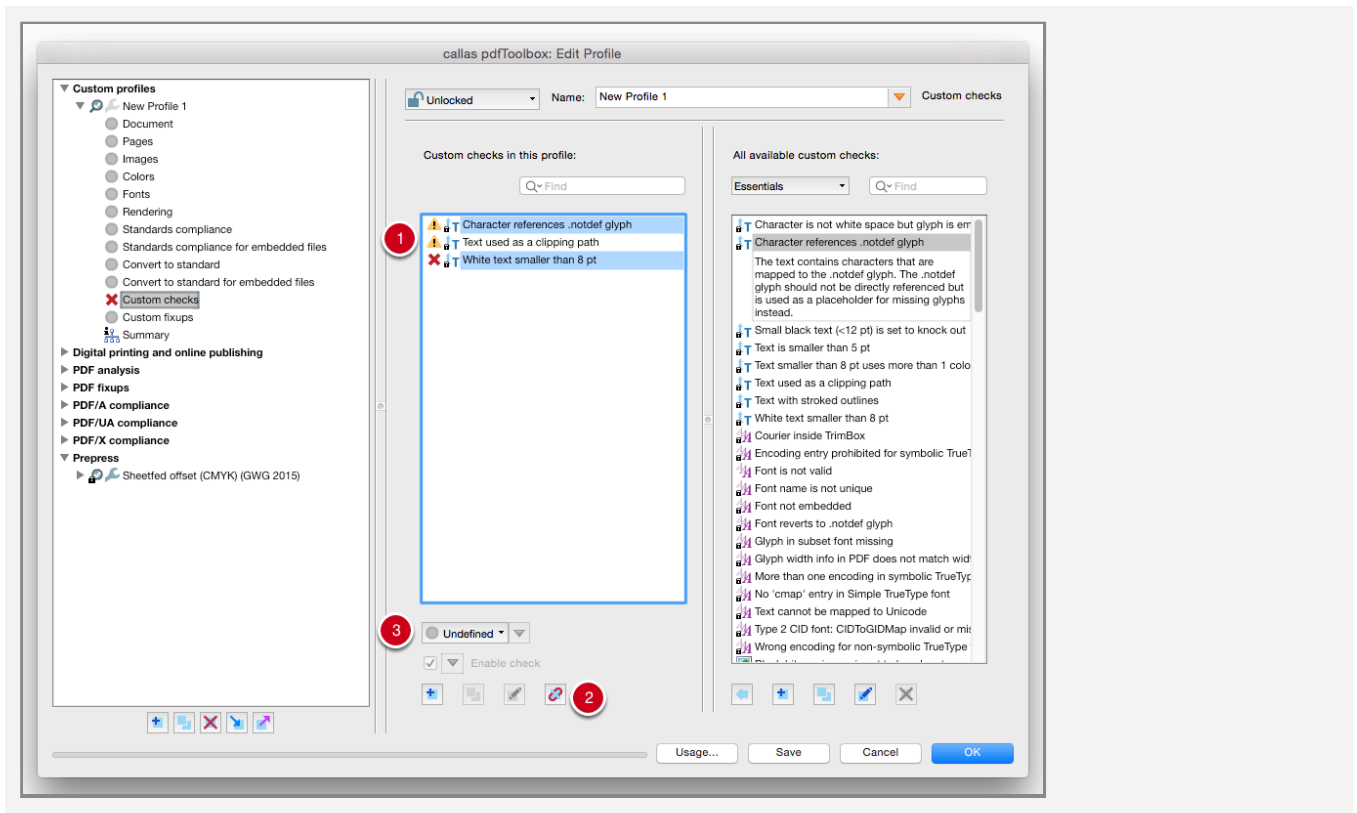
2. *Optional:* You can **Save** the profile overview. Click on the corresponding button to have the program save a **comprehensive report in PDF format**.
3. *Optional:* You can also **generate a customizable profile** based on the current Profile - a duplicate, in other words - which you can configure further.
4. Click on the **OK button** to finish setting up the Profile.

3.5 Deleting multiple Fixups or Checks at once

When you edit a profile, you can select multiple Fixups or Checks, deleting them all at once or adding more.

The screenshots are showing the sister product pdfToolbox.
The functionality in this part of pdfaPilot is identical to pdfToolbox.

Selecting multiple Fixups or Checks in the “Edit Profile” window



When setting up or modifying a profile, you can select and then delete (or change, or add) multiple Checks or Fixups at once.

1. To do so, simply select the desired Checks/Fixups in the desired column by pressing and holding **Shift** (for consecutive items) or **Ctrl** (for non-consecutive items).
2. You can then **Add / Delete** the selected items.
3. When selecting multiple Checks at once, you can also **change the message type** (Info, Warning, Error) for all of them at once.

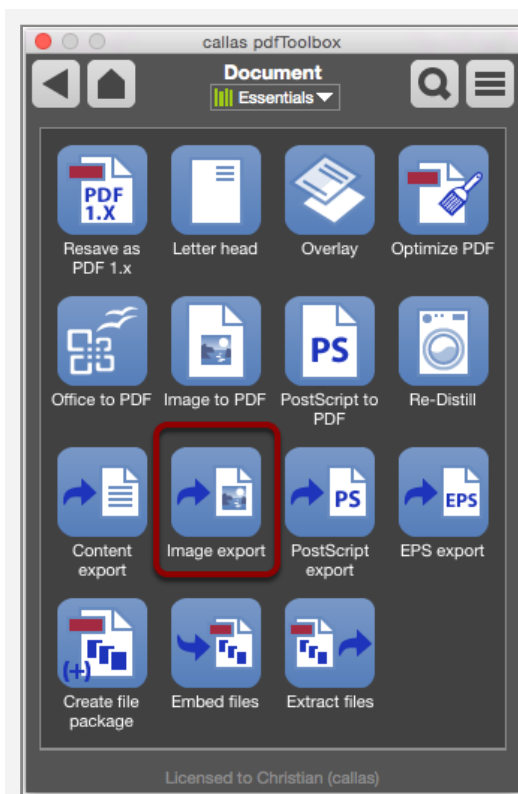
3.6 Actions in Process Plans

As well as containing Profiles, Checks and Fixups, Process Plans can also include some Actions. In this chapter, we'll show you how this can be done.

Note: Not all Actions can be used in Process Plans. Primarily, the Actions available are those which are required for certain workflows, such as for imposition.

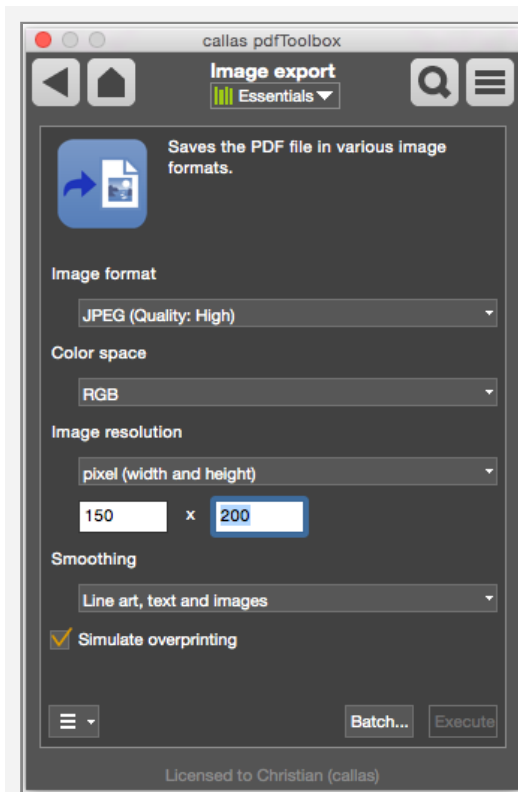
The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

Prepare the Action in the Switchboard

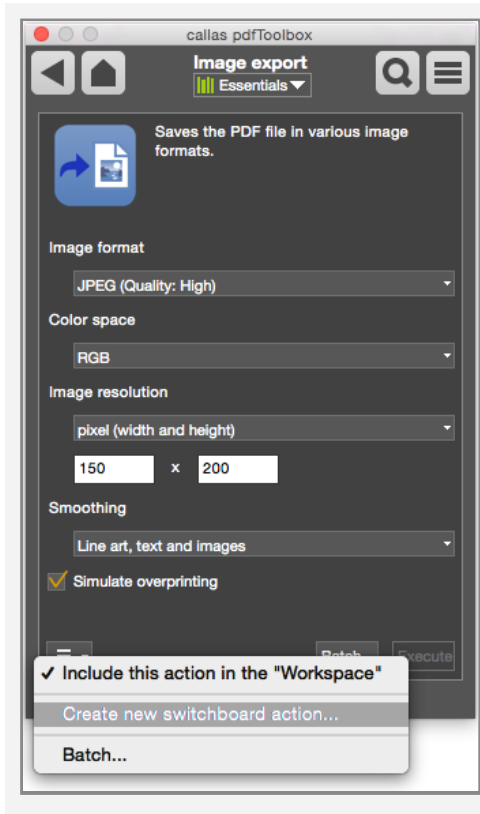


For our example, we will prepare the Image Export Action for use in a Process Plan.

This Action is a part of the **Document** group, which is contained in the **Essentials** library and elsewhere.

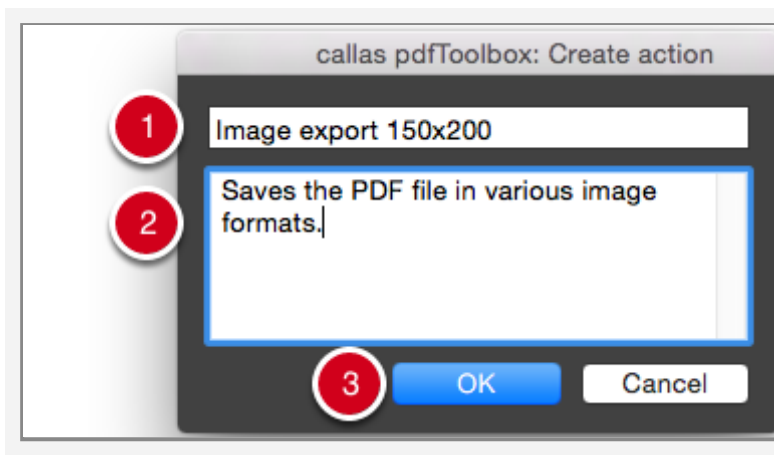


Here you can specify **Settings** for the Action which will later be used in the course of the Process Plan. (Among other things, the saved image file should have a fixed size; here, 150 x 200 pixels.)



The Flyout menu at the bottom-left provides a range of options for the Action.

Here you can also find the Create New Switchboard Action... command.

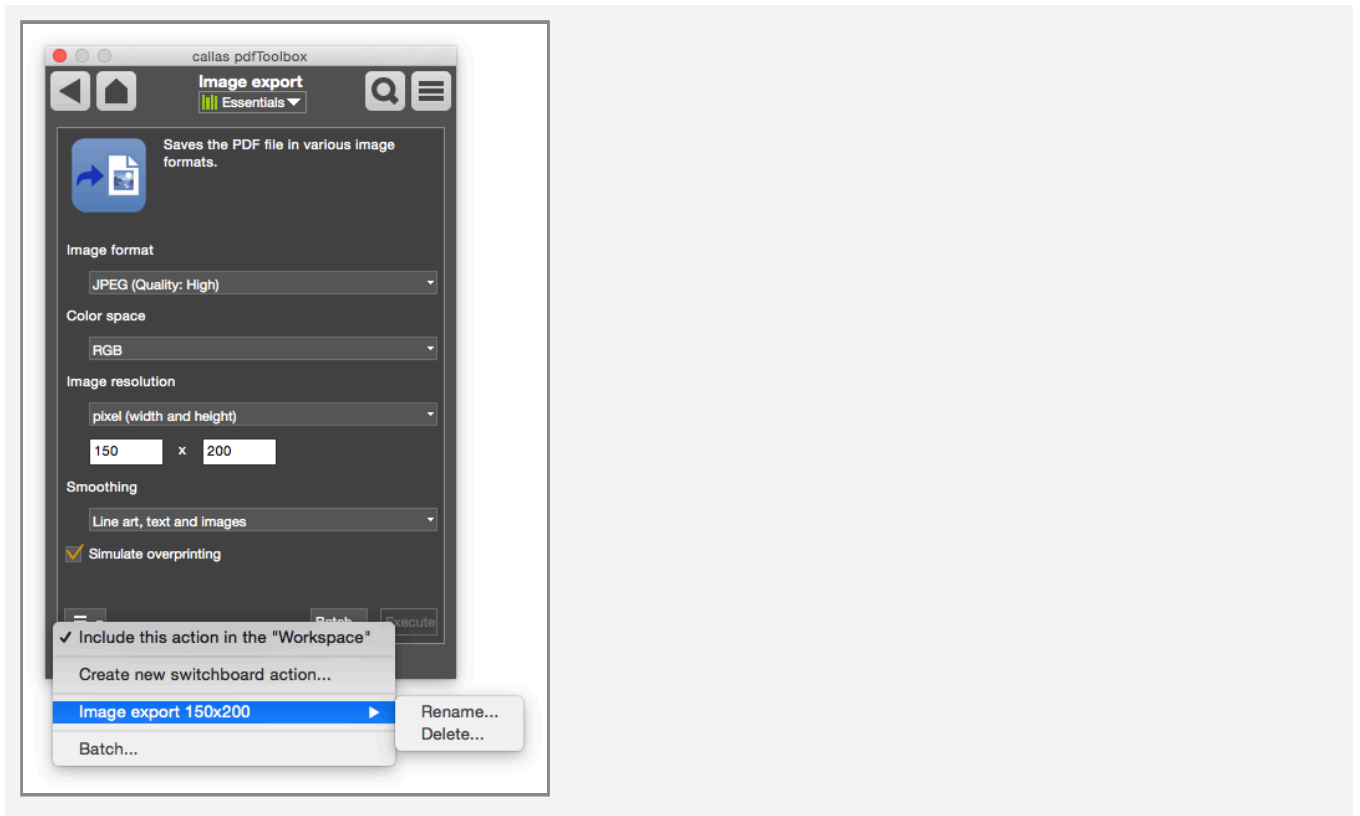


This will open the Create Action dialog.

Here, if required you can...

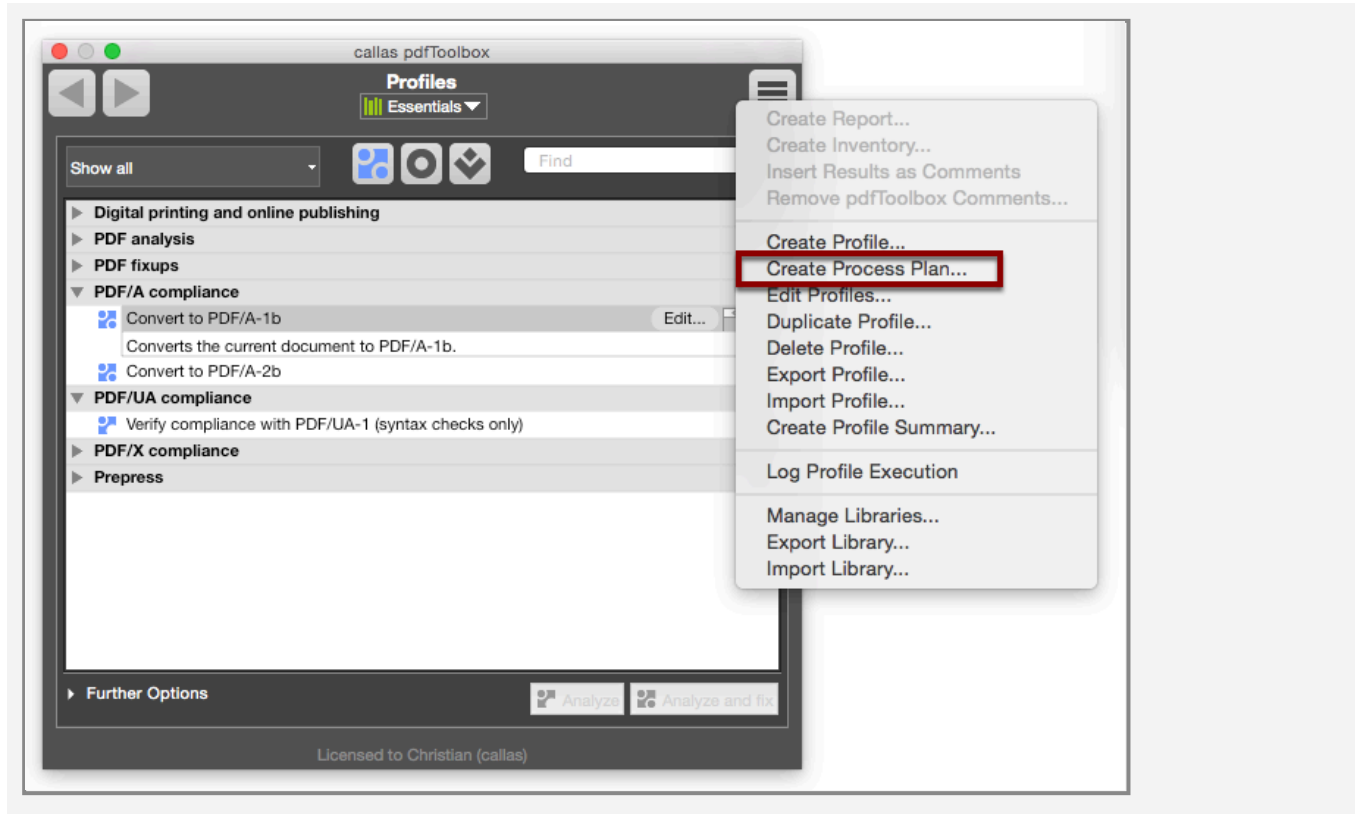
1. ... Change the Name...
2. ... And edit the Description.

3. Click on the **OK** button to finish creating the action.



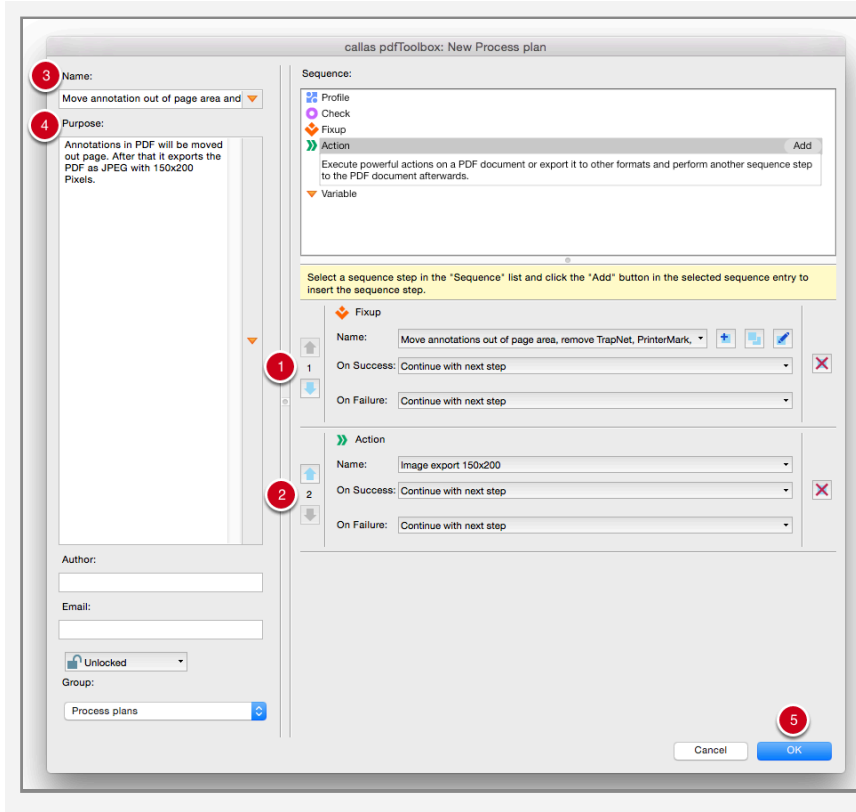
The newly created Switchboard Action **Image Export 150x200** can now also be accessed through the flyout menu. (You can also rename or delete it from here.)

Create a new Process Plan with the newly configured Action



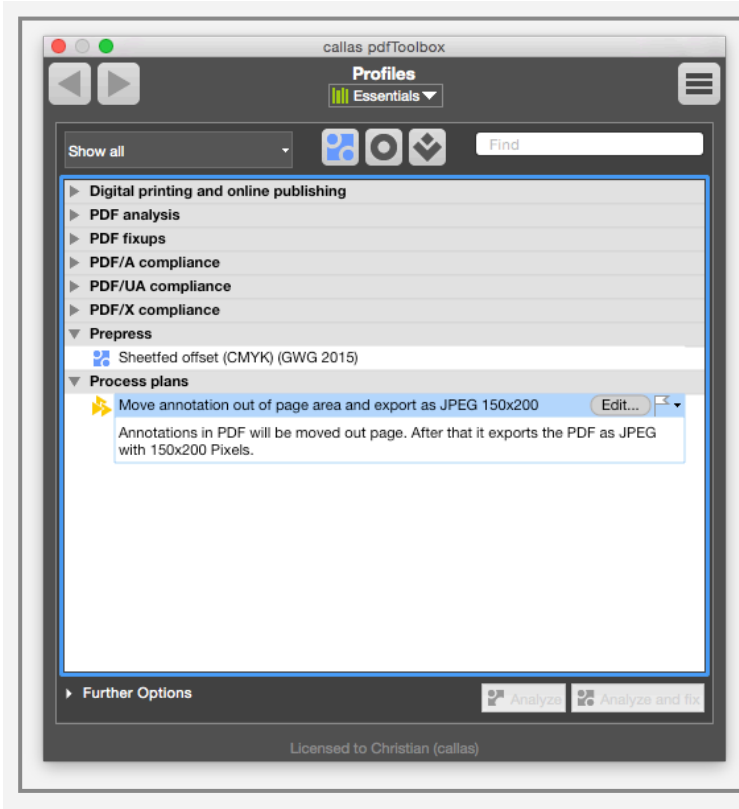
To create a new Process Plan, switch to the Profile window.

Here, from the pull-down menu at the top-right, you can see commands including Create Process Plan....



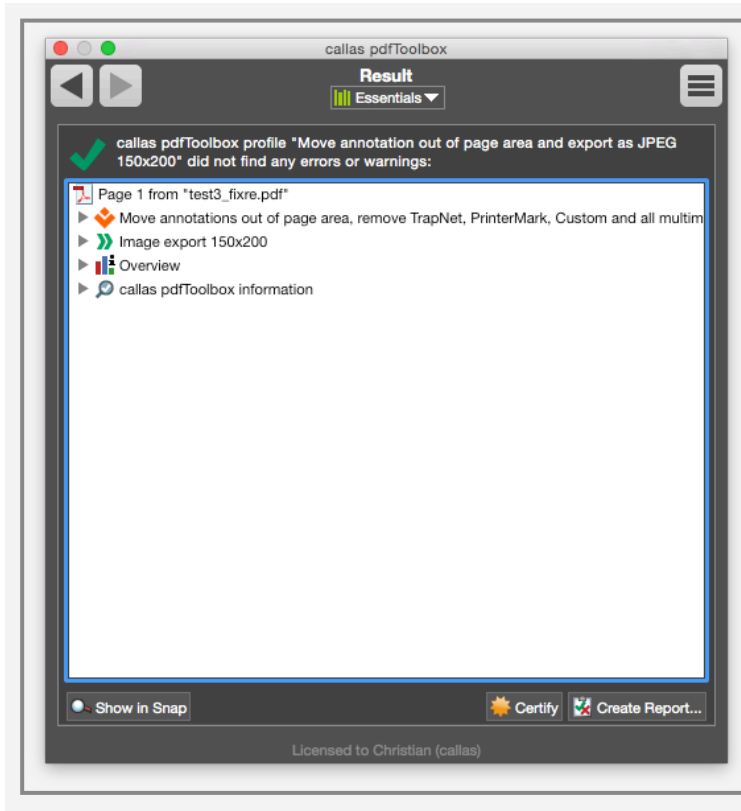
Process Plans allow you to use sequences from the following categories:

- Profile
 - Check
 - Fixup
 - **Action**
 - Variable
1. In the example shown, we should first use a **Fixup** to move any annotations in a PDF outside of the page area. (For each sequence step, you can specify what to do in case of success or failure.)
 2. The second step uses the **Action** we prepared earlier. Under the Name category, the Action **Image Export 150x200** can be selected.
 3. The new Process Plan should be given a meaningful **Name**.
 4. It is also useful to add a **Comment**. This will appear later in the Profile list when the Profile is activated.
 5. Click on the **OK button** to save the Process Plan.



The new Process Plan with the associated Action is now available in the Profile window under the name **Move annotations and export JPEG 150x200**.

Click **Check and fix-up** to execute the Process Plan.



The Results window shows all sequence steps executed by the Process Plan, including the integrated Action.

Switchboard Actions available as Actions for Process Plans

Not all Switchboard Actions are available as Actions for Process Plans. The reason is, that some Switchboard actions are based on Profiles of Fixups.

The following Switchboard Actions can be used as Actions for Process Plans:

- Arrange:
 - Booklet
 - N-Up
 - Fill page
 - Impose
 - Reader spreads
 - Split in half
 - Step & Repeat
 - New page

- Present:
 - Handout
 - Passe partout
- Document:
 - Overlay
 - Re-Distill
 - Content export
 - Image export
 - PostScript export
 - EPS export
- Prepress
 - Pre-separated pages

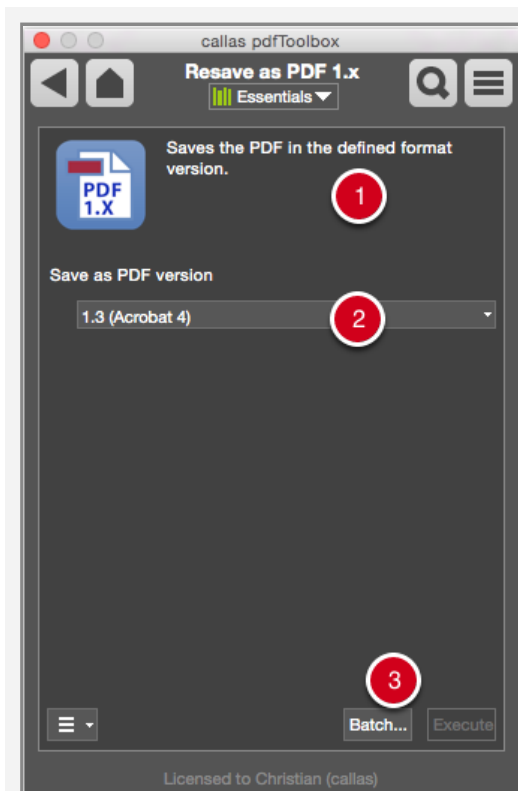
3.7 Batch processing for Actions

pdfaPilot Desktop can do more than just checking or fixing one file at a time; the Batch Processing mode allows it to process multiple PDF documents within a folder. The Switchboard allows you to apply any Action to all PDF files within a given folder.

The Batch function in pdfaPilot can process up to 100 documents one after another. For a high-volume approach using hot folders, pdfaPilot Server is recommended.

The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

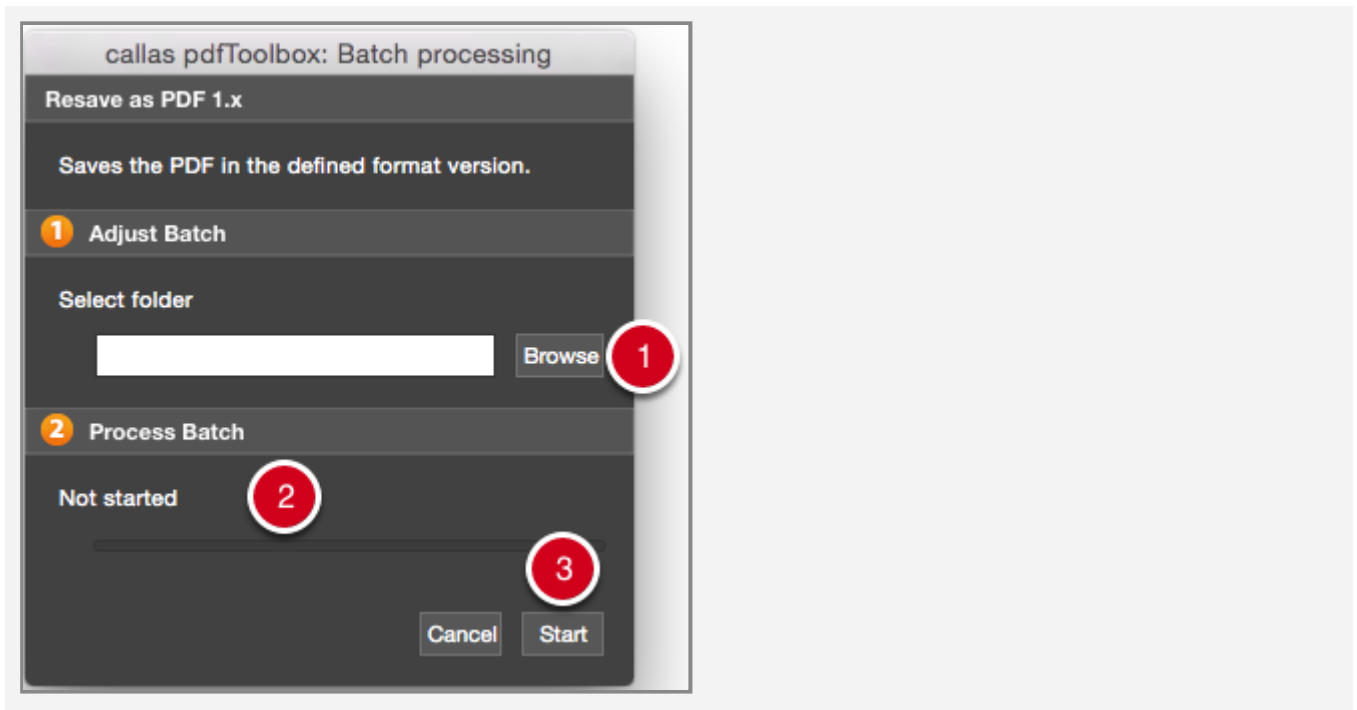
Select the Action from the Switchboard



1. Select the desired Action...

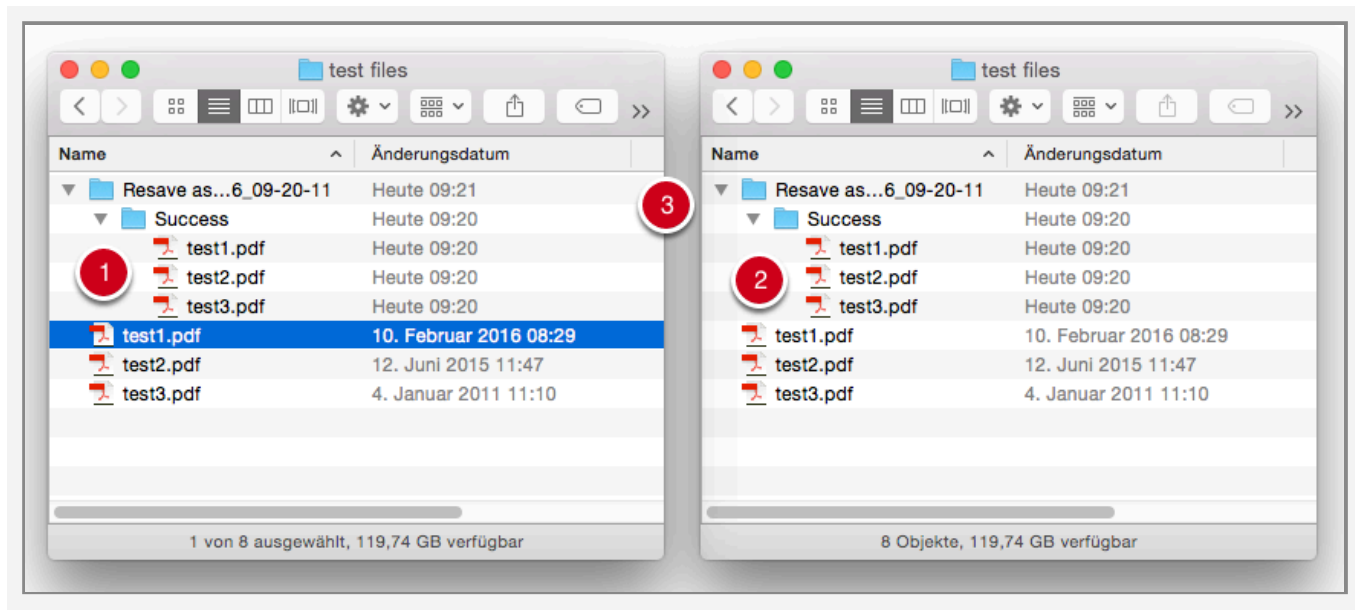
2. ... And apply any desired **Settings**.
(Here, under the Document category, the Action *Save as PDF 1.x* has been specified as *PDF Version 1.3*.)
3. Click on the **Batch** option to specify additional batch processing settings.

Batch processing: Organize and start the batch



1. Under **Heading 1**, select the folder containing the PDF files to be processed.
2. Under **2: Process batch** will display processing progress...
3. ... Once you click on the **Start** button.

After processing



Depending on the processing results, the files will then be stored either in a folder containing successfully processed (*Success* - 1) or unsuccessful results (*Failure* - 2).


These are located in a folder named with the *timestamp* - 3 for the processing point which will be automatically shown after processing is complete.

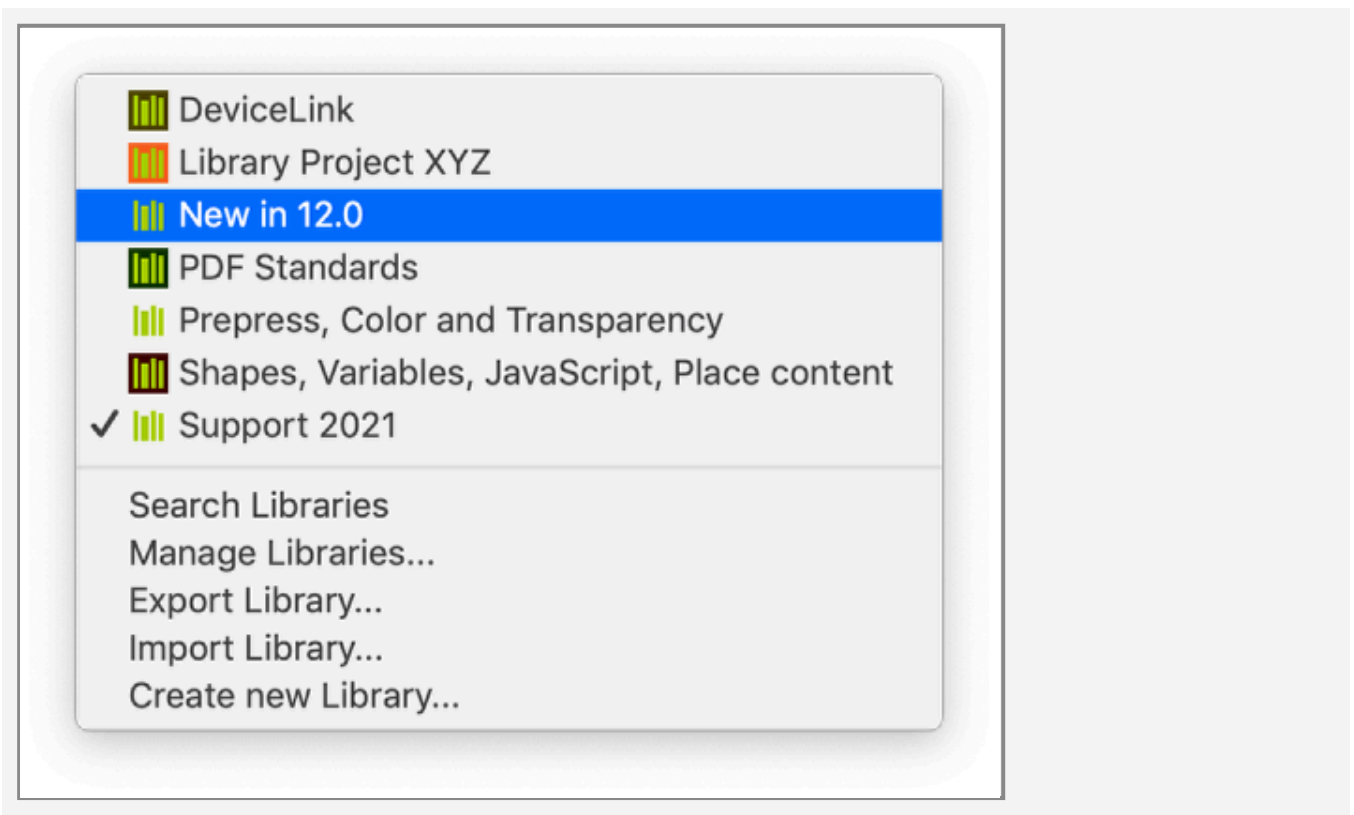
3.8 Libraries - Overview

What are Libraries?

Libraries are repositories for all the different items you can create in pdfaPilot (such as Checks, Fixups, Profiles, Process Plans, imposition layouts...).

You can use them to keep yourself organized on different projects, separate development from production profiles and much more.

 Please note that the screenshots below show the functioning of the sister product pdfToolbox. pdfaPilot works on the same grounds.



Use Libraries for versioning and project management

With pdfaPilot Libraries, versions and projects can easily be managed. Supposedly, a printing company has compiled records with Profiles, Checks and Fixups for a particular workflow or customer. Now it is possible to spread them and manage different versions across multiple workstations via Libraries.

1. For a customer or a workflow, a Library was created.
2. If the requirements change or if new pdfaPilot features are integrated, the existing Library can simply be duplicated, a new version number (possibly also with the current date) can be given and the additional Checks, Profiles and Fixups can be added.
3. The new, additional Library could then be called "customer-v1.1-2021-06-26" or "Printer v1.1-2021-06-26".
4. Further updates will be processed the same.

In this way, it is non-destructive and any changes in the procedure are tracked.

Use Libraries for task oriented workspaces

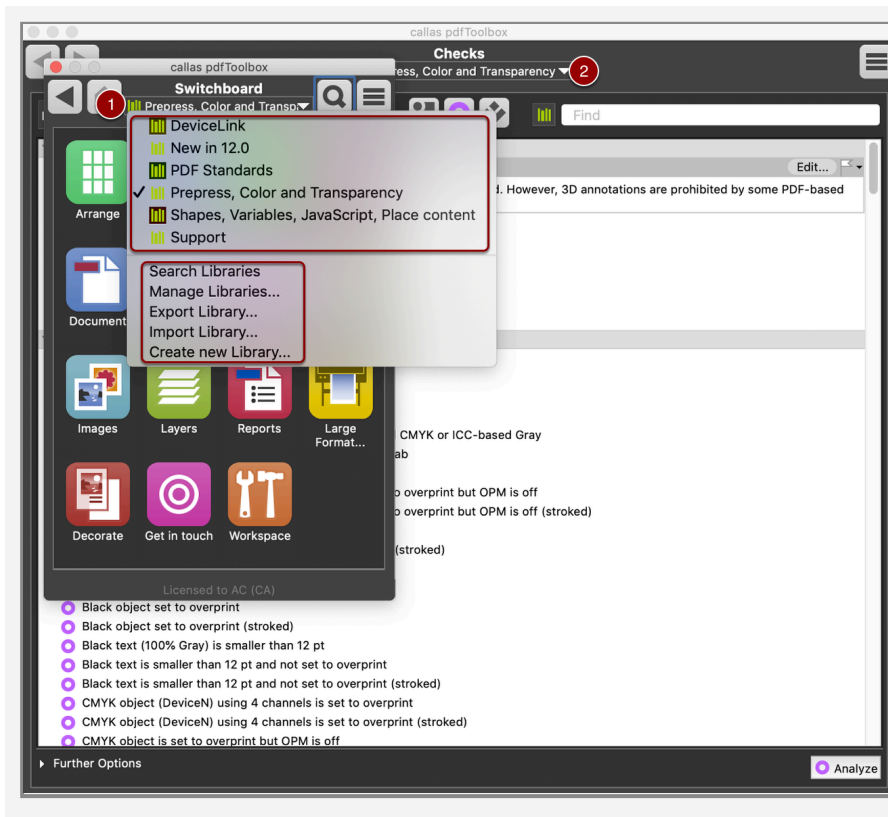
Libraries can be very useful for different employees in a company or for clients with whom you work, to get customized sets of Profiles, Checks and Fixups.

- It may be useful to first create an extensive Library, which you can adjust for the required workstations or partners.
- Smaller sets can be created as a Library for employees who require only certain functions.
- According to certain workflows, Libraries can be set up, such as for "Offset", "Digital", "Online PDF", "Large Format", and many more.

Where are the Libraries located?

You can access the Libraries via the dropdown (as shown below) in:

1. The 'Switchboard'
2. The main Profile pane



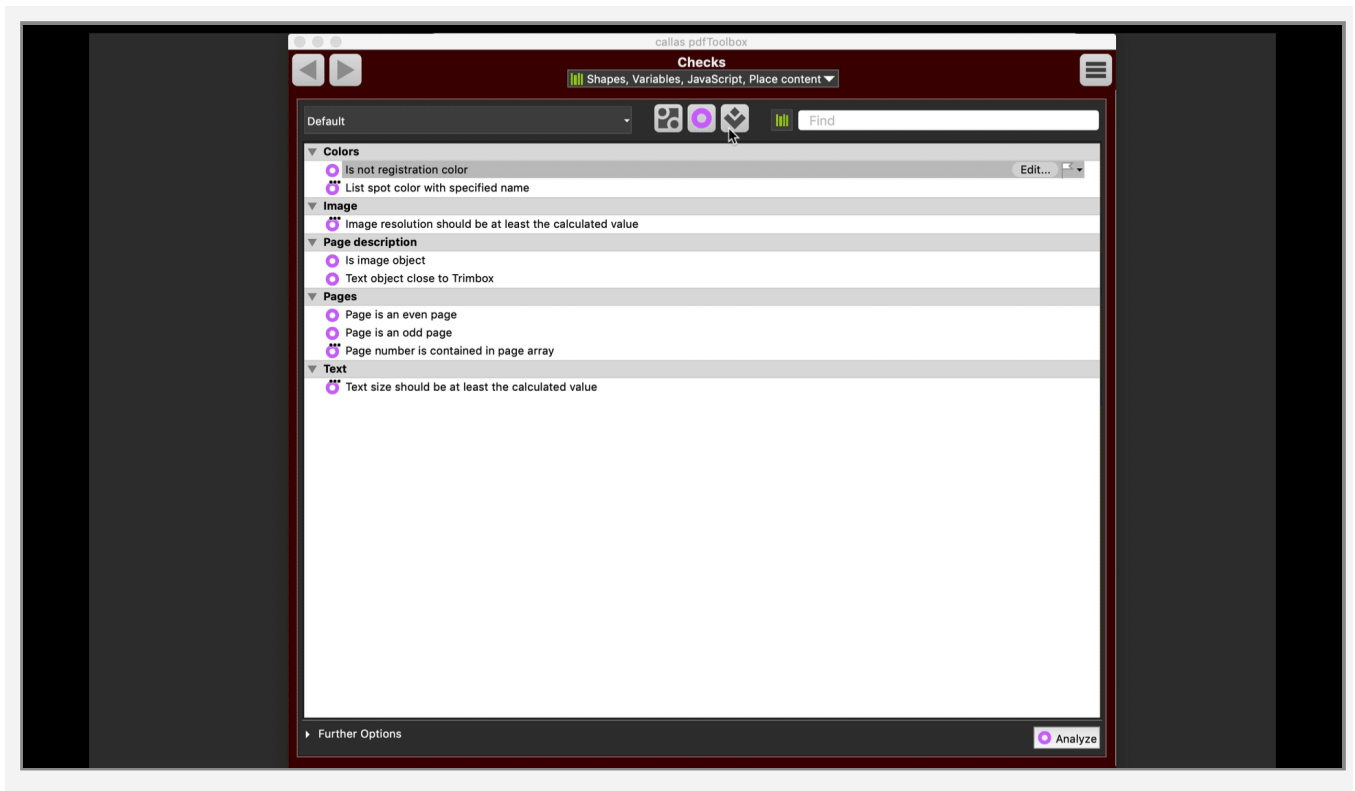
Apart from all the Libraries like 'PDF Standards' etc, you will see other options:

- Search Libraries (explained [here](#))

The rest explained in the next Chapter


- Manage Libraries
- Export Libraries
- Import Libraries
- Create new Library

Switch between different Libraries



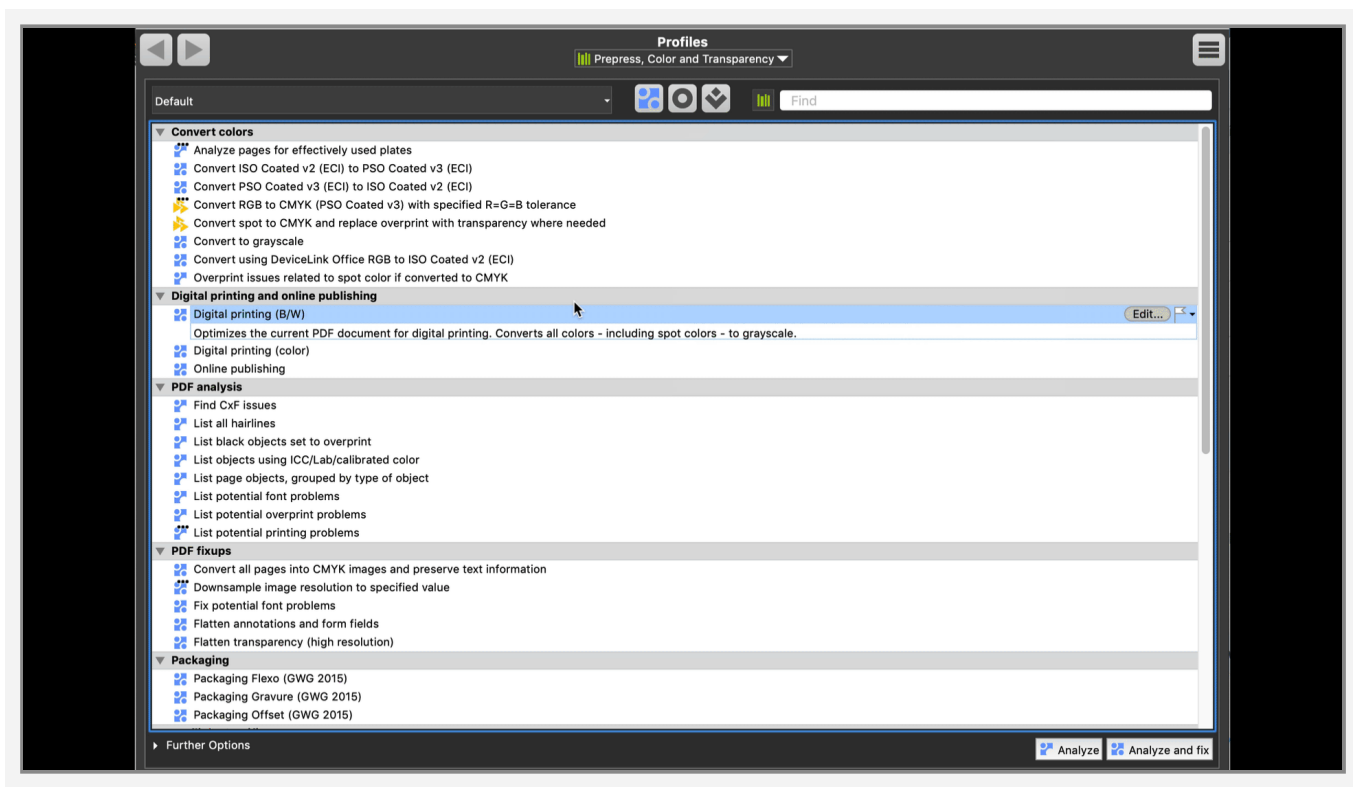
Watch the 10 minute video to know more:

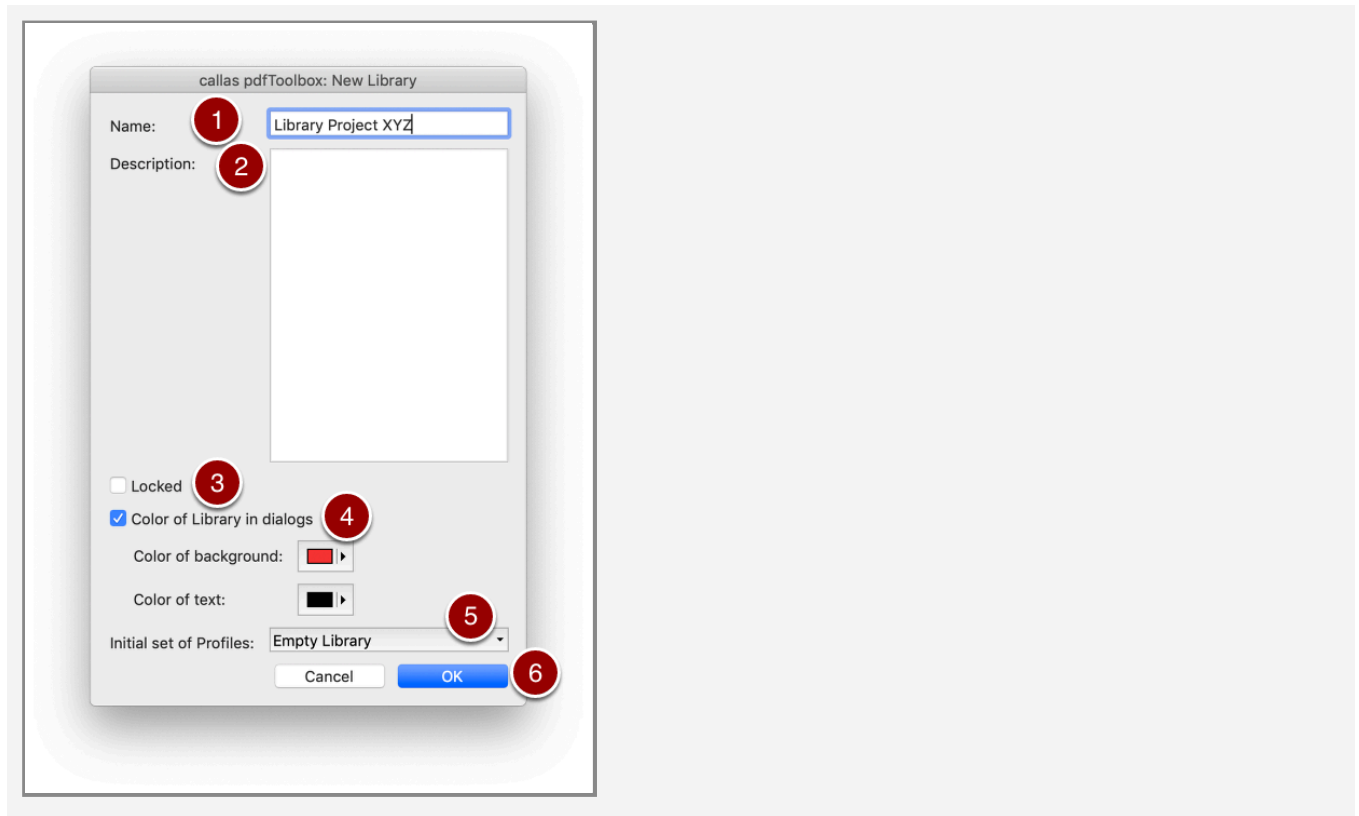
3.9 Create new, Manage, Import or Export Libraries

 Please note that the screenshots below show the functioning of the sister product pdfToolbox. pdfaPilot works on the same grounds.

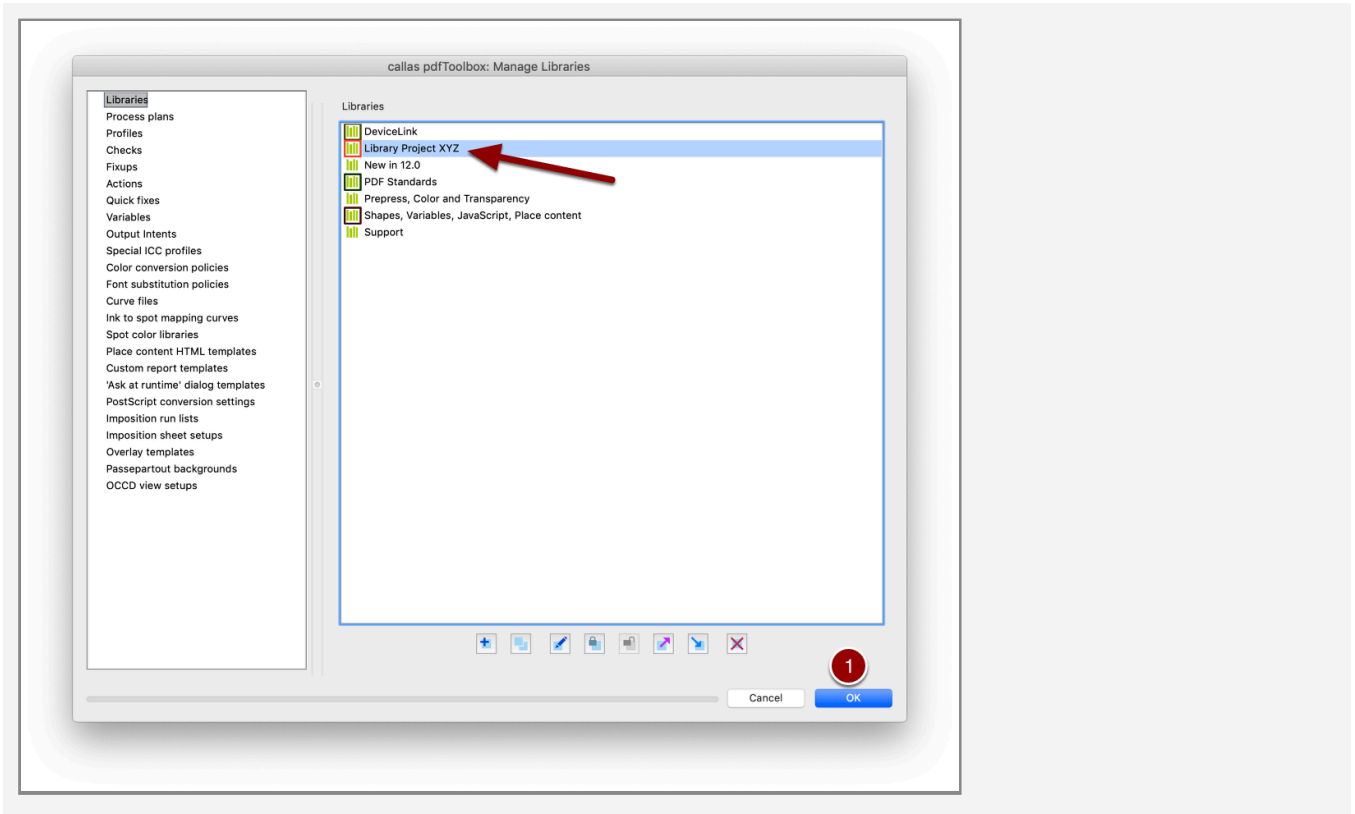
Create a new Library

Follow the below steps to create a new Library:





1. In the "Name" field save the new Library as "Library Project XYZ".
2. If necessary you can add a **description** for the new Library. No extra information is needed for this Library.
3. The new Library can be **locked** in order to protect them from changes. It is not done in this example.
4. For an improved Library overview, the **color of the background** and **text dialogs** can be customised.
5. You can prefer to start with an **empty library**, **empty library with resources** or using the **entire library with the default settings**. This example starts with a empty Library.
6. Click "OK" to save the new library.

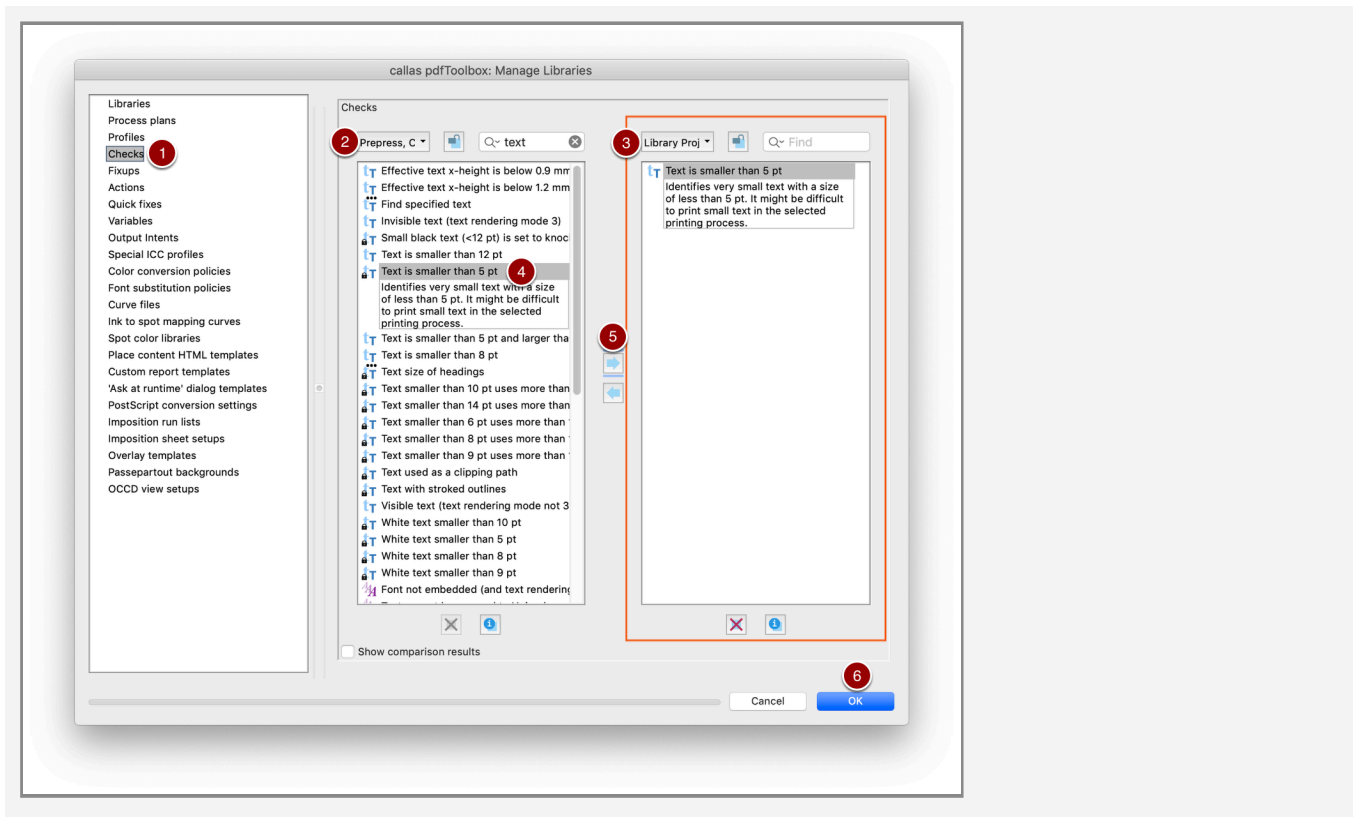


The new library "Library Project XYZ" is shown in the "Manage Libraries" list.

1. Click "OK".

You can now select this new Library in Switchboard and Profile pane.

Add a Check to created library



Now we want to investigate the created Library "Library Project XYZ". The new Library was constructed as an empty library. You can derive Process Plans, Profiles, Checks, Fixups and Actions from any other Library to the empty Library.

In this example we add the predefined Check "Text is smaller than 5 pt" from the 'Prepress, color and transparency' Library:

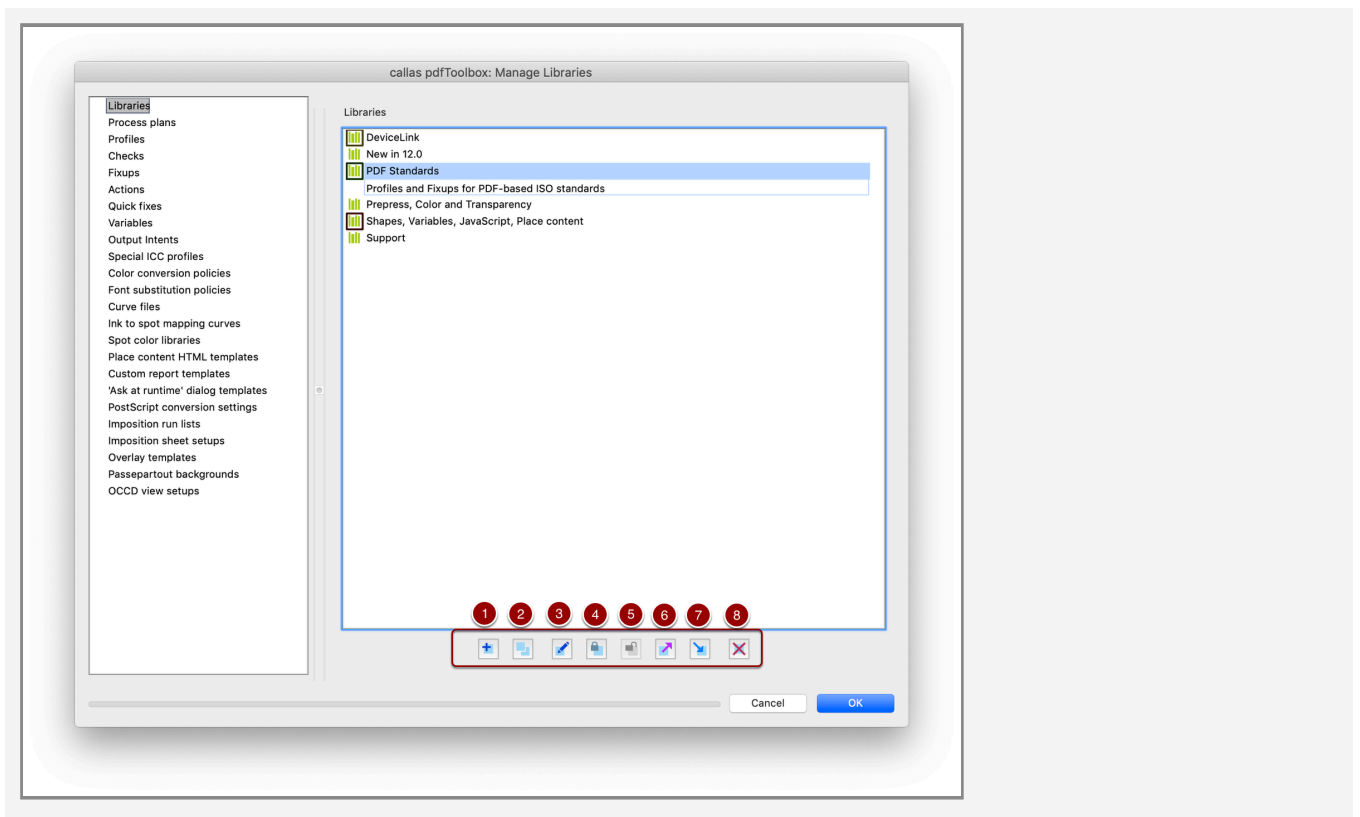
1. Go to 'Manage libraries' in Library dropdown > Select "Checks".
2. In the (left) dropdown list, select "a Library" from where you want to select the Check.
3. In the (right) dropdown list, select "Library Project XYZ" where you want to copy the Check.
4. Select the Check "Text is smaller than 5 pt".
5. Click on the blue arrow button to add the Check in "Library Project XYZ".
6. Click "OK".

Open 'Manage Libraries' dialog

1. Via Switchboard dropdown
2. Via Profile window dropdown
3. 'More options' button (top right)



Manage Libraries dialog

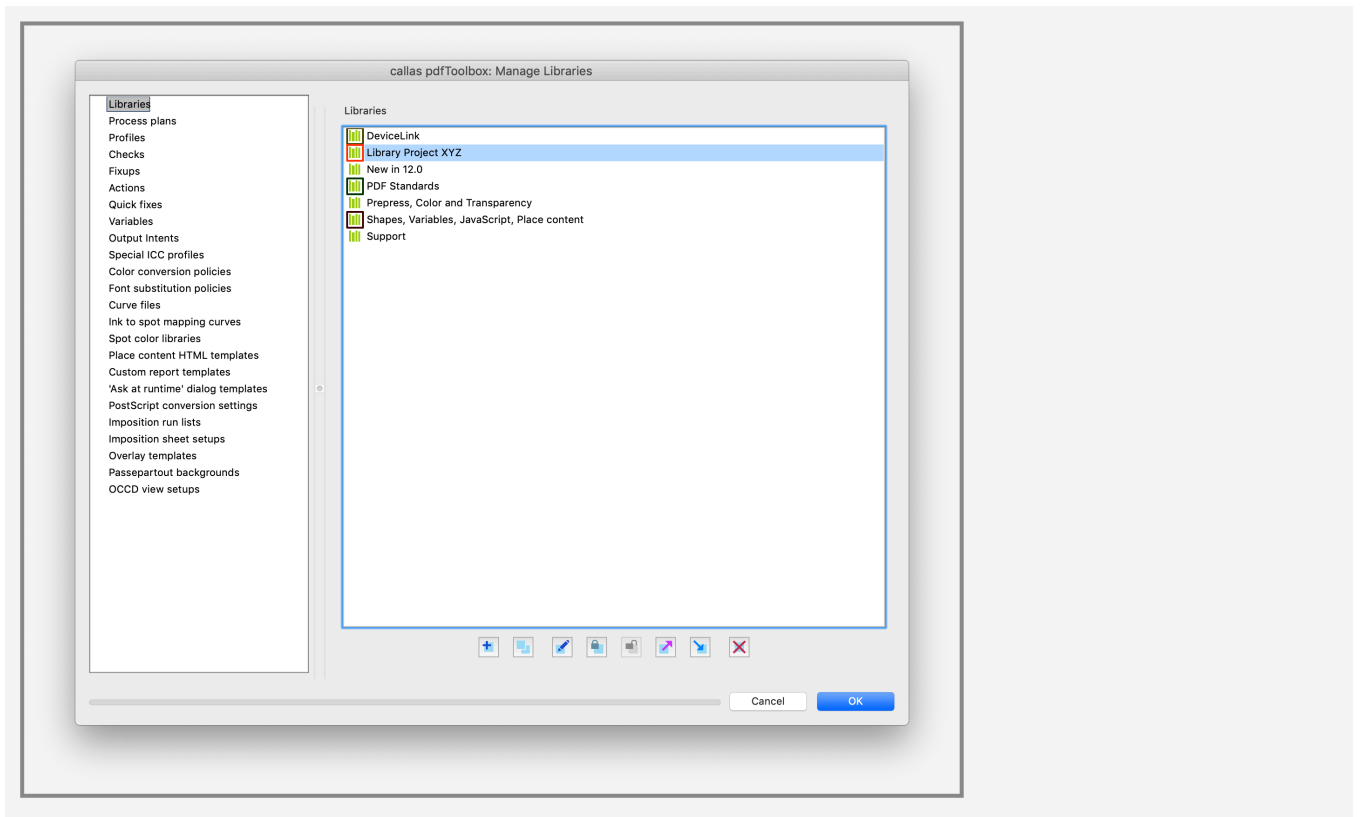


In the Libraries list above, all the default Libraries along with a custom Library 'Support' are shown.

In the lower part of the dialog, buttons for various functions can be found:

1. Adding a new Library.
2. Duplicate an existing Library.
3. Edit a selected Library.
4. Protect a selected Library.
5. Unprotect a selected Library.
6. Export a selected Library.
7. Import a Library.
8. Delete a selected Library.

Analyze the Manage Libraries dialog (left pane)



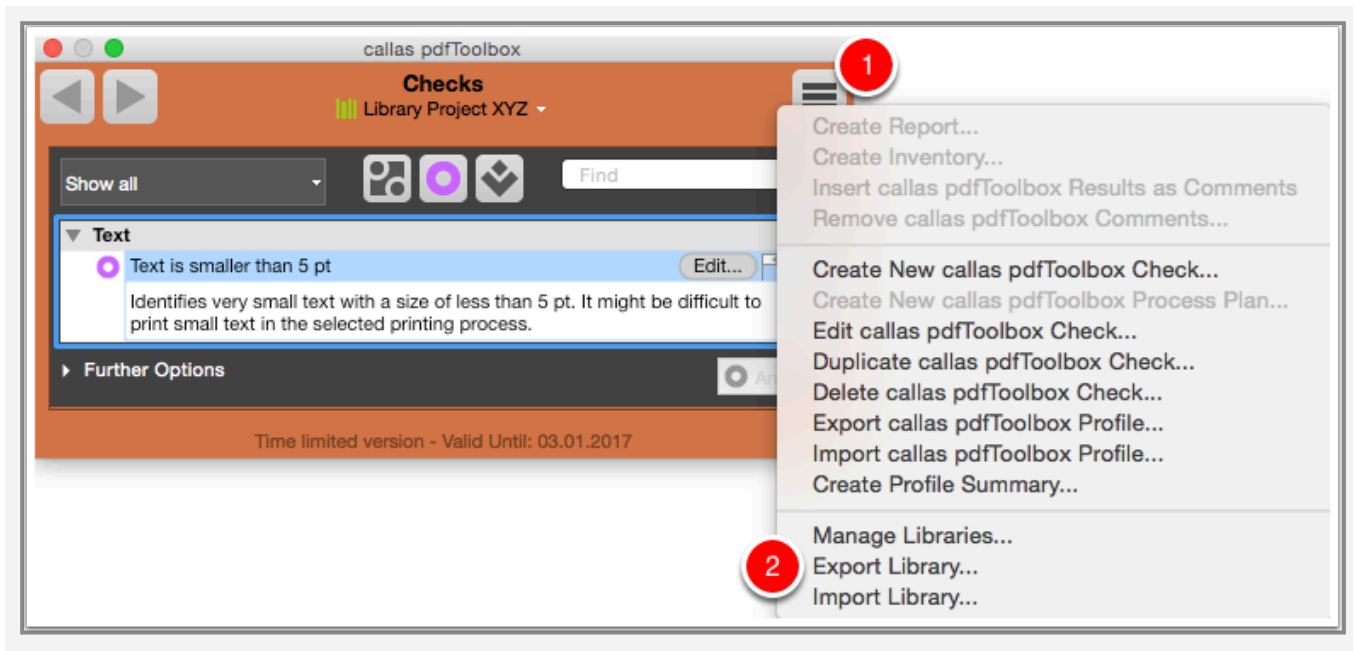
In the left column, under the entry "Libraries" all areas are listed that can be set up in the selected library.

Here are the details:

- **Libraries:** A list of available standard or custom libraries
- **Process plans:** Combined Profiles, Checks, Fixups and Actions to apply on a PDF file. Here the sequence is important
- **Profiles:** pdfaPilot Profiles that combines Checks and Fixups to apply on a PDF file. For example to convert a PDF to the PDF/X-4 standard
- **Checks:** pdfaPilot Checks that investigate a PDF file according to some criteria. For example the thickness of text
- **Fixups:** pdfaPilot Fixups that fixed a PDF file. For example converting all the text into outlines
- **Actions:** pdfaPilot Actions apply an action on a PDF file. For example creating an imposed PDF file.
- **Quick fixes:** Faster PDF manipulation feature

- **Variables:** Values given at run-time instead of predefined values
- **Output Intents:** The output conditions
- **ICC profiles:** ICC color profiles
- **Color conversion policies:** Policies to convert colors in a PDF file. For example convert to "Office RGB"
- **Font substitution policies:** Policies about missing fonts
- **Curve files:** Curve settings for tone value adjustments
- **Ink to spot mapping curves:** Mapping curve that looks at the ink amount values across the page and uses them to look up the tint value to use for the spot color plate
- **Spot color libraries:** A custom spot color library that stores spot colors and your alternate color space definitions
- **Place content HTML templates:** HTML templates to add PDF content. For example adding the file name.
- **Custom report templates:** A custom report templates for report generation.
- **'Ask at runtime' dialog templates:** A custom template to adjust the 'Ask at runtime' dialog to specify variables
- **PostScript conversion settings:** Quality levels.
- **Imposition run lists:** Imposition scheme plans. For example to plan 8 business cards, double sided.
- **Imposition sheet setups:** Sheet configuration plans. For example 8 business cards on a A4 page size.
- **Overlay templates:** Overlay templates such as "Draft" or "Watermark".
- **Passepartout backgrounds:** Passepartout backgrounds such as "Sand paper" or "Stone".
- **OCCD view setups:** Template to view Optional Content Configuration Dictionary

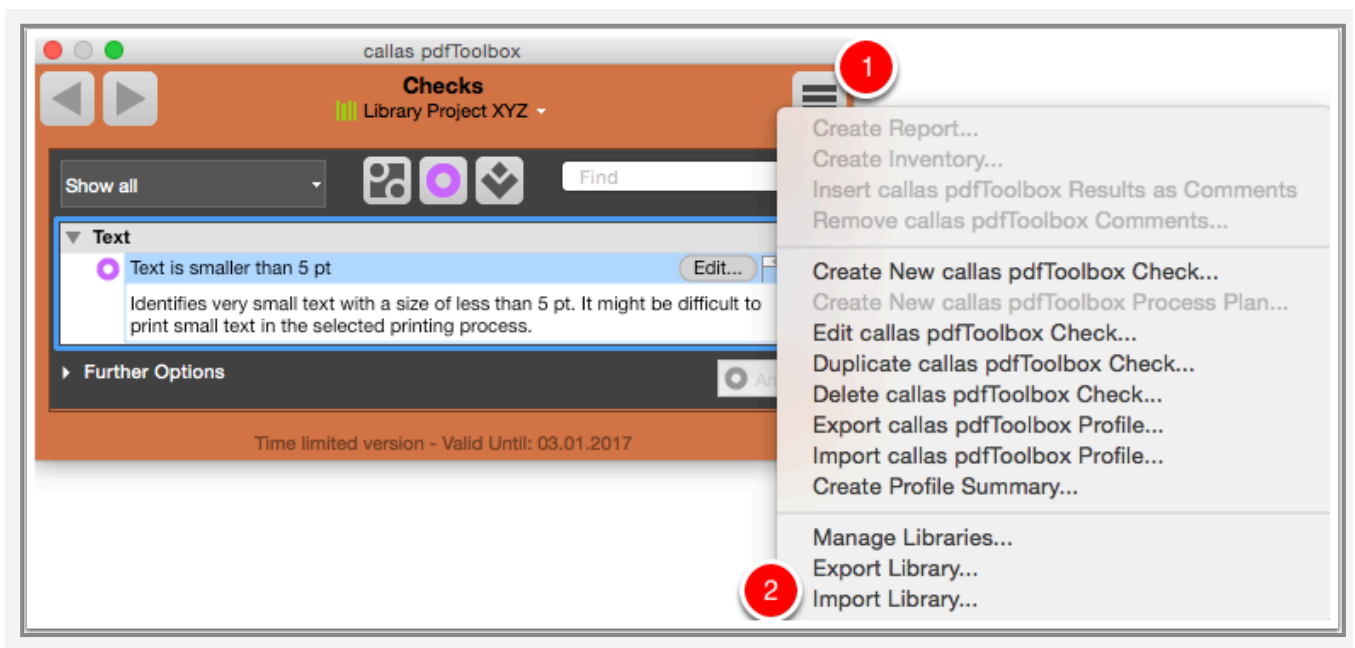
Export created library



1. Click on the Action button (of the Library dropdown).
2. Click "Export Library".

And save the Library in your system.

Import library



1. Click on the Action button (or the Library dropdown).
2. Click "Import Library".

Select a Library (.kfpl) from your system and import.

Watch this 10 minute video:

3.10 Search in or across Libraries



The illustrations show the related product pdfToolbox. The functionality is identical to that of pdfaPilot.

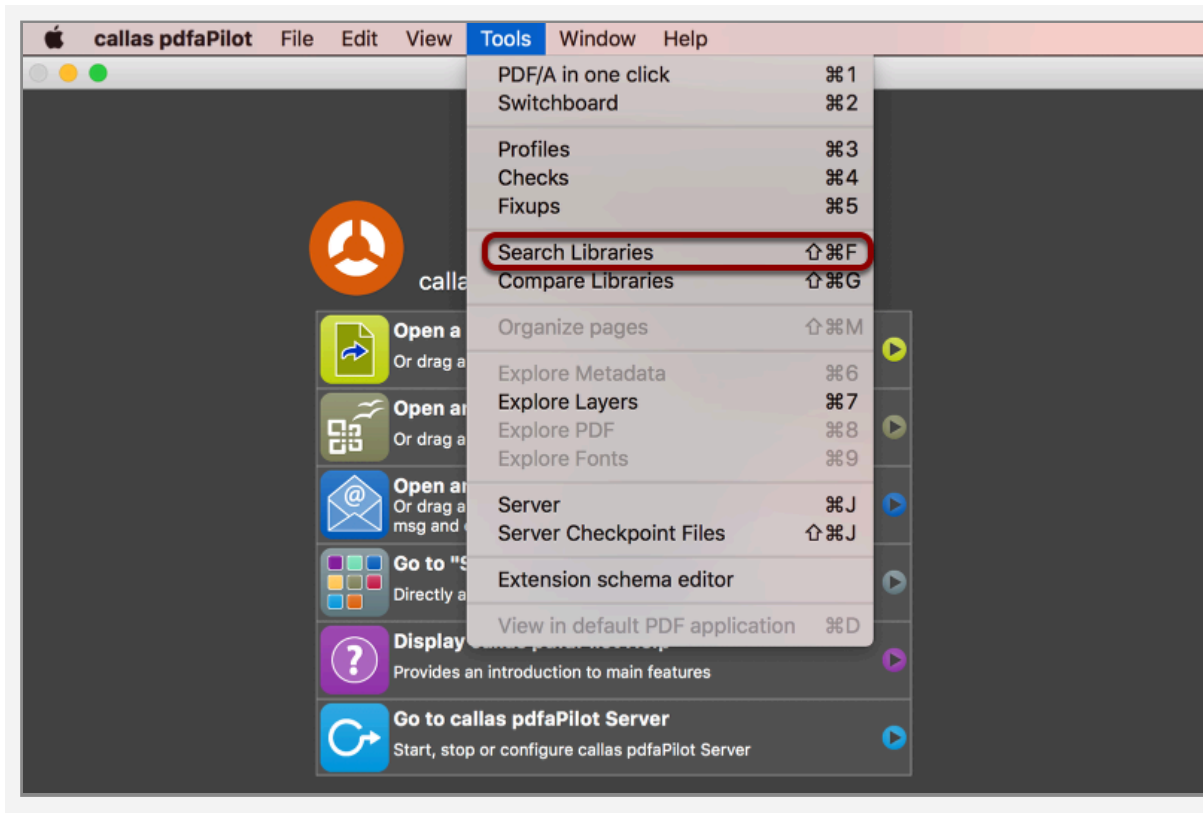
Libraries can contain a huge number of Profiles, Fixups, Checks or even Variables and Actions. Various search options help to find certain elements easily.

Global Search

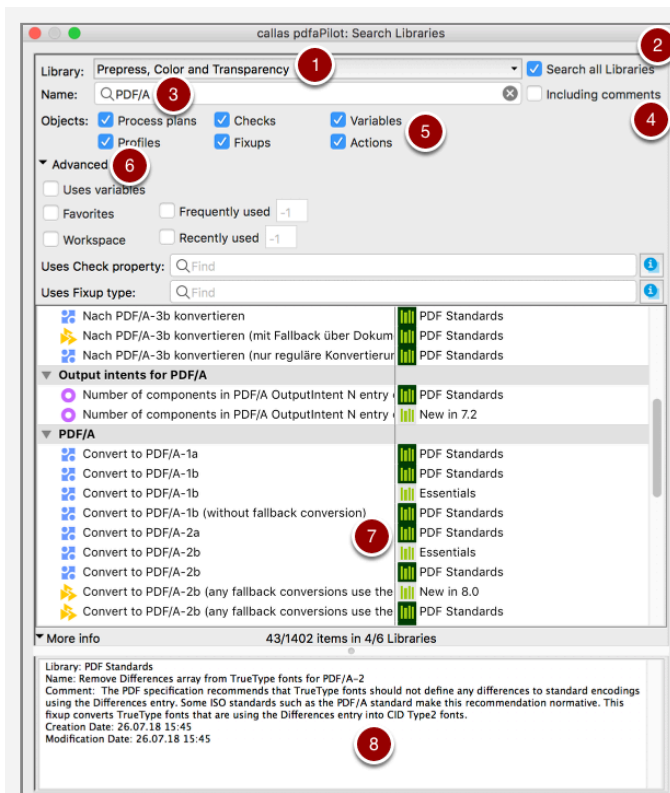
To search for Profiles, Fixups, Checks, Variables or Actions across the entire collection, use the 'Browse Libraries' menu item. You can access the function via Menu: Tools -> Search libraries.

Alternatively, you can use the keyboard shortcut Shift-Command-F.

You can browse libraries using the menu button in the Profiles window.



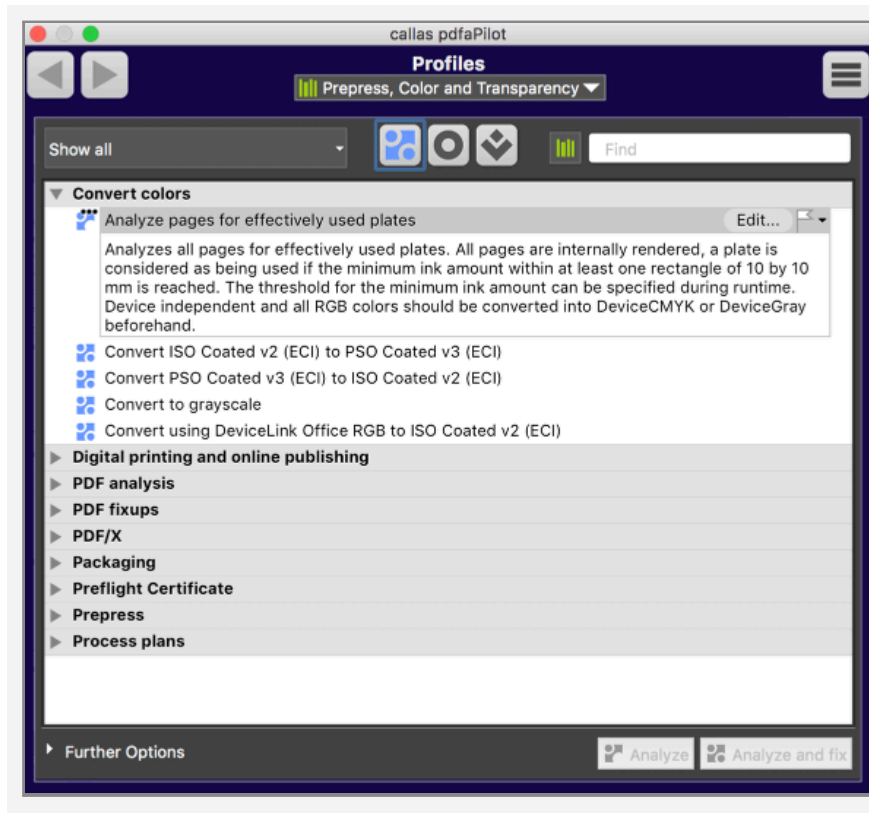
The search window contains various setting and filter options:



1. The Library to be searched can be selected from the pull-down menu.
2. The 'Search All Libraries' checkbox is used to search across Libraries.
3. Enter the search term under Name. As soon as you start typing, the first hits are displayed.
4. You can also include the comments used in the individual profiles etc. for the search.
5. Under 'Objects', you can specify the search individually for Process Plans, Checks, Variables, Profiles, Corrections and Actions.
6. Under 'Advanced', the search can be set in more detail (Favourites, uses Variables, Workplace as well as frequently or last used). The use of Check properties and Fix-up types can also be taken into account.
7. The hits are displayed in the list display, specifying the library.
8. Under 'More info' you will also find the creation date, author and other info of the selected entry.

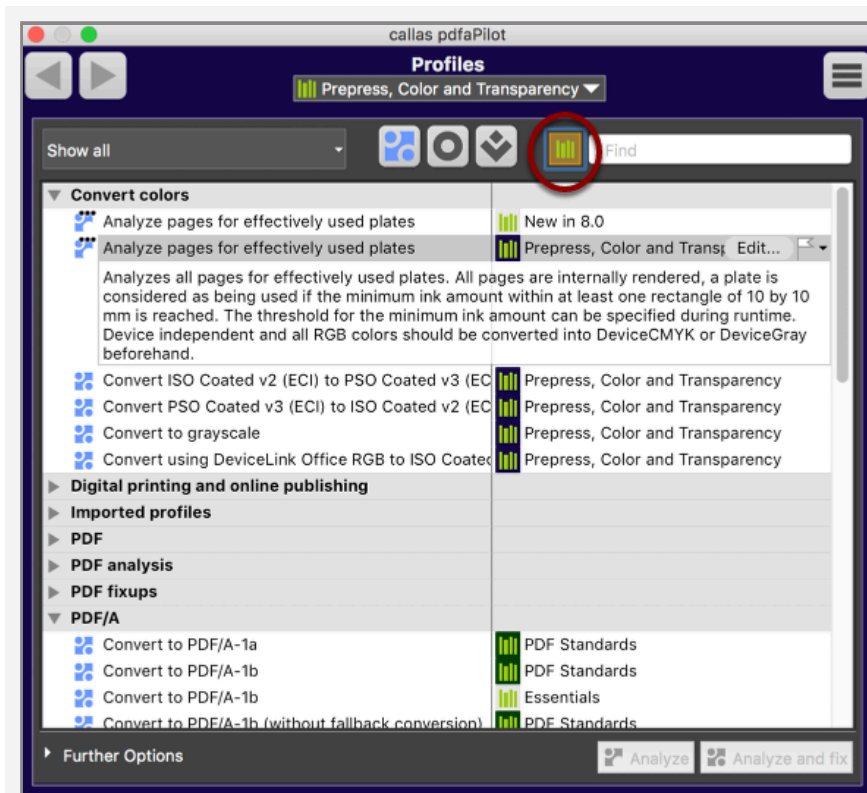
Library- Specific search in the Profiles/Fixups/Checks

You can also search in the windows for Profiles, Fixups or Checks.



Only profiles that are in the currently selected library are displayed.

Cross-library search for Profiles/Fixups/Checks



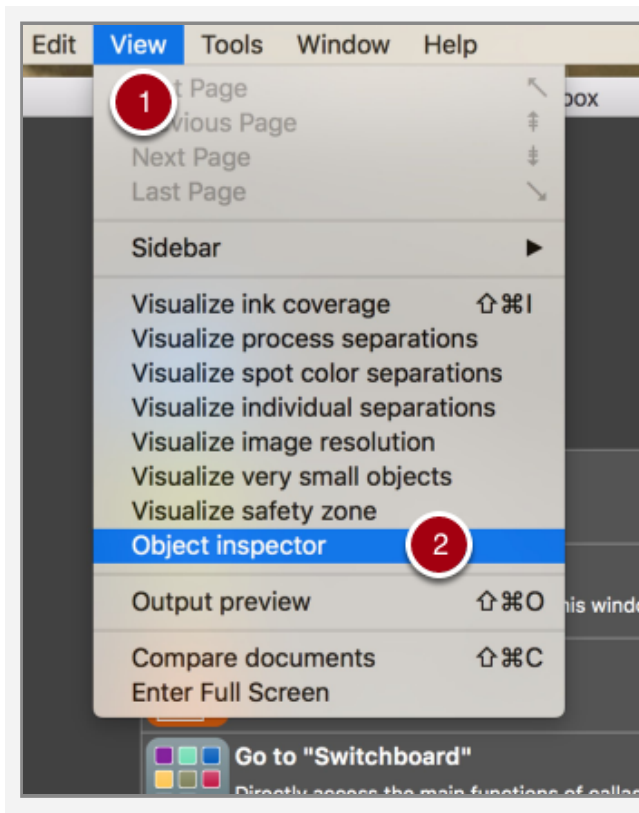
If you want to search all Libraries, first click on the Library icon next to the search field. The search term in the "Search" field will be searched across Libraries.

3.11 Examining page content: The Object Inspector

In the visualizer section of pdfaPilot Desktop, the Object Inspector allows identifying page content, both viewing which objects form a PDF page, and what their attributes are. Two additional concepts: "Wireframe viewing" and "Object type filtering" have been implemented here.

The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

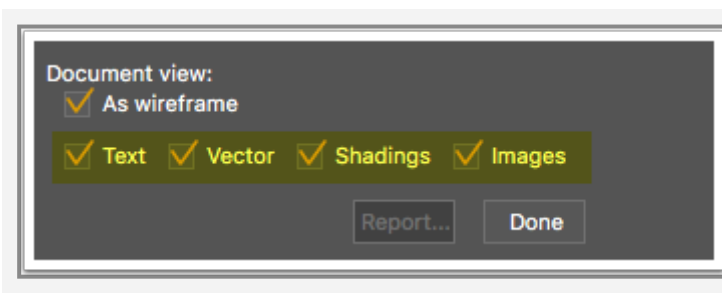
Showing the object inspector



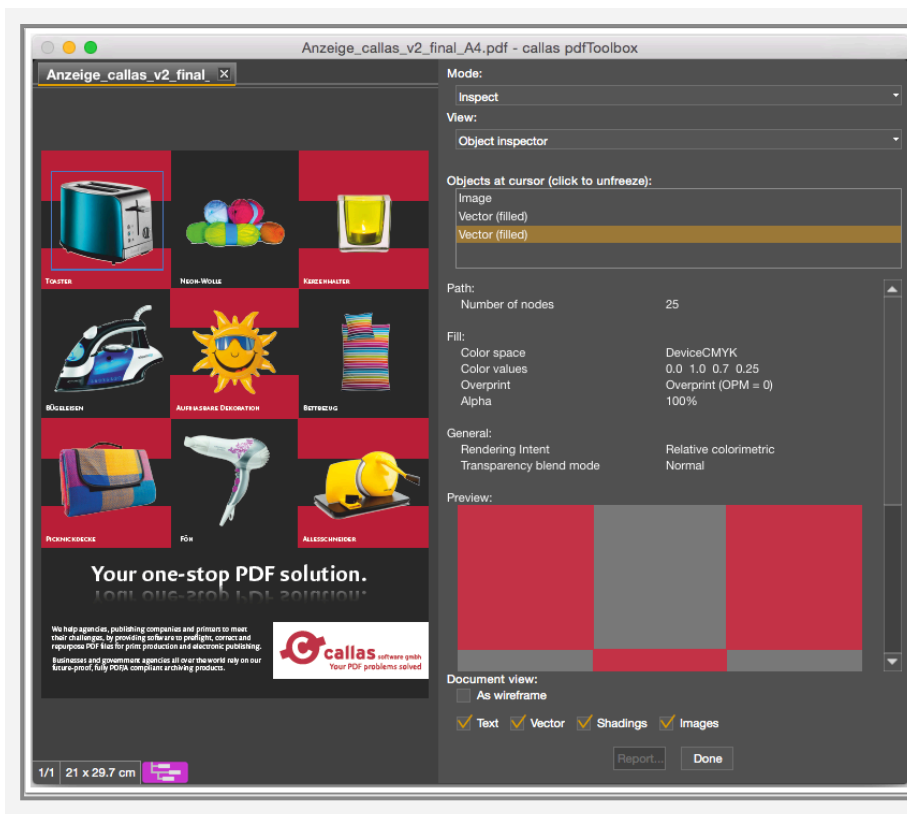
1. Click on the "View" menu
2. Select the "Object inspector" menu item

Object type filtering

By default, the object inspector shows all PDF objects on the page. This can be changed by disabling or enabling the checkboxes at the bottom of the Object inspector area. Deselecting "Text" for example, will cause all text objects on the page to be hidden. This allows examining whether objects are actually text for example, or allows viewing what is behind other objects (and normally hidden from view).



Object properties



Every single object can be selected by clicking on it inside the view on the left hand side. A click will fix the selection on the right hand side. Another click will release it.

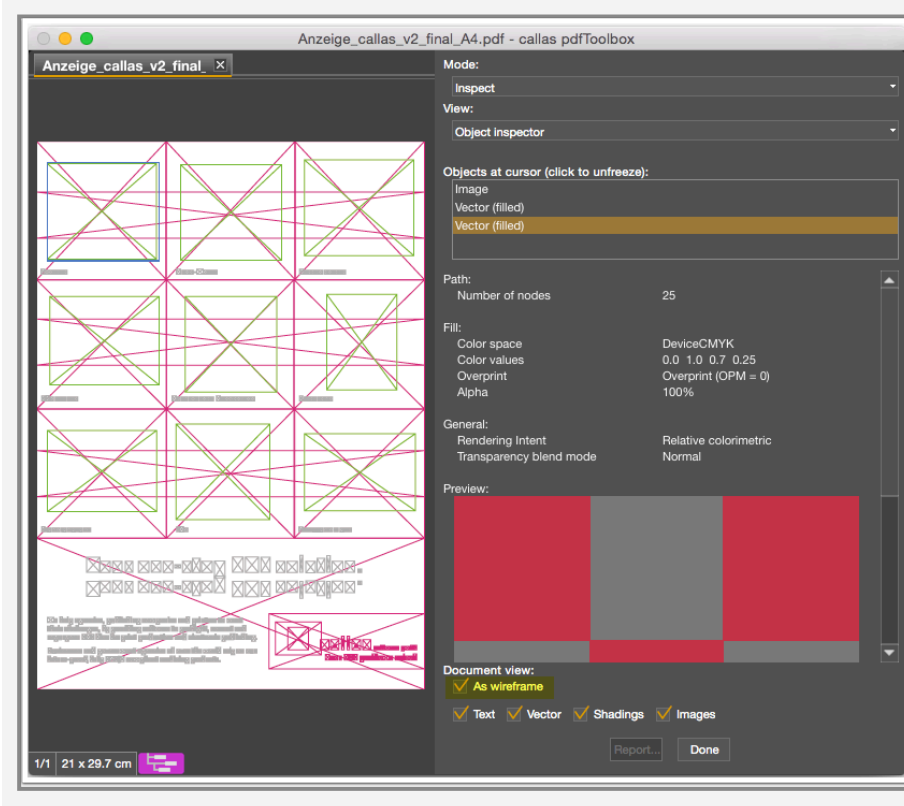
Selected objects are also visible in the preview on the right hand side.

Under "Objects at cursor", you will see a stack of all the objects in the same order as they appear in the PDF.

The topmost object is the topmost in the PDF. You can click at objects below the topmost object to see it's properties.

Wireframe viewing

At the bottom of the Object inspector area, check the "As Wireframe" checkbox, to show the displayed PDF page as wireframe. In this mode, all objects are shown with different-ly color rectangle outlines. This allows seeing the structure of the page (which objects are on the page, how they are layered etc...).

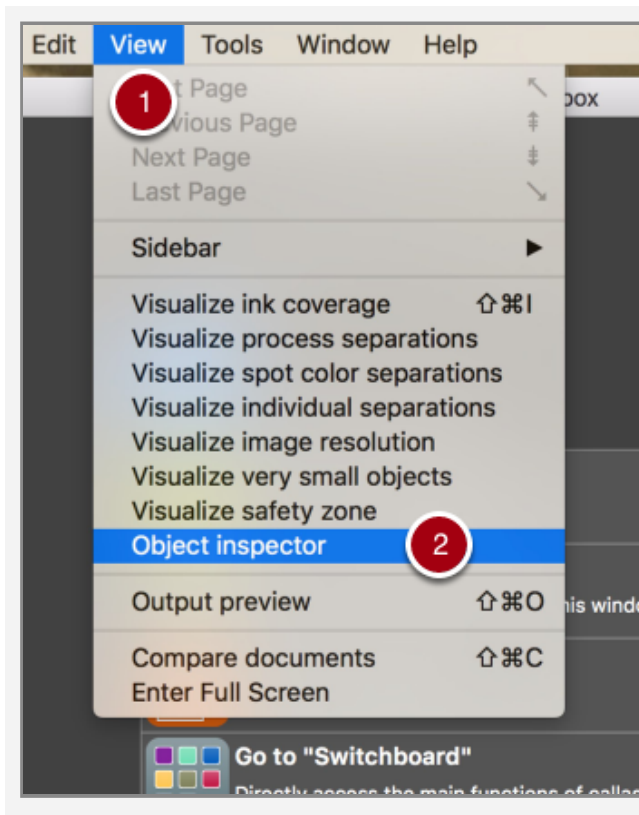


3.12 Examining page content: Filter in the Object Inspector

In the visualizer section of pdfToolbox Desktop, the Object Inspector allows identifying page content, both viewing which objects form a PDF page, and what their attributes are. Two additional concepts: "Wireframe viewing" and "Object type filtering" have been implemented here.

The screenshots are showing the sister product pdfToolbox. The functionality in this part of pdfaPilot is identical to pdfToolbox.

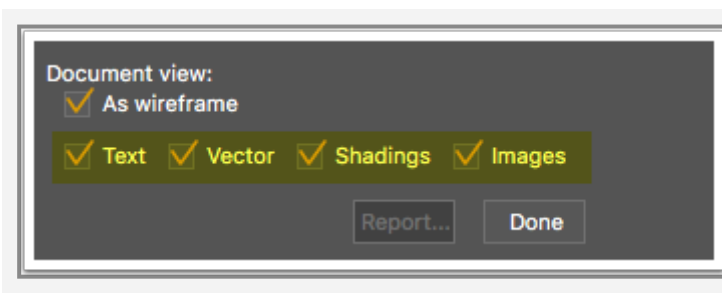
Showing the object inspector



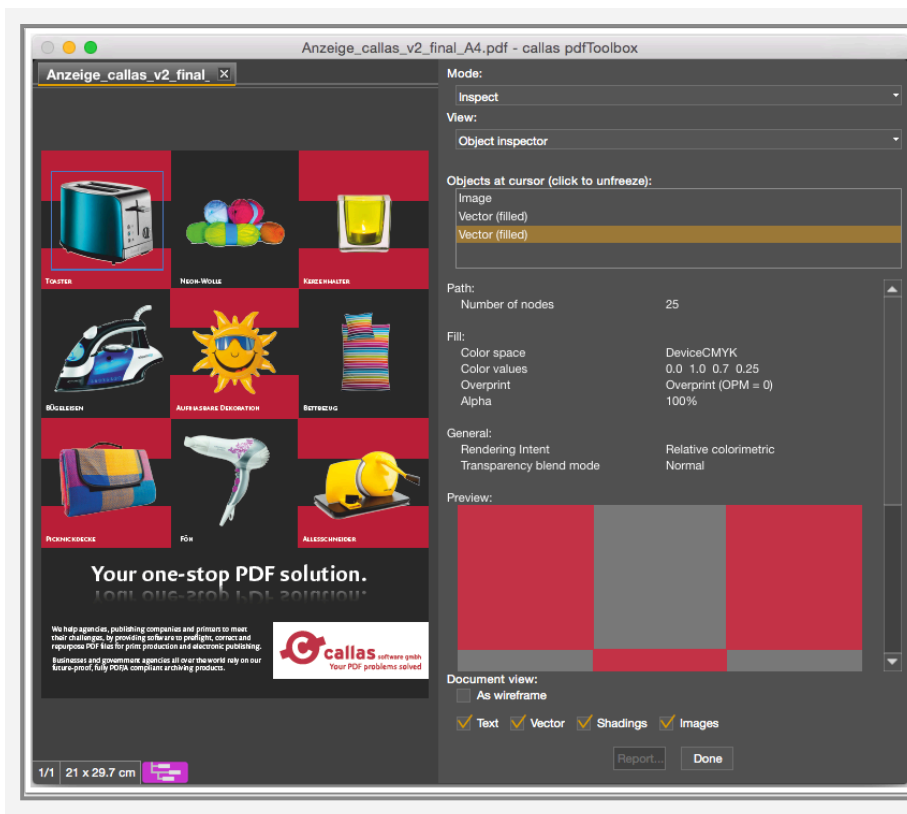
1. Click on the "View" menu
2. Select the "Object inspector" menu item

Object type filtering

By default, the object inspector shows all PDF objects on the page. This can be changed by disabling or enabling the checkboxes at the bottom of the Object inspector area. Deselecting "Text" for example, will cause all text objects on the page to be hidden. This allows examining whether objects are actually text for example, or allows viewing what is behind other objects (and normally hidden from view).



Object properties



Every single object can be selected by clicking on it inside the view on the left hand side. A click will fix the selection on the right hand side. Another click will release it.

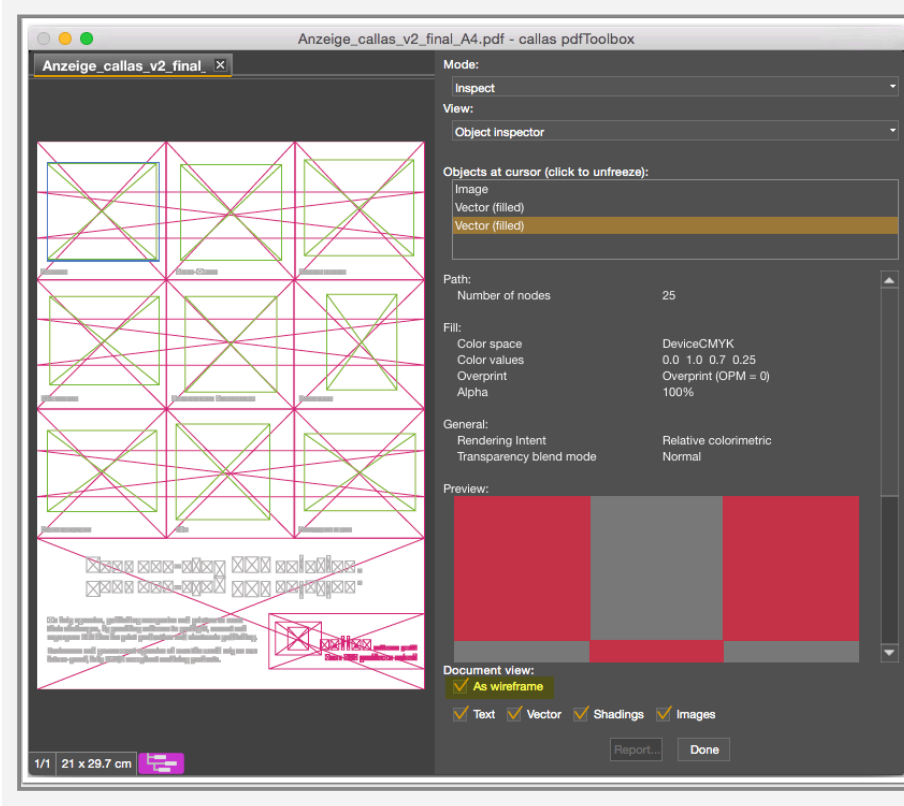
Selected objects are also visible in the preview on the right hand side.

Under "Objects at cursor", you will see a stack of all the objects in the same order as they appear in the PDF.

The topmost object is the topmost in the PDF. You can click at objects below the topmost object to see it's properties.

Wireframe viewing

At the bottom of the Object inspector area, check the "As Wireframe" checkbox, to show the displayed PDF page as wireframe. In this mode, all objects are shown with different-ly color rectangle outlines. This allows seeing the structure of the page (which objects are on the page, how they are layers etc...).



3.13 Requirements for conversions to PDF

pdfaPilot offers the possibility to convert several file formats to PDF. In order to receive the best results, the applications in which the files have been created are used for this conversion.

Supported applications and files

Application	Operating system	As of pdfaPilot version
<ul style="list-style-type: none"> • Microsoft Word 2007 • Microsoft Word 2010 (32bit and 64bit) • Microsoft Excel 2007 • Microsoft Excel 2010 (32bit and 64bit) • Microsoft PowerPoint 2007 	Windows	2.1
<ul style="list-style-type: none"> • Microsoft PowerPoint 2010 (32bit and 64bit) • Microsoft Publisher 2007 • Microsoft Publisher 2010 (32bit and 64bit) • Microsoft Visio 2007 • Microsoft Visio 2010 (32bit and 64bit) 	Windows	2.3

Application	Operating system	As of pdfaPilot version
<ul style="list-style-type: none"> • Microsoft Project 2010 (32bit and 64bit) 		
<ul style="list-style-type: none"> • Microsoft Word 2013 (32bit and 64bit) • Microsoft Excel 2013 (32bit and 64bit) • Microsoft PowerPoint 2013 (32bit and 64bit) • Microsoft Visio 2013 (32bit and 64bit) • Microsoft Publisher 2013 (32bit and 64bit) • Microsoft Project 2013 (32bit and 64bit) 	Windows	5.0
<ul style="list-style-type: none"> • Microsoft Word 2011 • Microsoft Excel 2011 • Microsoft PowerPoint 2011 • iWork '09 Pages 	Mac	2.4
<ul style="list-style-type: none"> • Microsoft Word 2016/365 • Microsoft Excel 2016/365 	Windows [MS Office on Mac until version 9.1.326]	7.0

Application	Operating system	As of pdfaPilot version
<ul style="list-style-type: none"> • Microsoft PowerPoint 2016/365 • Microsoft Visio 2016/365 • Microsoft Publisher 2016/365 • Microsoft Project 2016/365 		
<ul style="list-style-type: none"> • Microsoft Word 2019 • Microsoft Excel 2019 • Microsoft PowerPoint 2019 • Microsoft Visio 2019 • Microsoft Publisher 2019 • Microsoft Project 2019 	Windows [MS Office on Mac until version 9.1.326]	8.0
<ul style="list-style-type: none"> • Microsoft Word 2021 • Microsoft Excel 2021 • Microsoft PowerPoint 2021 • Microsoft Visio 2021 • Microsoft Publisher 2021 • Microsoft Project 2021 	Windows	11.0
<ul style="list-style-type: none"> • OpenOffice.org 3.2 	Windows, Mac and Linux	2.2

Application	Operating system	As of pdfaPilot version
<ul style="list-style-type: none"> OpenOffice.org 3.3 	Windows, Mac and Linux	2.3
<ul style="list-style-type: none"> OpenOffice.org 4.x 	Windows, Mac and Linux	5.0
<ul style="list-style-type: none"> LibreOffice 4.x 	Windows, Mac and Linux	5.0
<ul style="list-style-type: none"> LibreOffice 5.x 	Windows, Mac and Linux	6.2
<ul style="list-style-type: none"> LibreOffice 6.x 	Windows, Mac and Linux	8.0
<ul style="list-style-type: none"> LibreOffice 7.x 	Windows, Mac and Linux	9.0
<ul style="list-style-type: none"> PostScript and EPS files 	All Platforms	2.4
<ul style="list-style-type: none"> Image files 	All Platforms	2.4
<ul style="list-style-type: none"> XPS files 	Windows	6.0

Supported file types

Application	File type		
	Windows	Mac	Linux
Microsoft Word	.doc .docx .dot .dotx .dotm .rtf .txt .xml .vcf .ics .wpd	.doc .docx .txt	
Microsoft Excel	.xls .xlsx .xlsm .xlt .xltx .xltm	.xls .xlsx	
Microsoft PowerPoint	.ppt .pptx .pps .ppsx	.ppt .pptx	

Application	File type		
	.pot .potx .potm .ppa .ppam		
Microsoft Visio	.vsd .vdx .vdw .vss .vsx .vtx (v.11) .vsdx (v.11)		
Microsoft Project	.mpp .mpt .mpd .mpw .mpx		
Microsoft Publisher	.pub		
OpenOffice/LibreOffice Writer	.odt .ott .sxw .stw .doc .rtf .sdw .vor .txt .pdb .xml .psw .docx .docm .dotx .dotm .602 .wpd .hwp		
OpenOffice/LibreOffice Calc	.ods .ots .sxc .stc .dif .dbf .xls .xlt .sdc .vor .slk .csv .pxl .uos .xlc .xlm .xlw .sdc .xlsb .xlsm .xlsx .xltm .xltx .wb2		
OpenOffice/LibreOffice Impress	.odp .otp .sxi .sti .ppt .pot .sxd .sda .sdd .vor .uop .odg .cgm .pptm .pptx .potm .potx		
OpenOffice/LibreOffice Draw	.bmp .emf .pcx .pgm .wmf (from version 8.2)		
iWork '09 Pages		.pages	
PostScript and EPS files	.ps .eps .prn		
Image files	.tiff .tif .jpeg .jpg .png .gif .psd (from version 10.1) .psb (from version 11)		
HTML	.html .htm .svg		

File types that require additional parameters:

Application	File type		
	Windows	Mac	Linux
Email	.msg .eml .emlx When emails are converted also any attachments are converted, compressed archives (.zip, .gz, .7z, .rar) are uncompressed.		

Further information on the settings for email conversion [here](#).

General

- All Office-applications should be installed with the default options, in order to guarantee that all needed components are available.
- Install all the latest updates.
- Start the application at least once and make sure that the setup wizards have been executed successfully.
- pdfaPilot need to be activated. This means, a valid activation request must have been sent and the software must have been activated with the activation file received afterwards.
- The activation needs to be performed for the user who wants to execute the file conversion.
- On Windows systems, install the [Microsoft .NET Framework Version 4 or newer](#) - this is not automatically installed with the regular Windows Update but belongs to the optional updates.
- Please ensure, that the Microsoft Visual C++ Redistributable Packages for Visual Studio (2013, 2017, 2019) were properly installed by the callas installation package. Only the 32bit versions of the runtime libraries are needed because the internal conversion tool is a 32bit application.
- While conversions to PDF are performed, no other users on this machine should use the applications needed for this process.
- In particular the file to be processed should not be opened in the respective application.
- Only one instance of the application that creates PDF must be used at the same time. Therefore in pdfaPilot Server or in Enfocus Switch the number of instances needs to be set to 1.
- Although some file types can be processed with several applications it is recommended to process documents with the application where they were created with.
- Microsoft Office files are not processed via OpenOffice/LibreOffice by default under Windows and MacOS if no Microsoft Office is installed on the system. For Server/CLI, processing of such MS Office files by OpenOffice/LibreOffice can be forced using the `--topdf_forceopenoffice` parameter.
- Since in Microsoft Word, autocorrection is activated by default, we recommend to deactivate spellchecking,

grammar checking and hyphenation via the Word options to speed up the Office conversion.

Microsoft applications

- Please make sure that Microsoft Office is activated. In case Microsoft Office runs as an evaluation version, a warning dialog might pop up during conversion.
- Install the latest service pack for Microsoft Office.
- When updating the Microsoft Office Suite it is recommended to deinstall previously installed version before to ensure all registry entries are removed and properly set when the new installation of Office takes place. Especially when installing Microsoft Office 2010 on a system with an existing Microsoft Office 2007 such registry entries seems to remain. See [Microsoft knowledge base](#) for details how to uninstall.

Word

Word documents may contain annotations from the function “Track changes”. They are output into the PDF with the setting “Screen”. When outputting with setting “Print” they are not output. In the Desktop version of pdfToolbox the setting “Print” is default so that annotations are not output. Whereas in pdfaPilot Desktop the setting “Screen” is default so that annotations are output. On the Mac the output of annotations can only be omitted when running the english language version of Word. Moreover “Enable access for assistive devices” need to be enabled in “Universal Access” of the “System preferences”.

Excel

- If the contents exceed the “print area”, only the parts covered by this “print area” will be output whilst the other parts will not be output.
- The page format of the output PDF is optimized in order to fit as much content as possible in a well readable way onto a page. For large sheets the content may be spread over several PDF pages.
- To ensure conversion, the print spooler has to be active. This is the case when the Excel preview shows the content. Otherwise the spooler has to be started in the “Services” configuration of the system settings. After doing

this, the “Microsoft XPS Document Writer” shows up in the print dialog of Excel.

Powerpoint

It is not recommended to use parallel conversion of Power-Point files with MS Office, as this can sometimes lead to problems, instead sequential processing works stable.

Mac

- A page range can not be specified, the parameter Page range (pagerange) does not have an influence.
- Images are exported with a resolution of 72 ppi, the parameter Print (topdf_print) does not have an influence.
- If Microsoft Office 2008 and Microsoft Office 2011 are installed the Office application that is set as default for opening Office documents in the system or the Office application that is already open is used to process Office documents.

Dialog handling

- Under Windows dialogs appearing in the Office application during processing can be handled.
- Therefore an empty file named MTPGuiActions.log has to be placed in the user folder here:
- <user>\AppData\Roaming\axaio software\MS Office 12\MadeToPrint\Logfiles\
- If the Office document that opens a dialog is processed now an ID of the dialog is logged in MTPGuiActions.log
- This ID needs to be entered in MTPGuiActions.cfg which can be found in the etc/PDFOfficeTool folder next to the executable of pdfaPilot Desktop or Server
- Within MTPGuiActions.cfg the options of the dialog handling is described

Usage when running as a service

Sometimes problems may occur when running pdfaPilot as a service with Microsoft Office. Automation is not supported by Microsoft officially, but the following workaround was successful in most cases:

- Create the following folders (with appropriate permissions) on your system:
 - Windows Server 2012 + later:
 - C:\Windows\SysWOW64\config\systemprofile\AppData\Local\Microsoft\Windows\INetCache

- Especially when running Office 32Bit on x64 systems, the following folders are required as well:
 - Windows x64: C:\Windows\SysWOW64\config\systemprofile\Desktop
 - Windows x86: C:\Windows\System32\config\systemprofile\Desktop
 - **Please NOTE:**
For smooth operation, the folders named above need to be restored after a Windows Update.
- For the Office/Excel conversion in a service setting, it is necessary to set up a default printer, especially when working with the "System" account.
- For Microsoft Excel, Microsoft Word and Microsoft PowerPoint the DCOM Identity "The launching user" must be selected. For Microsoft Visio the DCOM Identity "The interactive user" needs to be selected.

OpenOffice and LibreOffice

- Close all open OpenOffice and LibreOffice instances before processing with pdfaPilot.
- Please keep in mind the known restrictions of OpenOffice.org regarding cases where umlauts from Type1 and OpenType fonts might not be output correctly. You will find more information on the current state of this issue on [OpenOffice.org](https://www.openoffice.org).
- Only LibreOffice or OpenOffice may be installed.
- For Windows only the 32-bit versions of LibreOffice and OpenOffice are supported (before pdfaPilot 9/pdfEngine 11).
- *From pdfaPilot 9/pdfEngine 11 onwards with the 64bit variant on Windows, the Libre/OpenOffice has to be 64bit as well (and 32bit when the 32bit variant of pdfaPilot/pdfEngine is installed).*
- For Non-Windows systems (Mac, Linux..) the JDK needs to be installed: <https://www.oracle.com/java/technologies/javase-jdk14-downloads.html>

Mac

Please verify that OpenOffice is installed inside the default installation path (e.g. "/Applications/OpenOffice.org" on Mac OS X systems).

If it is not, please re-install it to this location.

On Mac the following default installation/location are supported (in this order - first wins):

- /Applications/LibreOffice.app
- /Applications/OpenOffice.app
- /Applications/OpenOffice.org.app

Linux

- Support for Office Conversion has been tested thoroughly on Ubuntu 14.04 LTS (Server) with the default installation of LibreOffice and also with an OpenOffice installation.
- On Debian based Linux distributions (like Ubuntu or Mint), the required LibreOffice packages are installed via the package manager or via the command line as shown below:
 - `sudo aptitude install libreoffice`
 - (please substitute 'libreoffice' by 'openoffice' if you prefer OpenOffice)
- On other linux distributions or with other OS versions it might be required to search for packages with different names. E.g install everything found with the following search terms:
 - libreoffice (or openoffice)
 - uno
 - ure
 - headless
- Some distributions, like e.g. Debian 10 ("Buster") may require to install the additional "libreoffice-java-common" package:
 - `sudo apt install libreoffice-java-common`

iWork '09 Pages

- A page range can not be specified as it is also via manual PDF export, the parameter Page range (pagerange) does not have an influence.
- Images are exported with a resolution of 72 ppi as it is also via manual PDF export, the parameter Print (topdf_print) does not have an influence.
- Documents of iWork '08 Pages can not be processed with pdfaPilot Server application because of a changed file format (file packages of Pages version 3 and earlier).

PostScript and EPS files

- ICC profiles referenced in a PDF settings file (.joboptions) need to be copied into the operating system folder for ICC profiles, e.g.:
 - Windows:
C:\Windows\system32\spool\drivers\color
 - MacOS:
/MacOS HD/Library/ColorSync/Profiles
- The application will also look into the following folders for ICC profiles:
 - /Library/Application Support/Adobe/Color/Profiles/Recommended
 - /Library/Application Support/Adobe/Color/Profiles
 - /System/Library/ColorSync/Profiles
- Alternatively, you can put ICC-files for PostScript to PDF in the subfolder of the application:
.../etc/PDFPSTool/ICCProfiles
- A Color settings file (.csf) that is referenced in the PDF settings file is not necessary for the processing.

Additional settings with limited user rights

In general it is recommended to grant the respective service user administrator privileges. If this level of rights can not set due to internal regulations, some additional settings within the operating system are recommended.

The following folders should allow the user the respective access right:

For 64-bit machines	
C:\Windows\Temp	Modify
C:\Windows\syswow64\config	Read
C:\Windows\syswow64\config\systemprofile	Read
C:\Windows\syswow64\config\systemprofile\AppData	Modify

For 64-bit machines	
C:\Windows\syswow64\config\systemprofile\Desktop	Modify (Create it, if it does not exist)
C:\Windows\syswow64\config\systemprofile\AppData\Local\Microsoft\Windows\INetCache	Modify (Create it, if it does not exist)
For 32-bit machines	
C:\Windows\Temp	Modify
C:\Windows\system32\config	Read
C:\Windows\system32\config\systemprofile	Read
C:\Windows\system32\config\systemprofile\AppData	Modify
C:\Windows\system32\config\systemprofile\Desktop	Modify (Create it, if it does not exist)
C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\INetCache	Modify (Create it, if it does not exist)

Office conversion

Additional settings

- Set the 32-bit folder preferences (detailed above) in addition to the 64-bit preferences on 64-bit systems running 64-bit versions of Microsoft Office
- Set the default printer to XPS Document Writer

DCOM settings

- Launch DCOMCNFG by using: C:\WINDOWS\SysWOW64> mmc comexp.msc /32
- Go to Computers > MyComputer > DCOM Config.
- Right-click the application that you want to automate. The application names are listed in the table below:

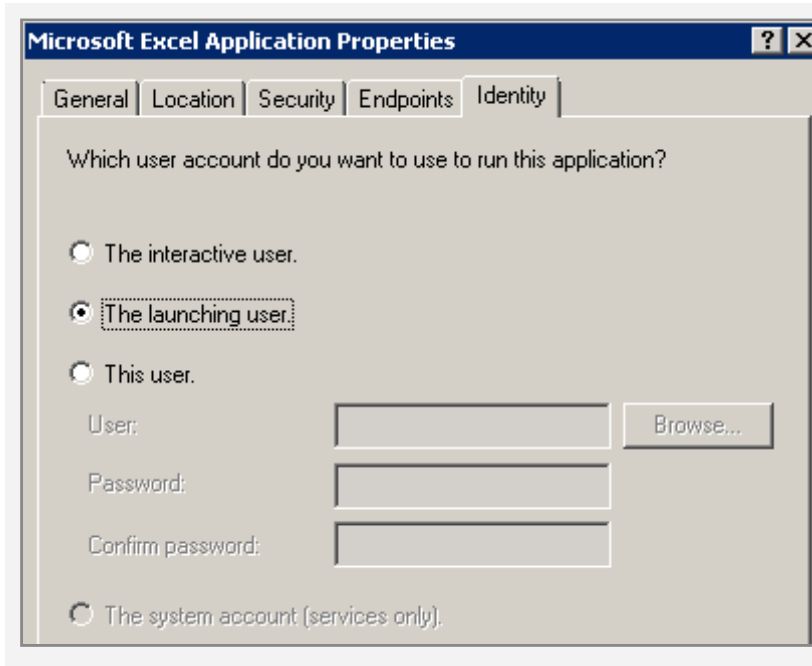
Application	DCOM Name
Microsoft Access 2007/2010/2013/2016	Microsoft Access Application
Microsoft Excel 2007/2010/2013/2016	Microsoft Excel Application
Microsoft Office Word 2007	Microsoft Office Word 97 - 2003 Document
Microsoft Word 2010/2013/2016	Microsoft Word 97 - 2003 Document

- On some systems Microsoft Word is not displayed and you will have to use {00020906-0000-0000-C000-000000000046} instead.
- Click Properties to open the property dialog box for this application.
- Verify Identity and Security tabs

 **Note**

For Microsoft Excel, Microsoft Word and Microsoft PowerPoint the DCOM Identity "The launching user" must be selected.

For Microsoft Visio the DCOM Identity "The interactive user" needs to be selected.



Slow conversion on systems without Internet connection

Some customers experienced long processing times on machines without an Internet connection, especially when using MS Office 365 or MS Office 2016/2019.

The following registry key did the trick for them:

```
[HKEY_CURRENT_USER\Software\Microsoft\Office\Common\ClientTelemetry] "DisableTelemetry"=dword:00000001
```

This might be related to:

<https://docs.microsoft.com/en-US/DeployOffice/compat/manage-the-privacy-of-data-monitored-by-telemetry-in-office>

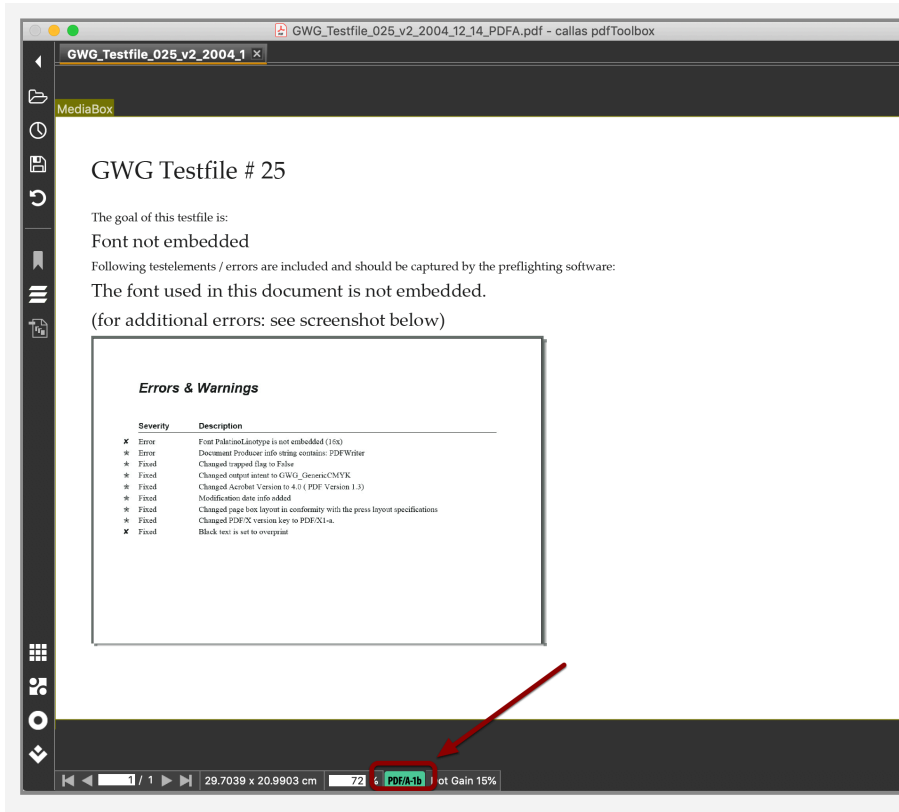
We haven't found if there are official settings available - therefore we recommend an intensive testing on such systems before this key is used in a production environment as we can not estimate if there are any side effects.

Still problems when converting?

In case you are having trouble with 'Convert to PDF', please contact our support team and simply fill out this [form](#).

3.14 Checking the ISO standards using callas Desktop products

This article explains how you can check if a PDF file is ISO standard compliant or not.

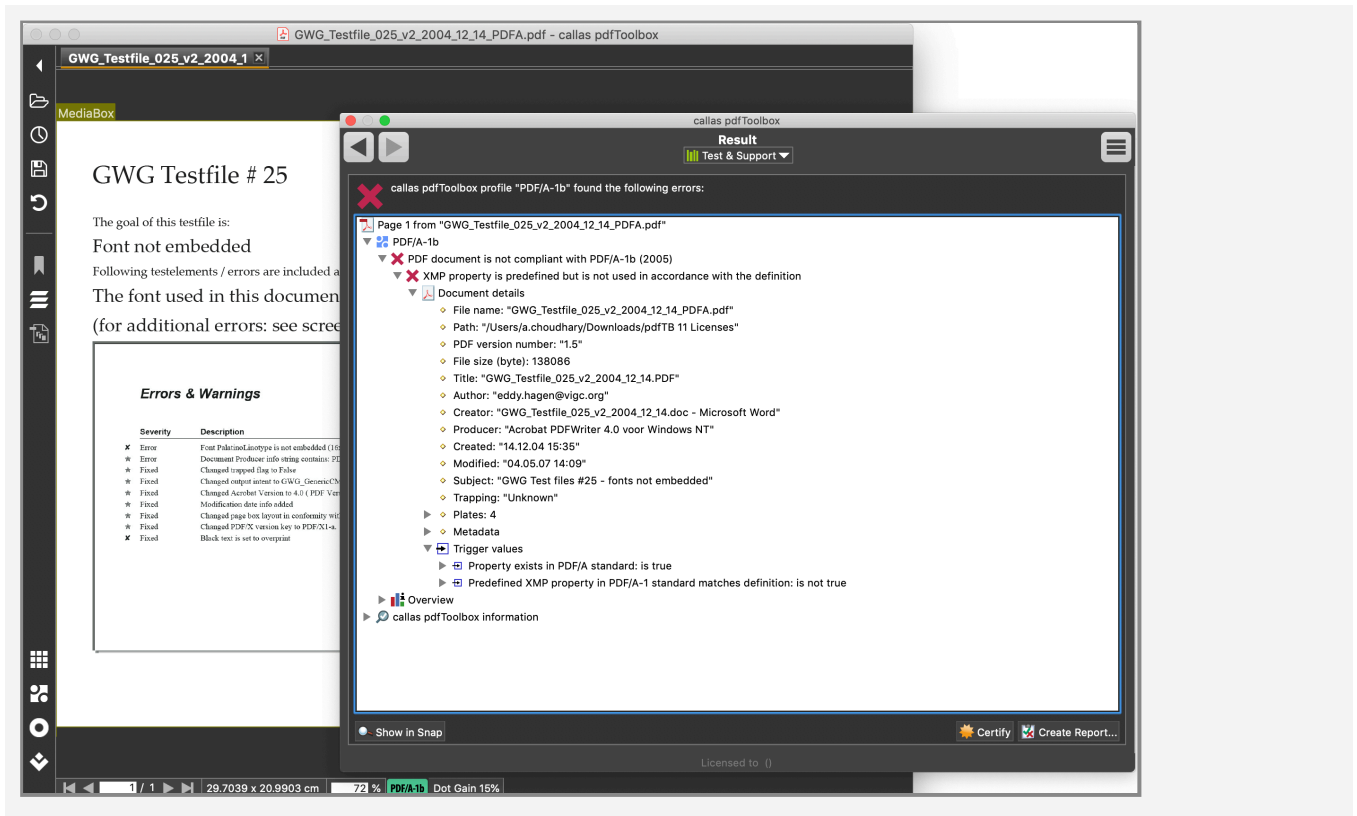


Upon opening a file in a callas Desktop product (pdfToolbox/pdfaPilot), the application will notify you at the bottom if the document is marked with a standard or not (marked as a rectangular selection in the screenshot above).

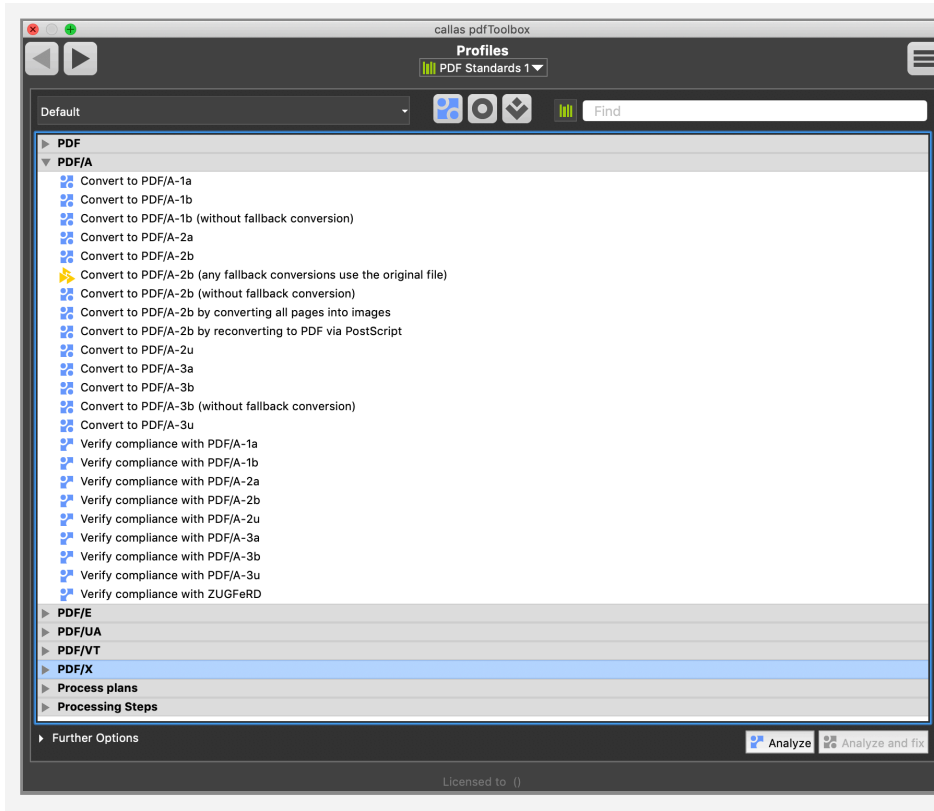
In the example above, the PDF document is marked as PDF/A1-b compliant. Upon clicking the coloured (green in this case but the colour depends on the type of standard compliance) button, a result window will pop up showing the compliance or non compliance result, like in the image below.




A PDF file with multiple standard compliance will be shown with multiple buttons, coloured individually based on the standard.



Another way to check the standards compliance is via the predefined Profiles. Under the 'PDF Standards' library, the desktop application comes with different Profiles for checking compliance with or even converting to a standard (PDF/A, PDF/X, PDF/UA etc).

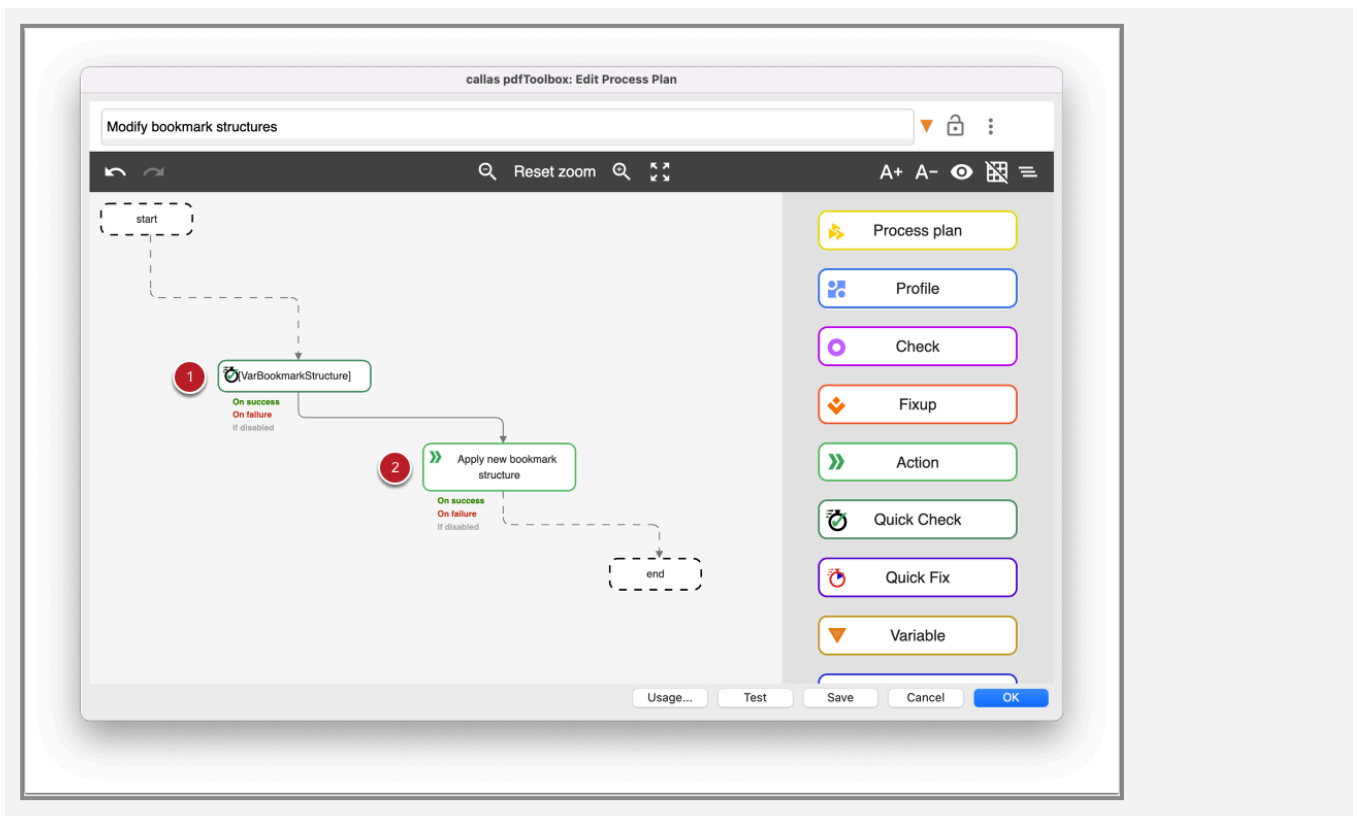


 Please note that the screenshots shown in this article are from the sister product of pdfaPilot.

3.15 Modifying structures in a Process Plan

pdfaPilot can extract existing DPart Metadata or bookmark structures in a JSON format so that you can modify them and inject them back into the PDF document.

This functionality is mainly used [on the CLI with the `--modifystructures` parameter](#), but it is also possible to modify the structures within pdfaPilot Desktop. Here you need to create a Process Plan where you extract the DPart or Bookmark structures using Quick Check and import them using the "Apply Structures" Action.

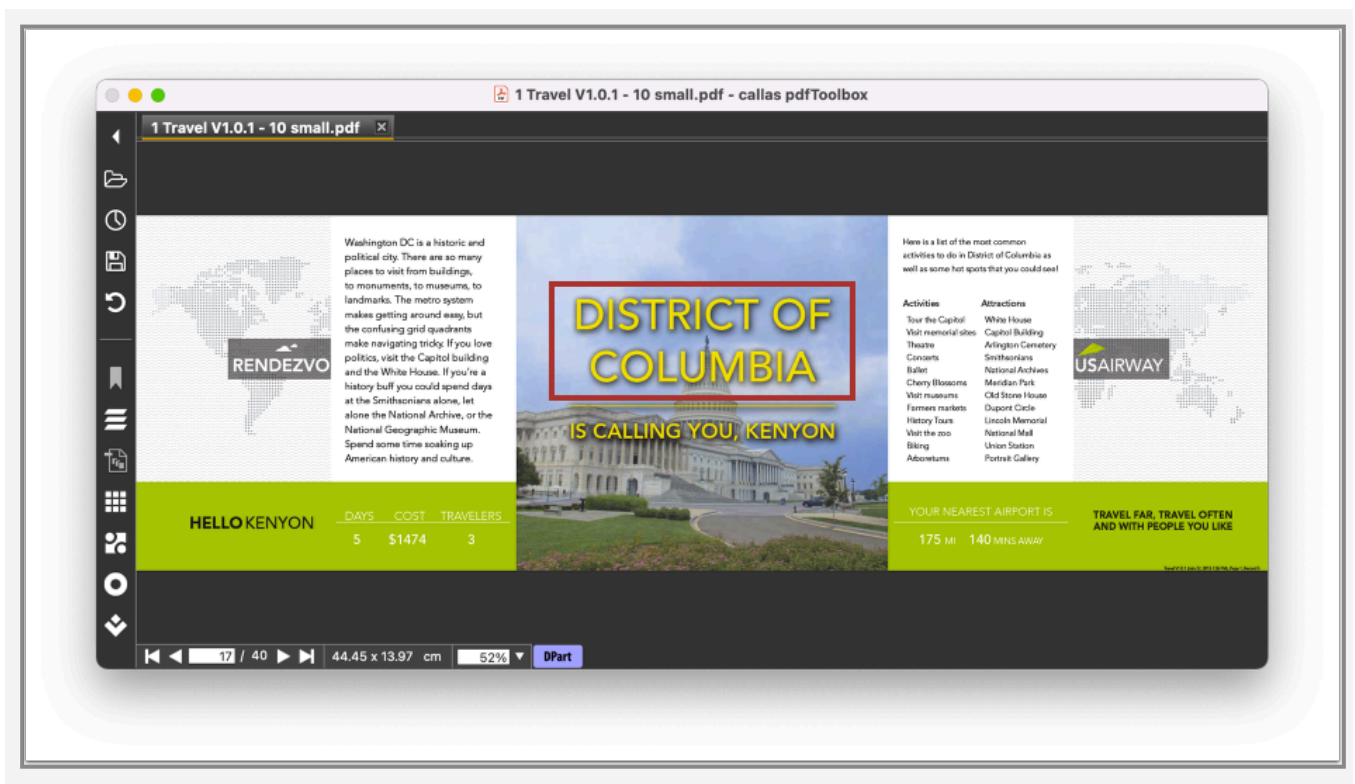


1. Quick Check to extract the existing structure to JSON. Relevant Quick Check filter expression:
 - DPart: "\$.aggregated.doc.dpartroot: true"
 - Bookmarks: "\$.aggregated.bookmarks: true"
2. "Apply structures" Action: JavaScript can be used to modify the existing structure from the Quick Check result and inject it into the PDF again.

3.16 Process Plan to create bookmarks from headings

This article provides a Process Plan that can be used to automatically generate bookmarks from the headings in a PDF.

The test file (taken from the PDF/VT sample files at <https://pdfa.org/resource/cal-poly-pdfvt-test-suite/>) contains several brochures for different cities. The Process Plan determines the headings (city name on the first page of each brochure) and uses them to create a bookmark structure.

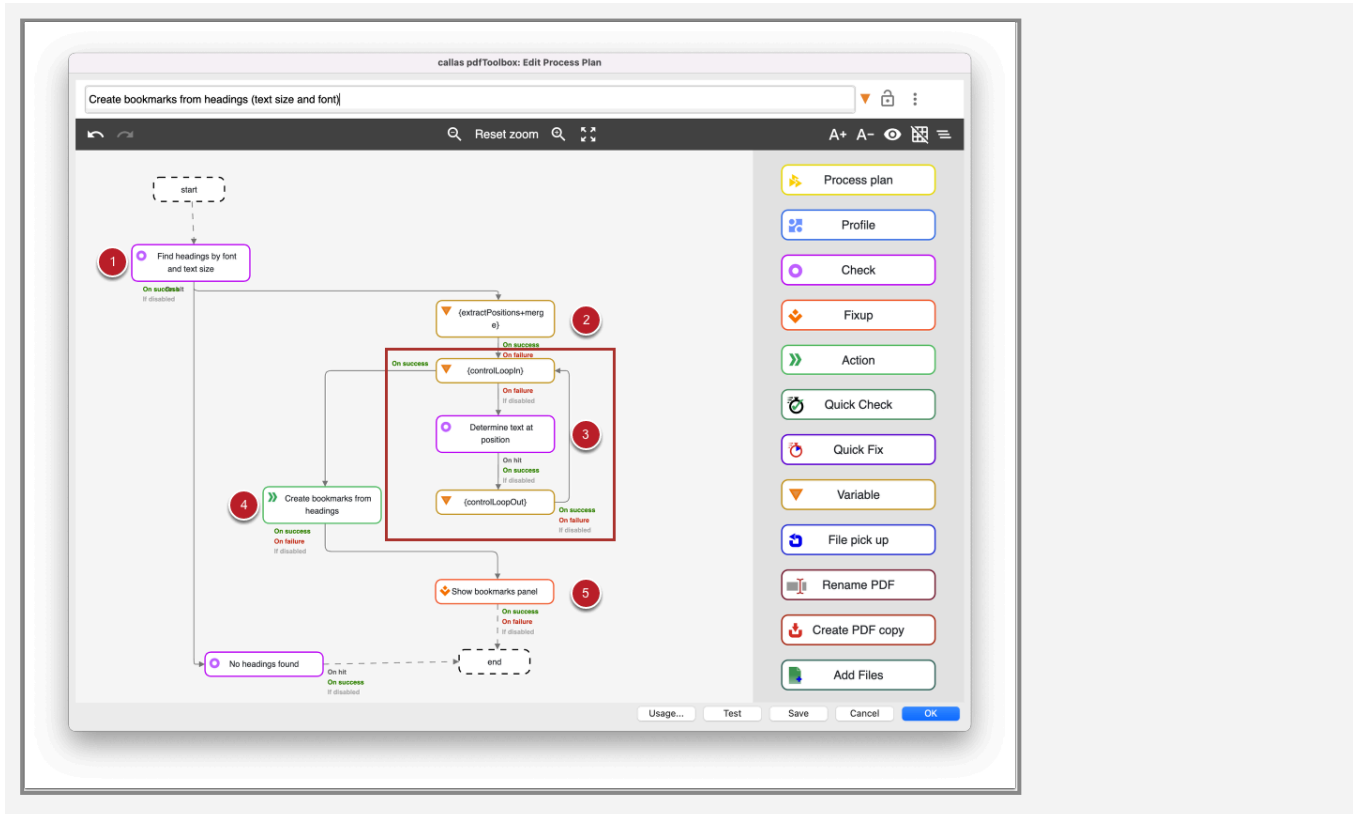


1 Travel V1.0.1 - 10 small.pdf



Create bookmarks from headings.kfpx

Let's have a look at the Process Plan:

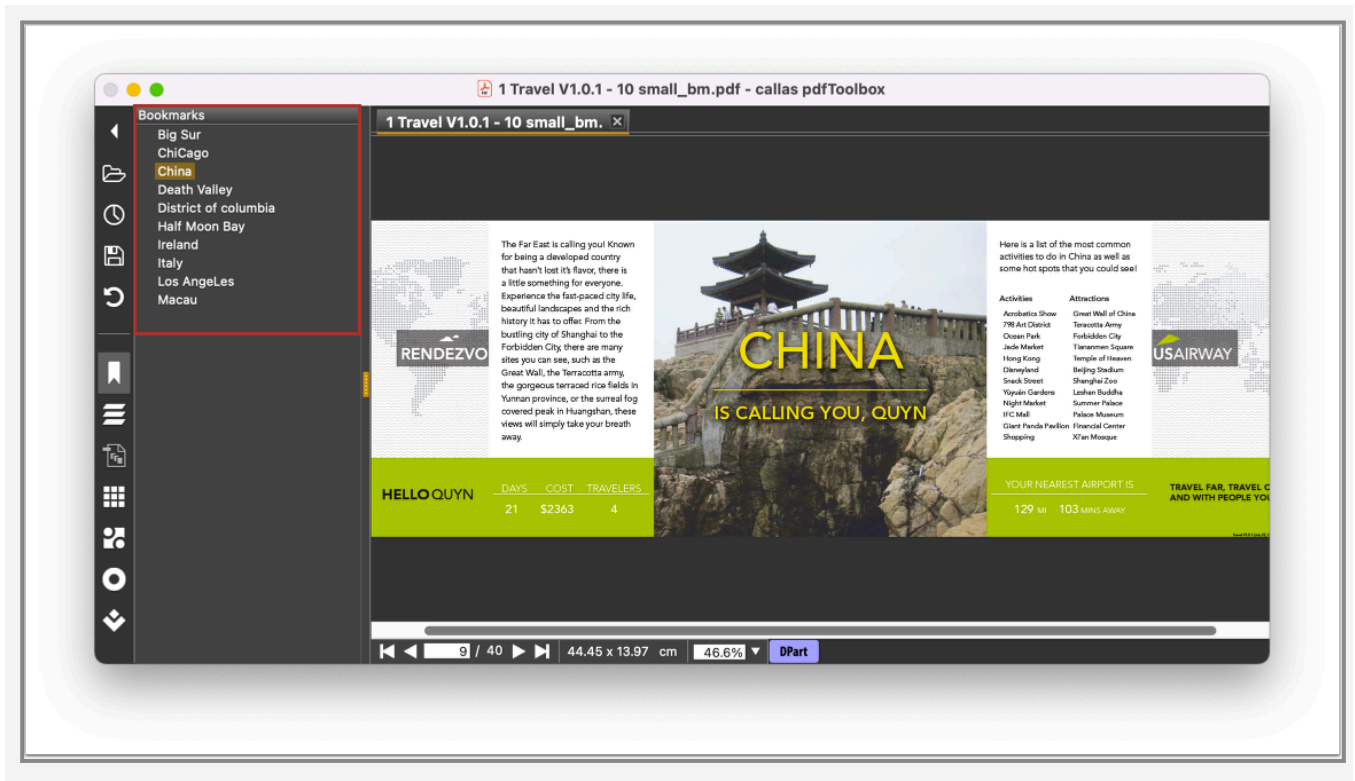


1. First, the positions of the headings are determined by a matching Check. This Check is individual and works only for the sample PDF file. It is important to properly detect all headings in the PDF using a combination of Check properties (font size, font type, font color, etc.).
2. The next step is the extraction of the text at the positions that have been detected in step 1. If the position of a text snippet touches another text snippet, they will be merged. This is important if you have headings that extend over two lines (like "District of Columbia" - see screenshot above) to receive only one bookmark for that heading.
3. A loop goes over an array to determine all headings. When all headings are determined, the next step starts.
4. To inject the bookmarks into the PDF the Process Plan uses the "Apply structures" Action. The information that has been gathered so far is converted into a proper JSON structure that can be used in this Action for bookmarks.
5. Finally, a Fixup defines that the bookmarks panel should be visible by default in the PDF viewer.

When executing the Process Plan, a Ask-at-runtime dialog will appear, to increase the search area for the text extraction

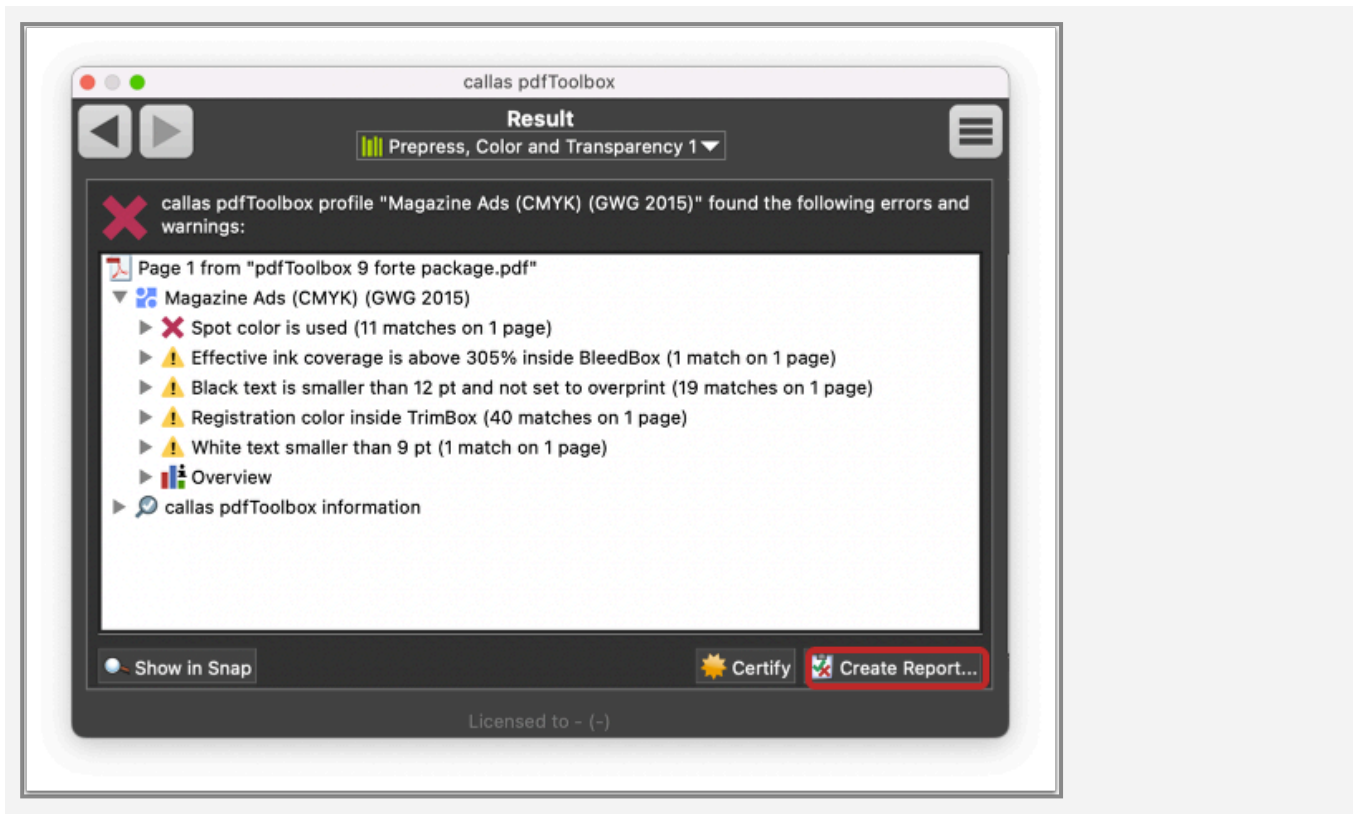
slightly in order to find all headings properly (the default value of 20 pt is suitable in most cases).

Result PDF with new bookmark structure



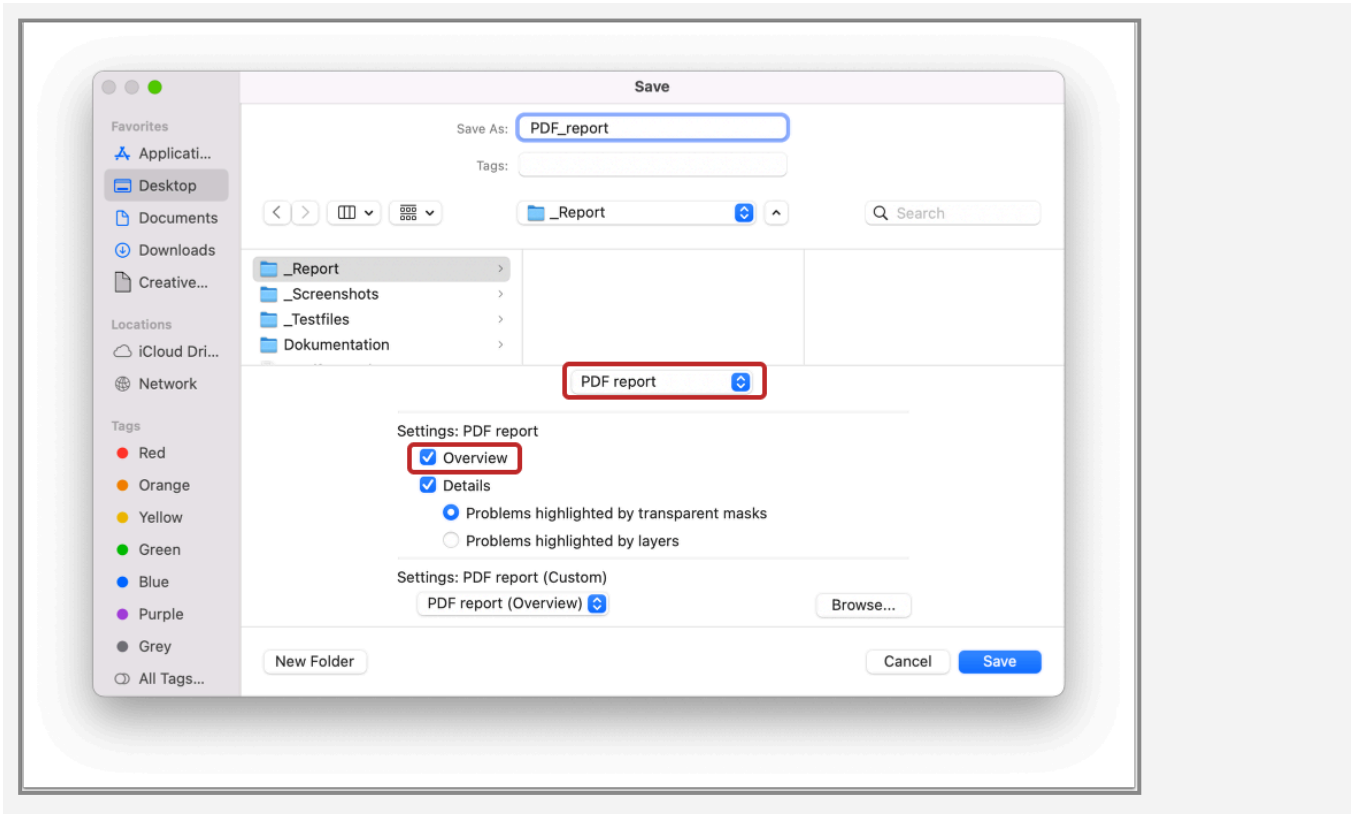
3.17 PDF reports (using masks or layers)

This article explains the different ways to create a PDF report. You have two different options to create a PDF report. Lets have a look at these.

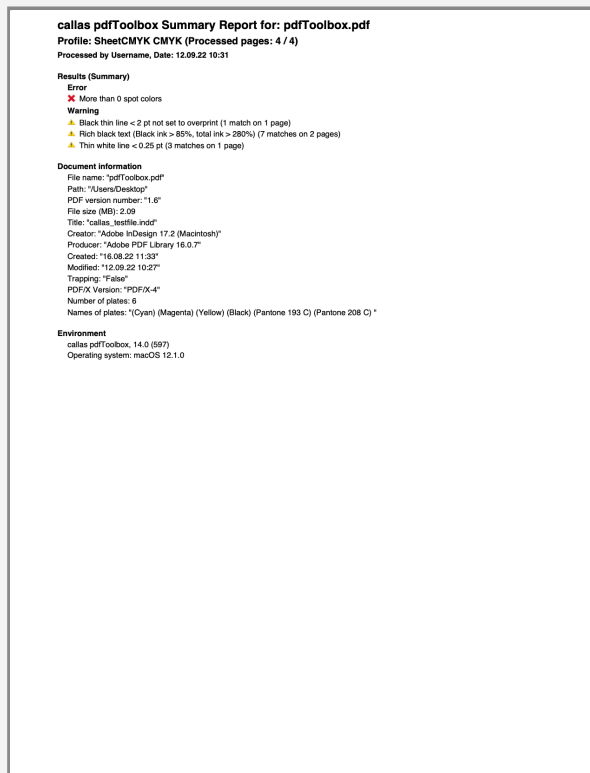


PDF report – Overview page

If "PDF report" is selected, a PDF document is created that can be opened in any PDF viewer. This PDF report optionally begins with an overview page; this page summarizes information about the preflighted document, the environment in which it was preflighted and provides a list of errors, infos and warnings. This PDF report is static, in the sense that you cannot change the layout, colors or text of the preflight report.

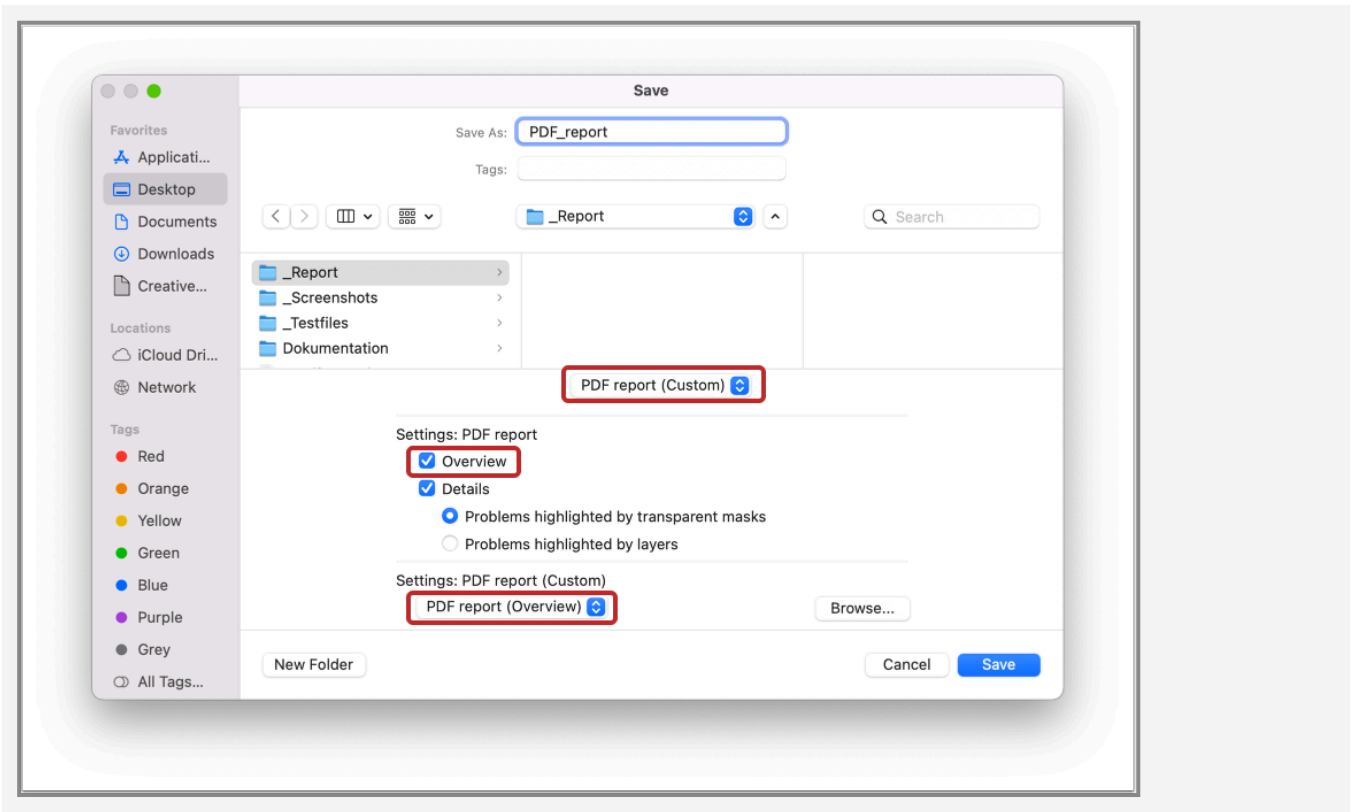


Example of the overview page of a PDF report

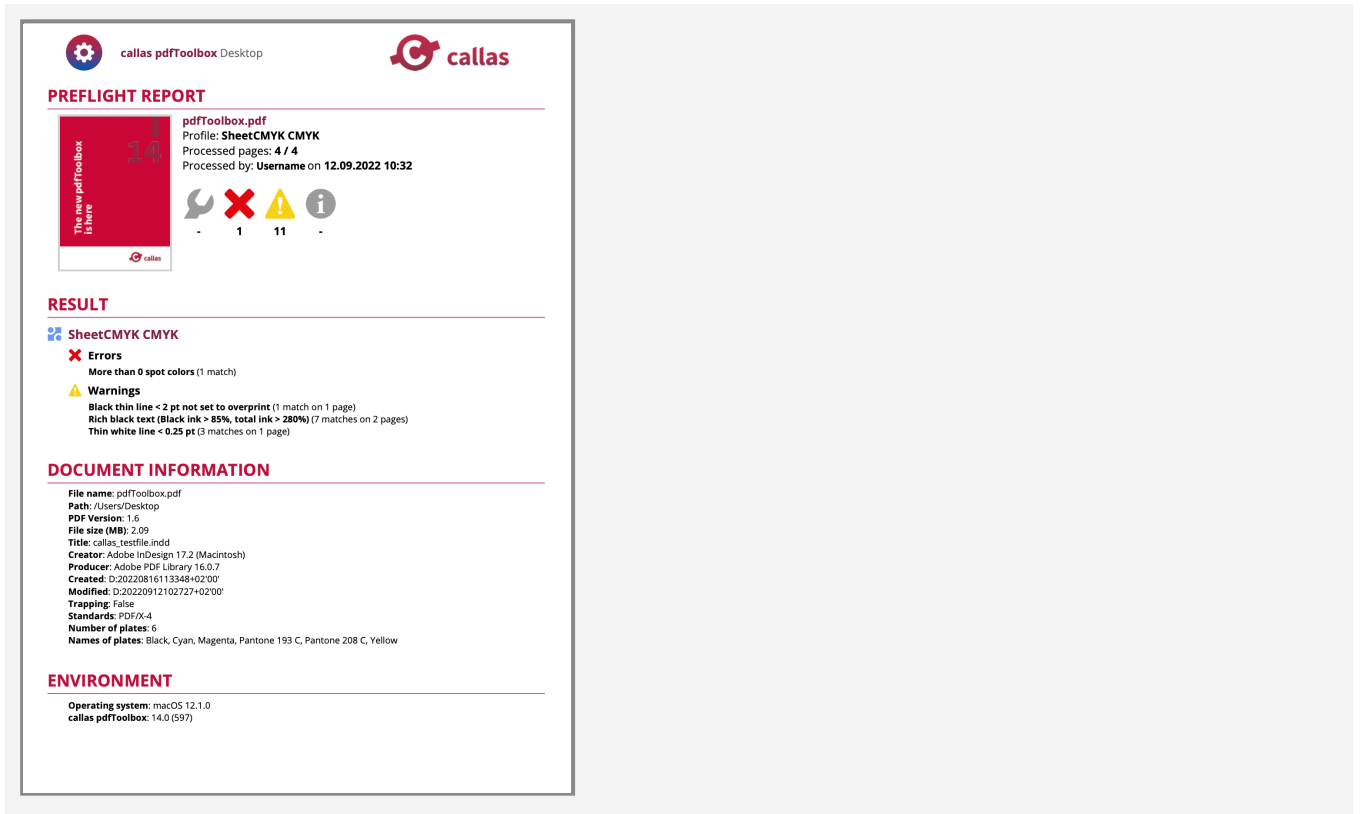


PDF report (Custom) – Overview page

If "PDF report (Custom)" is selected, a HTML template is used for the overview page layout. The product always comes with a predefined template called "PDF Report (Overview)", which can be selected at the bottom of the dialog.



Example of the overview page of a PDF report (Custom)

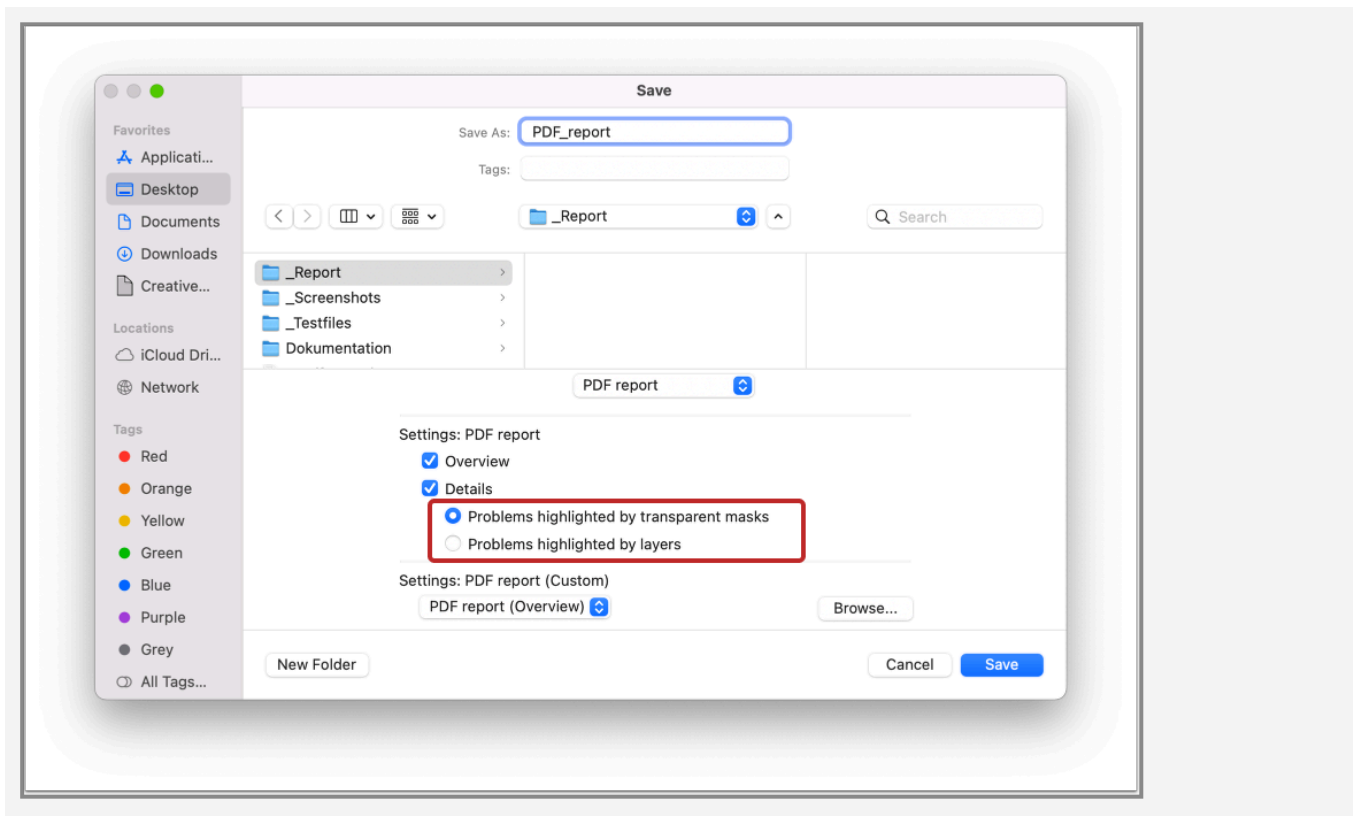


Unlike the static PDF report, the layout and content of the custom PDF report can be extended and customized:

- **Optional pages:** It is possible to add five optional pages to your report. These pages provide information about the spot colors, page information, ink coverage, ink amount and separation previews. How to add these pages to your report is explained in [this article](#).
- **Modify the template:** You can create your own templates, by modifying the predefined template. For example you can change colors, add your own logos or artwork and even modify what text is shown. The [next article](#) will give you an overview of all components of the report template.

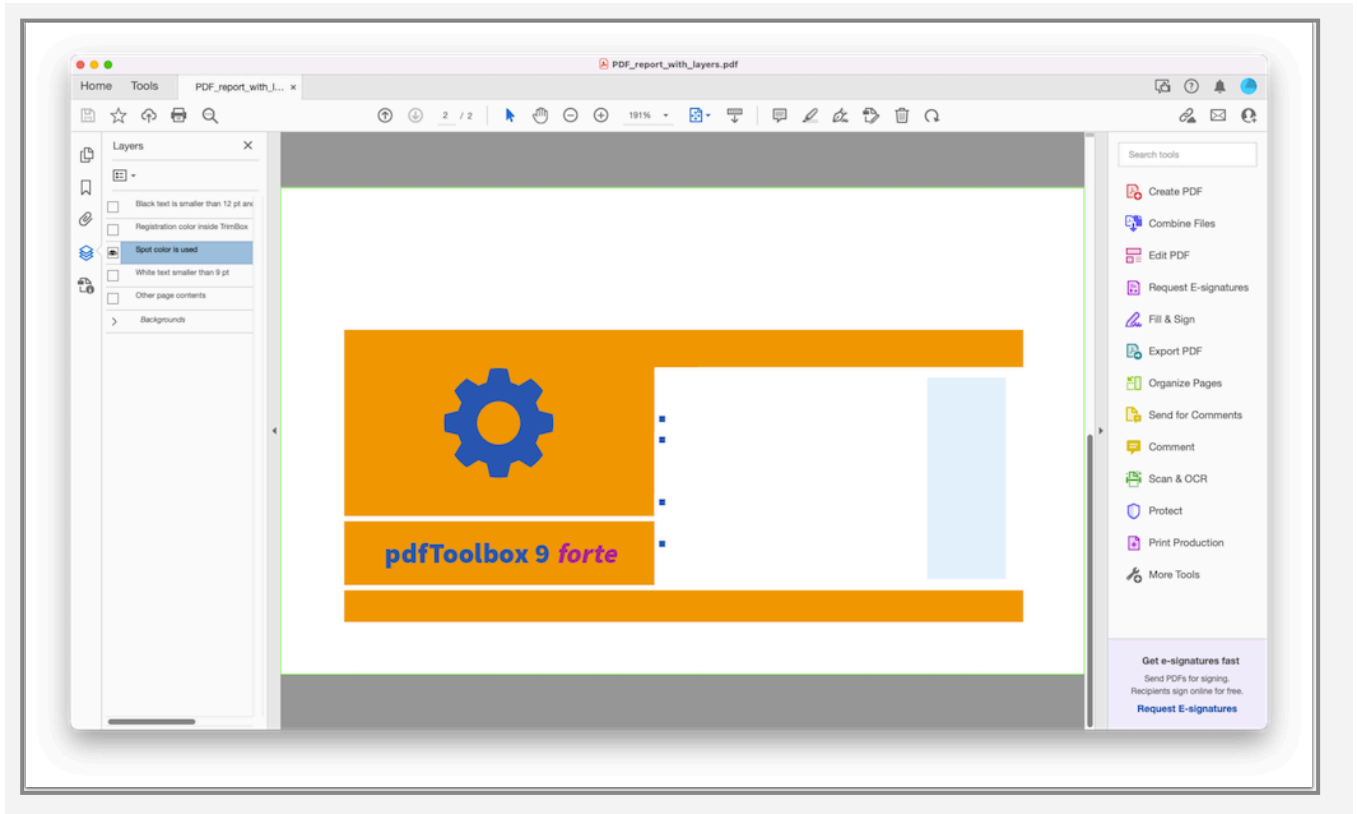
Highlighting problematic objects using masks or layers

In addition to the overview page you can generate a copy of the PDF file that was preflighted where all hits are identified and highlighted either by layers or masks. This copy will be added after the overview page.



Example of a layer report

The layer report puts objects with a particular problem on a layer; this allows toggling layers on and off to see all objects causing a Warning or Error. Because the report also adds background layers, you can easily see the problem objects on differently colored backgrounds (which is great to see white objects for example).



Example of a mask report

The transparency mask report uses a combination of transparent objects and comments to highlight all objects causing a Warning or Error. Additionally, bookmarks are added to make it easier to navigate through the highlighted areas.

The first level of the bookmark hierarchy has:

- The name of the Profile, Check or Fixup that was used. The destination is the first page of the report.
- The severities (Error, Warning or Info) of the hits encountered. The destinations are the first pages with a hit of the respective severity.
- "Switch highlighting off" stays on the current page but switches all masks off.

On the second level within Severities, there are the names of the respective Checks. Within each Check on the third level are all pages where the respective Check found a hit.

The screenshot displays the callas pdfaPilot interface. The main window shows a PDF report titled "PDF_report_with_masks.pdf". The report contains a table with the following data:

Date	Campaign	Batch Sequence ID	Approved
2016-05-31	08/15-B	KP-998-003-356	

Below the table is a large red warning box with a gear icon and the text "Spot color is used". The warning text includes:

- Black text is smaller than 12 pt and not set to overprint
- Effective ink coverage is above 305% inside BleedBox
- Registration color inside TrimBox
- White text smaller than 9 pt

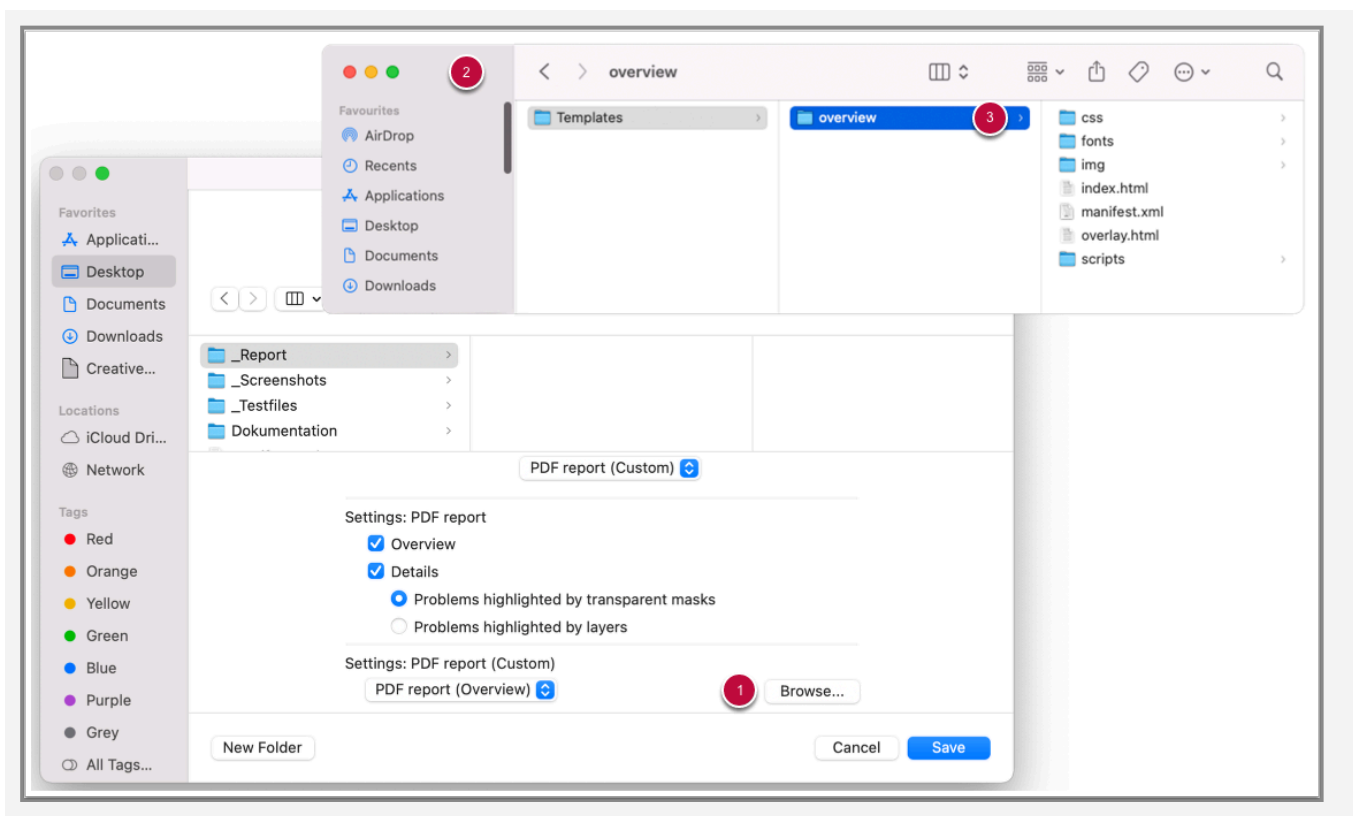
The right-hand side of the interface shows a "Comments" panel with 71 comments. The first comment from "callas pdfToolbox" at 11:25 states: "Black text is smaller than 12 pt and not set to overprint". The second comment states: "Spot color is used". The third comment states: "There are objects using spot color." The fourth comment states: "Black text is smaller than 12 pt and not set to overprint".

3.18 HTML based custom reports

As mentioned in the previous article, it is possible to generate a custom PDF report using a HTML template. The visual appearance is controlled by a HTML-Template and Custom Style Sheets (CSS), while the reported details are directly requested from the software itself or (optionally) parsed from an internally created JSON or XML report.

Structure of the custom HTML report

A predefined HTML template is always contained in all installer packages for Desktop and Server/CLI. The following screenshot shows how to get to the template folder:



1. When generating a report, click on the "Browse..." button in the settings section.
2. A new window into the file system opens where the Template folder is stored.
3. The **overview** folder is the predefined Template folder.

The predefined template contains several folders and files

- **index.html**: the template in HTML format that references a number of stylesheets, JavaScripts as well as a number of images
- **overlay.html**: contains the company and product logo that is placed on top of every report page
- **manifest.xml**: a XML file which defines information needed as content for the report, to be delivered by the engine
- **/css**: contains a style sheet
- **/fonts**: contains used fonts
- **/img**: contains used images
- **/scripts**: contains used JavaScripts



It is highly recommended to create a copy of the original template in a separate folder when starting to adjust a HTML-template based report.

The manifest.xml

The manifest.xml defines the set of information to be provided by the engine. This information will be used to fill up the details in the report based on the HTML-Template. Basic document information as well as all results of the processed profiles and a preview image are provided by default. Other parts like comparison images, a JSON or XML report, or parameters for optional document information can be also requested to enable picking up additional information about the PDF. These parts are commented out in the manifest by default. Here an example of an optional part of the manifest.xml:

```
1  <!--
2      Create a xml report
3
4      The path to the xml report is stored in cal_s_res_info.xml_report'
5
6      If either inkcovres or inkcovbox is specified, ink coverage information is added to the report
7
8      path: Destination path for XML report, Default: xml/report.xml
9      inkcovres: Resolution for ink coverage. Default: "10"
10     inkcovbox: Page box for ink coverage. Default: "CropBox"
11
12 <x:xmlreport path="xml/report.xml" inkcovres="10" inkcovbox="CropBox"/>
13 -->
```

As you can see, the whole XML report section is commented out by surrounding the code with `<!--` and `-->`. Down at the bottom you can find the XML report parameter. If you want to request a XML report, all you have to do is to uncomment the XML report parameter:

```
1  <!--
2      Create a xml report
3
4      The path to the xml report is stored in cal_s_res_info.xml_report'
5
6      If either inkcovres or inkcovbox is specified, ink coverage information is added to the report
7
8      path: Destination path for XML report, Default: xml/report.xml
9      inkcovres: Resolution for ink coverage. Default: "10"
10     inkcovbox: Page box for ink coverage. Default: "CropBox"
11
12 -->
13 <x:xmlreport path="xml/report.xml" inkcovres="10" inkcovbox="CropBox"/>
```

Parameters of the manifest.xml

Keep the temporarily generated files

```
<x:keeptemp>false/x:keeptemp>
```

If `true` the temporarily generated files like the filled index.html, CSS-files, images, JSON or XML reports and used JavaScripts will not be deleted after finishing the PDF report. They will be saved in a new folder with the extension .html instead. The folder name is identical to the PDF file. The folder is saved in the same location where the PDF file is located.

Options

- `false`: temporarily generated files will be deleted (default)
- `true`: temporarily generated files will be saved

Resolution for preview images

```
<x:defaults resolution="20"/>
```

This parameter allows you to set the default value for the resolution of all preview images. This value may be overwritten by the individual settings.

Options

- `resolution`: resolution used in ppi (default: "20")

Request basic information about PDF

```
<x:dict>  
  <x:overview/>  
</x:dict>
```

If contained, document information and results of the performed profile will be available for using them in the HTML template.

Visual comparison of original and processed file

```
<x:compare>
  <x:document_a resolution=20/>
  <x:document_b resolution=20/>
  <x:diffresult resolution=20/>
</x:compare>
```

Include compare tree if comparison resources are used inside index.html.

Attribut

- `resolution`: resolution used in ppi for rendering the comparison (default: "20")

Create a XML report

```
<x:xmlreport path="xml/report.xml" inkcovres="10" inkcovbox="CropBox"/>
```

Requests a XML report of the performed profile to extract additional information using JavaScript which can be used in the report. The ink coverage is determined only if one of the corresponding parameters is present.

Options

- `path`: Destination path for XML report (Default: "xml/report.xml")
- `inkcovres`: Resolution for ink coverage (Default: "10")
- `inkcovbox`: Page box for ink coverage (Default: "CropBox")

Create a JSON report

```
<x:jsonreport path="json/report.json" quickcheck="default" />
```

Requests a JSON report to extract additional information.

Options

- `path`: Destination path for JSON report (Default: "json/report.json")
- `quickcheck`: Optional relative path to a custom quickcheck config in the template folder. This quickcheck config will replace the internal default config (quickcheck="none" will disable the default config). More about the default Quick Check configuration file can be found [here](#).

Page previews

```
<x:preview resolution="20" pageselector="1"/>
```

If this parameter is present in the manifest.xml, page previews are generated for the selected pages. The preview of the first selected page is added to the overview page of the preflight report for visual representation of the PDF file. The preview images are exported into the file system as png files. The folder in the intermediate HTML is:

```
./img/cals_pages/../../cals_src_a.png
```

Options

- `resolution`: Resolution in ppi for ink amount previews (Default: "20")
- `pageselector`: Selected pages for previews (Default: "1")

Optional information

```
<x:inkamountheatmaps ... />  
<x:inkcoverage ... />  
<x:spotcolors ... />  
<x:separations ... />  
<x:pageinfo ... />
```

There are five new parameters in the results section, that generate optional information about the document. Read more about that in the next article: [HTML based custom reports – generate optional information](#).

3.19 HTML based custom reports – generate optional information

The previous article ([HTML based custom reports](#)) explained the different components of the HTML based custom report. In this article you will learn how to add additional pages to your custom HTML report.

i **Note:** The optional pages are available since pdfToolbox 14 (and pdfaPilot 12) – The optional pages "Image resolution" and "Small objects" are available since pdfToolbox 15 (and pdfaPilot 13)

The optional information

The custom PDF report contains an overview page that summarizes information about the preflighted document, the environment and information about the executed profile with a list of Errors, Infos, Warnings and Fixups. Until pdfToolbox 14 the report did not provide any information about spot colors, page size, ink amount, ink coverage or separation previews. Now you have the possibility to order seven additional pages in the **manifest.xml** that are then added to your custom PDF report:

- Page information
- Ink coverage
- Ink amount
- Spot color information
- Separation preview
- Small objects
- Image resolution

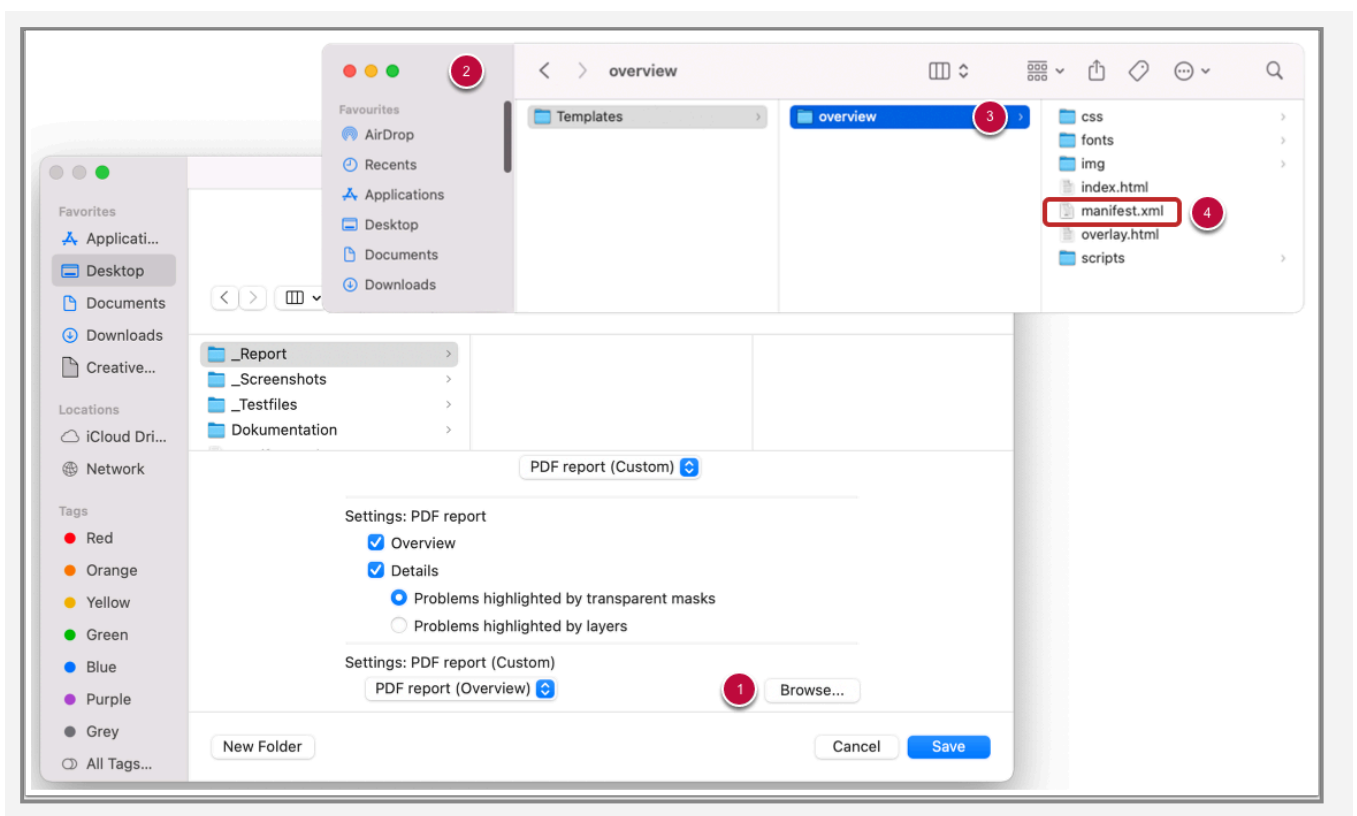
All the parameters are part of the `<x:results>` section of the manifest.xml.

i **Note:** If you do not find the additional information listed below in your manifest.xml, or if you still get

the old layout for the overview page, this may be because you imported the old libraries and resources when you upgraded to pdfToolbox 14/pdfToolbox 15. If this is the case, you can simply [create a new library](#) (including resources) to get the new template, or you can replace the old templates folder with the [folder attached below](#).

How to open the manifest.xml?

The additional information can be ordered manually via the manifest.xml. The following screenshot documents how to get to the manifest.xml:



1. When generating a report, click on the "Browse..." button in the settings section.
2. A new window into the file system opens where the Template folder is stored.
3. The **overview** folder is the predefined Template folder.
4. In this folder the **manifest.xml** is stored.

How to order the optional information in the manifest.xml?

Since these pages are optional, you decide if you want to add one or all pages to your report or not. The default is that none of these information will be generated and added to the report. All five parameters are commented out in the manifest.xml with `<!--` and `-->`:

```
1 <!-- - Create ink amount heatmap previews and show ink amount page in report
2           If present page ink amount previews are generated for the selected pages.
3
4           Paths to the preview files are stored in cal_s_doc_info.docs[0].pages[i].ink_amount
5
6           resolution : Resolution in ppi for ink amount previews. Default: "20"
7           threshold  : Threshold in % for ink amount previews. Default: "300"
8           pageselector: Selected pages for previews, default: "all"
9           (see online documentation for full pageselector syntax)
10          show       : Add ink amount section to custom report. Default: "1"
11
12          <x:inkamountheatmaps resolution="20" threshold="300" pageselector="all" show="1"/>
13          -->
```

To add one page to your report all you have to do is uncomment the parameter:

```
1 <!-- - Create ink amount heatmap previews and show ink amount page in report
2         If present page ink amount previews are generated for the selected pages.
3
4         Paths to the preview files are stored in calx_doc_info.docs[0].pages[i].ink_amount
5
6         resolution : Resolution in ppi for ink amount previews. Default: "20"
7         threshold  : Threshold in % for ink amount previews. Default: "300"
8         pageselector: Selected pages for previews, default: "all"
9         (see online documentation for full pageselector syntax)
10        show       : Add ink amount section to custom report. Default: "1"
11
12        -->
13        <x:inkamountheatmaps resolution="20" threshold="300" pageselector="all" show="1"/>
14
```

Now the parameter is uncomment and present in the manifest.xml.

You want to add all optional pages to your report?

If you want to add all pages to your PDF report, we provide a template for download, where all optional information in manifest.xml is already uncommented. Just follow the instructions:

1. Download one of the templates below:

- overview_with_optional_pages.zip: earliest support version = pdfToolbox 14
- overview_with_optional_pages_pdfToolbox15.zip: earliest support version = pdfToolbox 15

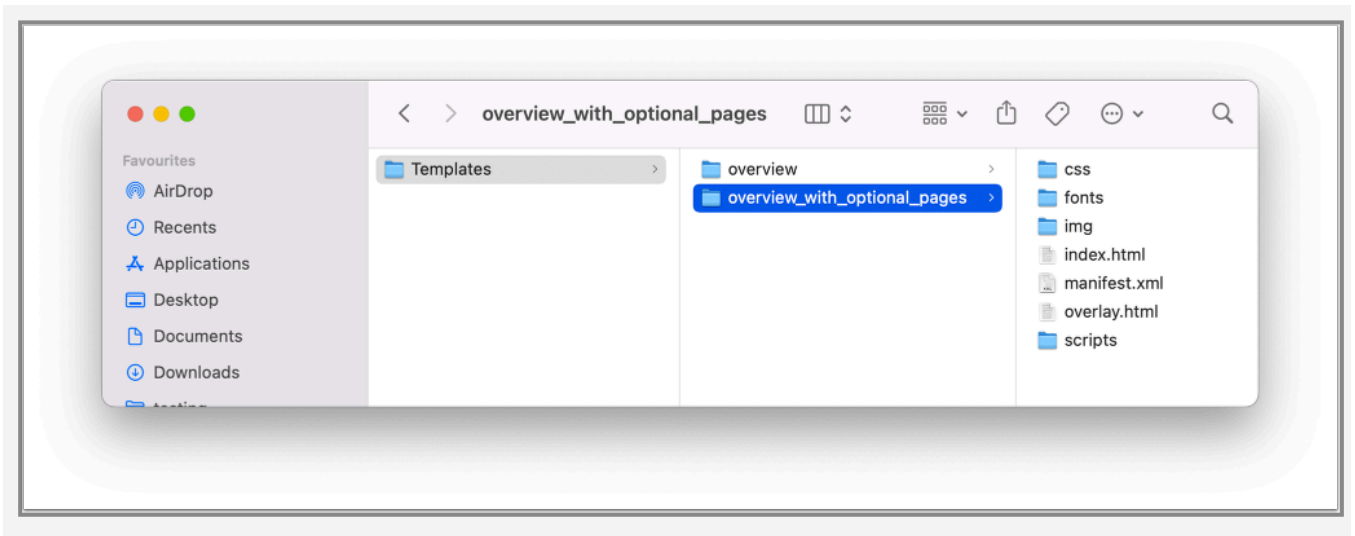


overview_with_optional_pages.zip

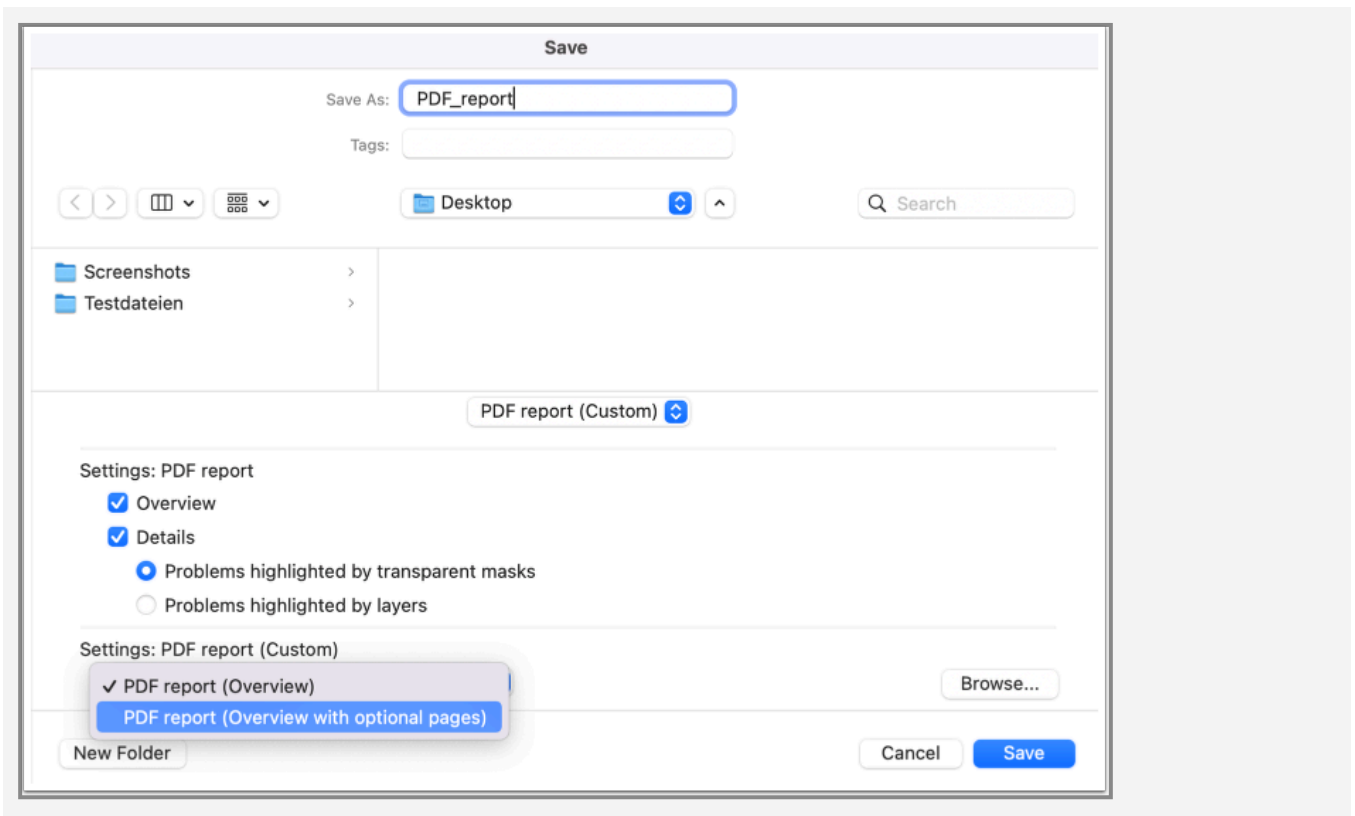


overview_with_optional_pages_pdfToolbox15.zip

2. Add the downloaded template to the "Templates" folder.
The [screenshot](#) at the beginning of the article shows you how to get to the template folder.



3. In order to display the new template in pdfToolbox, the "Save" dialog must be opened again. Then the template "PDF report (Overview with optional pages)" can be selected in the settings options.



Parameters for the optional report information

The show option

All optional information parameters have a show option.

The default for the show option is `show="1"`. This means that the data will be generated and the optional page will be added to the report.

If you want to use your own template you can set `show="0"`. This will generate the data for the specific parameter, but not the optional report page.

Ink amount heatmaps

```
<x:inkamountheatmaps resolution="20" threshold="300" onlyproblems="1" pageselec-  
tor="all" show="1"/>
```

Creates ink amount heatmap previews and show ink amount for all pages or selected pages by highlighting any area of the file that goes over a certain threshold. The visuals that are created are similar to the visualizer mode "Ink coverage".

Parameters

- `resolution` : Resolution in ppi for ink amount previews (Default: "20")
- `threshold` : Threshold in % for ink amount previews (Default: "300")
- `pageselector` : Selected pages for previews (Default: "all") (click [here](#) for full pageselector syntax)
- `onlyproblems` : Only emit pages that have problems (Default: "1") (earliest support version = pdfToolbox 15)
- `show` : Add ink amount section to custom report (Default: "1"), `show="0"` will only generate the data

Example ink amount report page



Ink coverage

```
<x:inkcoverage resolution="10" pagebox="CropBox" pageselector="all" listpages="1"
show="1"/>
```

If present the page "Ink coverage" is added to the PDF report. The ink coverage information can refer to all or selected pages. For large files, the creation of this page may take some time, since all pages must be rendered.

Parameters

- `resolution`: Resolution in ppi for ink coverage determination (Default: "10")
- `pagebox`: Page box for ink coverage (Default: "Crop-Box")
- `pageselector`: Selected pages for ink coverage information in custom report (Default: "all") (click [here](#) for full pageselector syntax)

- `listpages`: Add ink coverage information per Page (Default: "1"), listpages="0" will only list ink coverage for the hole document (total)
- `show`: Add ink coverage section to custom report (Default: "1"), show="0" will only generate the data

Example ink coverage report page

callas pdfToolbox Desktop		callas	
INK COVERAGE			
Total			
Cyan	0.60 %	15.97 cm ²	2.48 inch ²
Magenta	0.82 %	21.87 cm ²	3.39 inch ²
Yellow	0.72 %	19.05 cm ²	2.95 inch ²
Black	2.19 %	58.22 cm ²	9.02 inch ²
Pantone 193 C	28.14 %	562.13 cm ²	87.13 inch ²
Pantone 208 C	2.93 %	77.98 cm ²	12.09 inch ²
Per page			
Page 1			
Cyan	0.22 %	1.48 cm ²	0.23 inch ²
Magenta	0.56 %	3.72 cm ²	0.58 inch ²
Yellow	0.45 %	2.96 cm ²	0.46 inch ²
Black	0.28 %	1.86 cm ²	0.29 inch ²
Pantone 193 C	79.24 %	527.59 cm ²	81.78 inch ²
Pantone 208 C	3.01 %	20.02 cm ²	3.10 inch ²
Page 2			
Cyan	1.05 %	6.97 cm ²	1.08 inch ²
Magenta	1.35 %	9.01 cm ²	1.40 inch ²
Yellow	1.13 %	7.55 cm ²	1.17 inch ²
Black	2.41 %	16.05 cm ²	2.49 inch ²

Spot color information

```
<x:spotcolors pageselector="all" show="1"/>
```

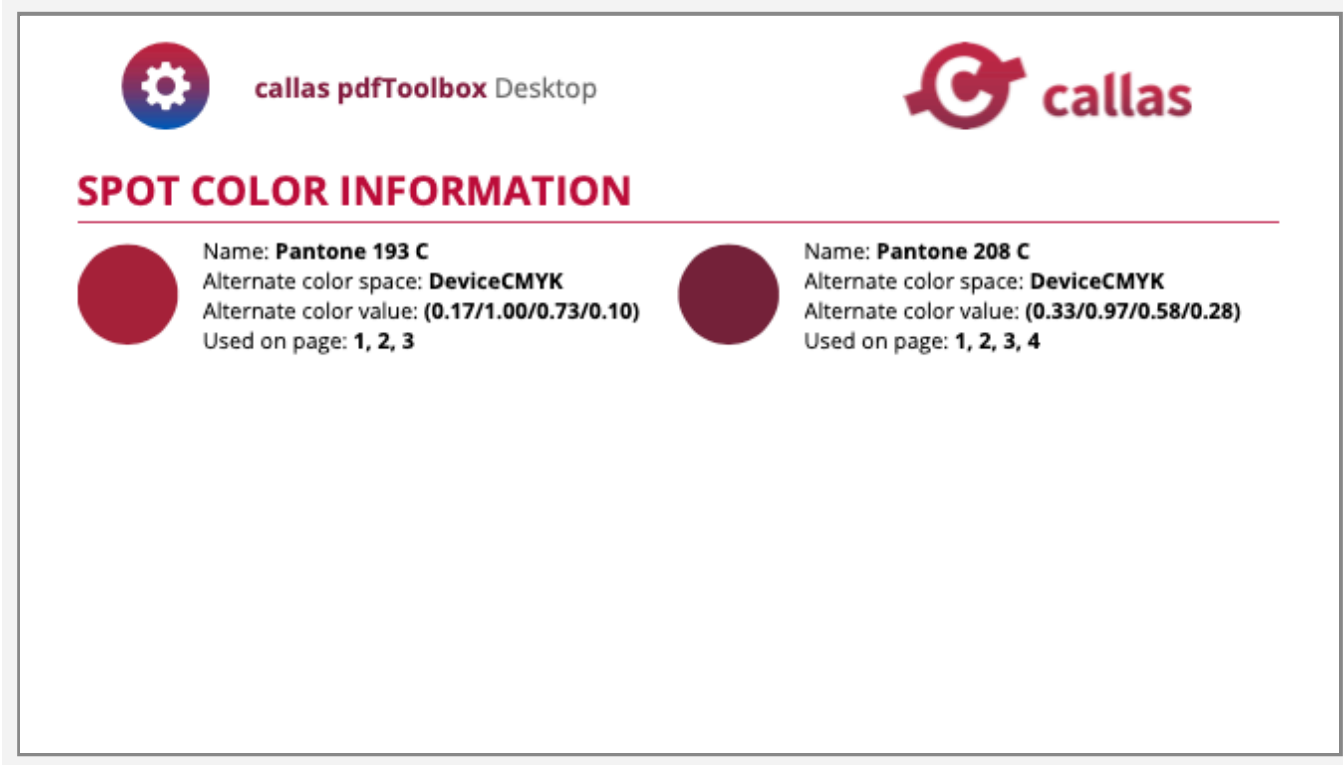
If present the spot color information is generated. The spot color information page will contain all used spot colors including the spot color name, the alternate color space and value and on which pages the color has been used.

Parameters

- `pageselector`: Selected pages for spot color information in custom report, default: "all" (click [here](#) for full pageselector syntax)

- `show`: Add spot color section to custom report (Default: "1"), `show="0"` will only generate the data



Example Spot color report page



callas pdfToolbox Desktop

callas

SPOT COLOR INFORMATION

	Name: Pantone 193 C Alternate color space: DeviceCMYK Alternate color value: (0.17/1.00/0.73/0.10) Used on page: 1, 2, 3		Name: Pantone 208 C Alternate color space: DeviceCMYK Alternate color value: (0.33/0.97/0.58/0.28) Used on page: 1, 2, 3, 4
---	---	---	--

Seperation preview for Process and Spot colors

```
<x:separations type="spotifpresent" resolution="20" pageselector="all" show="1"/>
```

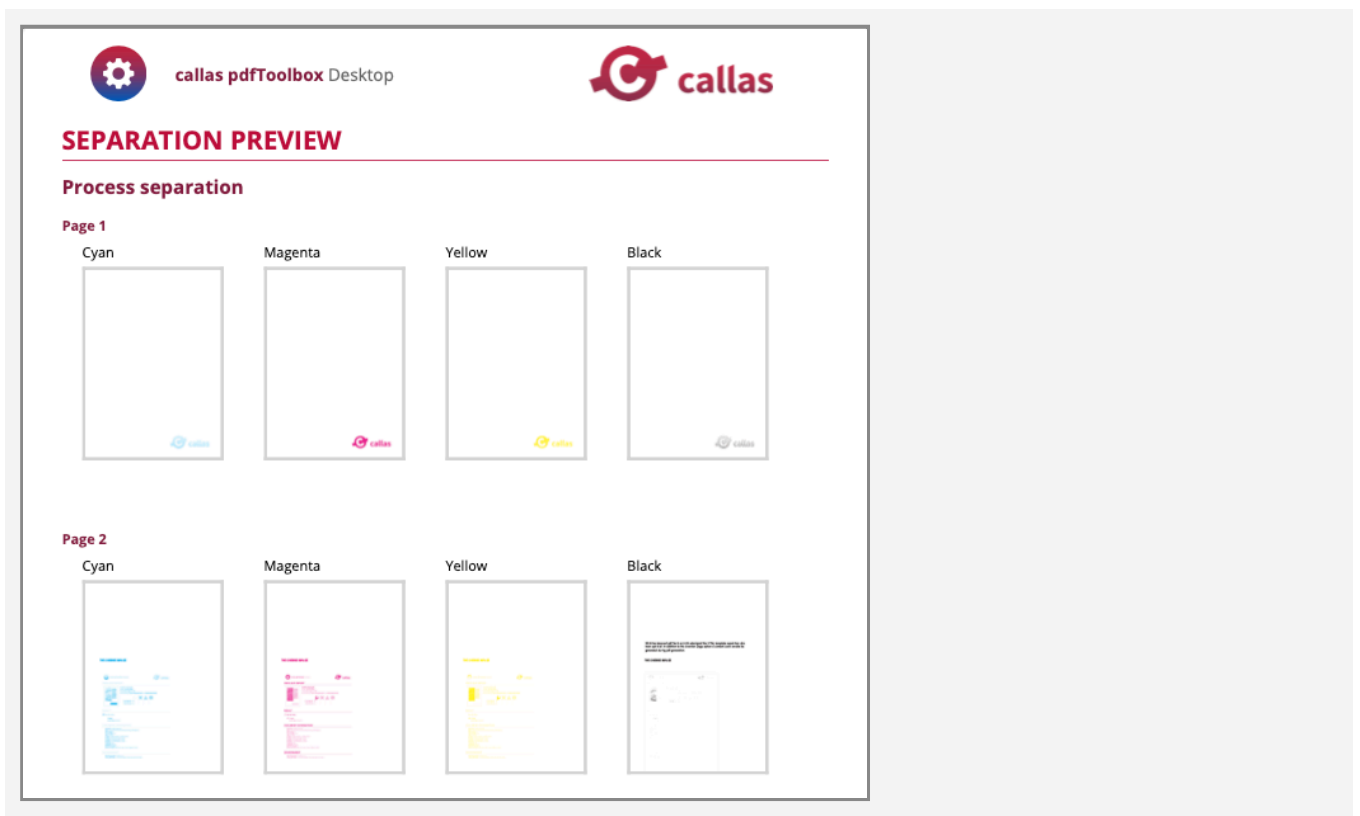
Creates separation previews for process separation (Cyan, Magenta, Yellow and Black) and spot color separation.

Parameters

- `resolution`: Resolution in ppi for separation previews (Default: "20")
- `type`: Type for separation views, any of (earliest support version = pdfToolbox 15):
 - "all": all separation views (Process separation and spot color separation pages are always created)
 - "process": Only process color separation views
 - "spot": Only spot color separation views

- "spotifpresent": Process color separation views and only spot color separation views if document has spot color (Default)
- `pageselector`: Selected pages for previews (Default: "all") (click [here](#) for full pageselector syntax)
- `show`: Add separation preview section to custom report (Default: "1"), `show="0"` will only generate the data

Example separation preview report page




Page information

```
<x:pageinfo pageselector="1" resolution="20" safetyzoneinside="3 mm" safetyzoneoutside="3 mm" usebleedbox="0" unit="mm" show="1"/>
```

Generates a page information for the first page by default. This provides the sizes of all page geometry boxes, as well as an information about the page rotation and the page scaling factor. If a page scaling factor is set, you also get the effective sizes of the geometry boxes. A preview images with a predefined safety zone and highlighted TrimBox and BleedBox is


also part of the page. Additionally you get the information if all pages of the document have the same size or not.

-  If multiple pages are specified in the pageselector, only the first selected page will be considered for the template report.


Parameters

- `resolution`: Resolution in ppi for safety zone preview (Default: "20")
- `pageselector`: Selected pages for previews (Default: "1") (click [here](#) for full pageselector syntax)
- `safetyzoneinside`: Safety zone inside (Default: "3 mm")
- `safetyzoneoutside`: Safety zone outside (Default: "3 mm")
- `usebleedbox`: Use bleed box if existing (Default: "0")
- `unit`: Unit for page geometry boxes. Options: mm, pt, cm, inch (Default: "mm")
- `show`: Add page information section to custom report (Default: "1"), show="0" will only generate the data

Example page information report page




callas pdfToolbox Desktop



PAGE INFORMATION

Page geometry boxes & safety zone

Page 1



Safety zone

Safety distance inside 3 mm

Safety distance outside 3 mm

Page geometry boxes

	Defined size	Effective size
MediaBox	220.00 x 307.00 mm	440.00 x 614.00 mm
CropBox	220.00 x 307.00 mm	440.00 x 614.00 mm
TrimBox	210.00 x 297.00 mm	420.00 x 594.00 mm
BleedBox	220.00 x 307.00 mm	440.00 x 614.00 mm
ArtBox	220.00 x 307.00 mm	440.00 x 614.00 mm

Page scaling factor 2

Page rotation 0

All pages have the same size

Image resolution

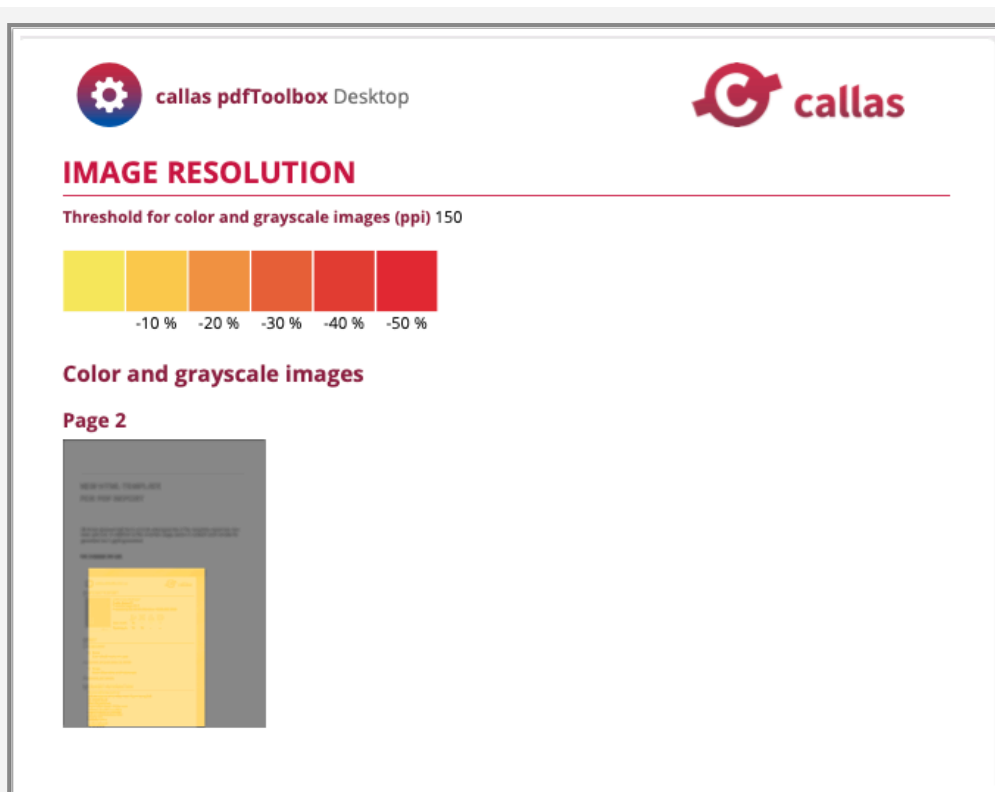
```
<x:imageresolution type="img" resolution="20" imgthreshold="150" bmpthreshold="550" onlyproblems="1" pageselector="all" show="1"/>
```

Creates the Image Resolution report page, which shows all pages where an image resolution is below the specified threshold by highlighting those images. The visuals that are created are similar to the visualizer mode "Image resolution".

Parameters

- **type**: Image type for image resolution views, any of:
 - "all": all image resolution views
 - "img": image resolution views (Default)
 - "bmp": bitmap resolution views
- **resolution**: Resolution in ppi for image resolution previews (Default: "20")

- `imgthreshold`: Threshold in ppi for image resolution previews (Default: "150")
- `bmpthreshold`: Threshold in ppi for bitmap resolution previews (Default: "550")
- `onlyproblems`: Only emit pages that have problems (Default: "1")
- `pageselector`: Selected pages for previews (Default: "all") (click [here](#) for full pageselector syntax)
- `show`: Add page information section to custom report (Default: "1"), `show="0"` will only generate the data





Small objects

```
<x:smallobjects type="all" threshold="medium" resolution="20" onlyproblems="1"
pageselector="all" show="1"/>
```

Creates the Small Objects report page and displays thin line objects and/or small text objects for all pages or selected pages by highlighting any object that is below a specified threshold. The visuals that are created are similar to the visualizer mode "Small objects".

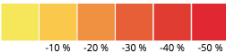


Parameters

- `type`: Type of small objects, any of:
 - "all": all small object views (Default)
 - "text": small text objects below threshold
 - "lines": small vector objects below threshold
- `resolution`: Resolution in ppi for image resolution previews (Default: "20")
- `threshold`: Threshold for small object coverage highlighting, any of:
 - "low": low resolution output (0.2 pt single color lines & 0.5 pt multicolour lines, 8 pt single color text & 10 pt multicolour text)
 - "medium": medium resolution output (Default) (0.125 pt single color lines & 0.25 pt multicolour lines, 5 pt single color text & 9 pt multicolour text)
 - "high": high resolution output (0.125 pt single color lines & 0.25 pt multicolour lines, 5 pt single color text & 8 pt multicolour text)
- `onlyproblems`: Only emit pages that have problems (Default: "1")
- `pageselector`: Selected pages for previews (Default: "all") (click [here](#) for full pageselector syntax)
- `show`: Add page information section to custom report (Default: "1"), show="0" will only generate the data


SMALL OBJECTS

Threshold (%) Medium resolution output

	Single color  0.125 pt	T 5 pt
	Multicolor  0.25 pt	T 9 pt


Small line objects

Page 2



Small text objects

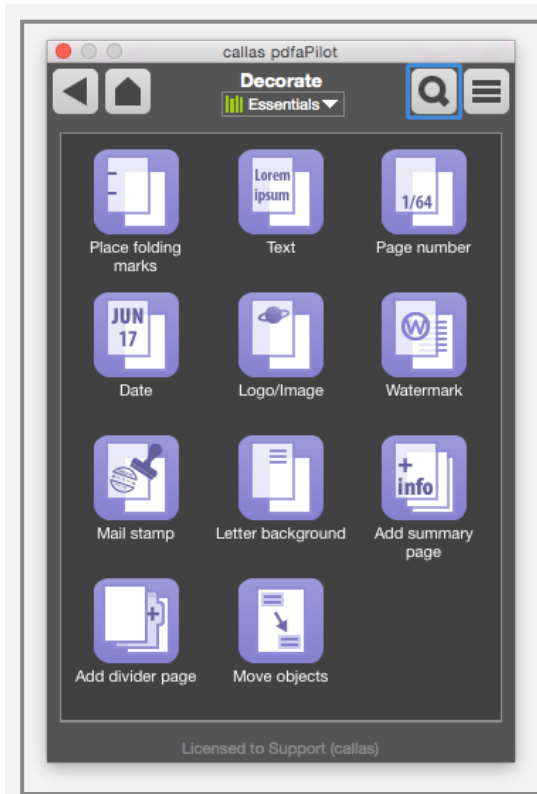
Page 1



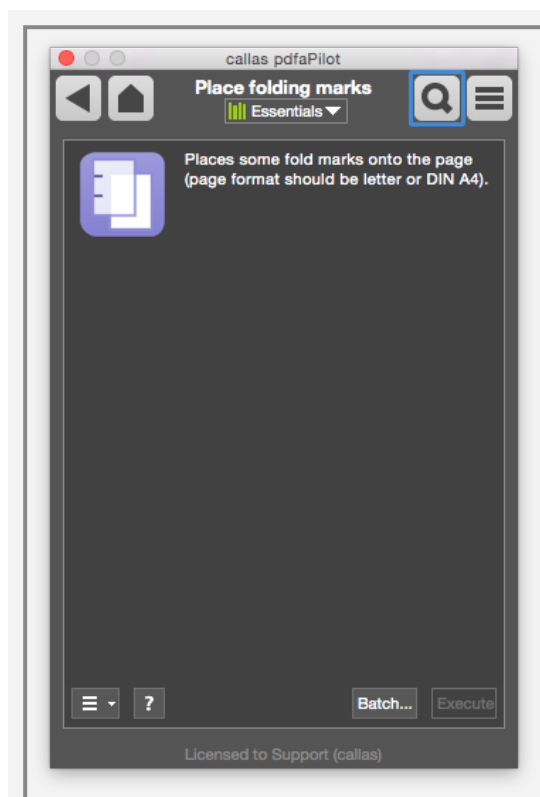
4. Decorate pages by adding content

4.1 Actions: Decorate

The Switchboard Actions under “Decorate” let you place a range of different object types within a PDF.



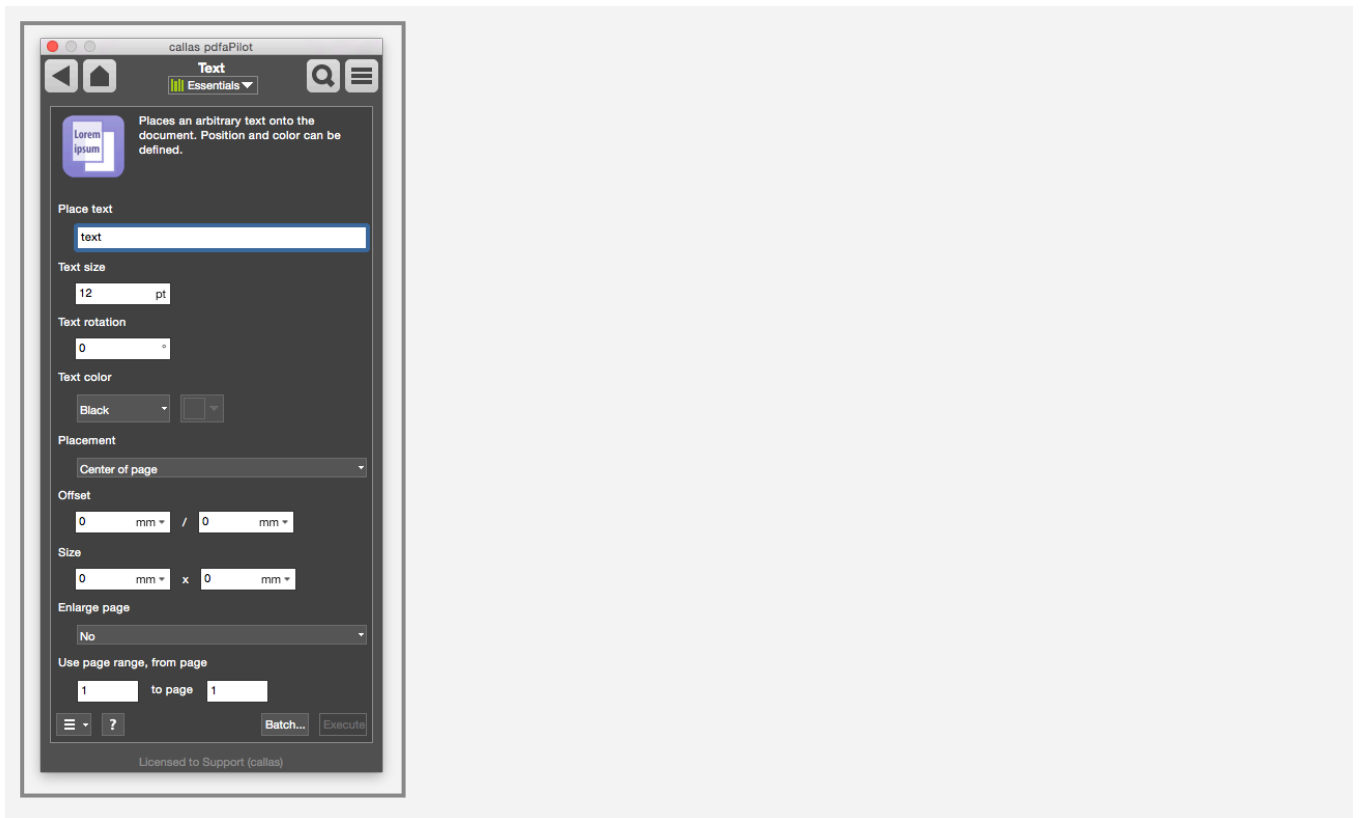
1. 1. Place folding marks



Adds two folding marks to pages.

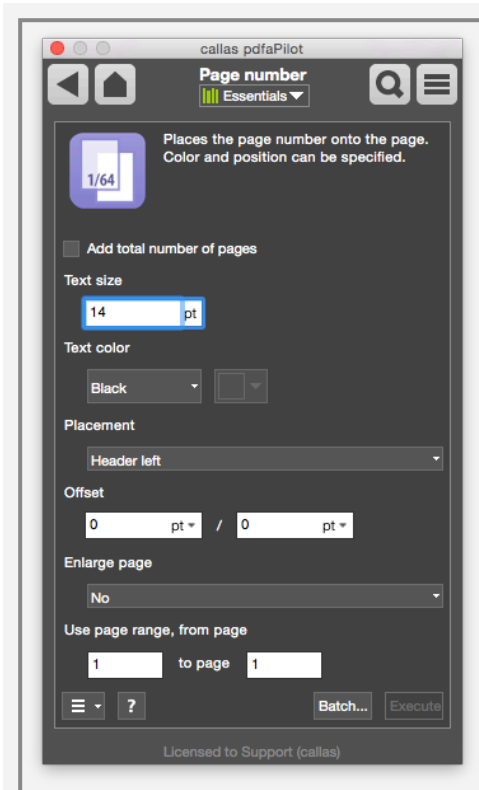
These will be generated and positioned based on the page height.

2. 2. Text



Places user-defined text on the page.
You can specify settings such as the text size and position.

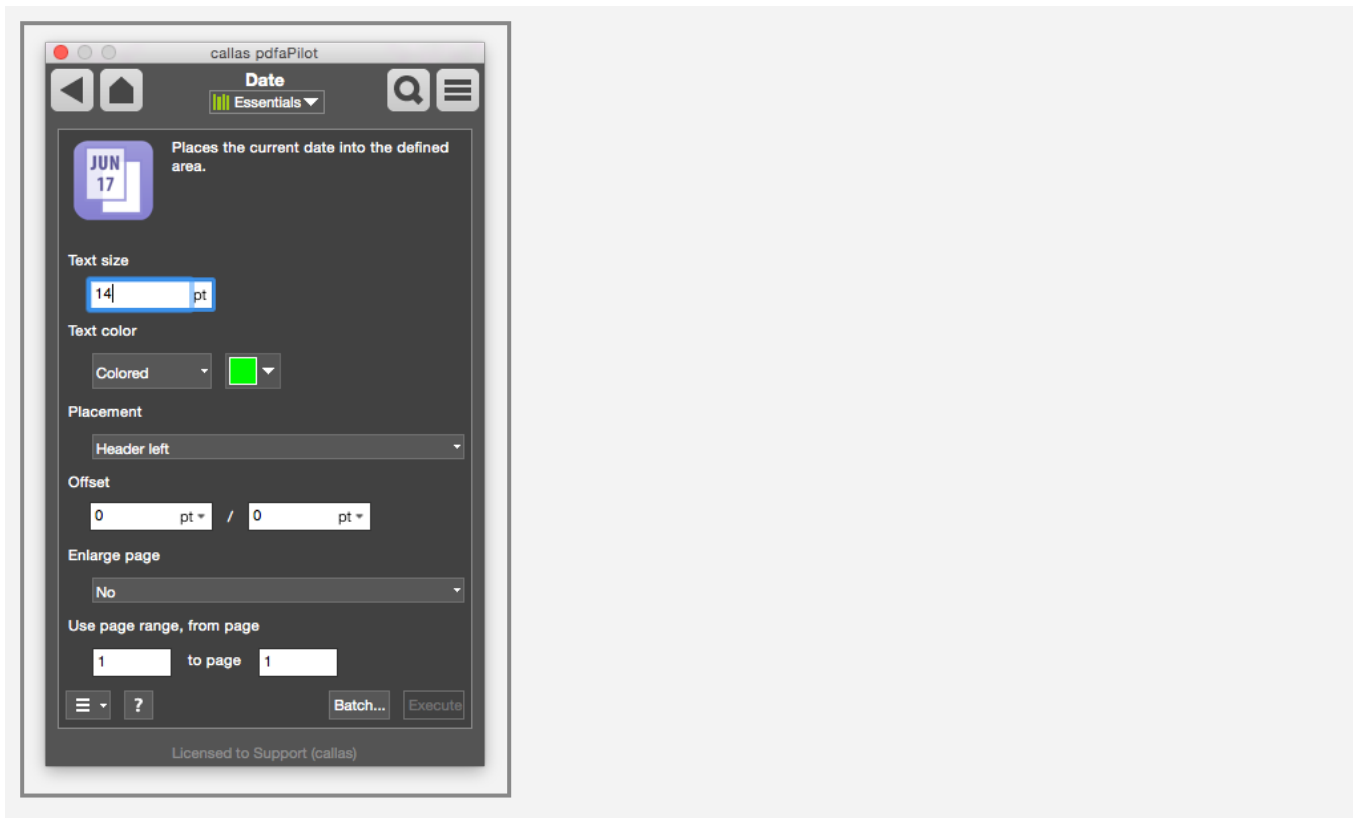
3. 3. Page number



You can number pages.

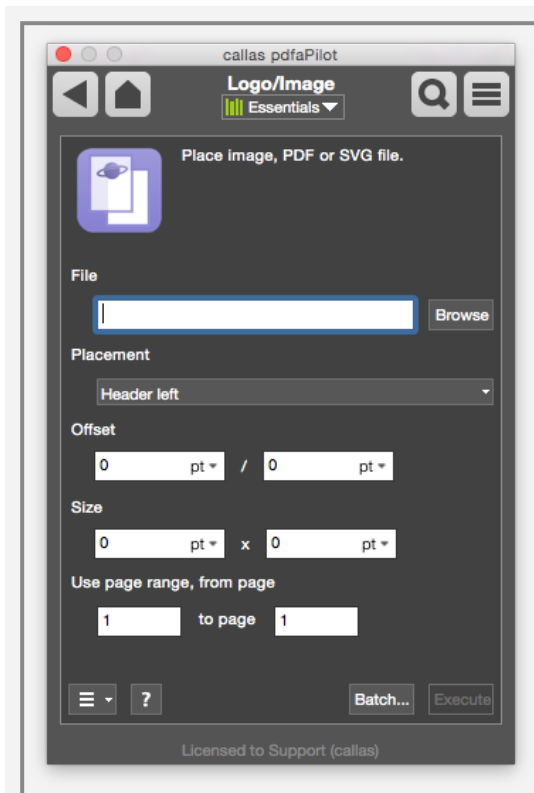
The text color, position and other settings can be specified.

4.4. Date



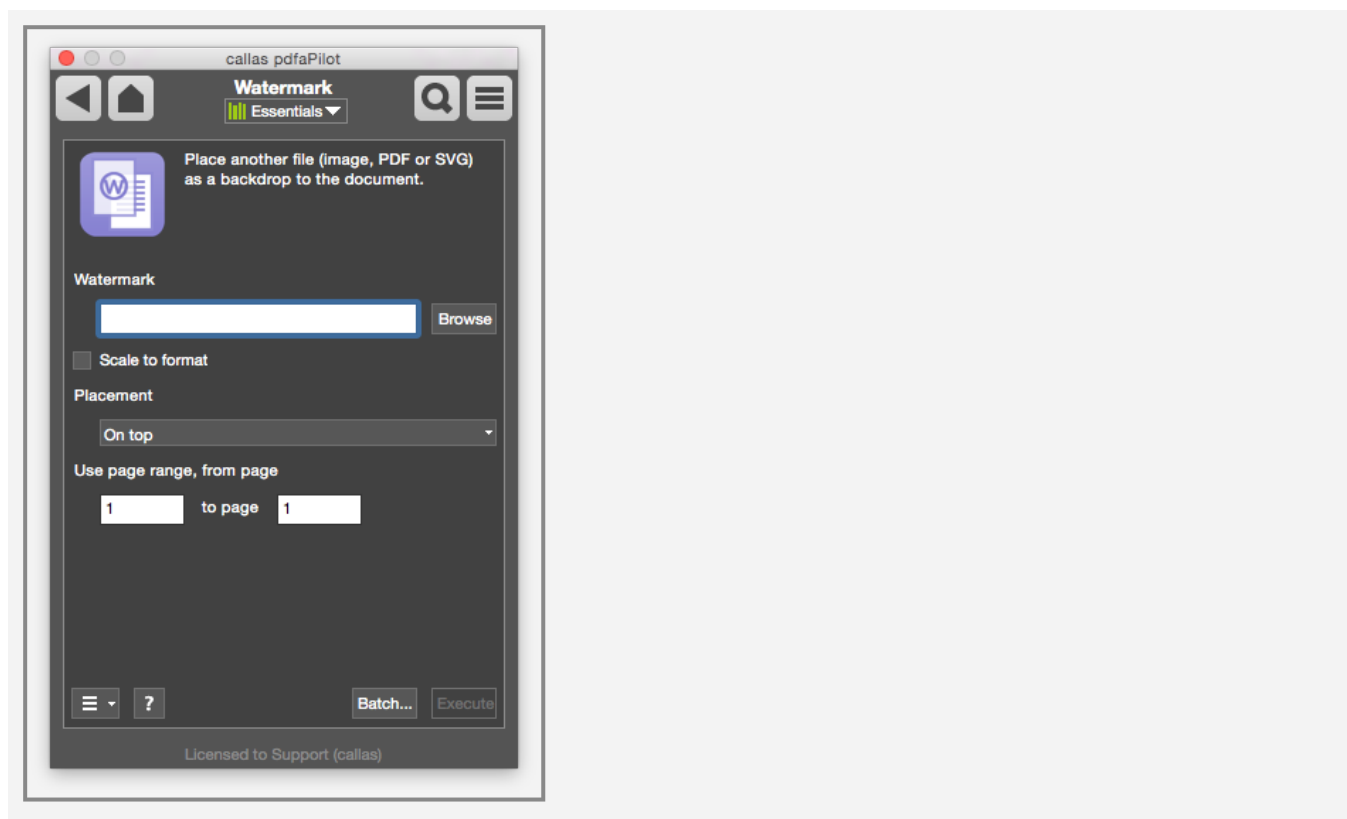
Places the current date on the page.
The text color, position and other settings can be specified.

5. 5. Logo/Image



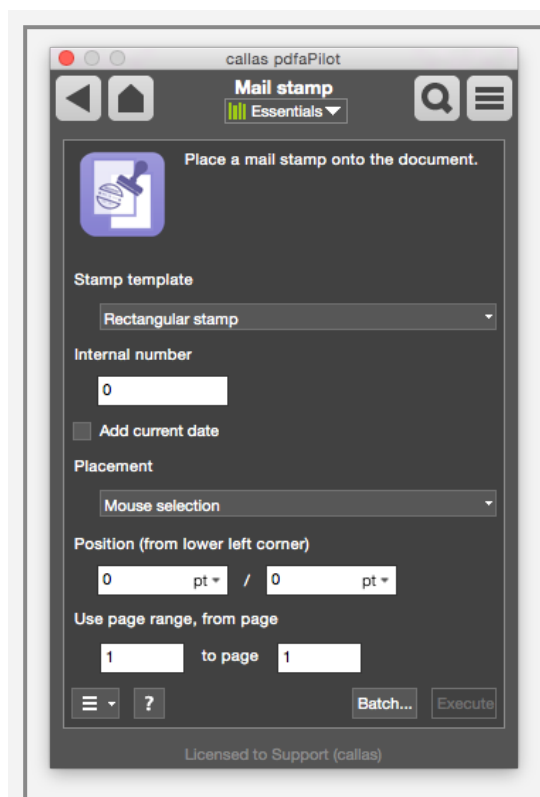
Places an image, PDF or SVG on pages.
You can specify the position as well as the page range.

6. 6. Watermark



Places an image, PDF or SVG with transparency.
This produces a see-through effect similar to a watermark.

7. 7. Mail stamp



You can add a mail stamp to the document.

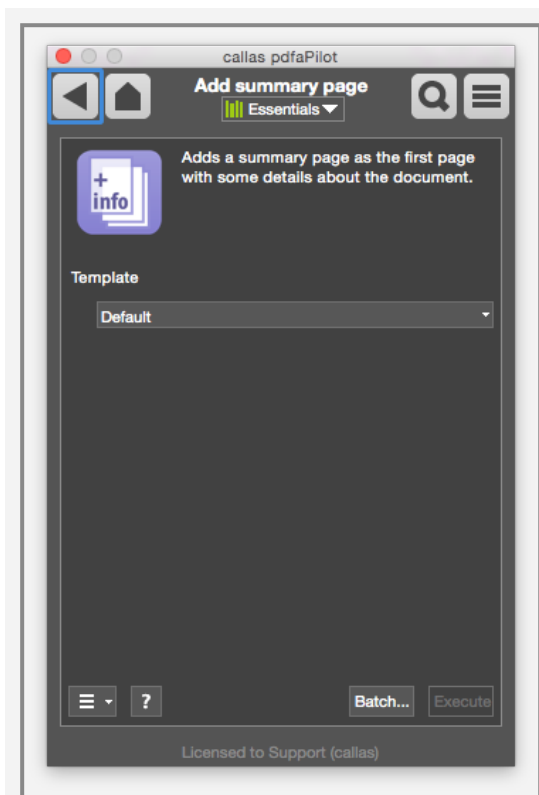
An HTML template can be used to customize the stamp's format.

8. 8. Letter background



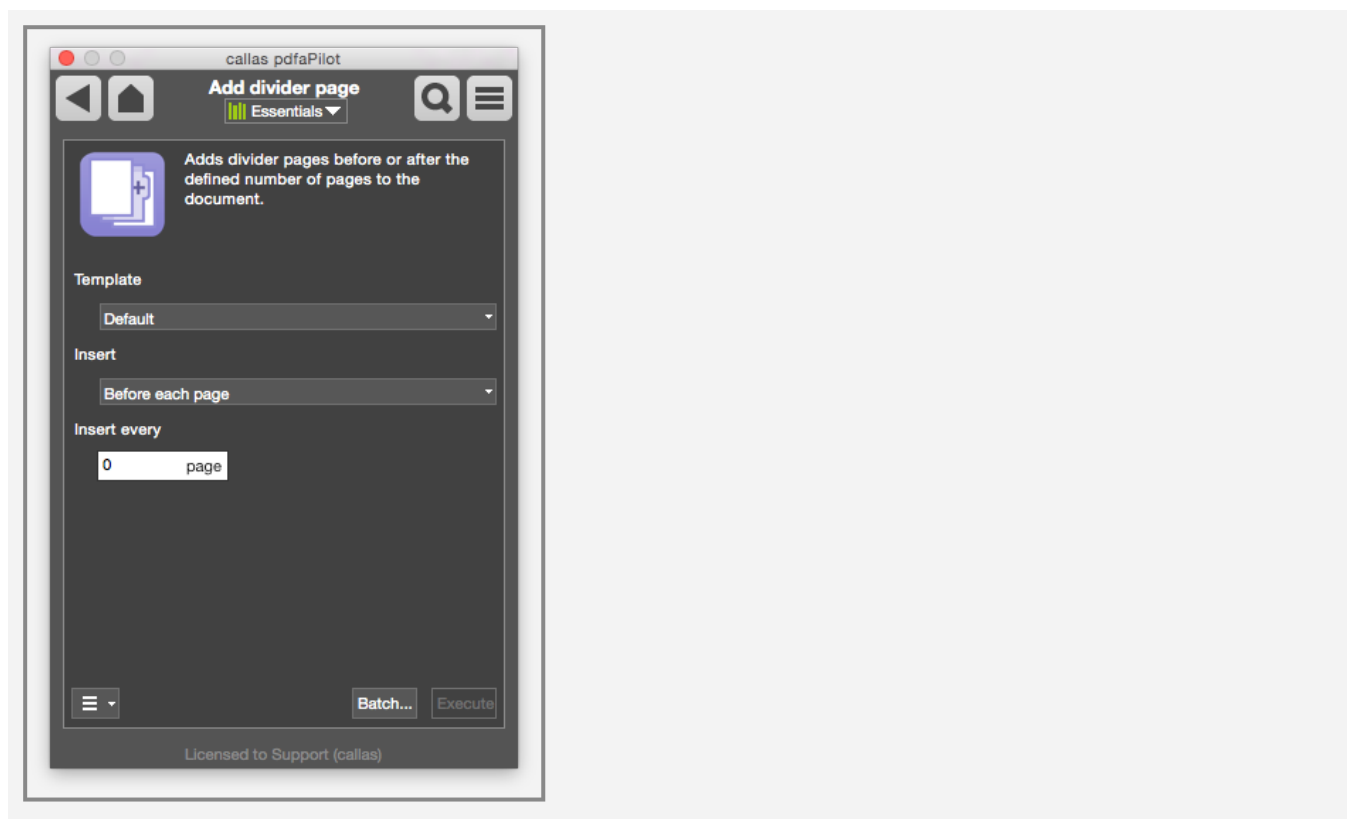
A letter background template can be placed on top of or below the content in the input file.

9.9. Add summary page



Adds a page to the document summarizing its contents.

10. 10. Add divider page



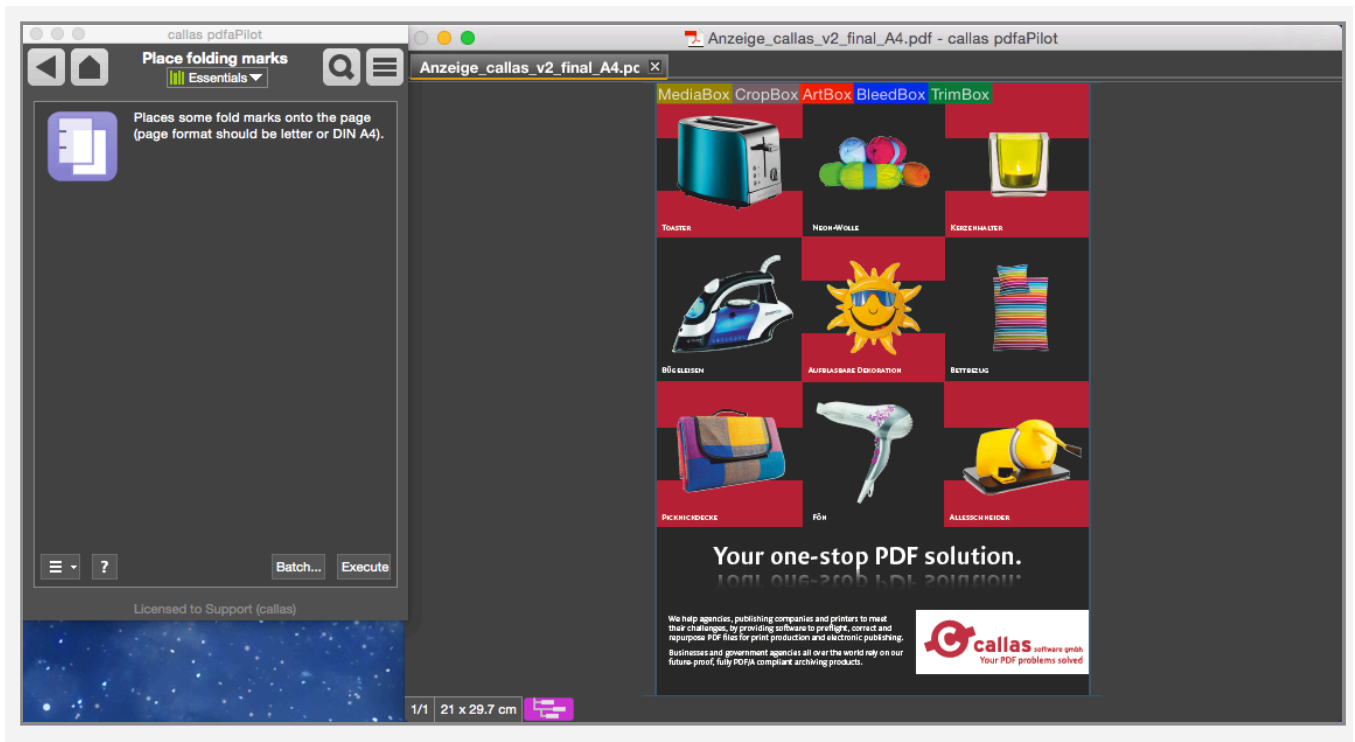
Divider pages can be added before or after a given page count.

11. 11. Move objects



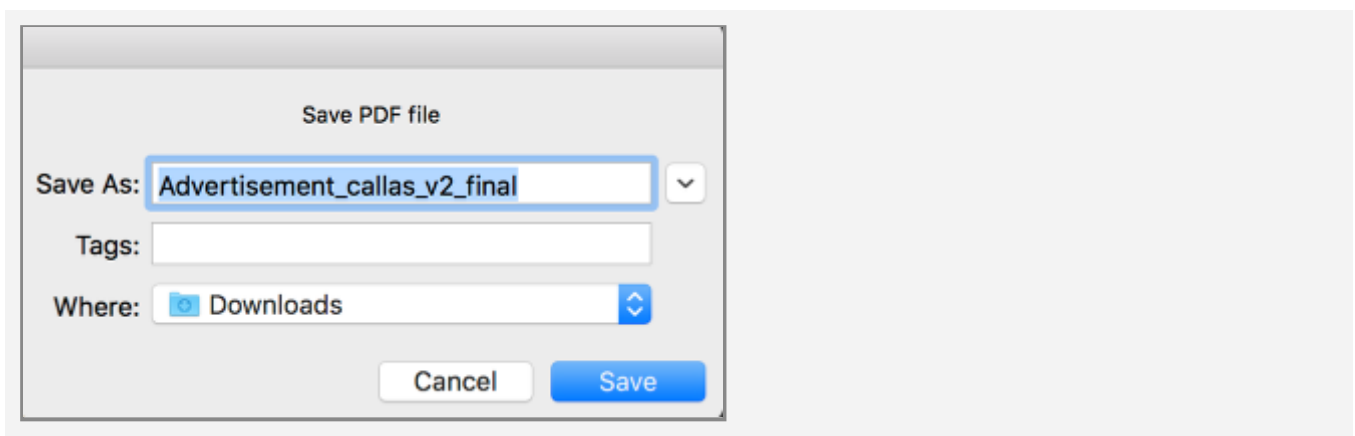
Objects of a given type can be moved to another position.

4.2 Place folding marks



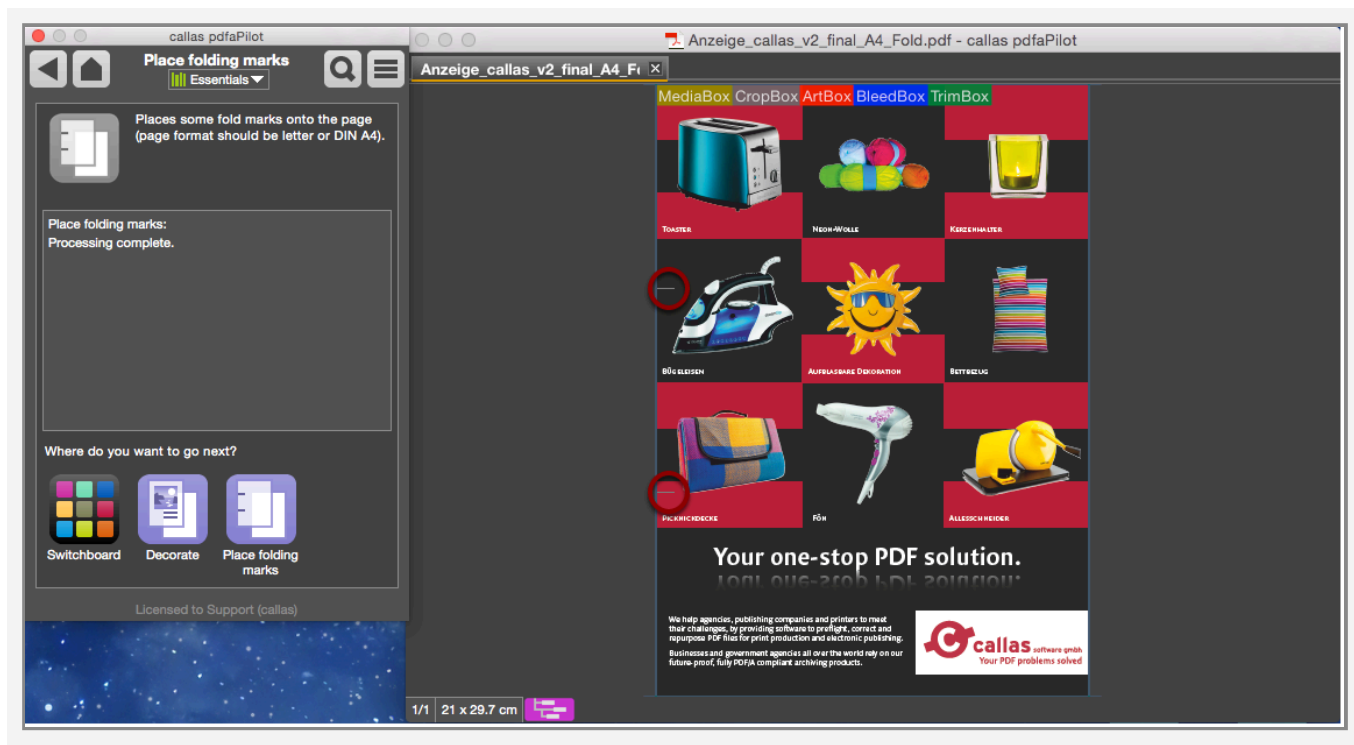
Call up the Switchboard Action and start the process by clicking “Execute”.

Specifying output destination



The “Save” dialog shown lets you specify where the results will be saved.

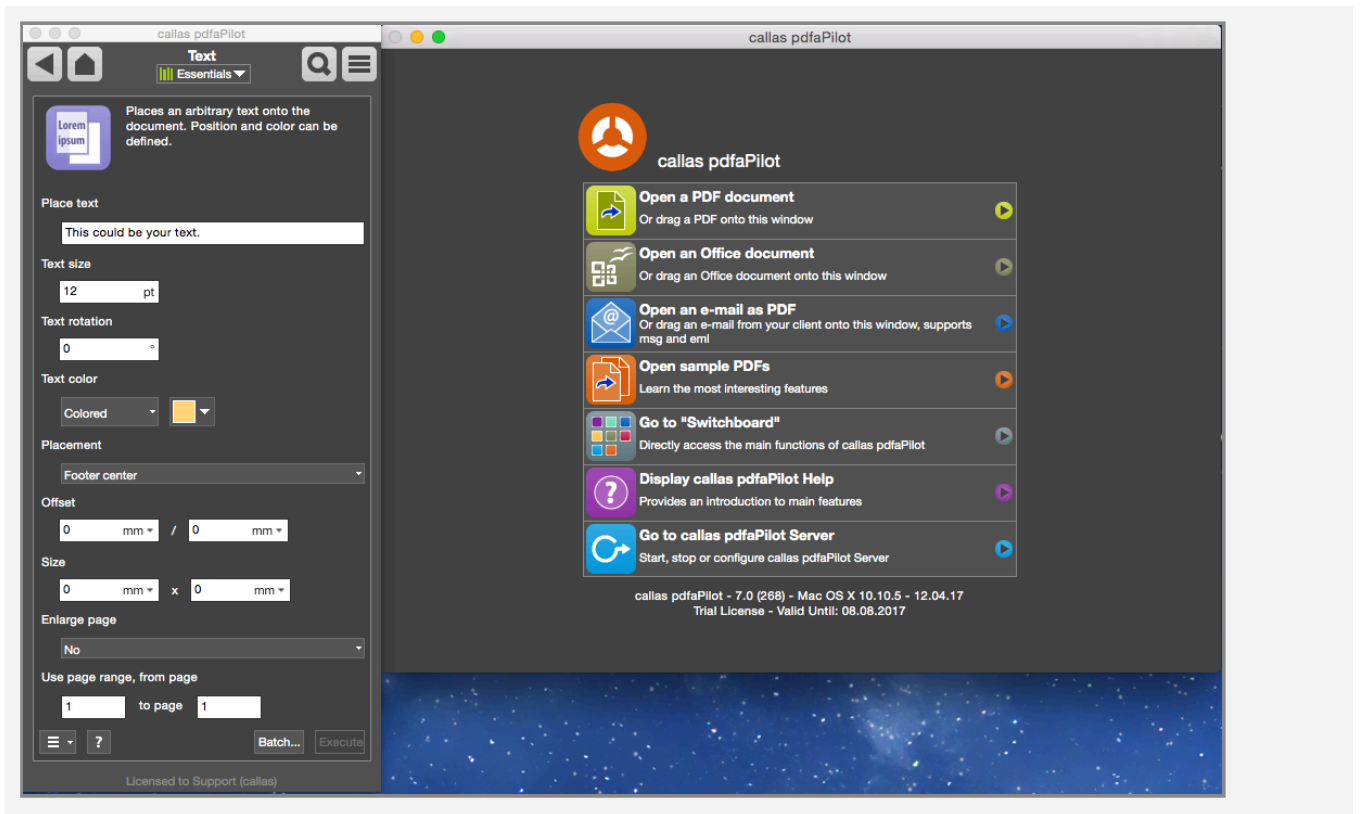
Results with folding marks placed



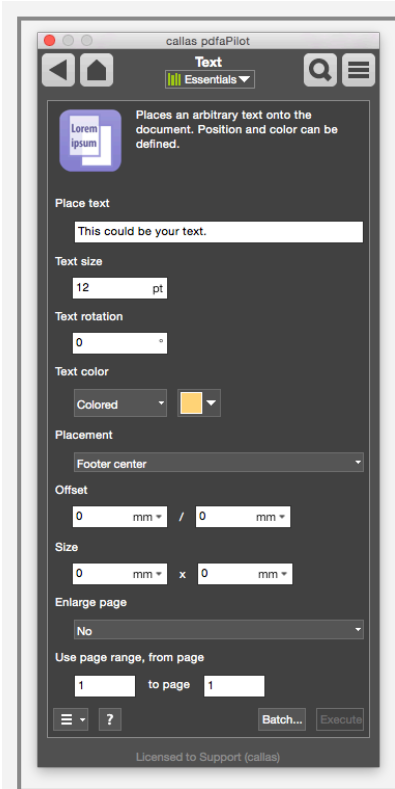
You will then see the folding marks overlaid on the document.

Here, two folding marks have been spaced out based on the height of the documents. This is ideal for DIN A4- and letter-format documents, among others.

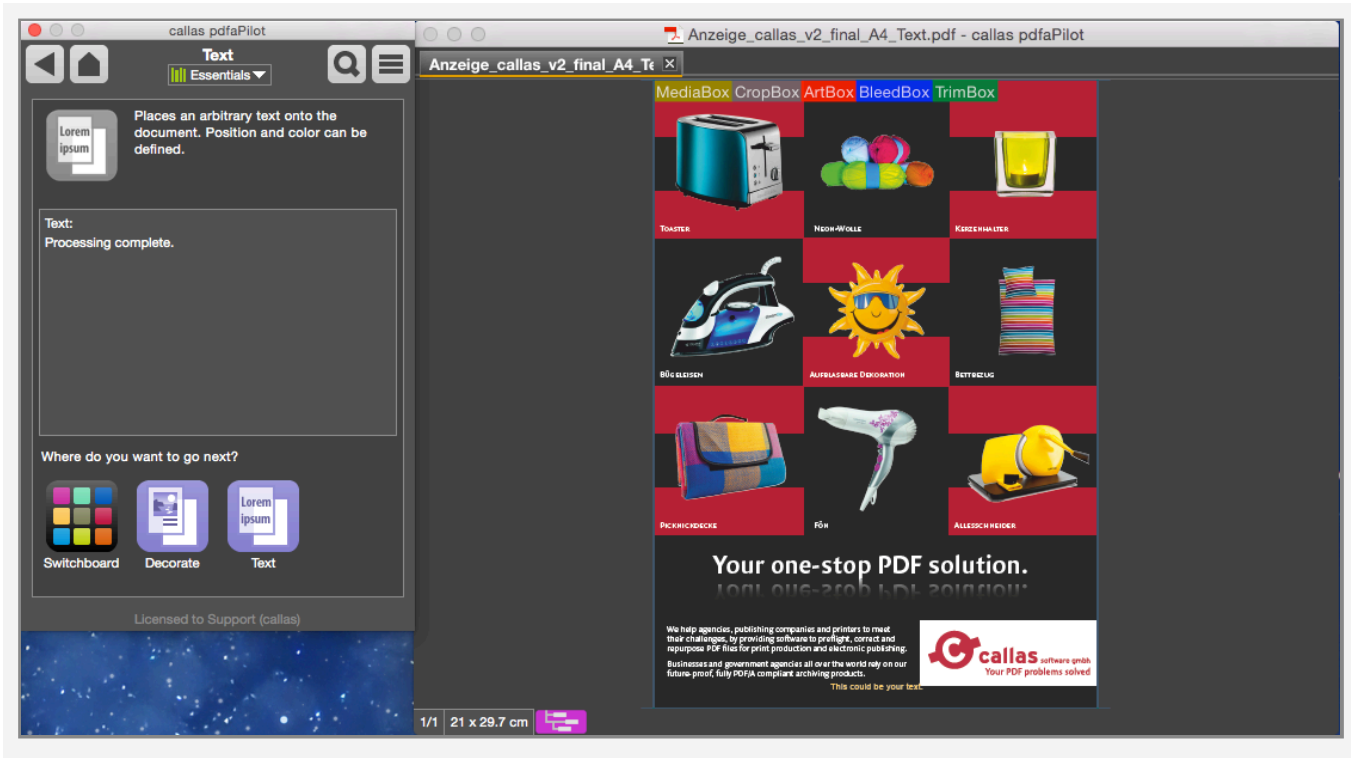
4.3 Add text



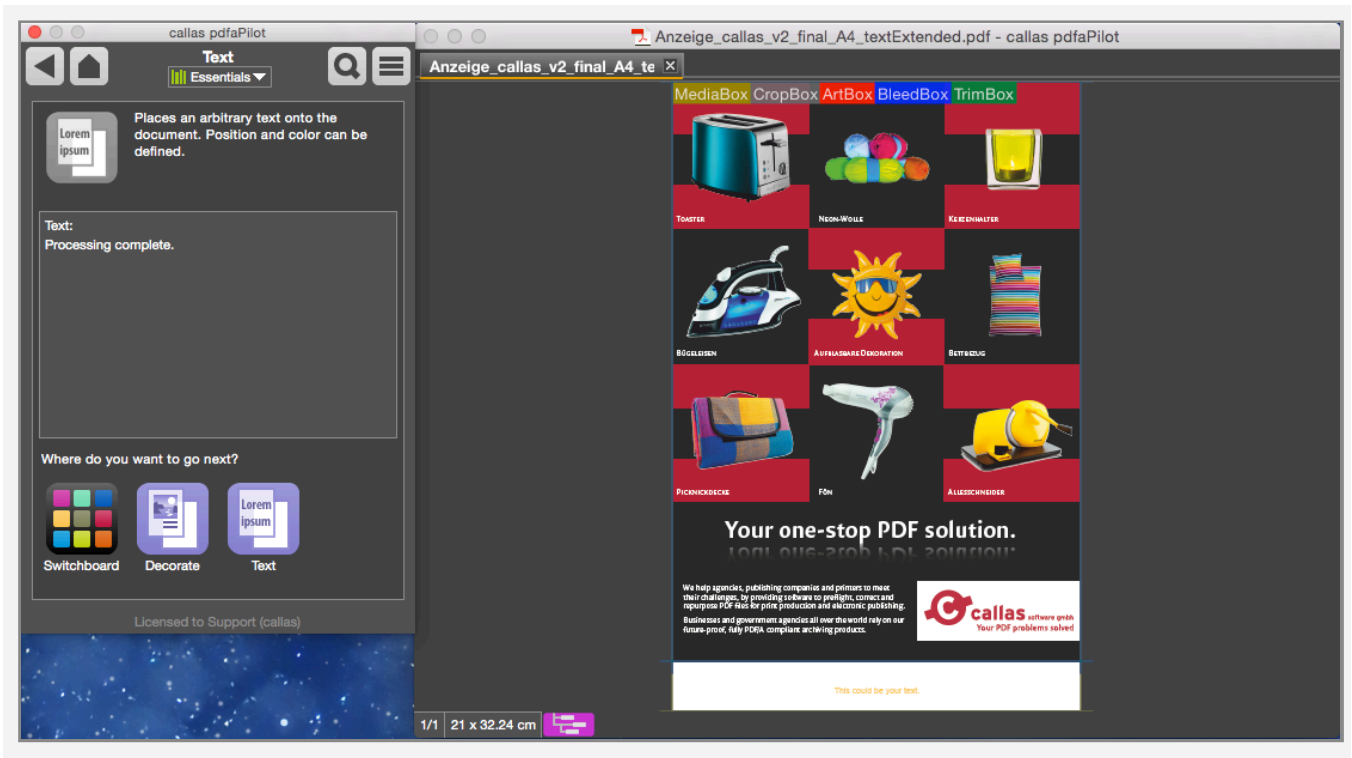
The “Text” Action lets you place text on a page.



Place text	Enter the text to be placed.
Text size	Defines the size of the text.
Text rotation	Specifies the angle at which the text will be created
Text color	Use the color selector to specify the color of the text to be placed.
Placement	Position where the object is to be placed.
Offset	Offsets the object relative to the parameters given under “Placement”. Values should be entered in the form of coordinates, giving x and y values relative to the reference point given under “Placement”. This means that value such as “-5” is also valid.
Size	Specifies the size of the text field
Enlarge page	If the text does not appear on the page, the page region can be enlarged.
Page range	For multi-page documents, specifies the range of pages on which the text will be placed.

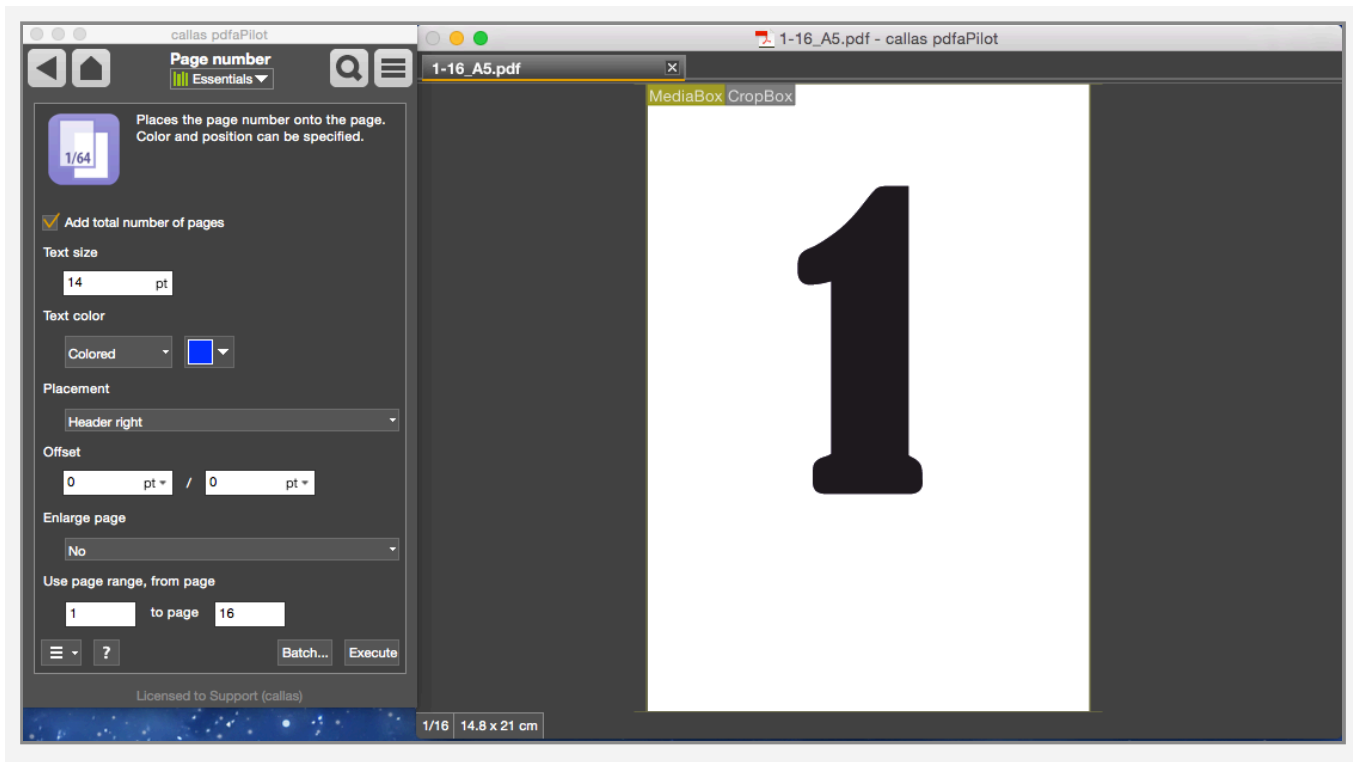


The result will then be shown in the visualizer. (Here it is the yellow text at the bottom of the page.)



This is the result with “Enlarge page” enabled, causing the page to become slightly larger.

4.4 Place page number

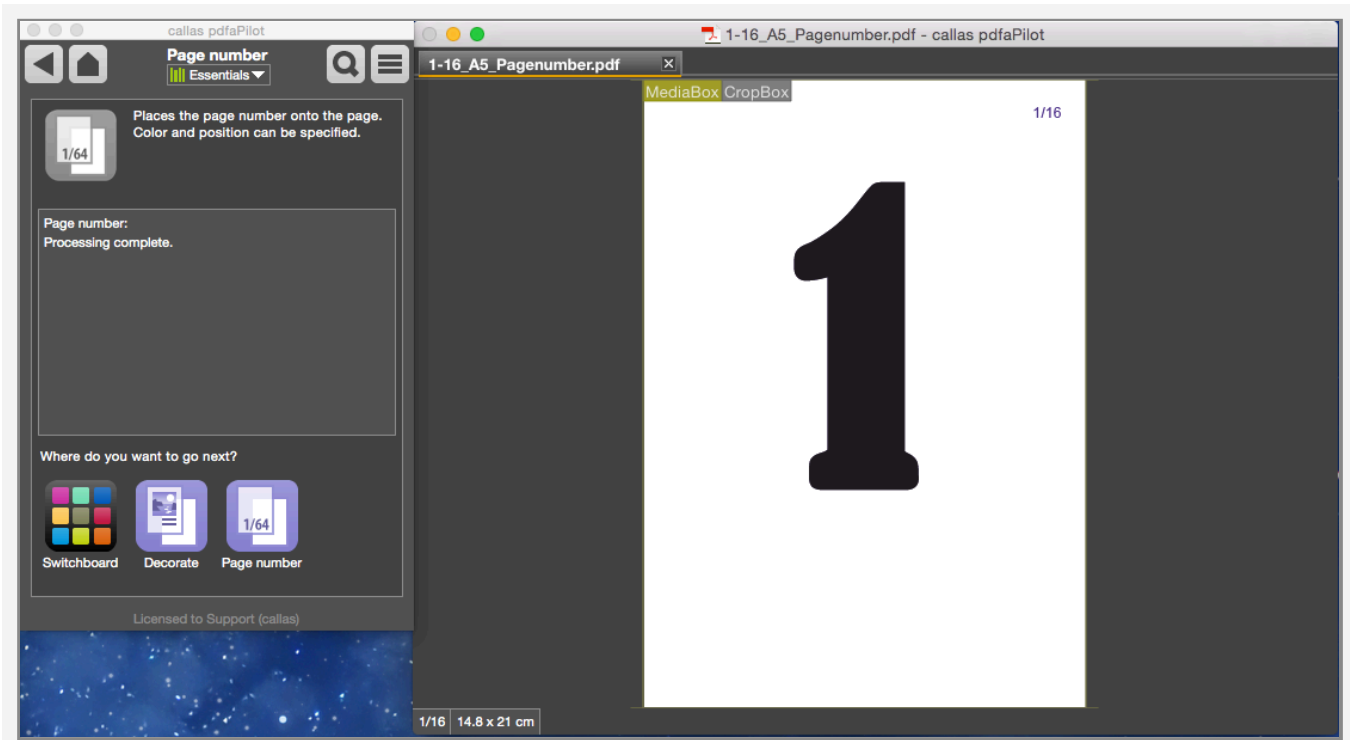


Add total number of pages	Uses the current page number as well as the total number of pages, using the format “<Page number>/<Total pages>”
Text size	Defines the size of the text.
Text color	Use the color selector to specify the color of the text to be placed.
Placement	Coordinates where the object is to be placed.
Offset	Offsets the object relative to the parameters given under “Placement”. Values should be entered in the form of coordinates, giving x and y values relative to the reference point given under “Placement”. This means that values such as “-5” are also valid.
Enlarge page	If the text does not appear within the page region, the page can be enlarged.
Page range	For multi-page documents, specifies the range of pages on which the text will be placed.

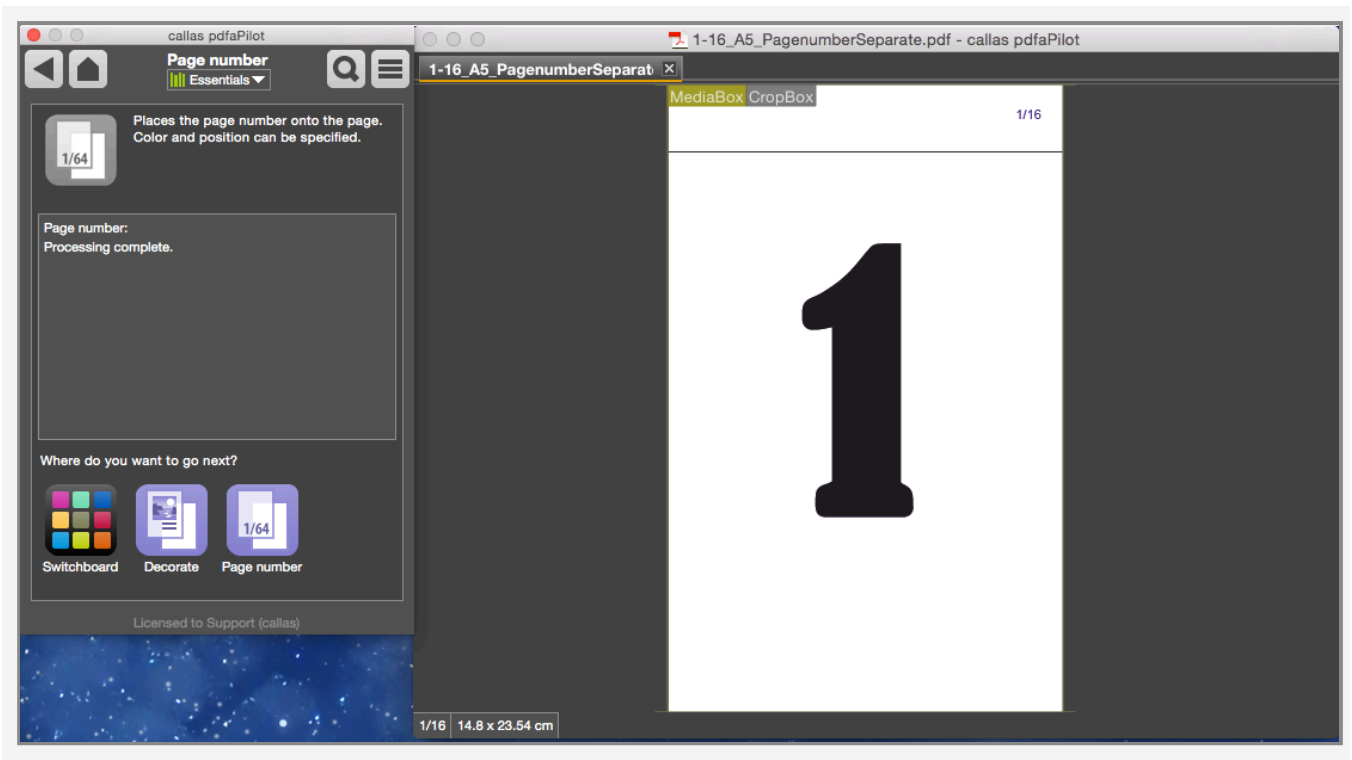
For usability reasons, the Switchboard Action has a limited range of functionality.

If you wish to customize additional page numbering parame-

ters, you can also use the “Place page number” fixup (see below for further details).

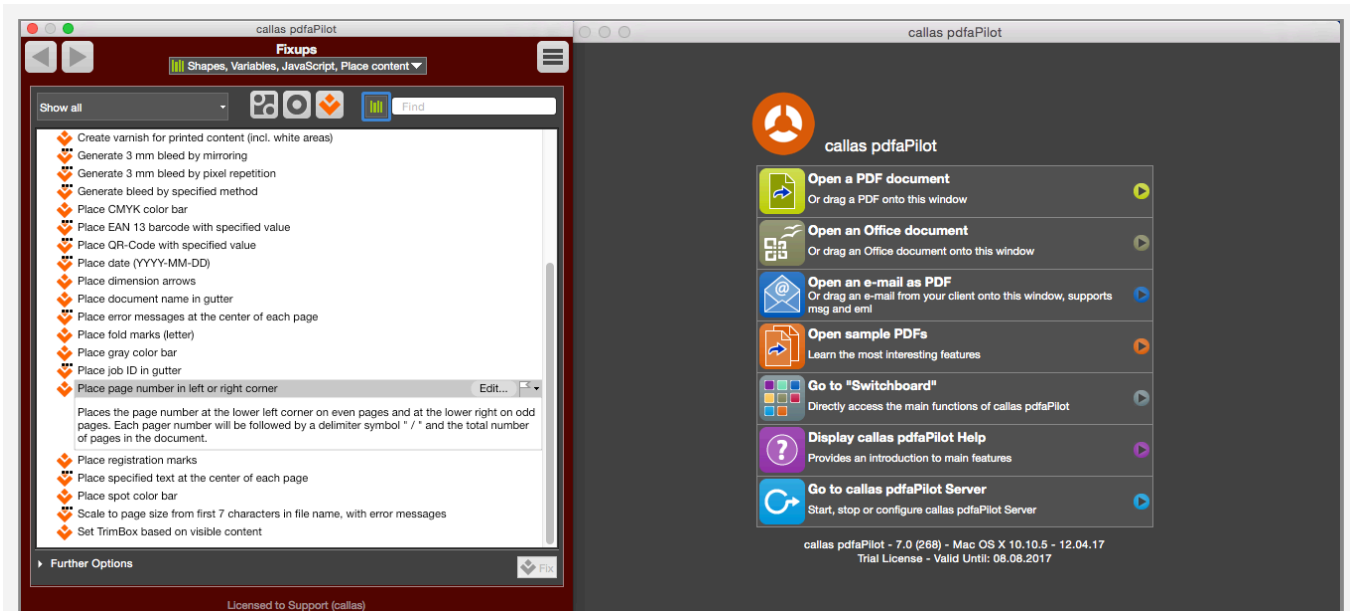


Result without using “Enlarge page” option: The page number is placed at the top right.

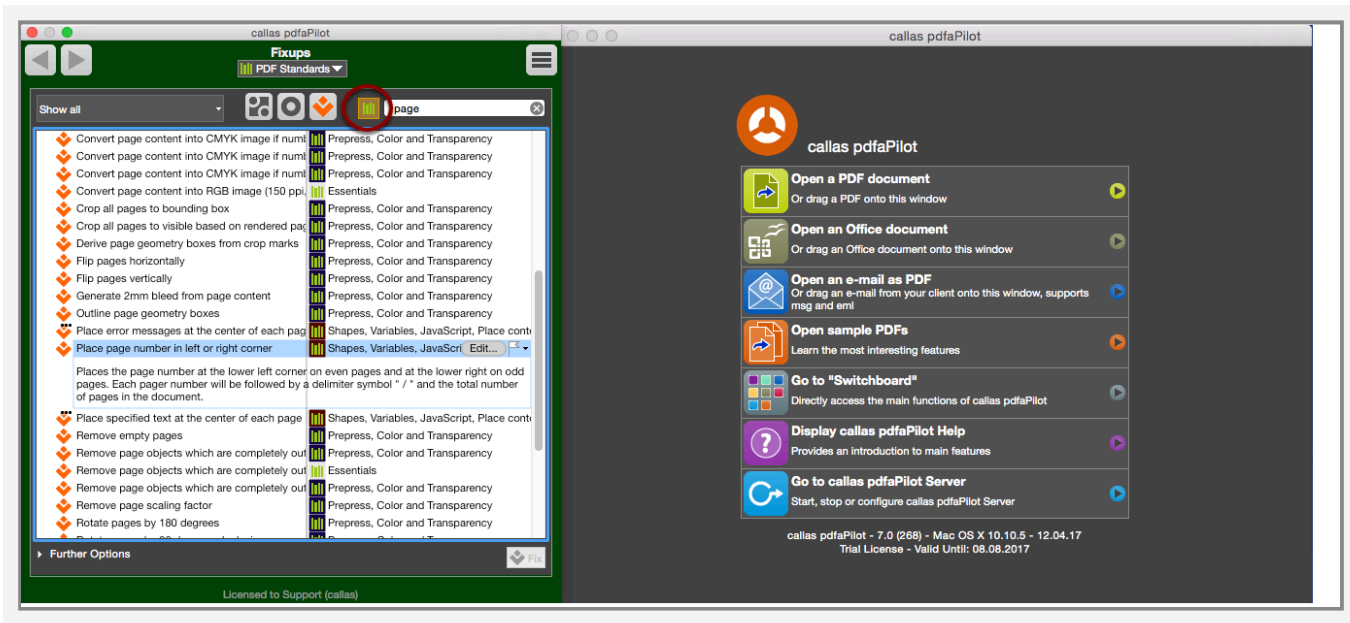


Result of “Enlarge page and indicate original size” This will draw an additional line allowing you to identify the original page size.

Using the “Place page number” Fixup

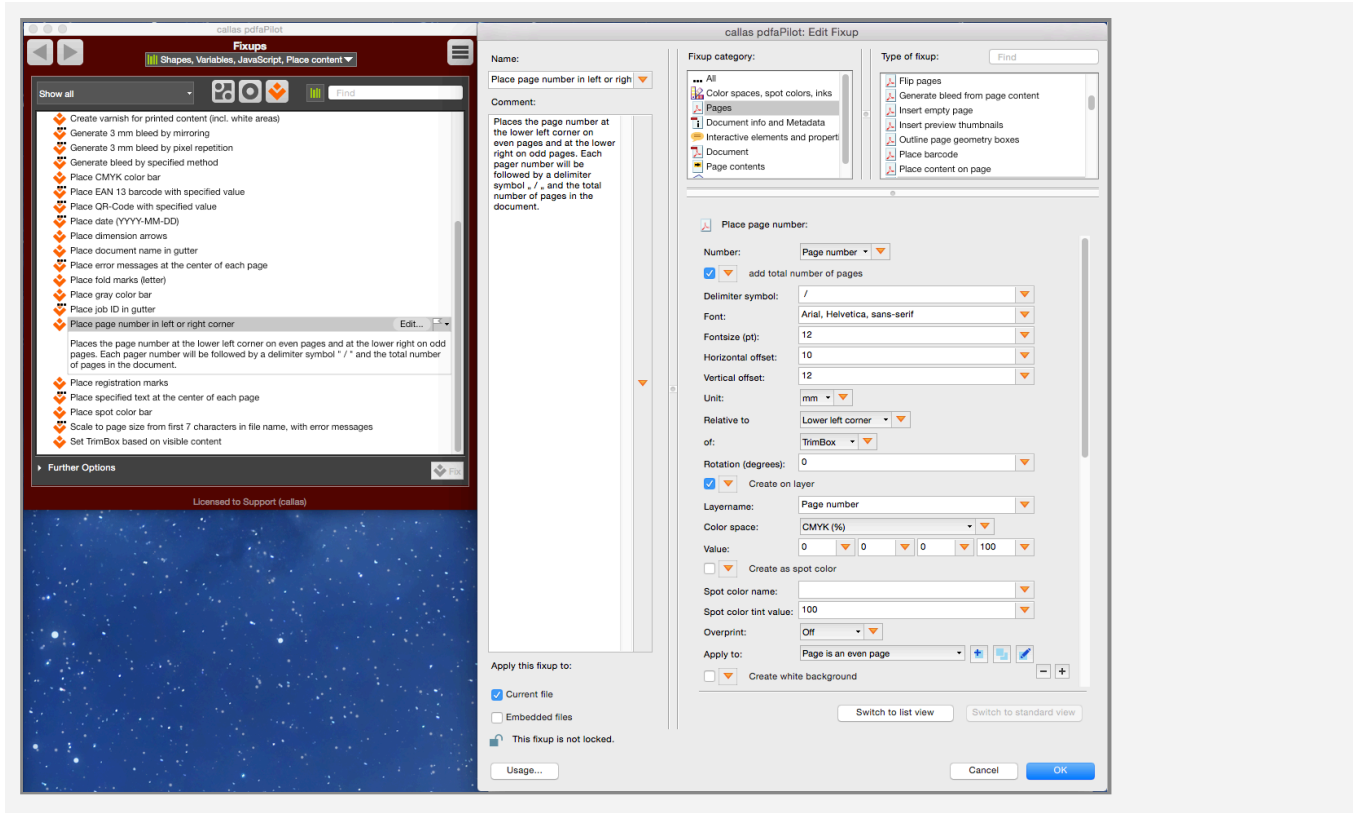


First, call up the Fixup window under Tools - Fixups.

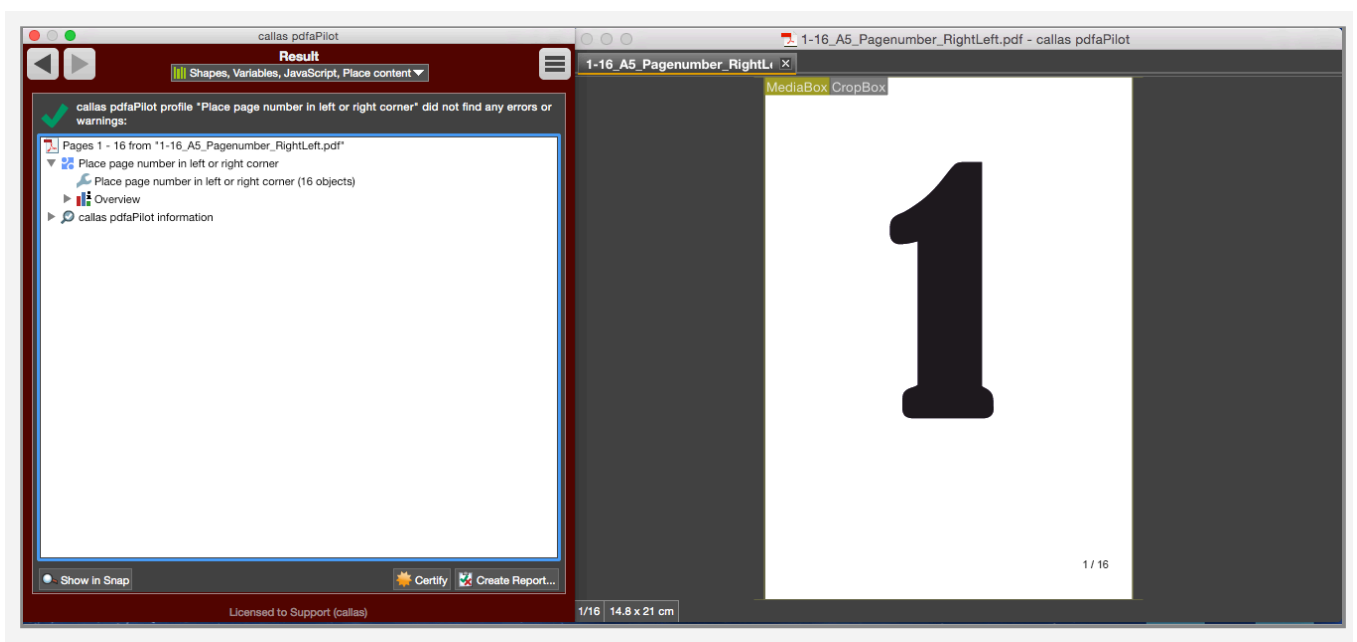


A predefined Fixup using “Place page number” can be found under the “Shapes, Variables, JavaScript, Place content” library.

If another library is currently selected, you can also search all available libraries using the green icon. The predefined Fixup is named “Place page number in left or right corner”.

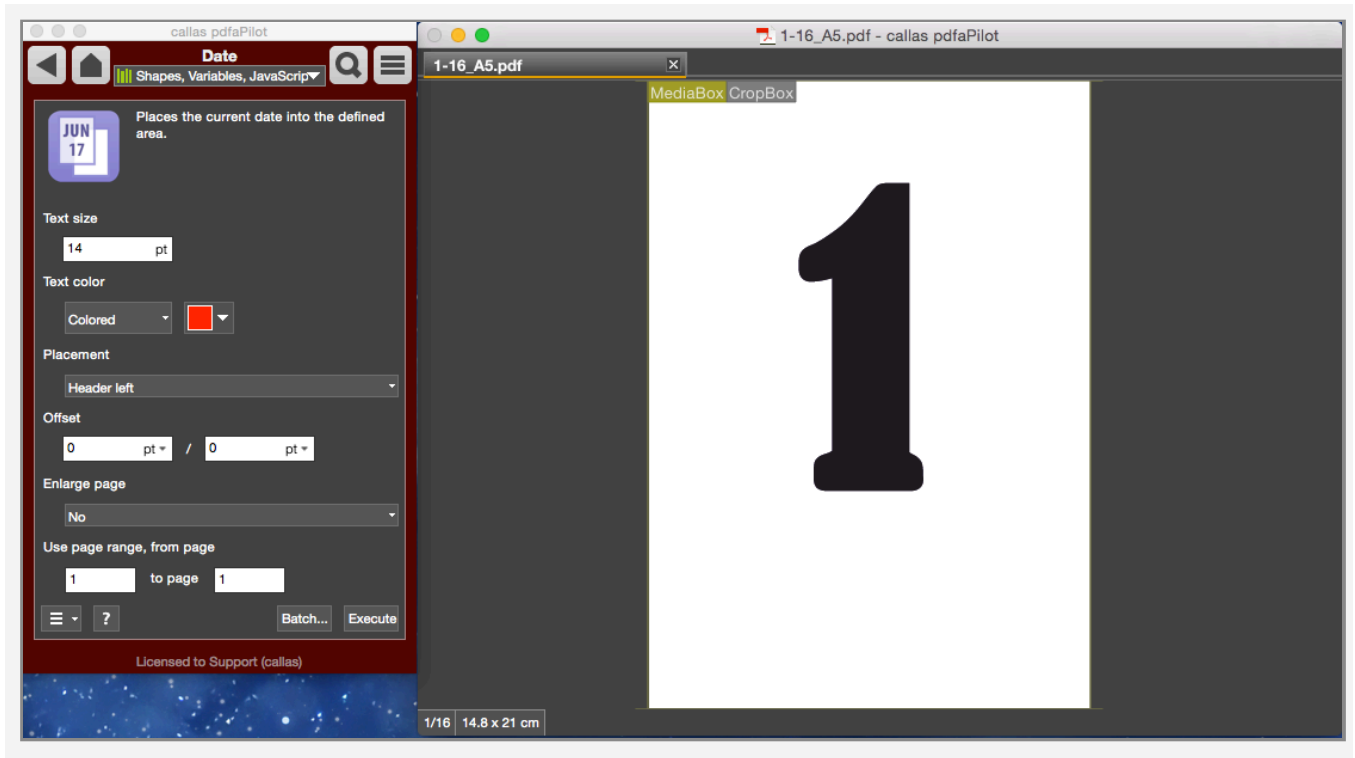


The Fixup lets you create the object on a layer or as a spot color. It also lets you create a white background.

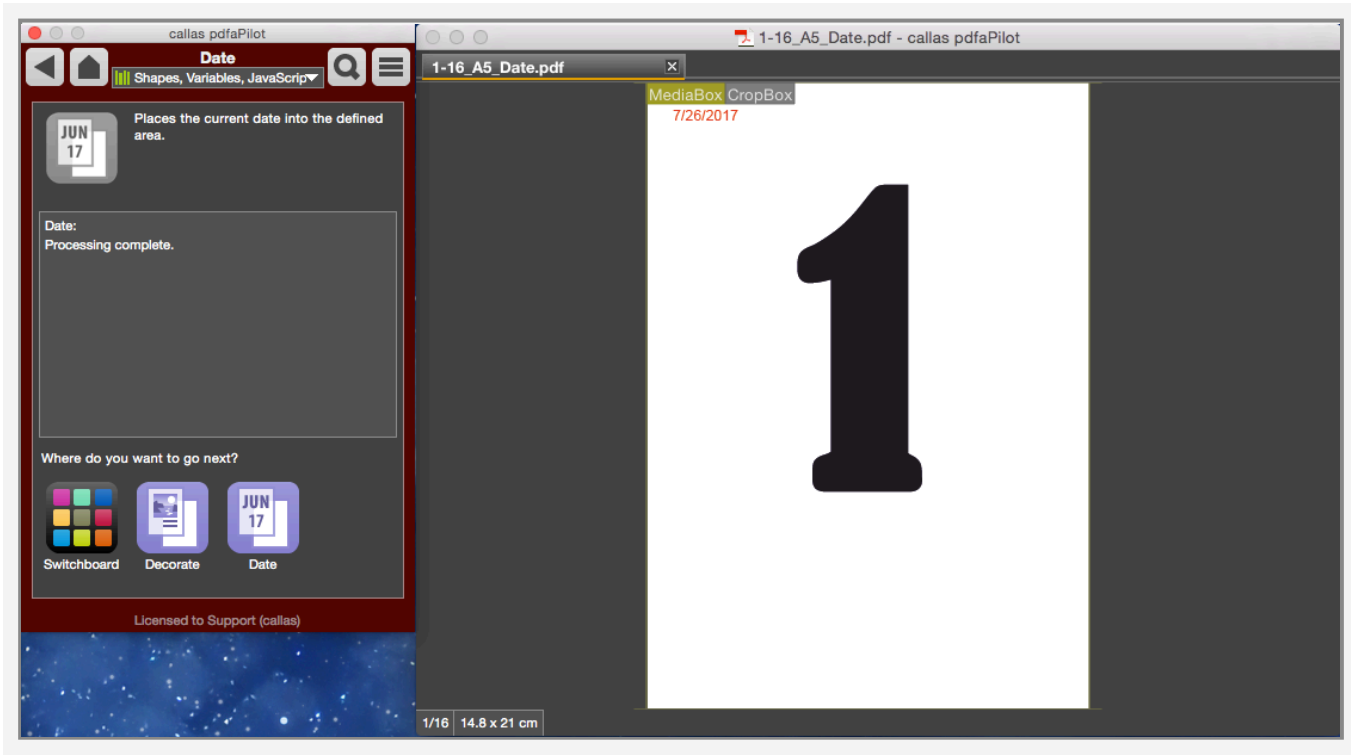


4.5 Place date

This Action allows you to place the current date in a position of your choice.

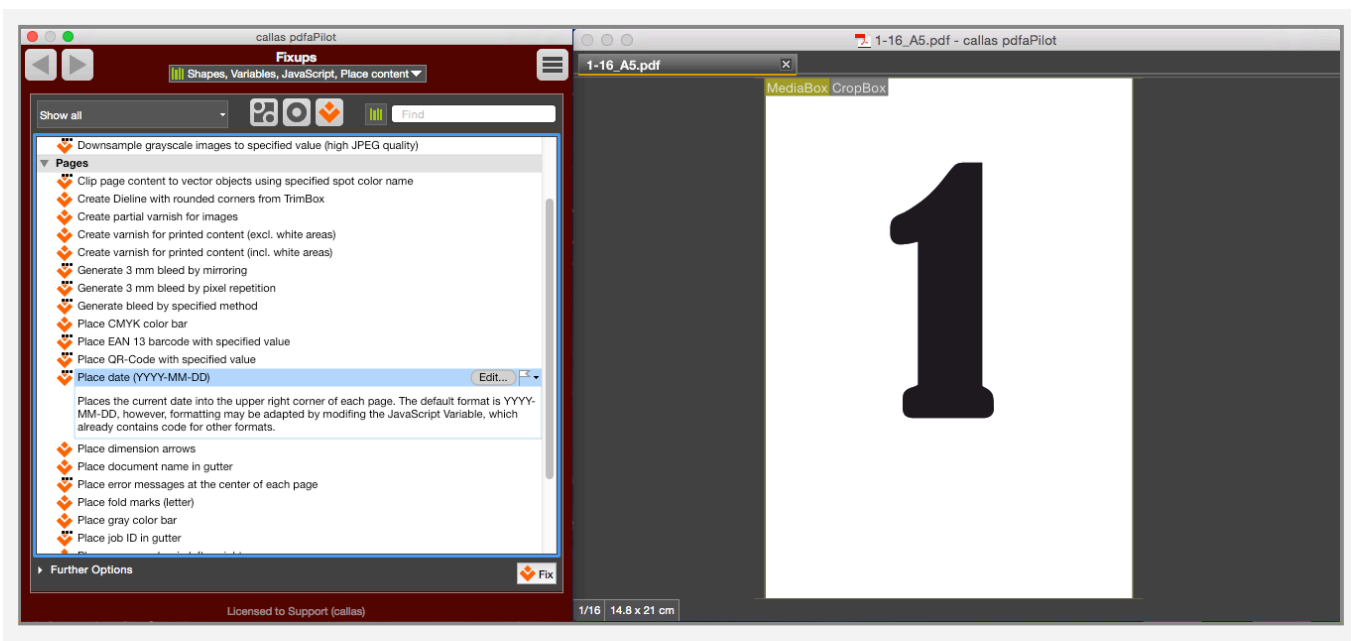


Text size	Defines the size of the text.
Text color	Use the color selector to specify the color of the text to be placed.
Placement	Coordinates where the object is to be placed.
Offset	Offsets the object relative to the parameters given under “Placement”. Values should be entered in the form of coordinates, giving x and y values relative to the reference point given under “Placement”. This means that values such as “-5” are also valid.
Enlarge page	If the date does not appear within the page region, the page can be enlarged.
Page range	Specifies the range of pages on which the date will be placed.

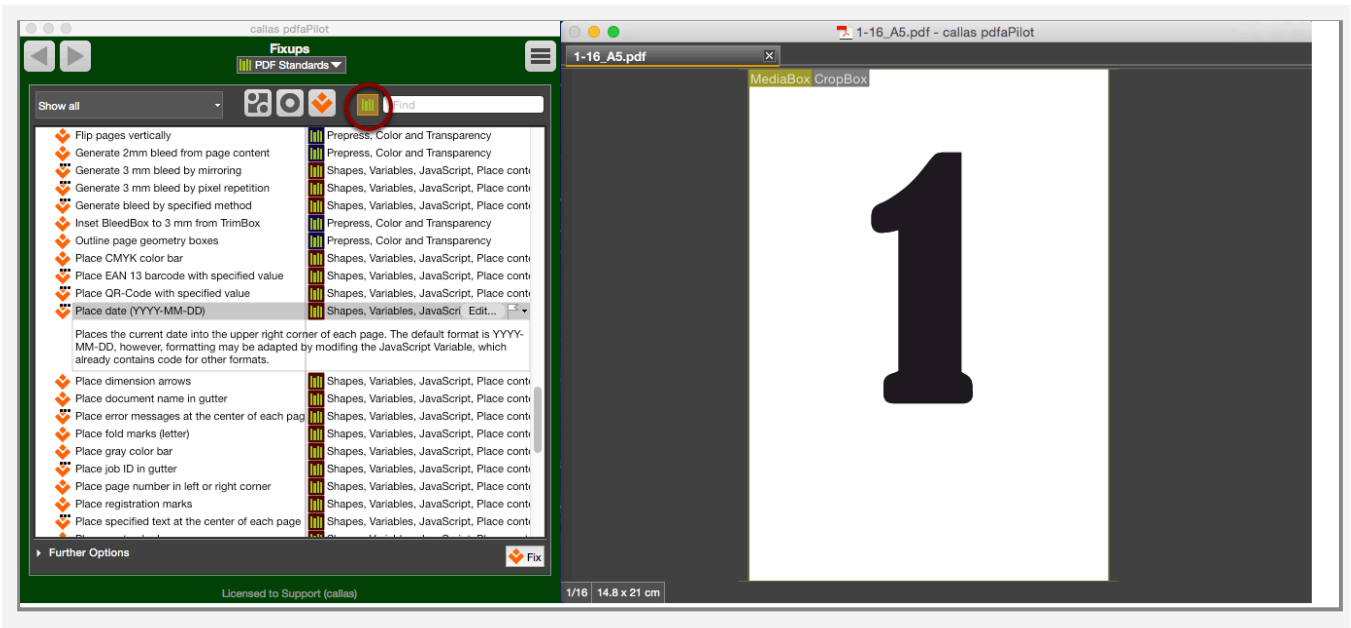


For usability reasons, the Switchboard Action has a strictly limited range of functionality. If you wish to fine-tune the date format, you can also use the applicable Fixup.

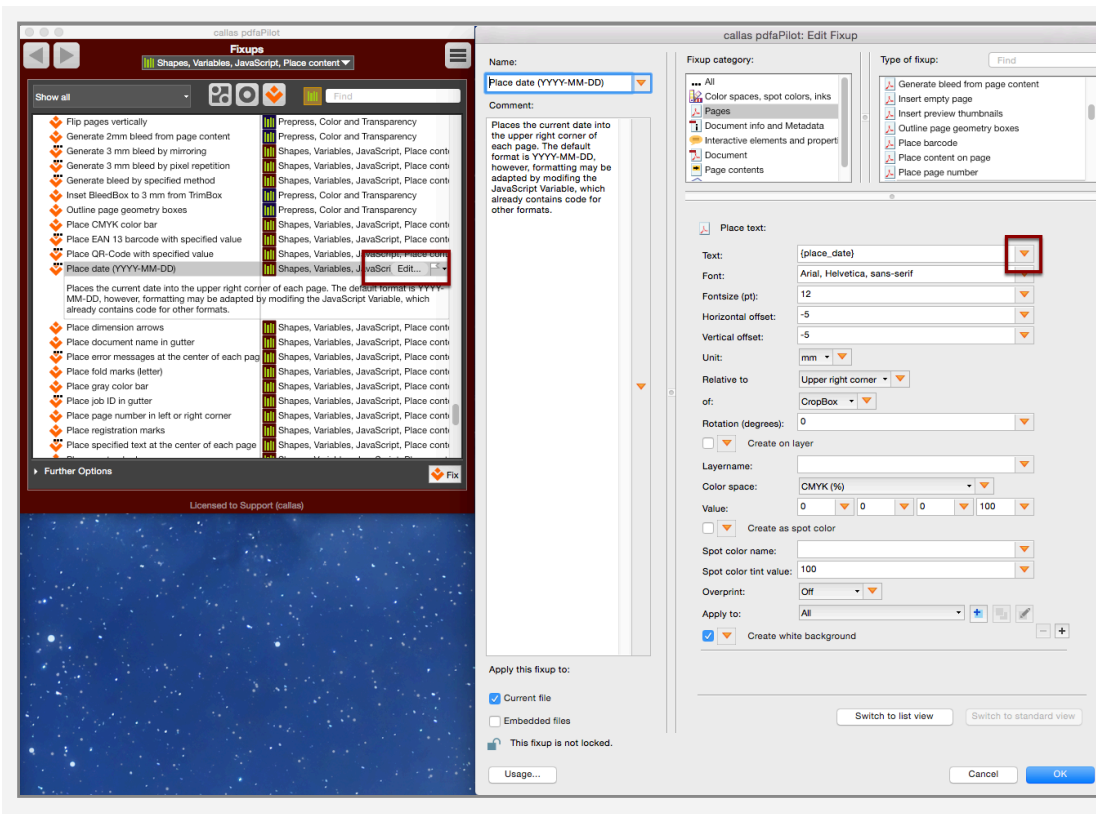
Using the “Place date” Fixup



First, call up the Fixup window under Tools - Fixups.

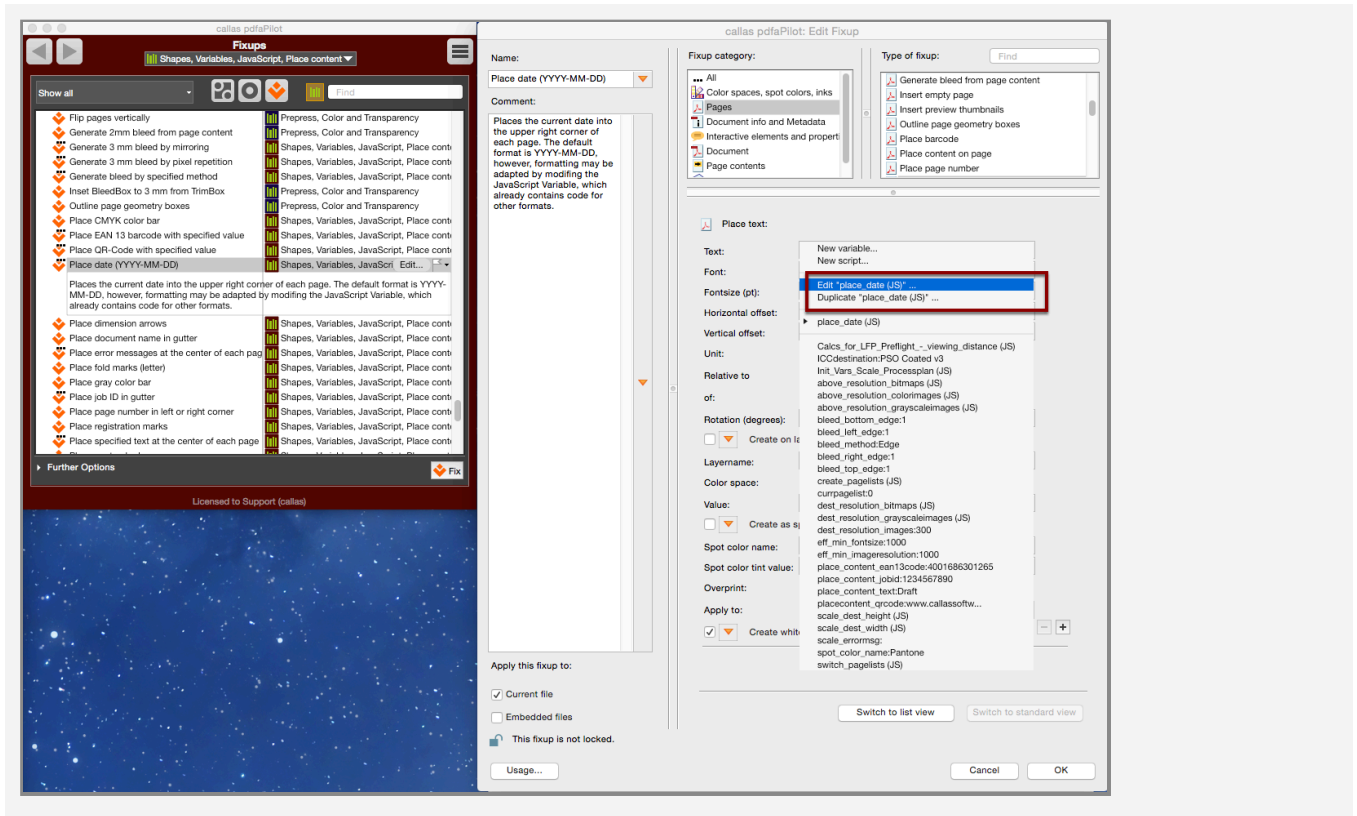


The “Place date” Fixup can be found under the “Shapes, Variables, JavaScript, Place content” library. If another library is currently selected, you can also search all available libraries using the green icon.

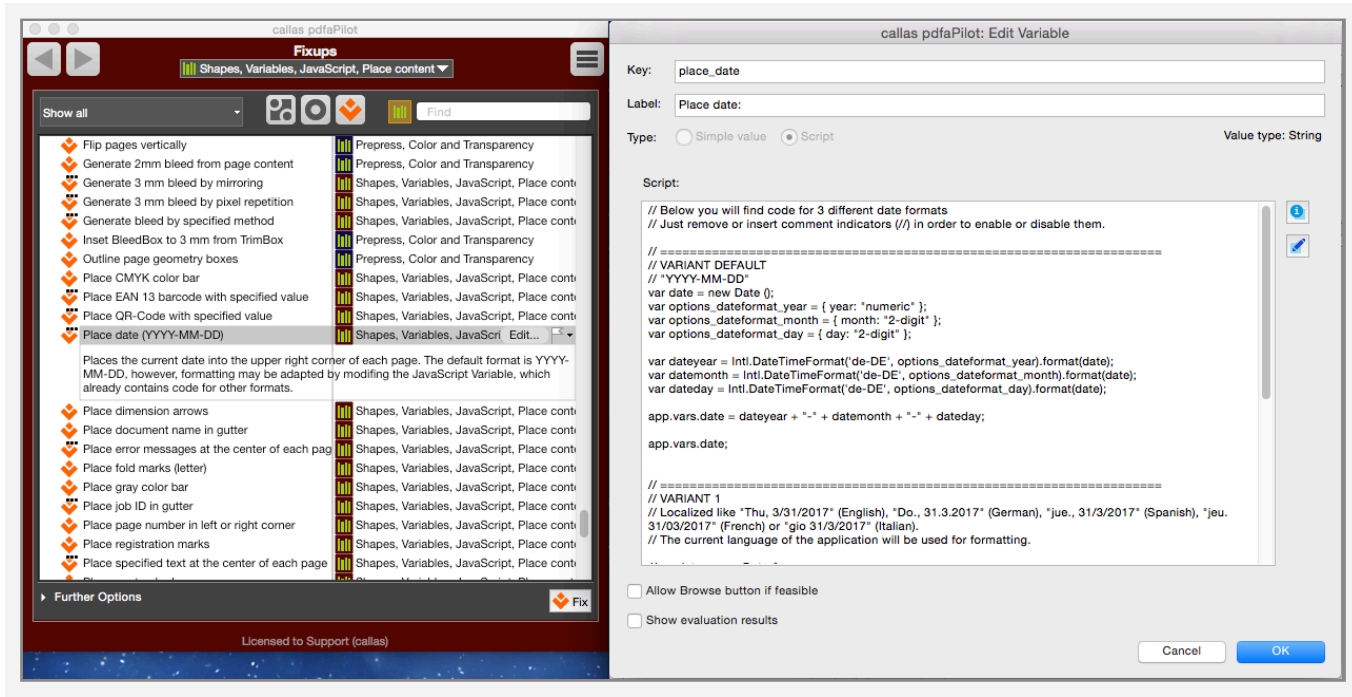


Click “Edit” to view and alter the parameters used for the Fixup.

It is advisable to create a copy of this Fixup to make changes of your own. You can do this easily by selecting “Duplicate Fixup” from the menu button in the top right.



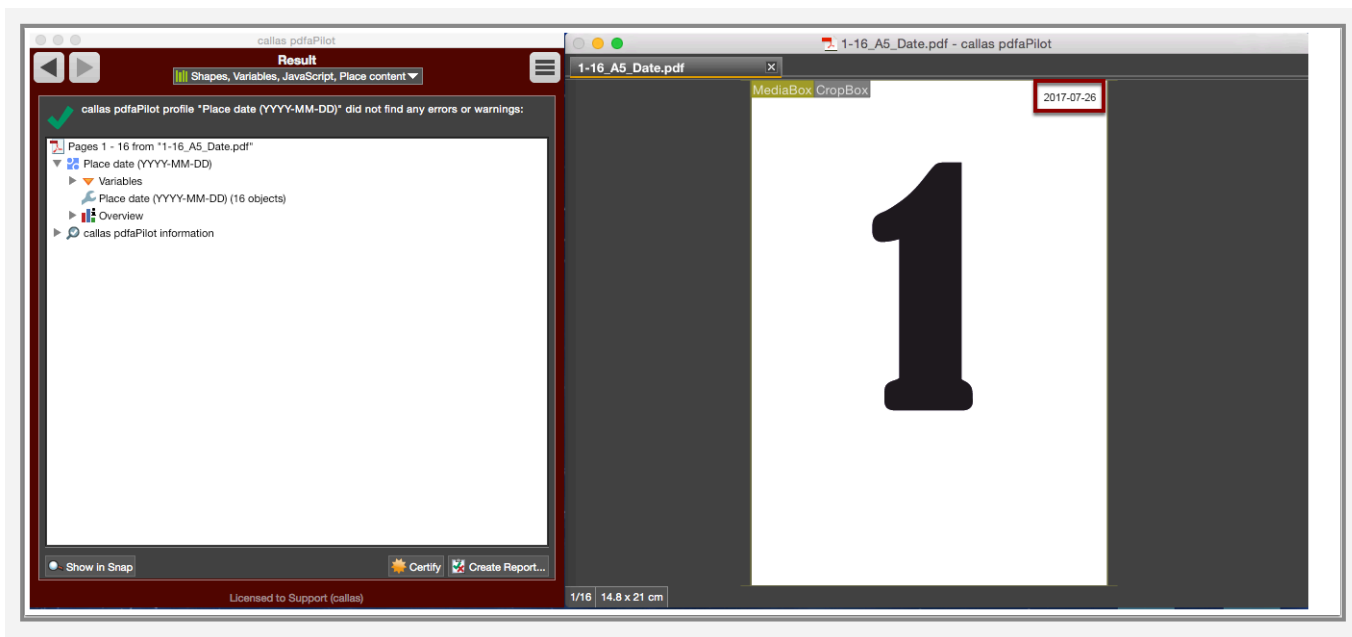
JavaScript is used to provide the date text. You can edit the version given under “place_date (JS)”. Alternatively, you can simply duplicate this variable and work with a copy of it.



The JavaScript can be edited as required within the text box.

Some options for different date formats are already included as examples. These are commented out by default and can be reactivated by deleting the “//” characters at the start of the line.

Unused JavaScript sections should then be disabled or deleted.



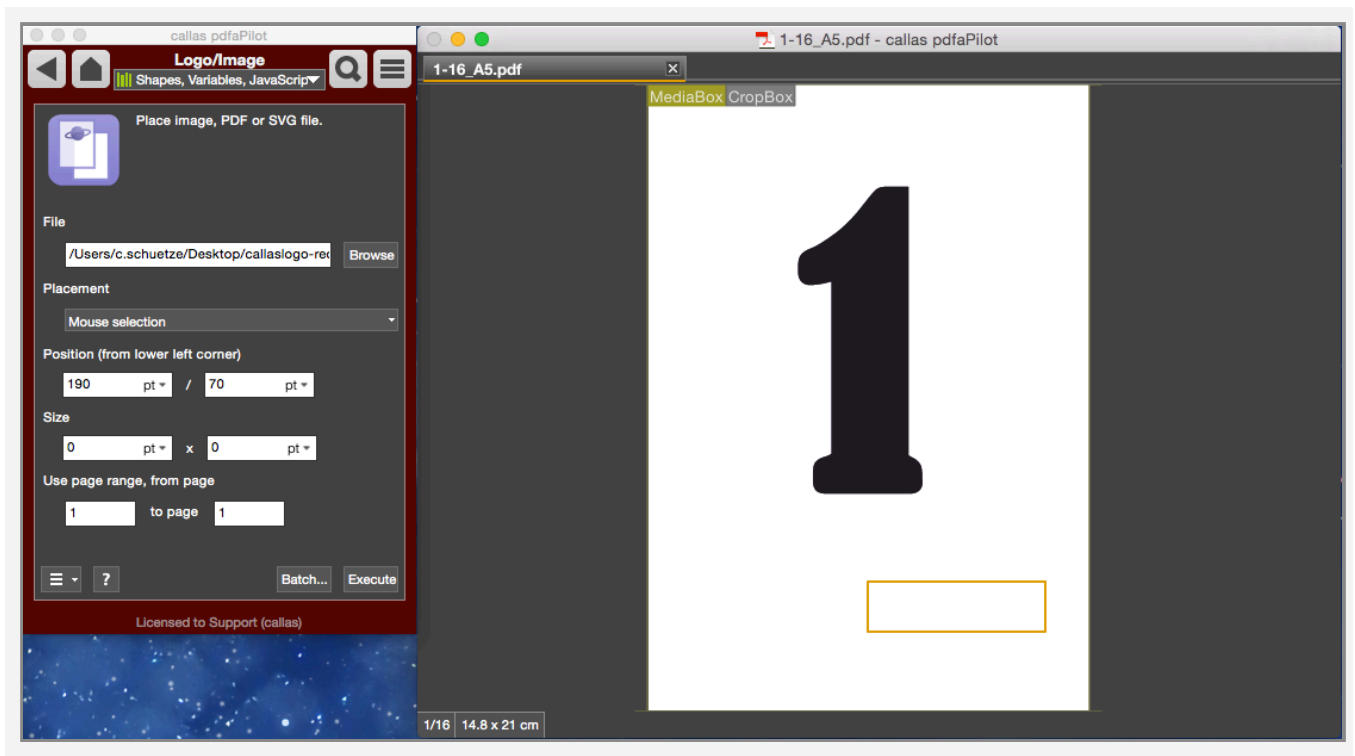
The Fixup has been successfully applied.

4.6 Add logo/image

This Action lets you add any logo or image to individual pages or all pages within a PDF.

You can specify both the position and desired size of the image.

File	The file to be placed (image, PDF or SVG)
Placement	Coordinates where the object is to be placed.
Position (from lower left corner)	Offsets the object relative to the parameters given under “Placement”. Values should be entered in the form of coordinates, giving x and y values relative to the reference point given under “Placement”. This means that values such as “-5” are also valid.
Size	Specifies the size of the region. If “Placement” is set to “Mouse selection”, the dimensions of the region can be specified.
Page range	Specifies the range of pages on which the object will be placed.

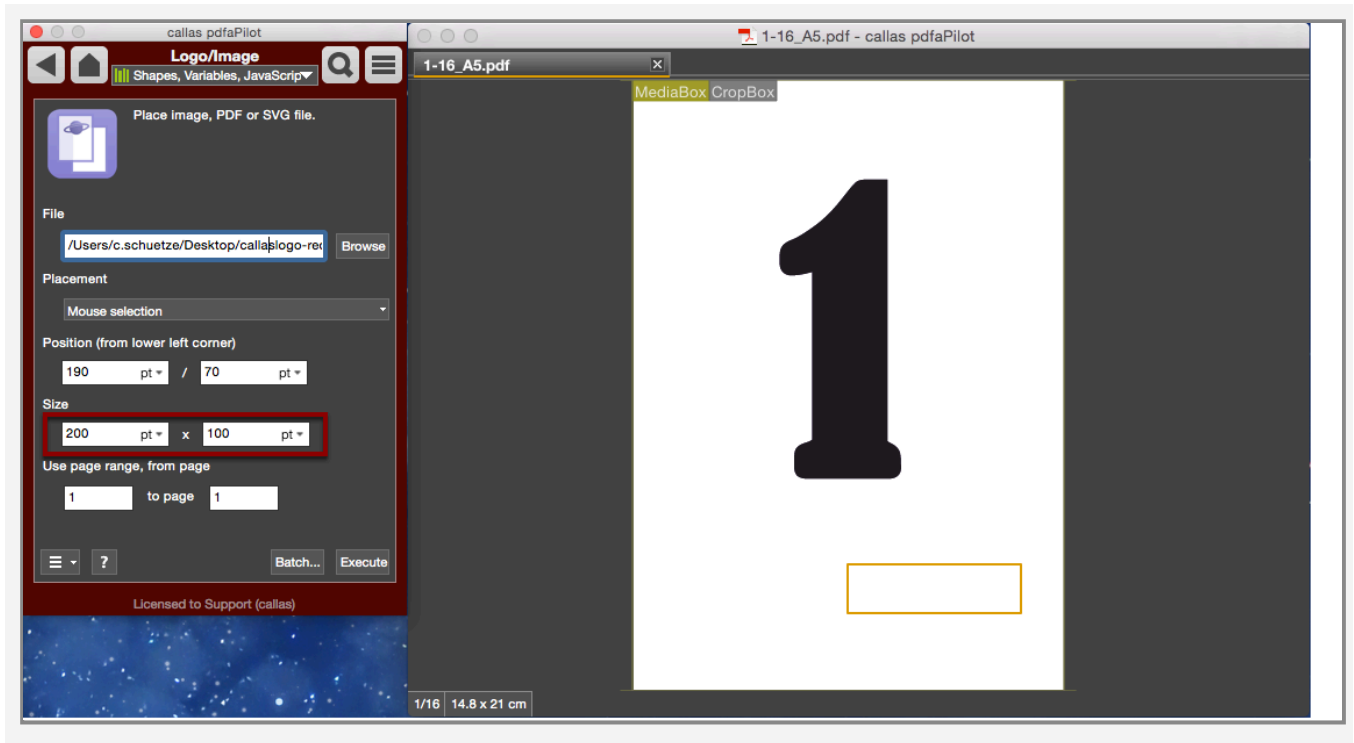


If “Placement” is set to “Mouse selection”, you can click and drag to select a region with the mouse to define the placement.

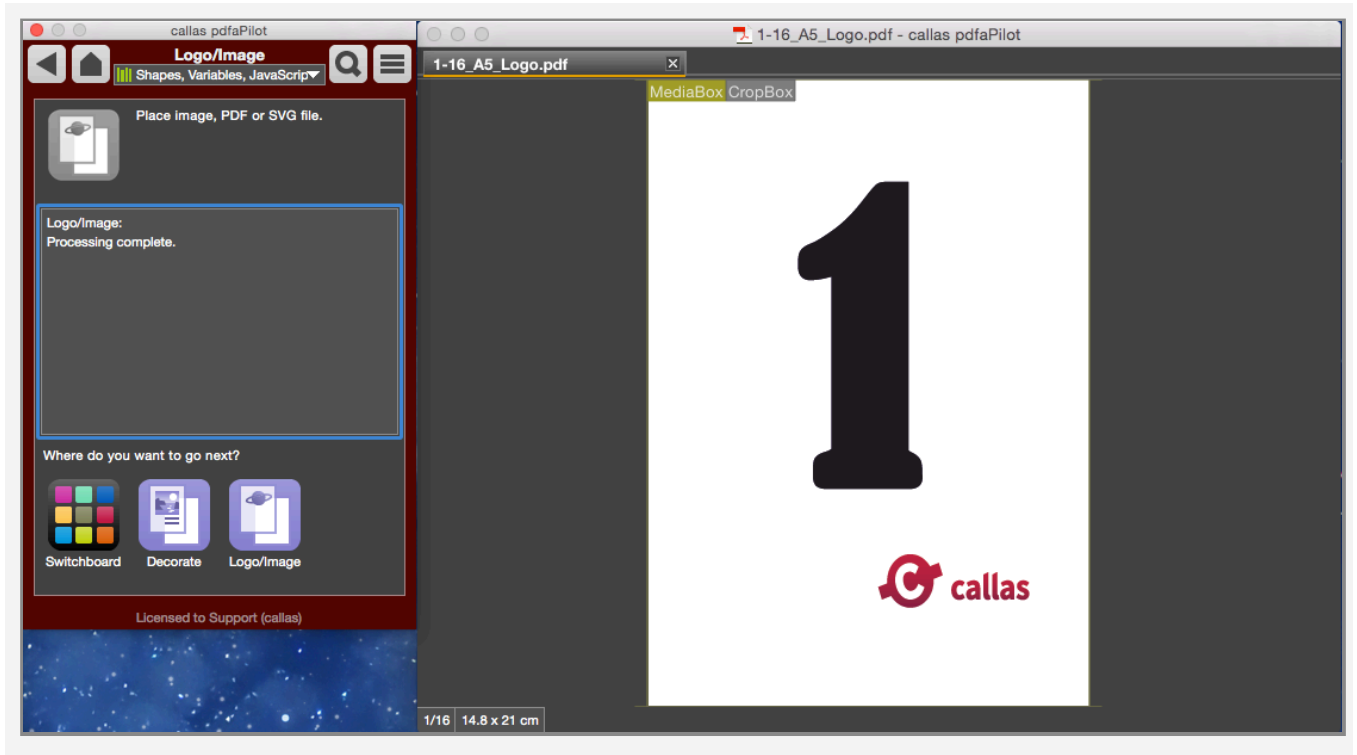
The value for the size of the region can also be altered at any

time.

However, the highlighted region will only be drawn once when you click and drag. It will not be changed if you edit the details. (If you want to adjust the way the region is drawn, you can select a new Placement type and then re-select “Mouse selection”.)



Region with values altered.

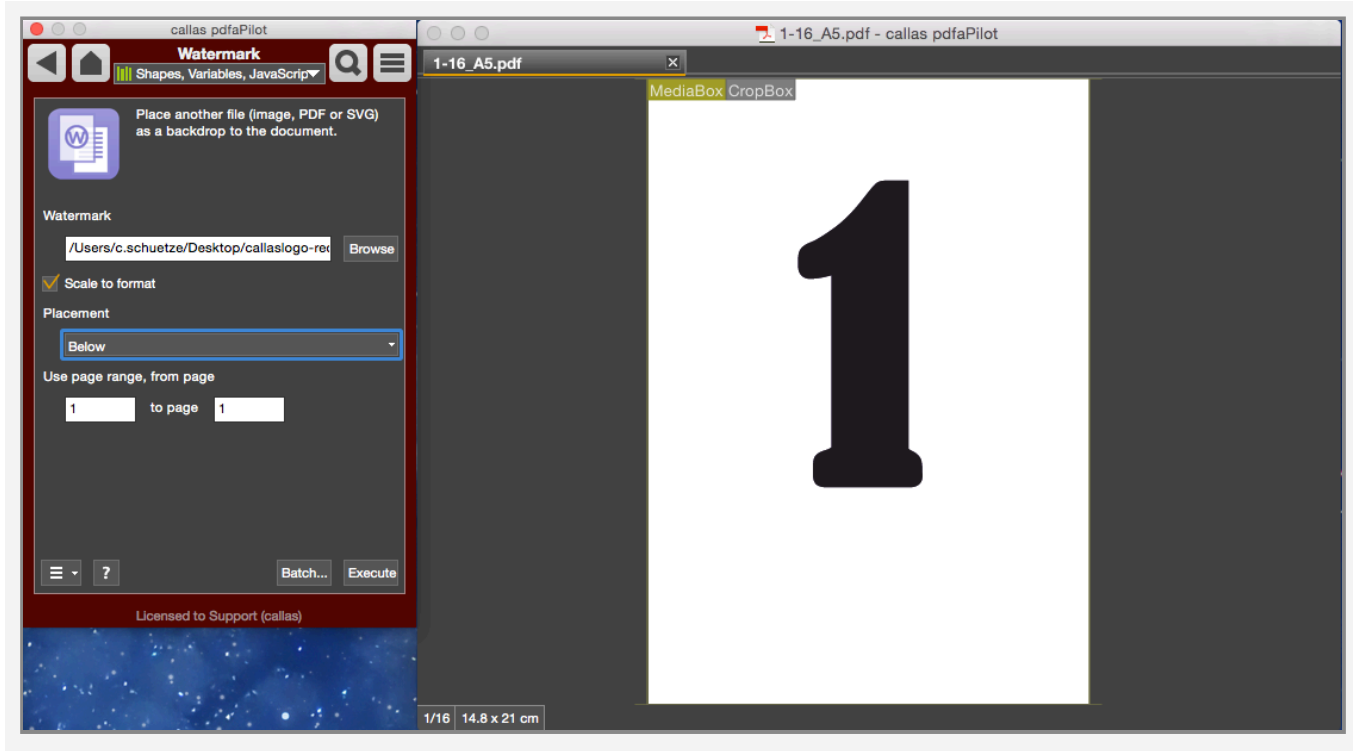


Logo placed after execution.

4.7 Add watermark

This Action lets you add any logo or image to one or all pages within a PDF, with a transparent effect. The image will be vertically and horizontally centered.

You can specify whether the object should be placed on top of or below the existing content.



Watermark	Click “Browse” to choose the file to be placed.
Scale to format	Scales the placed object to the page size.
Placement	Specifies whether to place the object on top of or below the existing content.
Page range	For multi-page documents, specifies the range of pages on which the object will be placed.



A “scaled-to-format” result.

The logo has been scaled to the page size during placement.



An example of a non-scaled-to-format result.

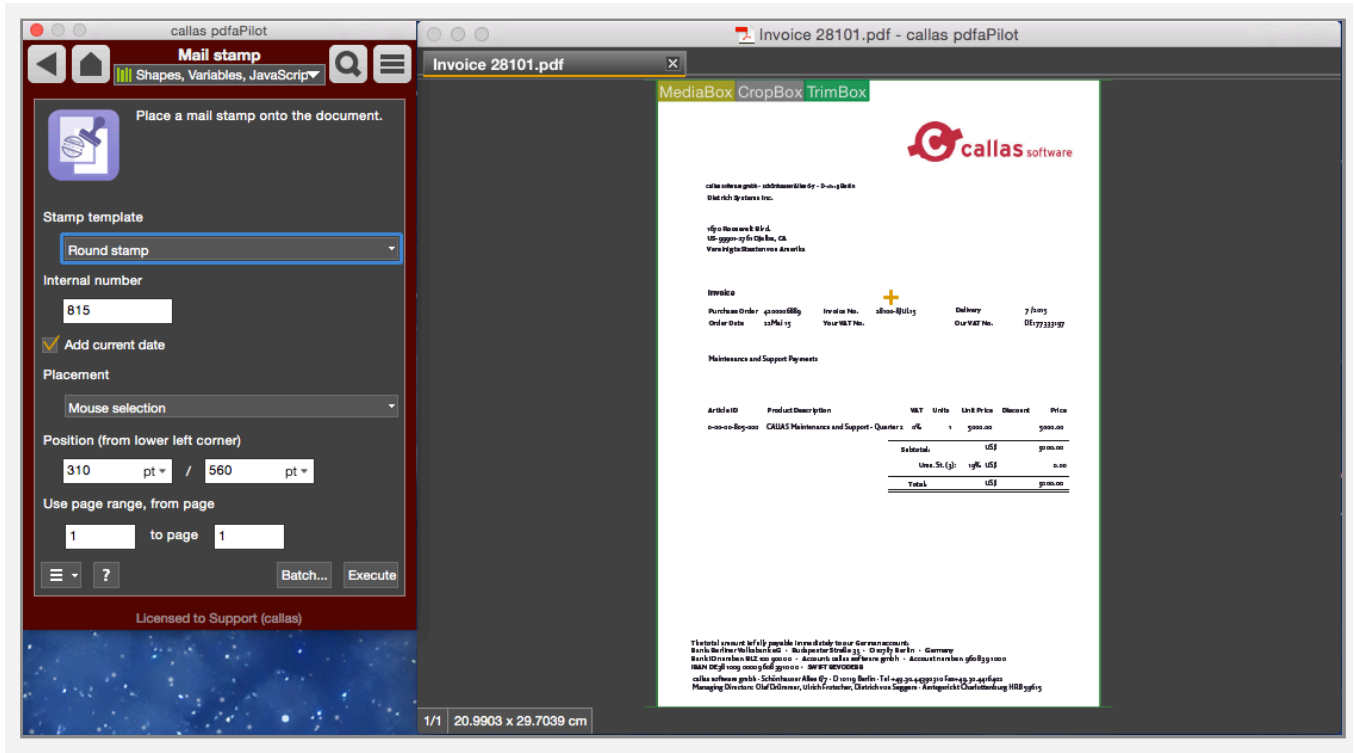
The logo has been placed in the document at its original size.

4.8 Place mail stamp

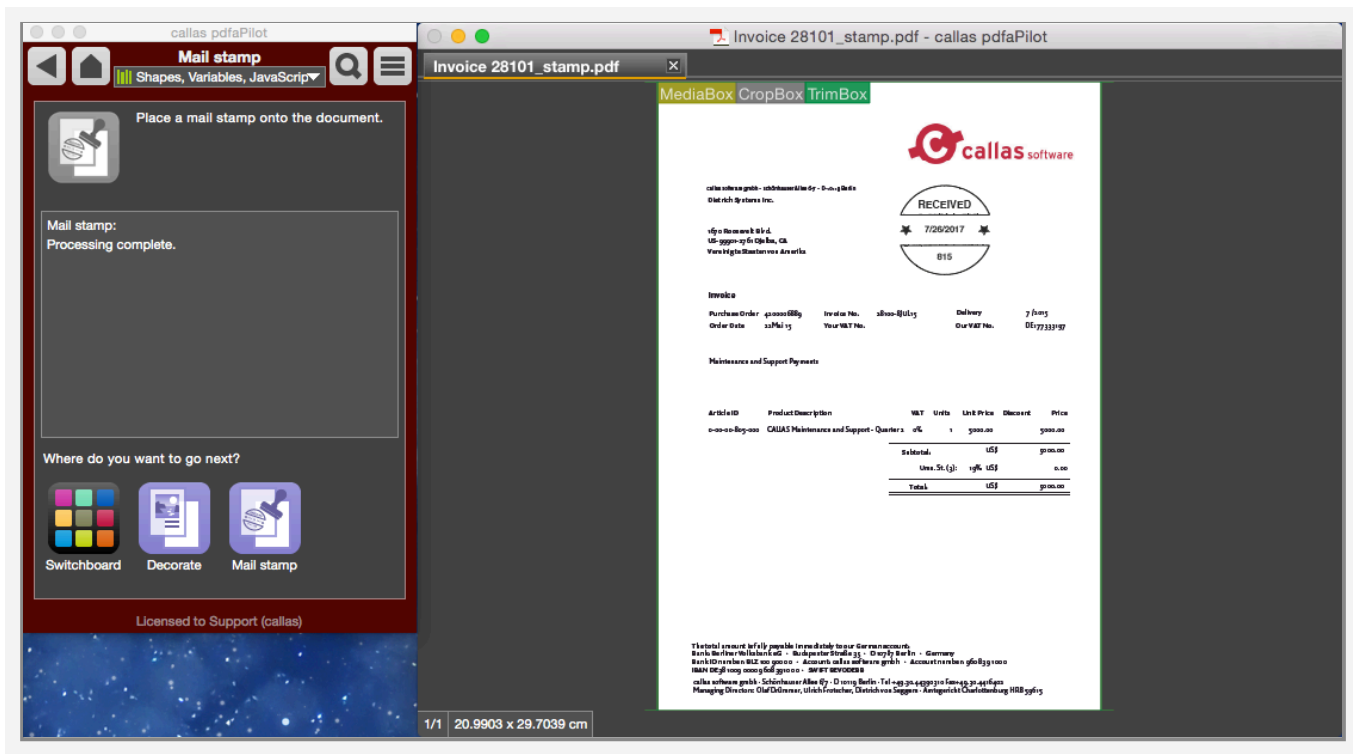
This Action adds a mail stamp to the document. You can also specify whether to output a user-definable number and/or the current date at the same time.

As this Action uses HTML templates as a basis, you can also create your own layouts or edit existing ones. The corresponding template folder can be found in the menu at the bottom left of the Switchboard Action.

Stamp template	Round or square stamp
Internal number	You can enter any number here
Current date	Adds the current system date
Placement	Coordinates where the object is to be placed.
Offset	Offsets the object relative to the parameters given under "Placement". Values should be entered in the form of coordinates, giving x and y values relative to the reference point given under "Placement". This means that values such as "-5" are also valid.
Page range	For multi-page documents, specifies the range of pages on which the object will be placed.



Choose “Mouse selection” to set the placement. The selected reference point will be indicated with an orange “plus” symbol.



The final result with stamp placed.

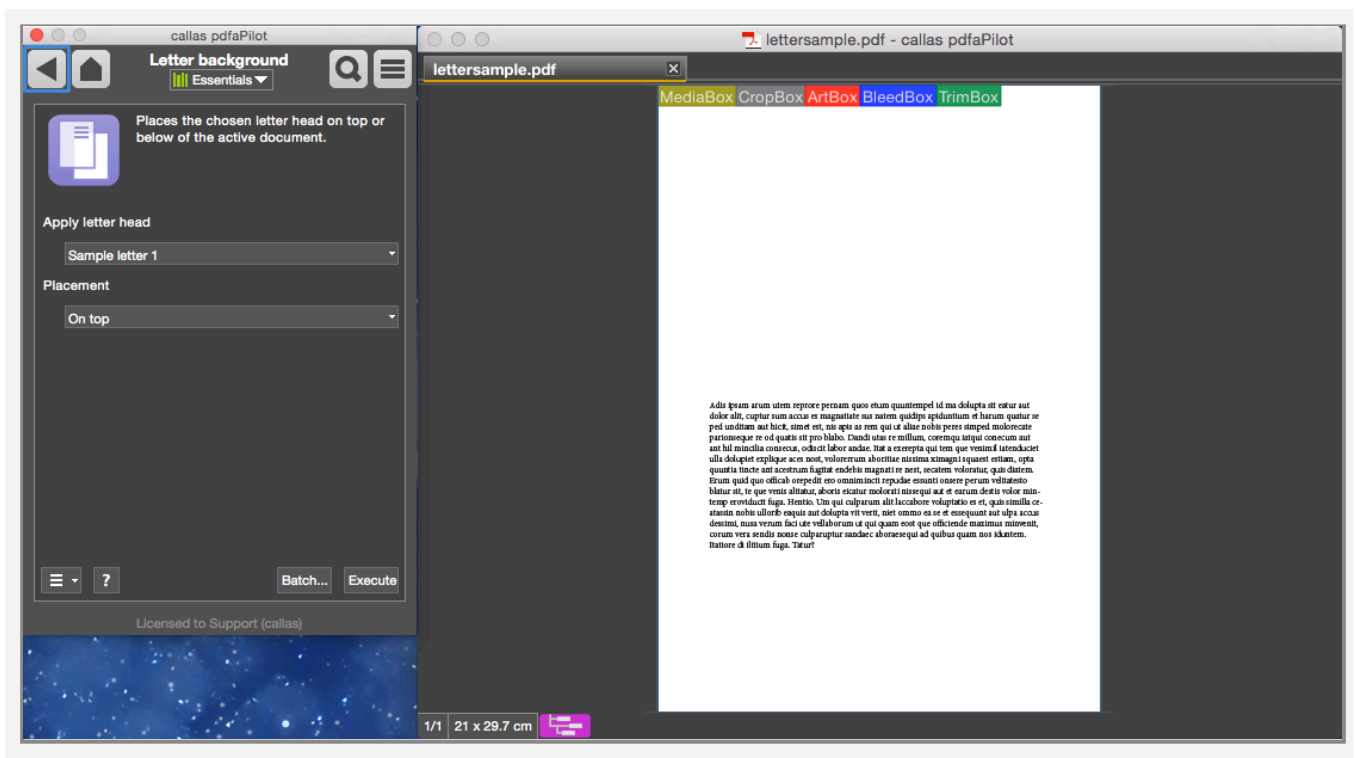
4.9 Place letter background

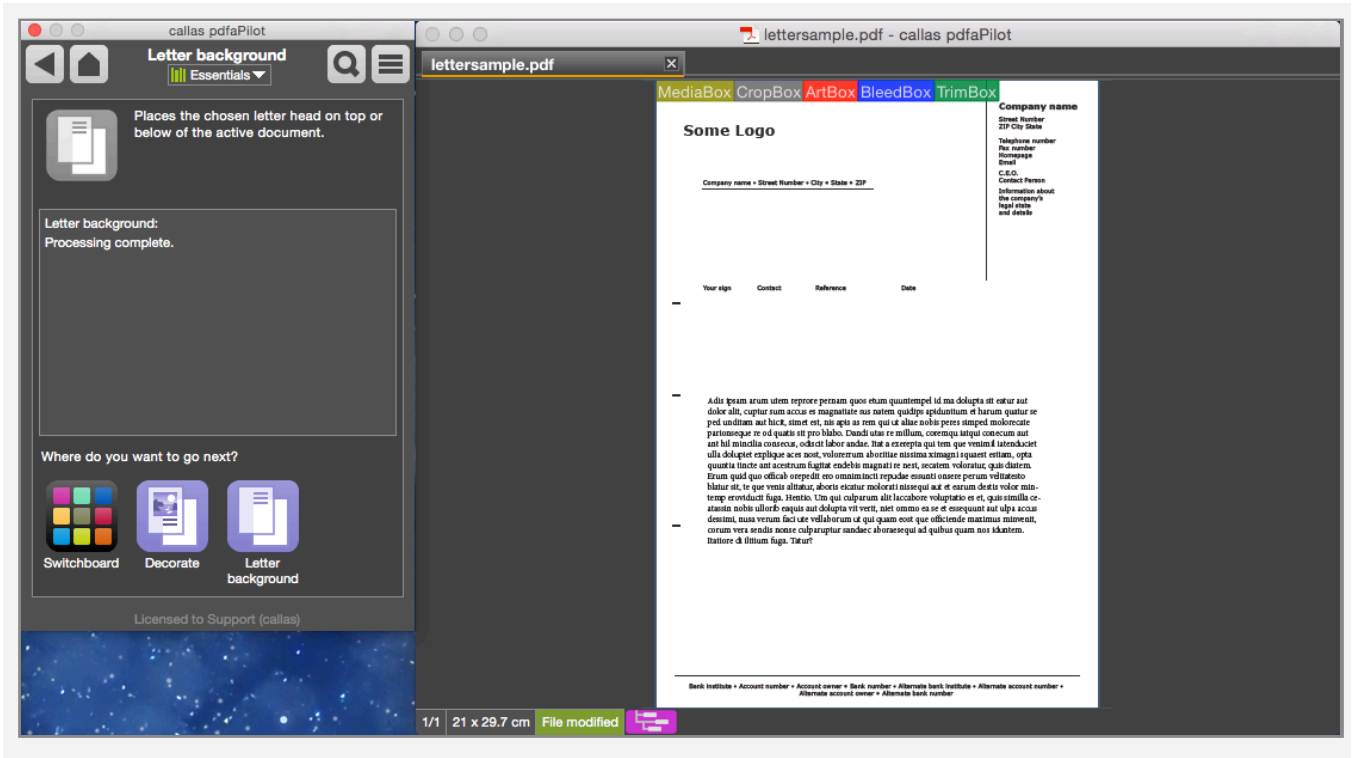
The “Letter background” Action allows you to add letter templates to existing documents.

These can be added on top of or below the document.

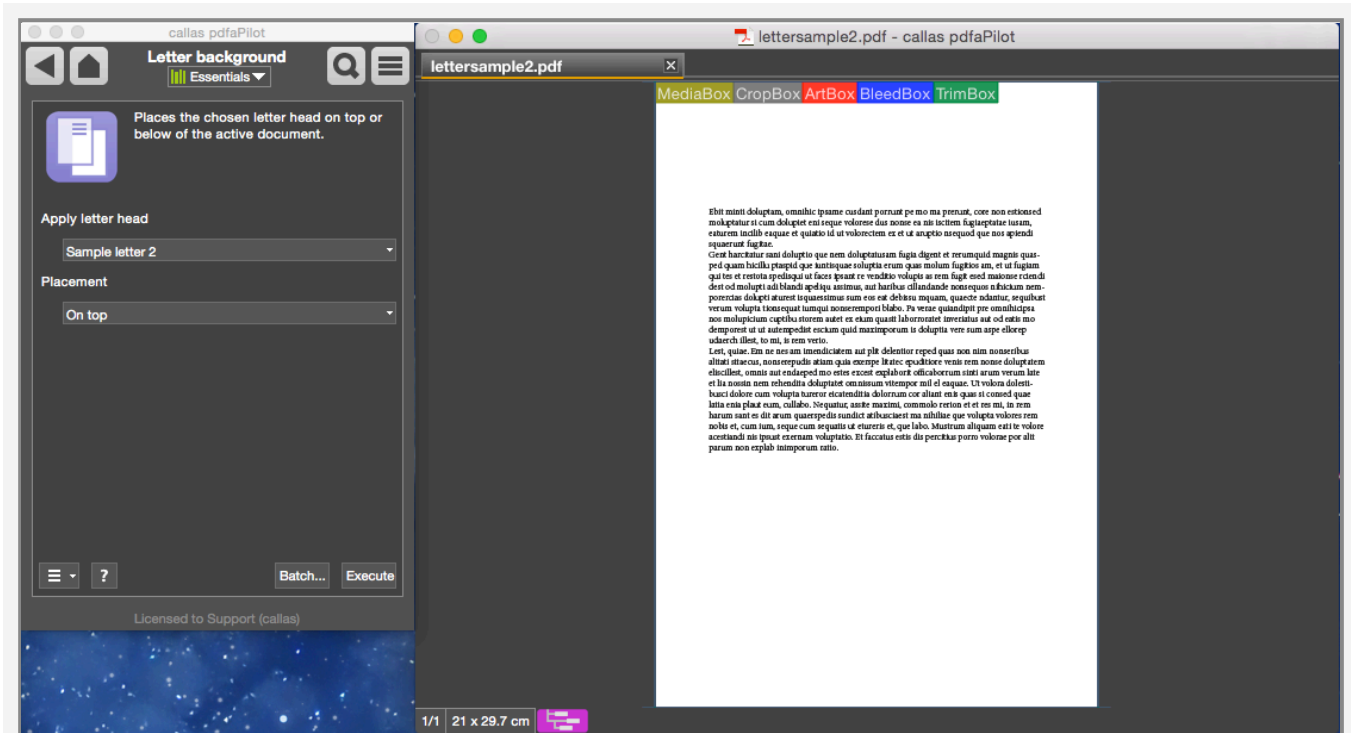
You can also provide a first and second page in the letter template. The first page of the template will then be placed only on the first page of the PDF being processed, while the second template page will be used for all subsequent pages.

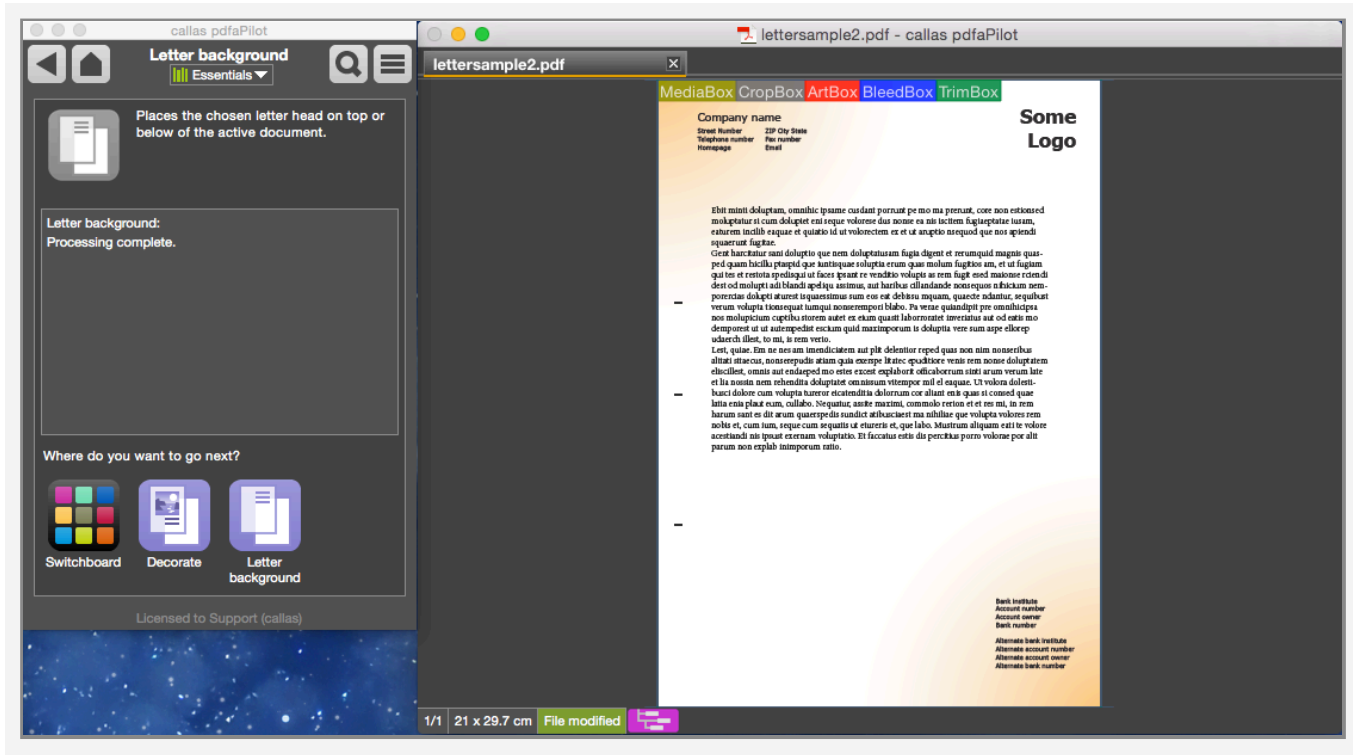
Before/after example with sample letter 1



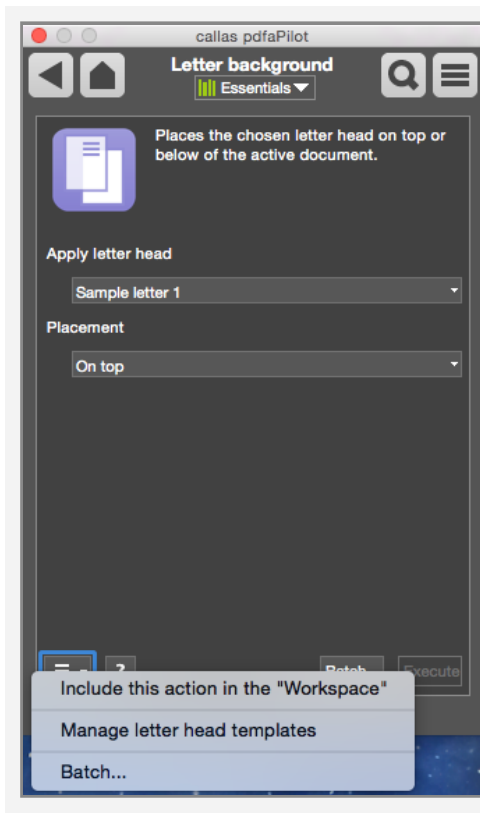


Before/after example with sample letter 2





Managing letter head templates



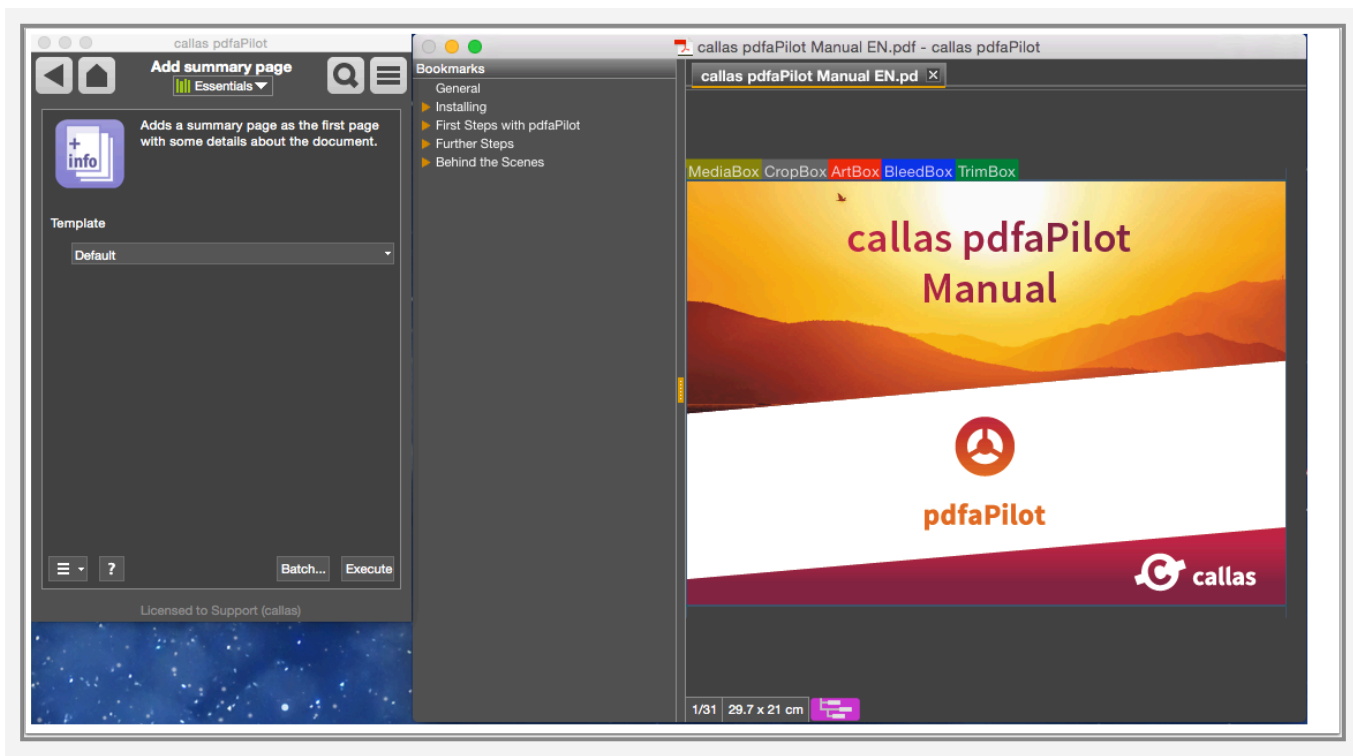
Using “Manage letter head templates” in the menu at the bottom left of the Switchboard Action, you can add your own templates.

To load newly added templates, you must first click Back to leave the Action, then open the Action again.

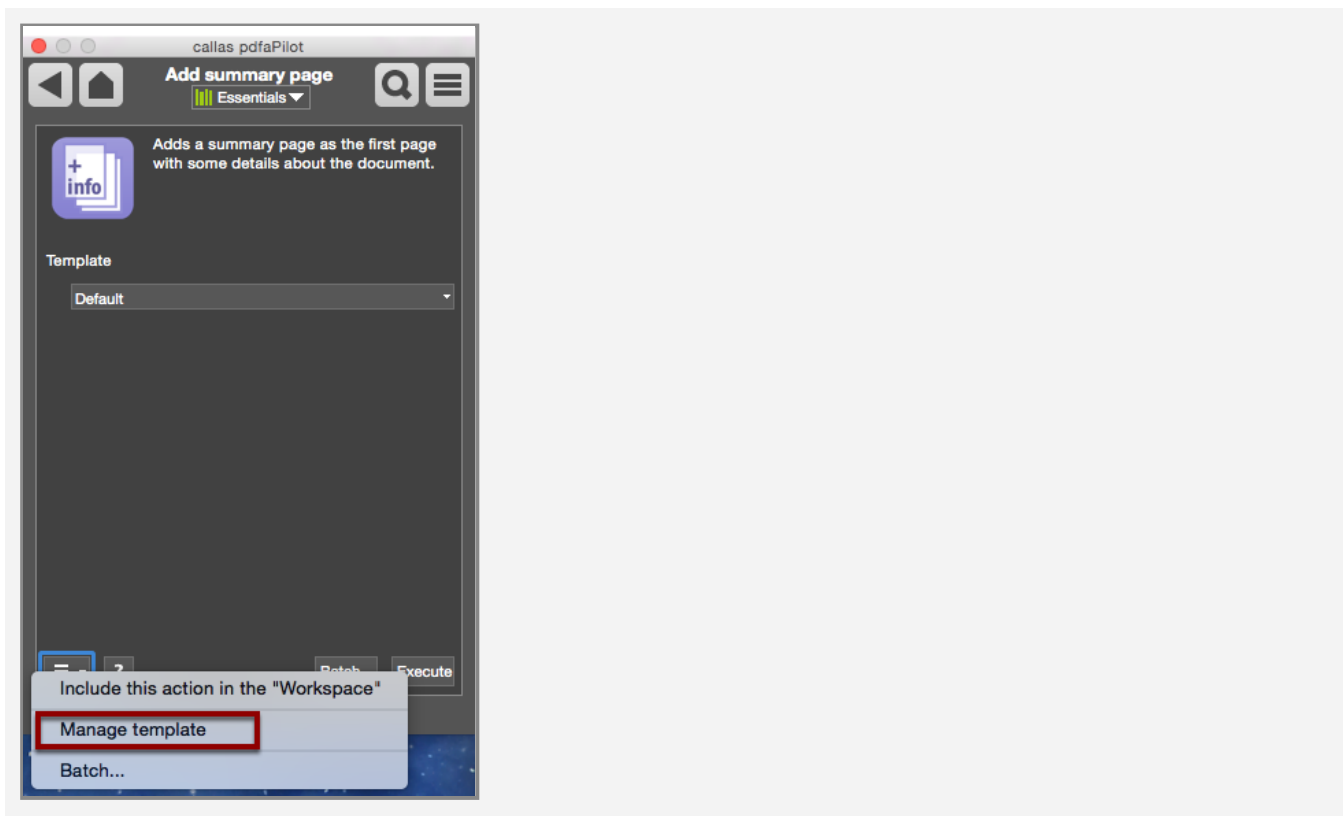
4.10 Add summary page

The “Add summary page” Switchboard action creates a cover sheet with information about the document as a whole:

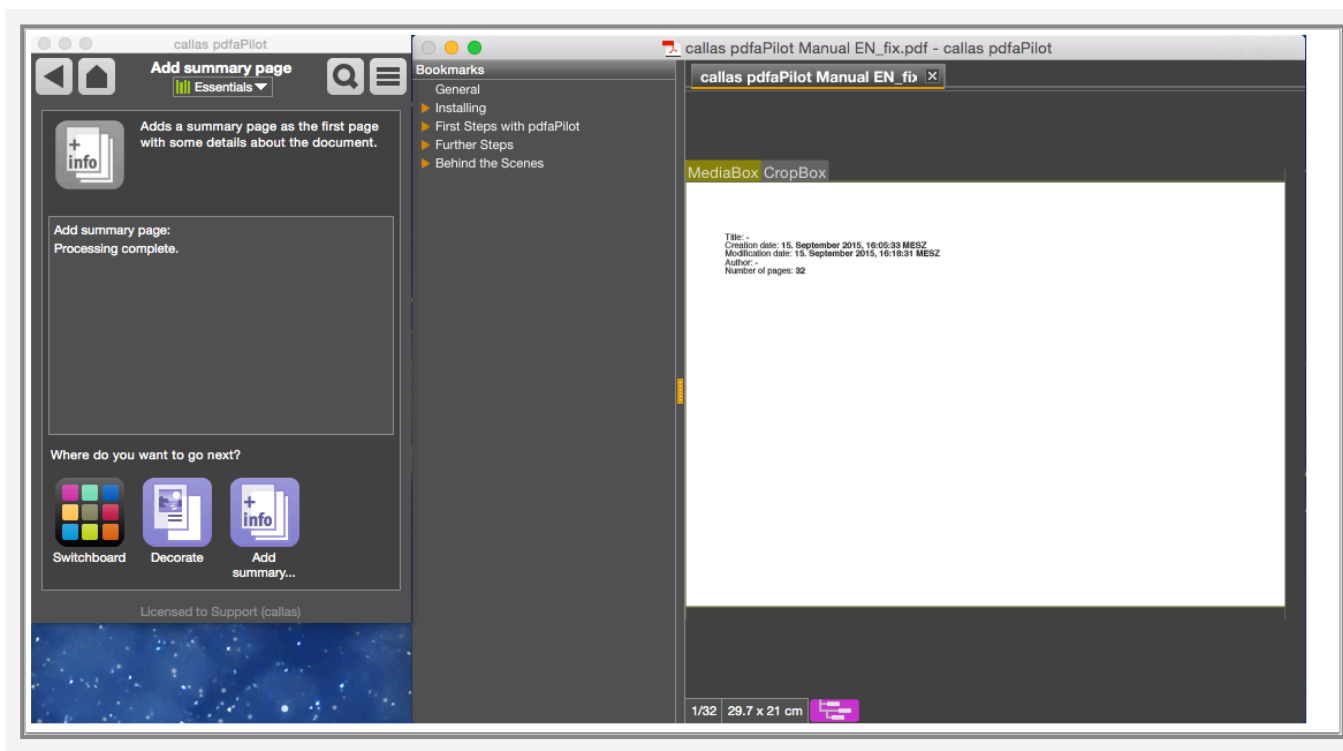
- Title
- Creation date
- Modification date
- Author
- Number of pages



Choose a template to use.



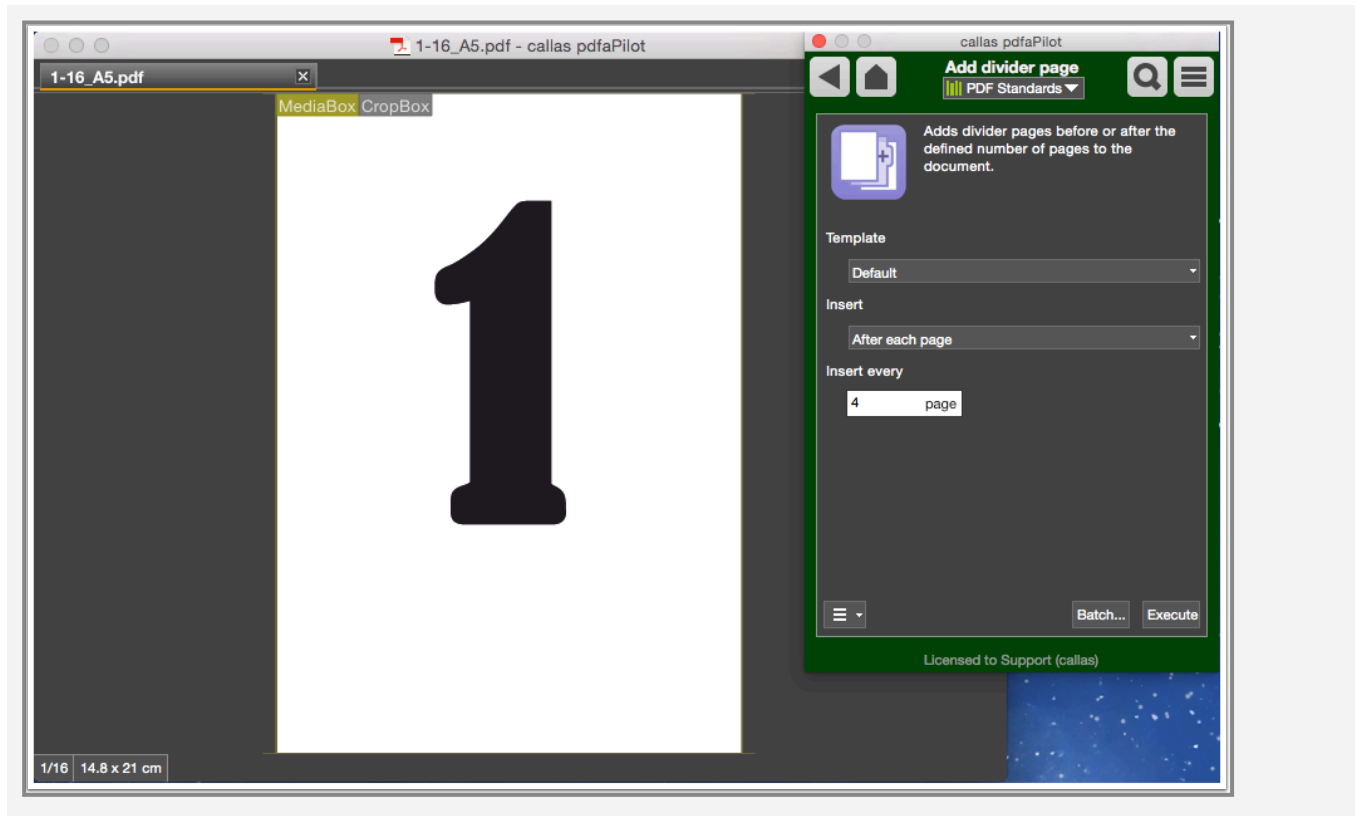
Click “Manage template” from the menu in the lower-left corner to open the HTML template being used and to edit it as required.



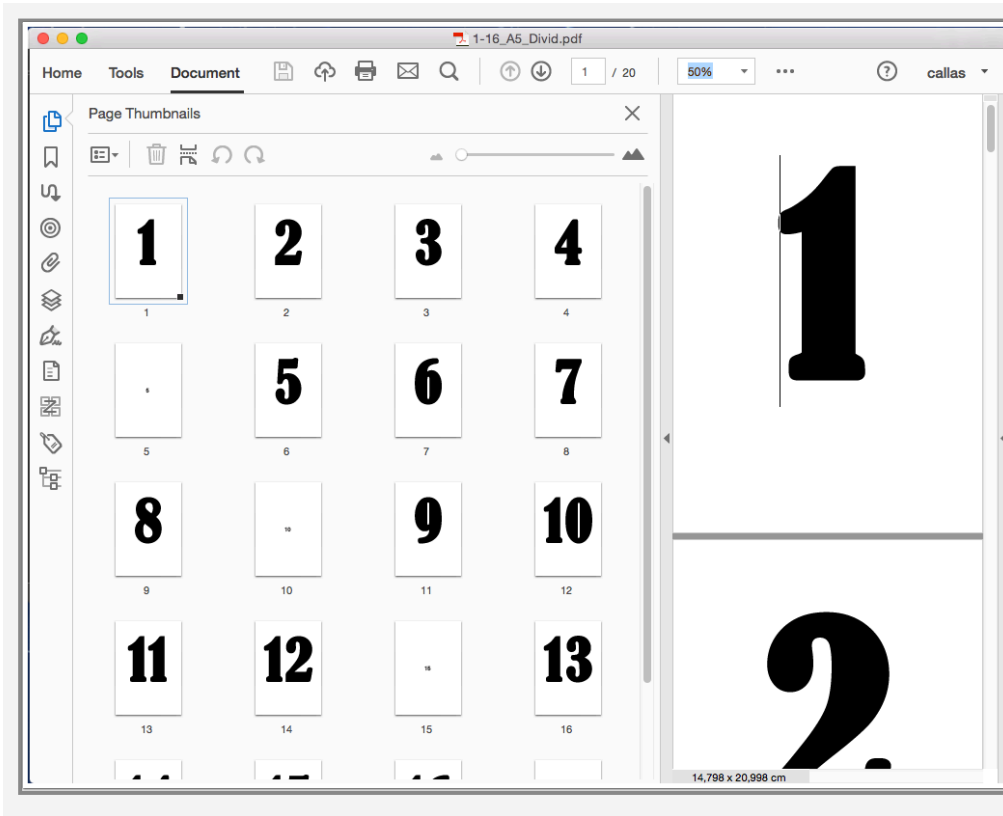
4.11 Add divider page

The “Add divider page” Switchboard Action lets you add pages at specific points within a document.

You can customize the page to be inserted by editing a HTML-based template file. The corresponding folder can be found in the menu at the bottom left of the Switchboard window.



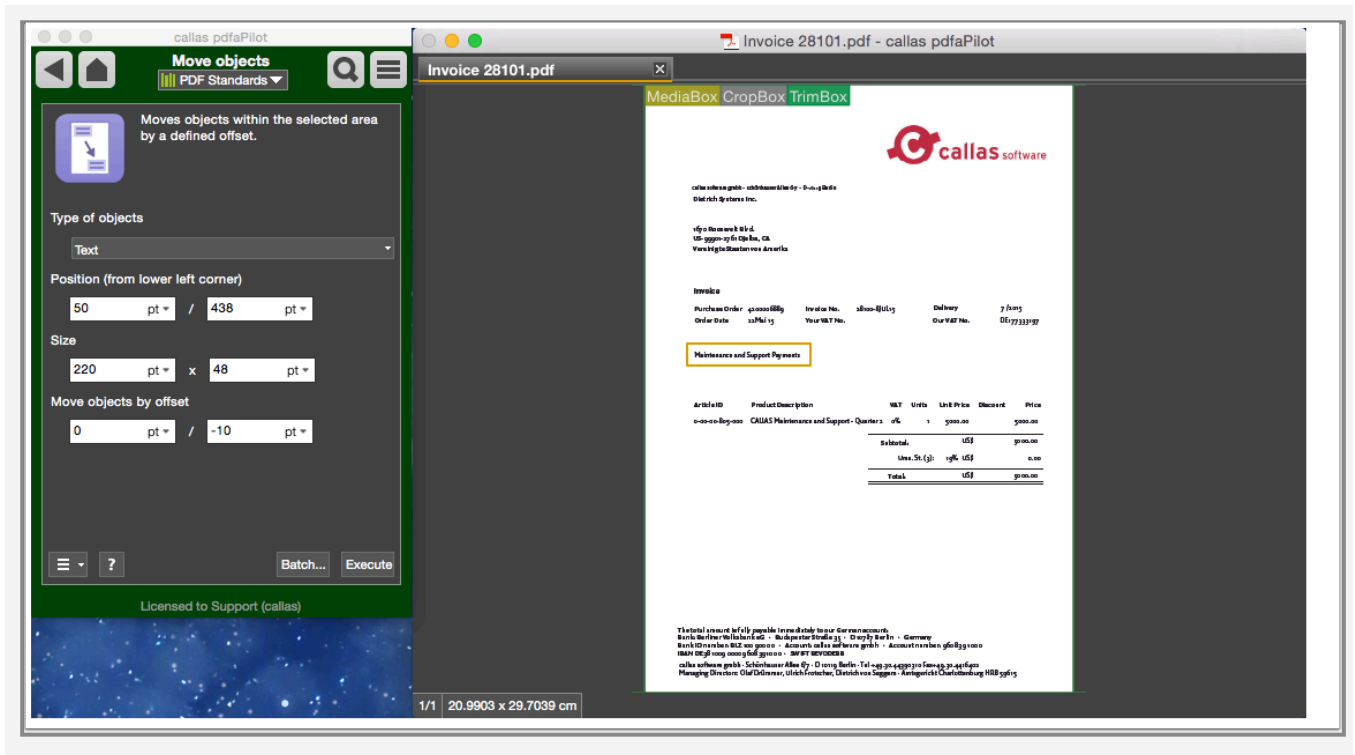
Results with divider page shown in Acrobat:
A new page has been added after every four pages.



4.12 Move objects

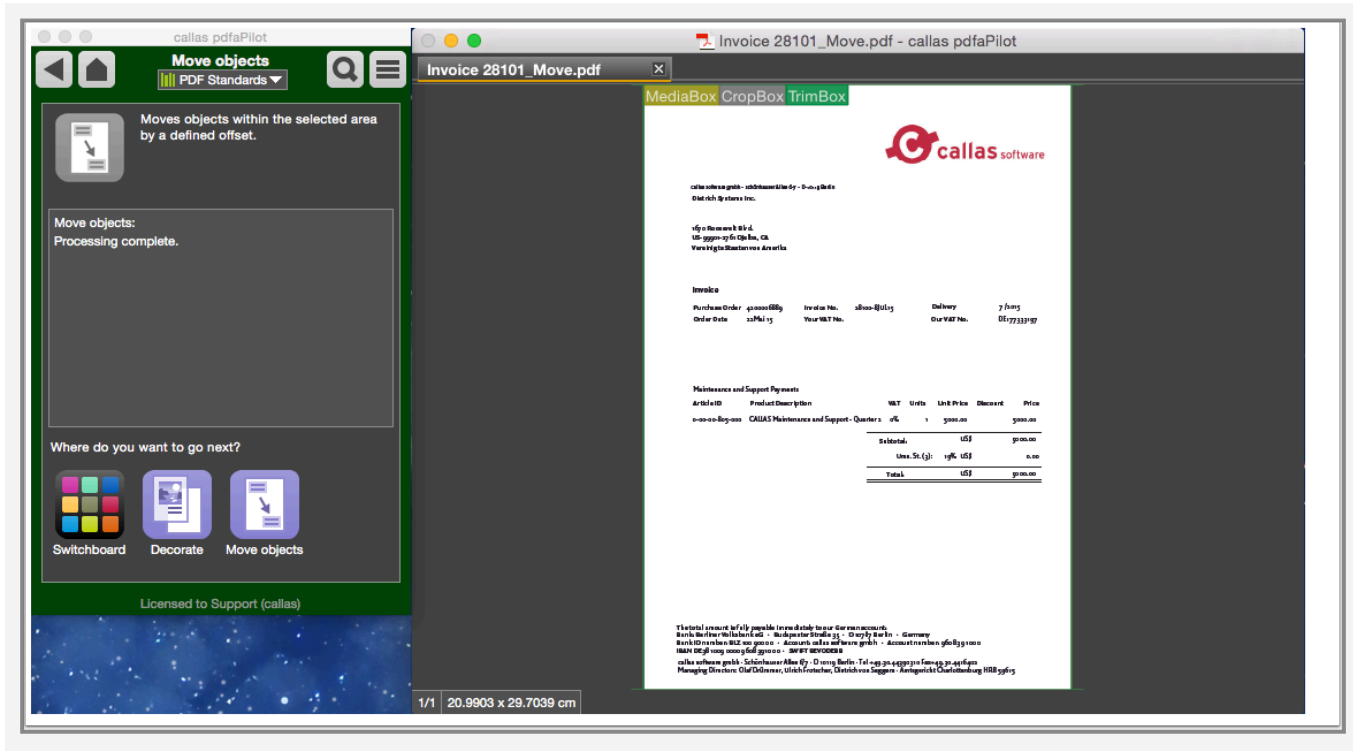
The “Move objects” Switchboard Action lets you move the following object types:

“Text”, “Vectors”, “Text and vectors” and “Images”.



First, use the cursor to draw a box around the objects to be moved. Next, define the offset by which the selected objects are to be moved.

By filtering the object types, the Action can also be applied only to specific types of objects.



Once the Fixup is applied, the objects will be moved by the specified offset .

5. Working with XMP Metadata and PDF/A files

5.1 Extension schema editor

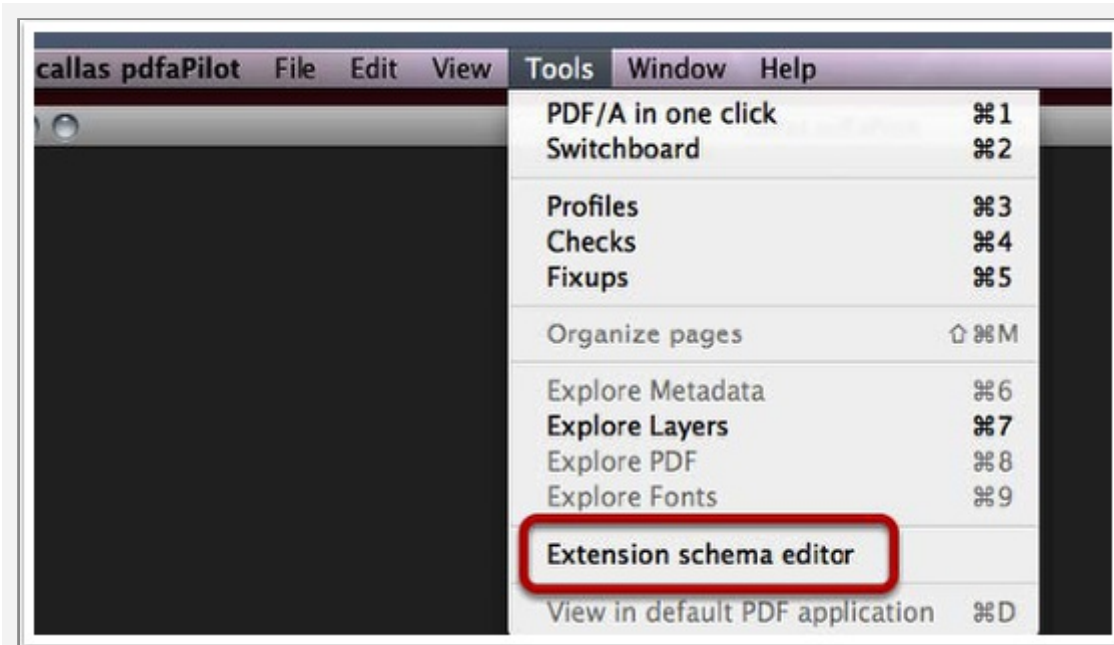
About custom metadata and pdfaPilot's XMP Extension Schema editor

If a user wants to add custom metadata to a PDF/A file, an extension schema is required that covers this custom property.

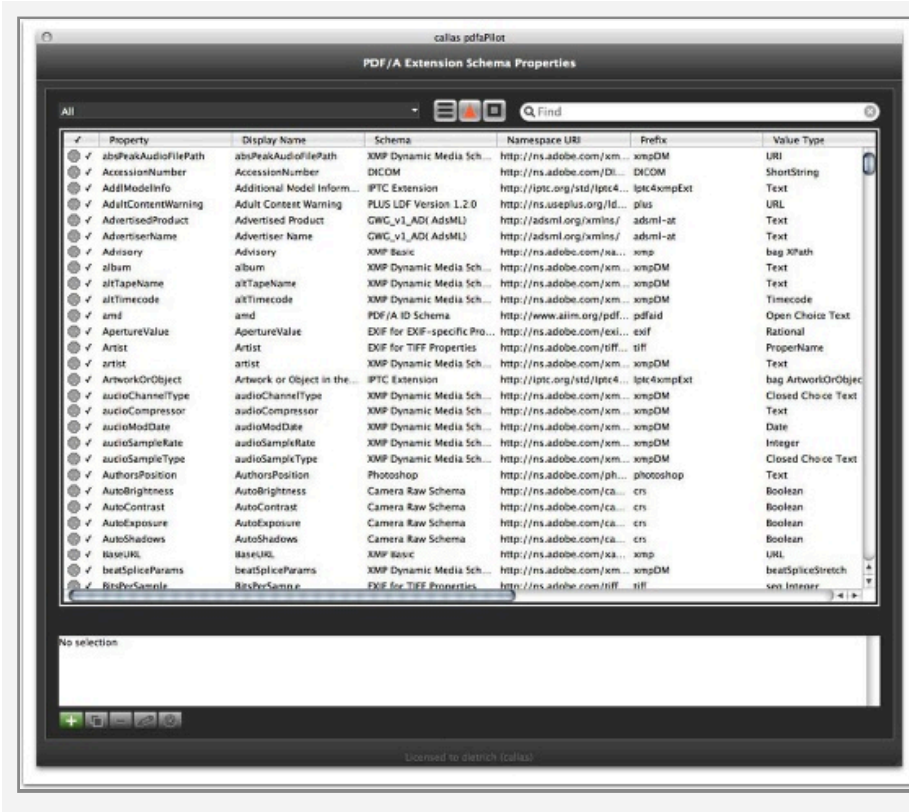
callas pdfaPilot comes with its own XMP Extension schema editor, which can be accessed via the action "Extension schema editor" in the "Metadata" section in the Switchboard. Here you can have a look at predefined schemas, data fields and value types or create your own to be embedded during conversion to PDF/A. You can also capture properties which are present in the document metadata.

Create and Manage PDF/A compatible XMP extension schemas

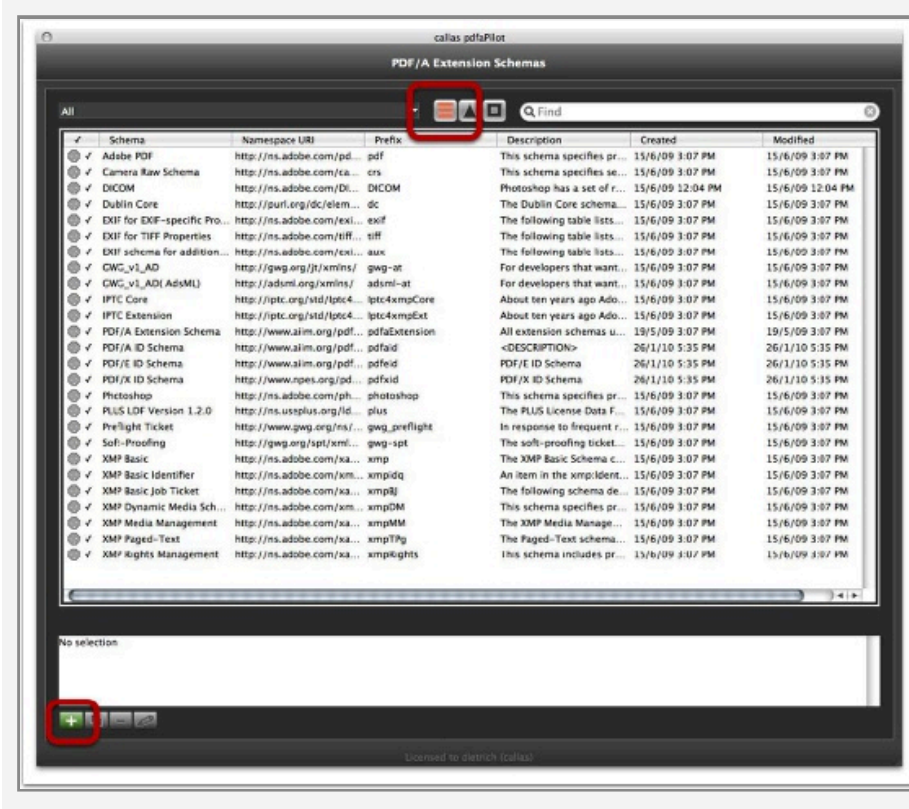
Open the Extension Schema Editor in callas pdfaPilot.



A new window opens with a long list of Extension Schema Properties:



Switch to the list of Extension Schemas:

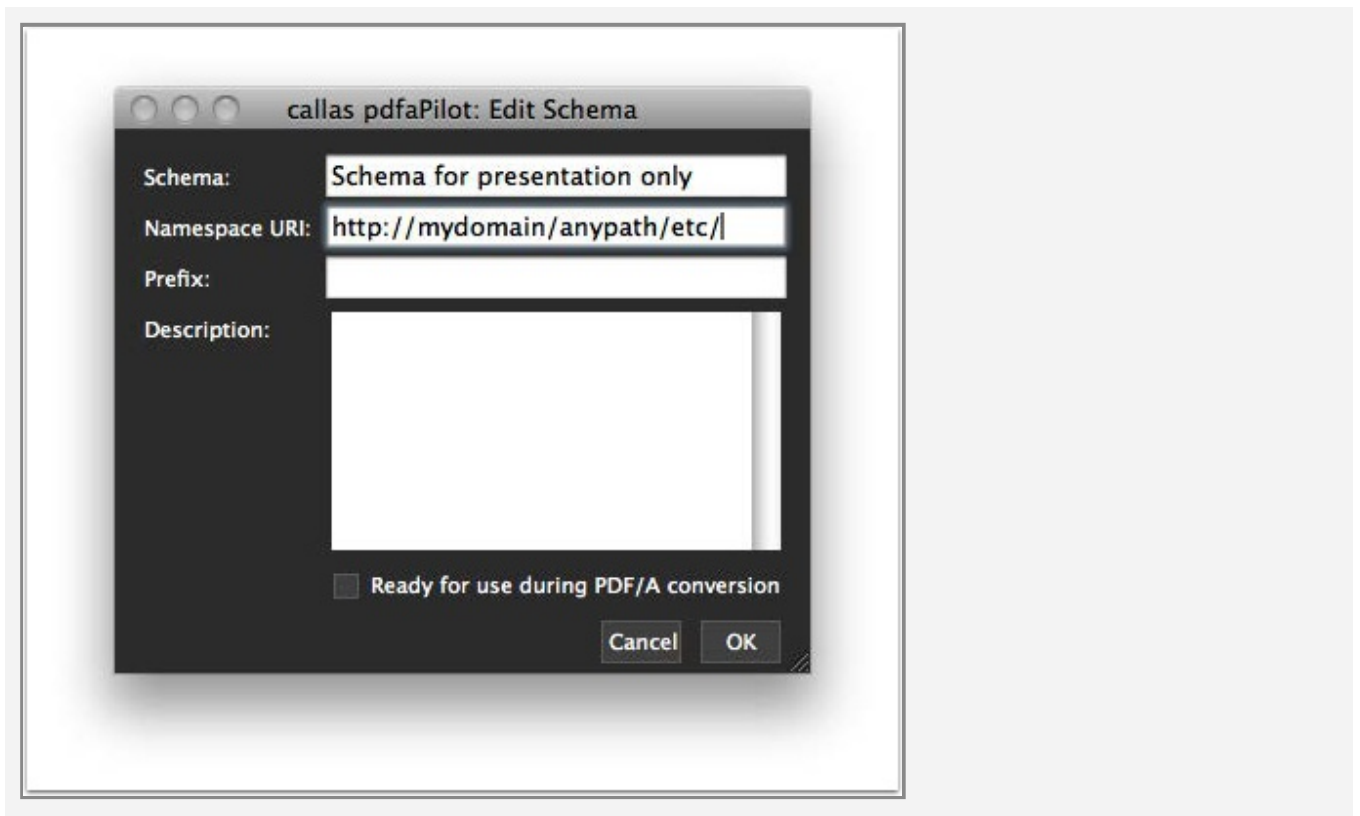


A Schema is a container for an arbitrary number of properties. First you will have to create your own schema that will later take your properties. Click on the left of the three icons on top of the list in order to switch to the list of Extension Schemas. Then click on the green [+] icon at the bottom of the window.

Define your Schema

A new window opens that lets you define your schema.

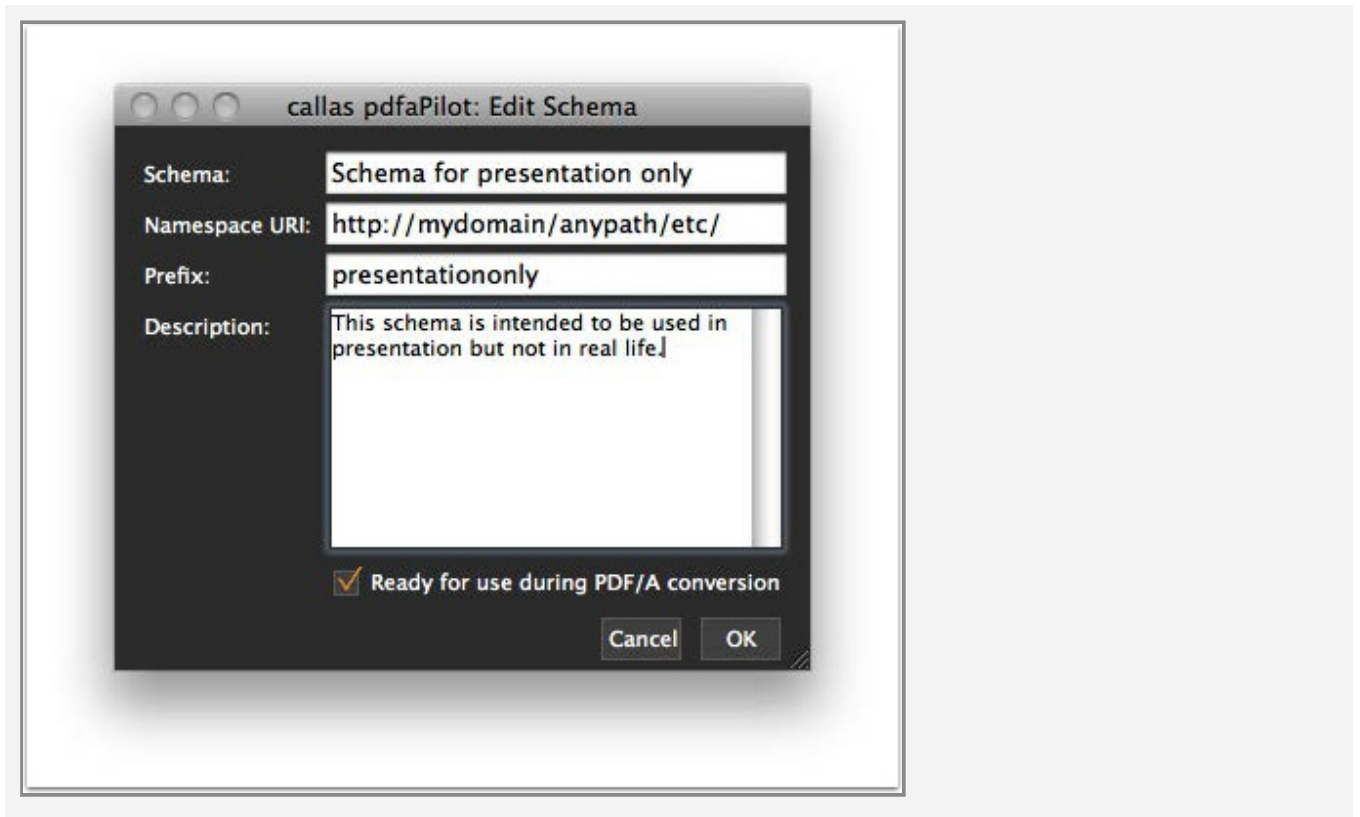
Give the schema a name that describes its area of application and a unique namespace URI:



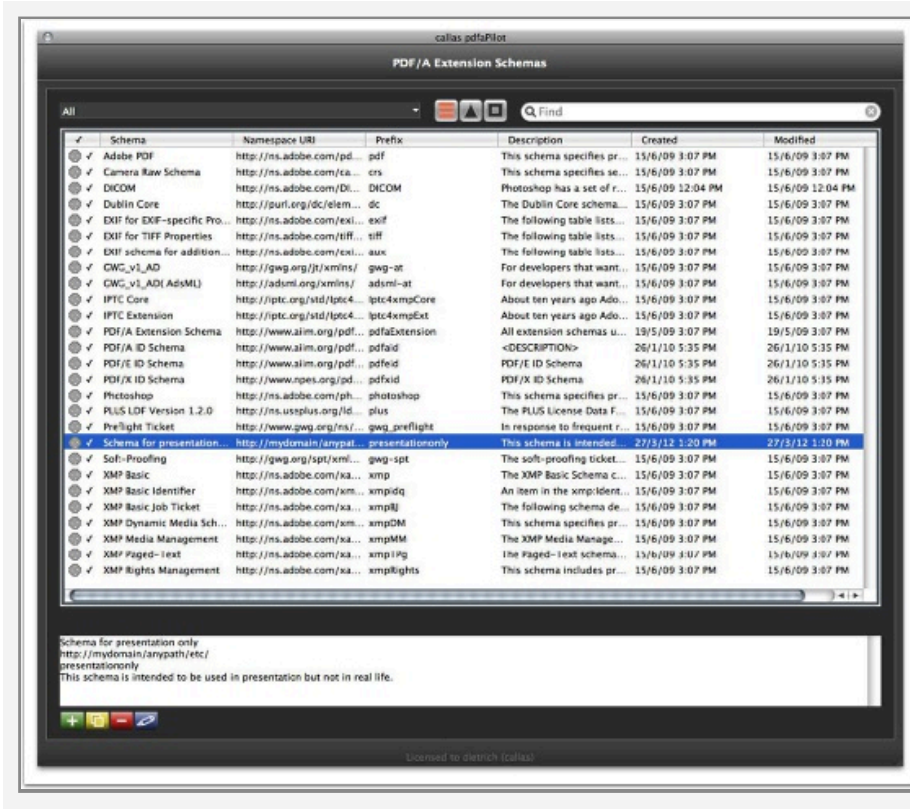
The namespace URI is intended to be a unique string so that the schema can be identified. Therefore it is common practice to use a domain name that belongs to the organisation which is defining the Extension Schema. That reduces the probability another organisation would use the same namespace. Note that it is neither required nor recommended that the namespace URI is in fact a website.

Define a prefix for the Extension Schema that will work as an abbreviation within the XMP metadata stream and enter a

description for the Extension Schema. Mark the Extension Schema as ready to use, otherwise pdfaPilot would not use it. Save the Extension Schema:

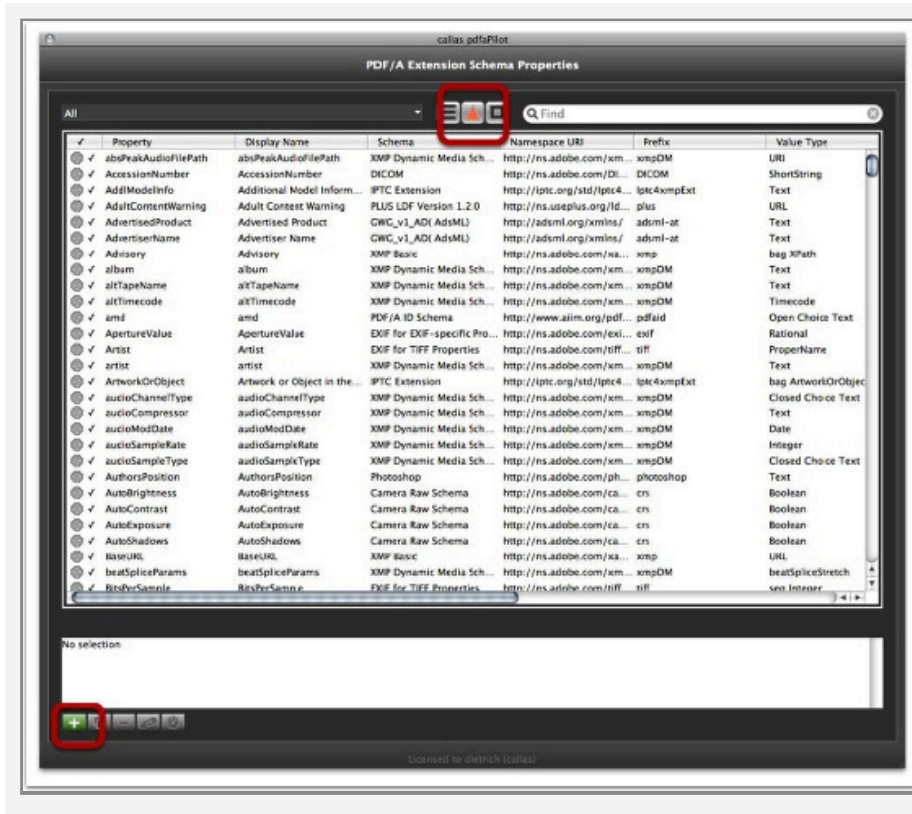


The new schema will now show up in the Extension Schema list:



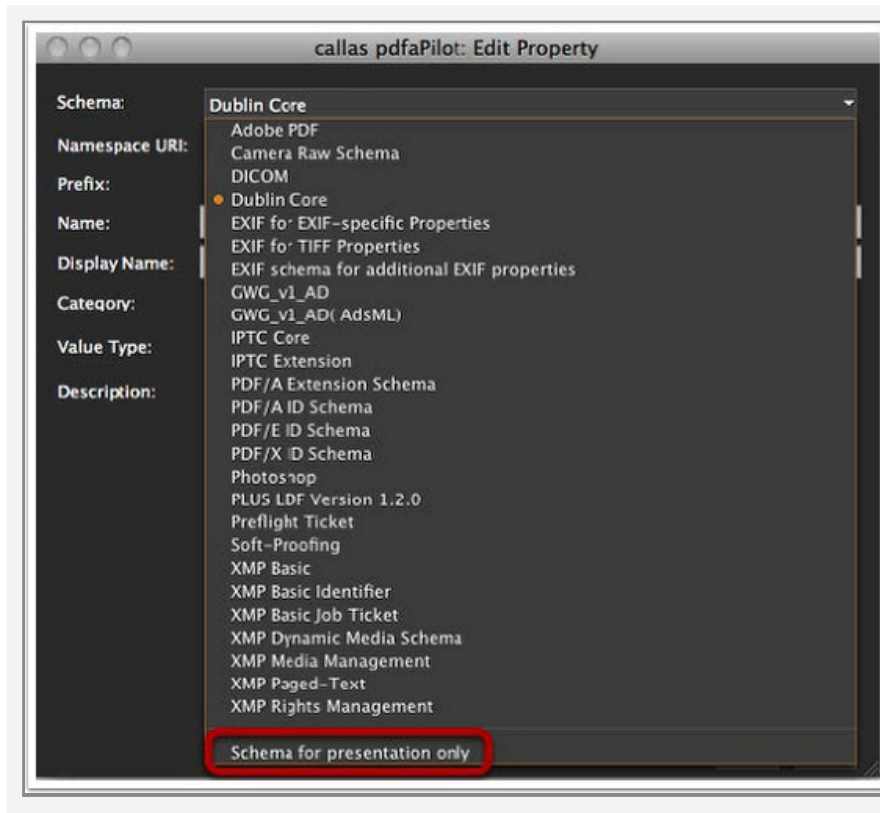
Extension Schema Property View

Switch to the Extension Schemas Properties view:



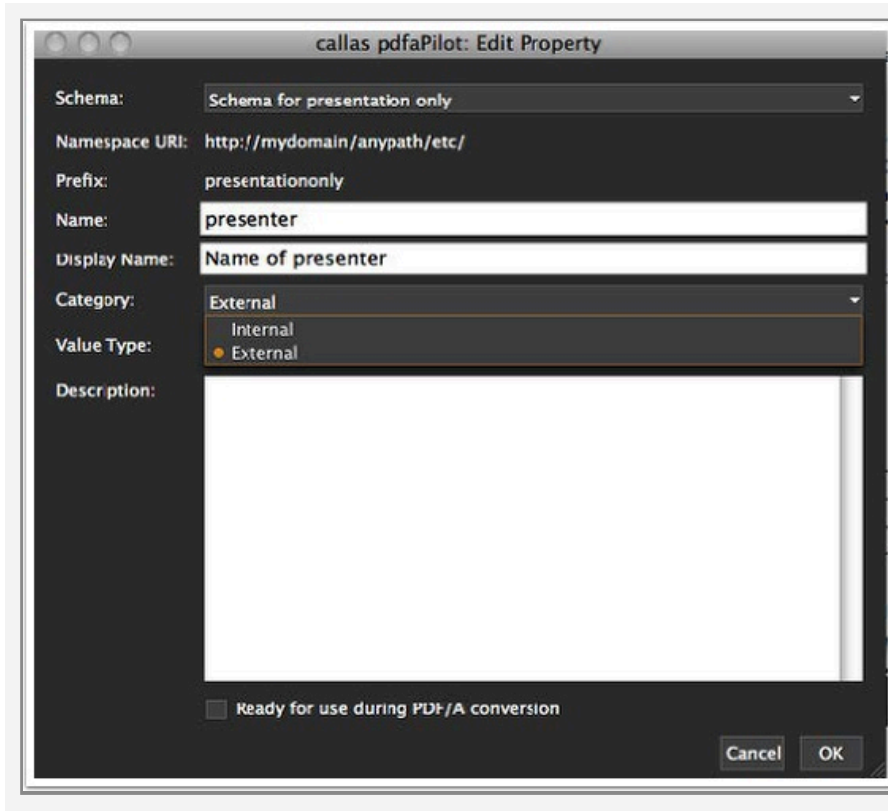
Click on the middle icon above the list of Extension Schemas in order to switch to the list of properties. Click at the green [+] icon in order to create a new property.

A new window opens that lets you define a property. Select the Extension Schema that you have just created:

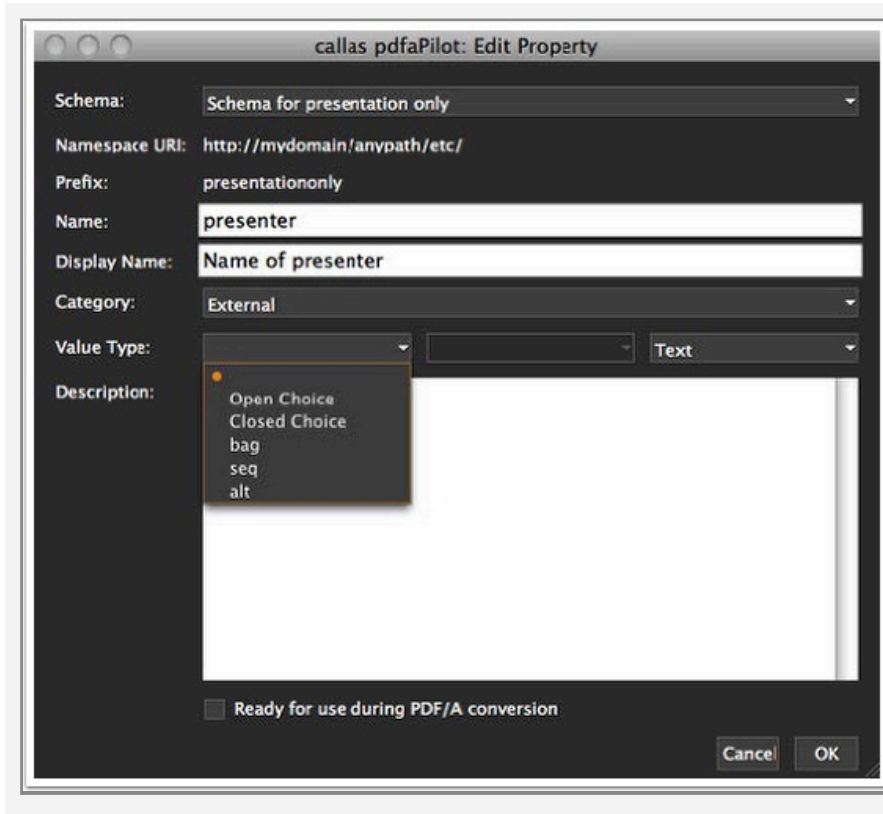


The Namespace URI and the Prefix are listed accordingly. Enter name, display name and category for the new property:

Note: The category defines whether it is user data (External) or rather technical internal information (Internal).

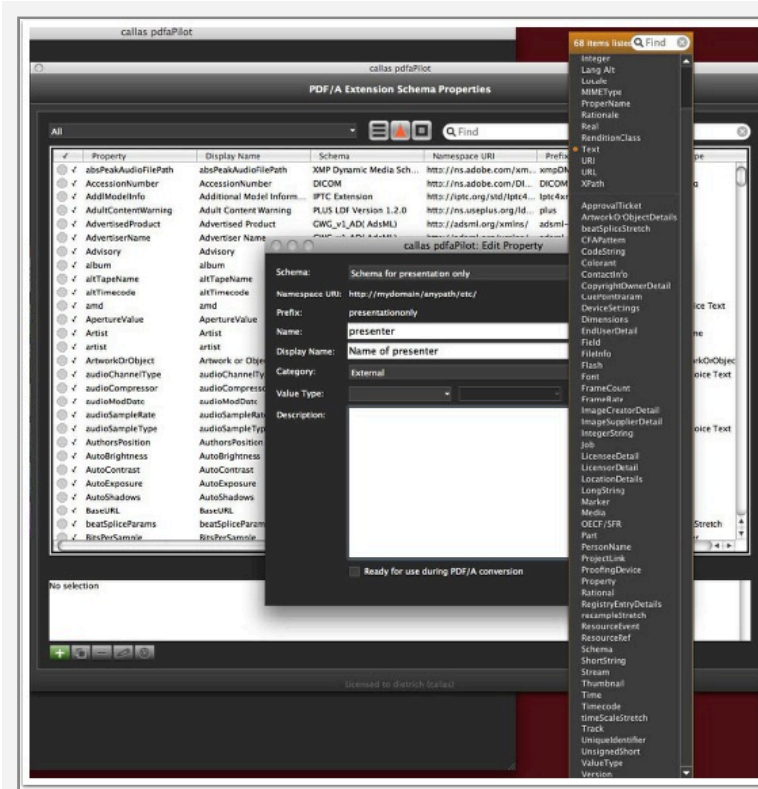


The value type is either a simple value type, an open or closed choice or a structured type. For a simple value type leave the first pop up empty:

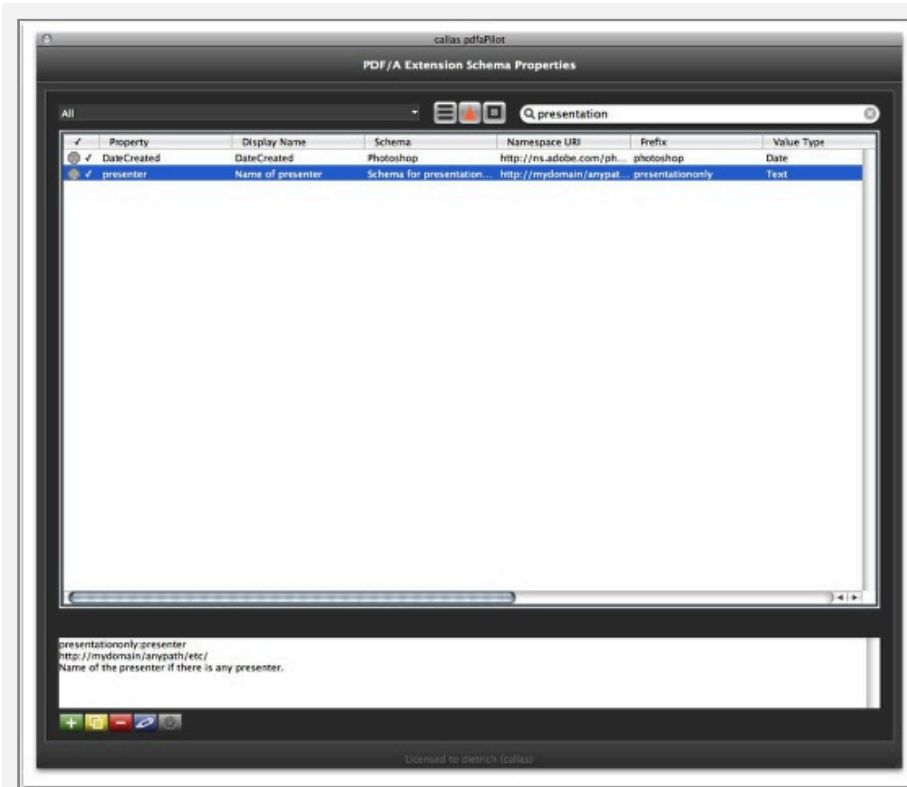


callas pdfaPilot lets you even define your own structured value types, .e.g in order to save machine data or an address (consisting of a name, a street, a city etc.). In order to define your own types you will have to use the right icon above the list of properties. Any type you will define there can be selected here. This is, however, not described in this document.

The last pop up is the actual type. In most cases "Text" is correct:



You can use any word of the Extension Schema or the property in order to filter the properties in the list:

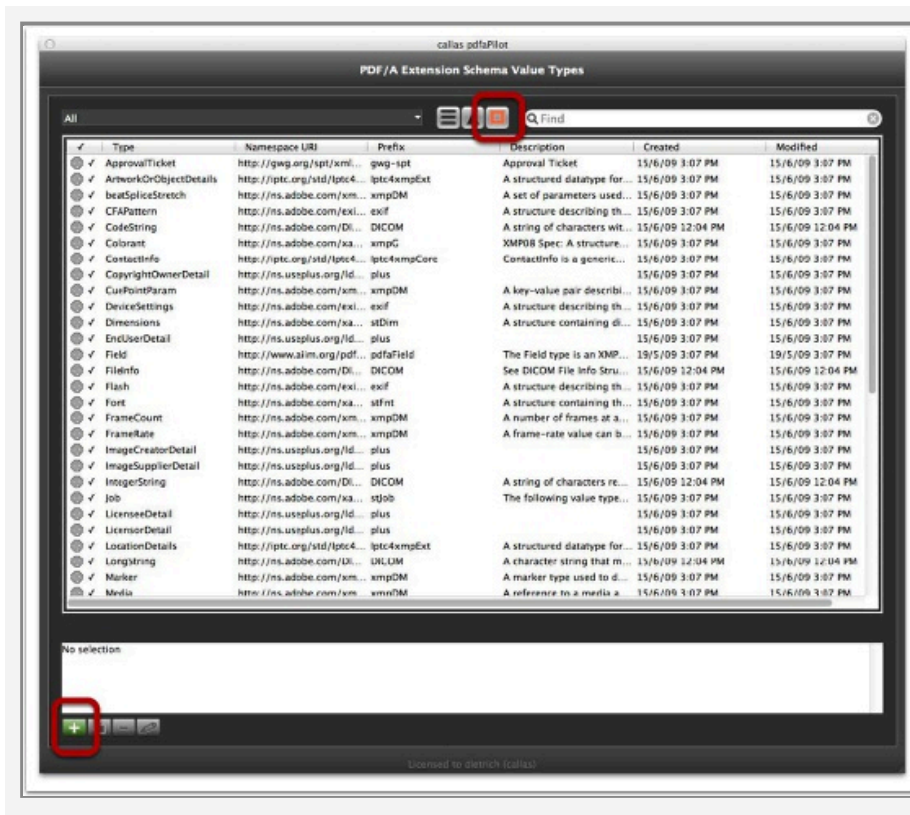


You can now use the Metadata fields in a PDF file and pdfaPilot will automatically embed the Extension Schema for all entries that show up in a PDF file when it is converted to PDF/A.

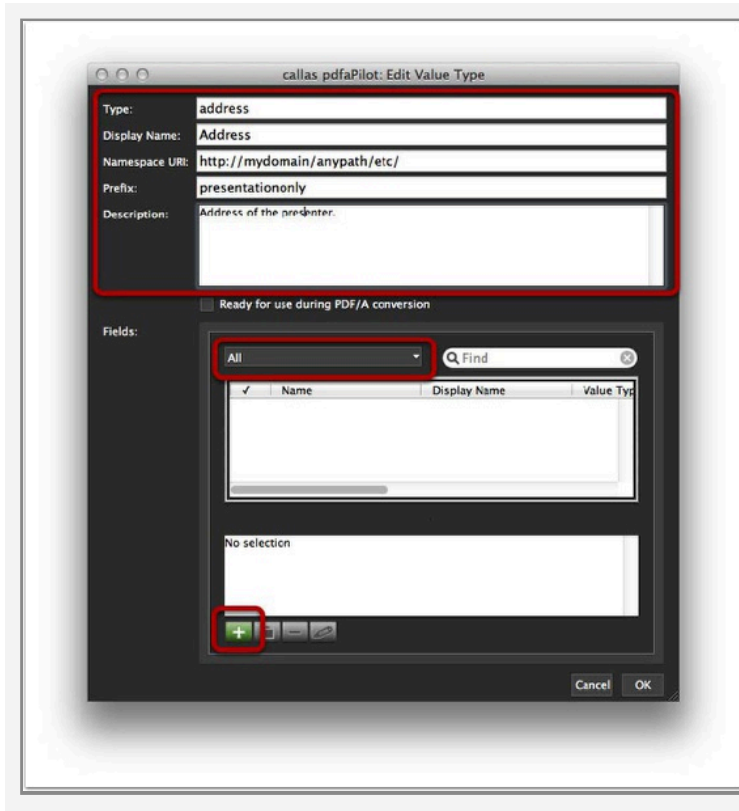
But you can even define your own data types or structures in XMP, read below how you can do that with pdfaPilot.

Defining data types or structures in XMP

Select the value type list in the Extension Schema editor and click on + in order to create a new value type:

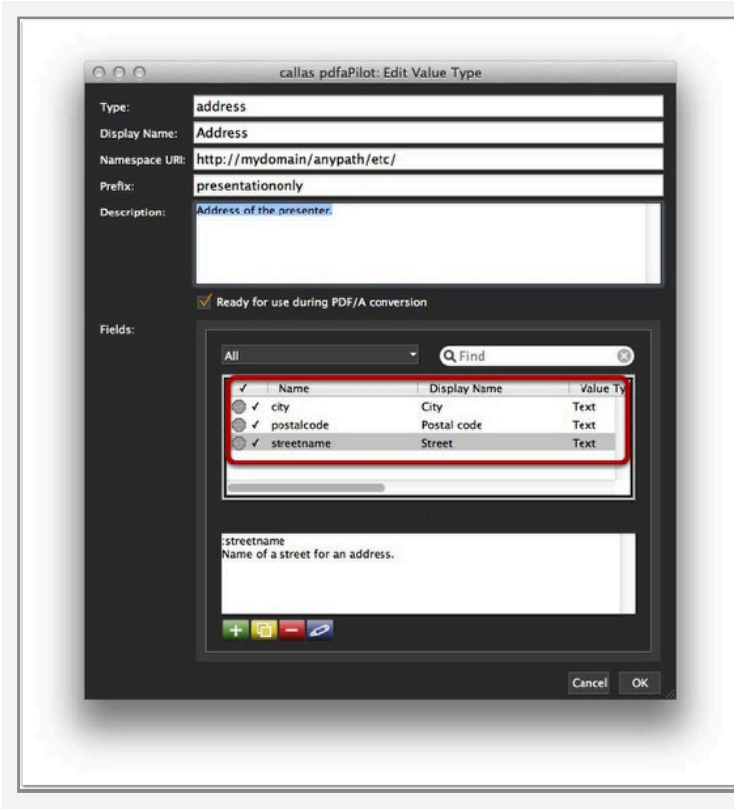


A new window opens that lets you define the value type properties:

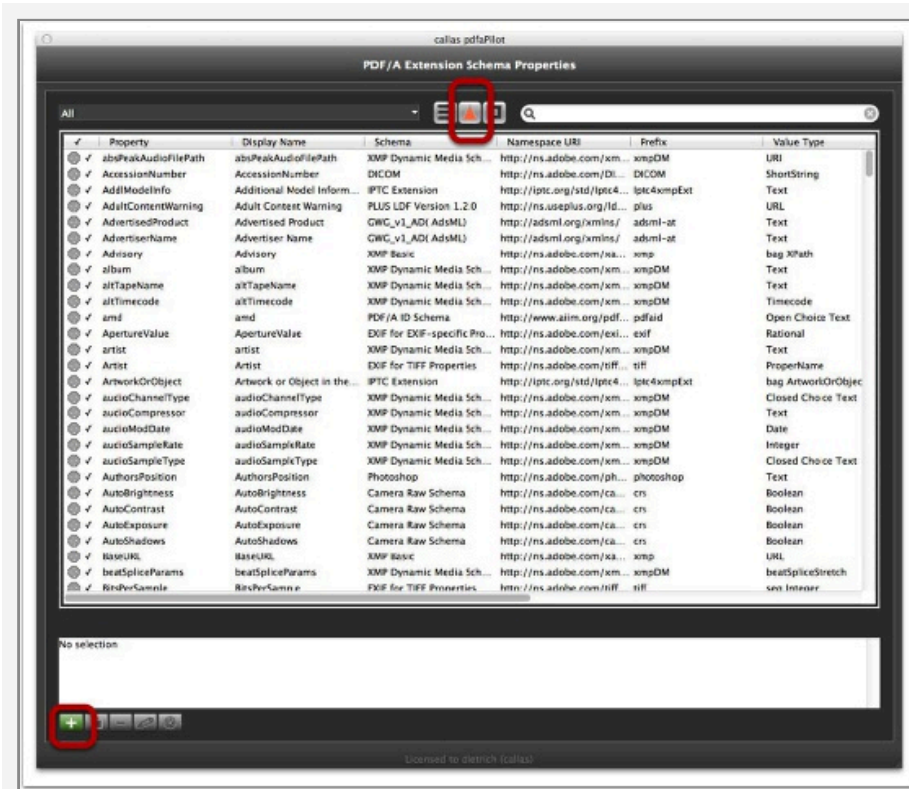


Fill out the first rows with proper values. Lets create a value type for the address of the presenter. The address should consist of three properties. Click on the + in order to add field to the address type. You may also use the Pop-Up in order to add an existing field.

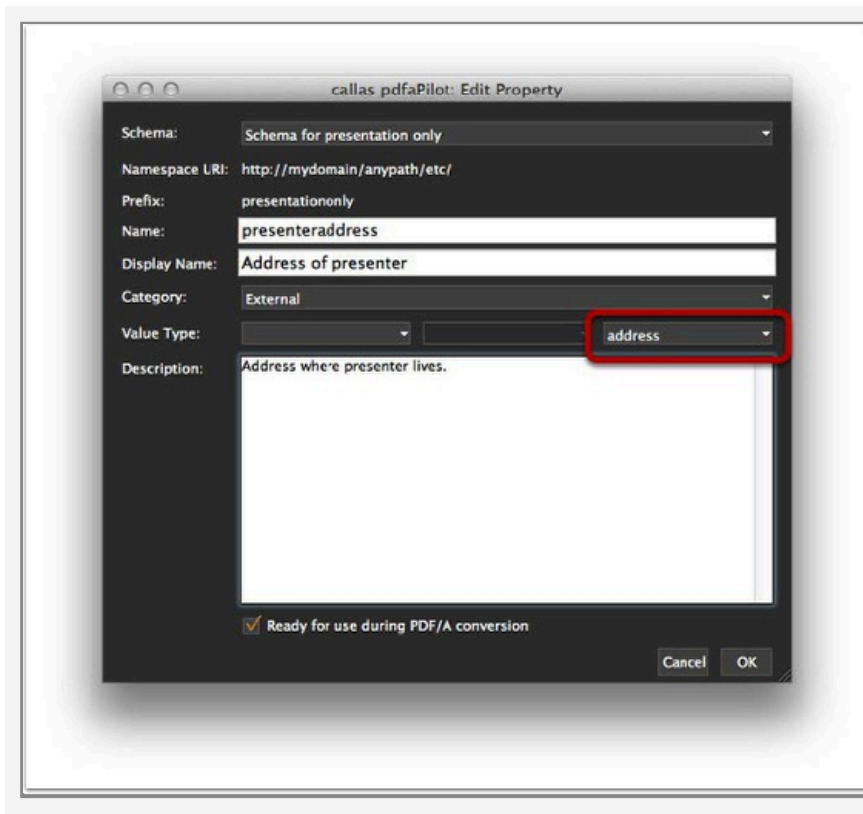
Enter the values for a street name. Use the + again in order to add more fields, for example, postal code and city:



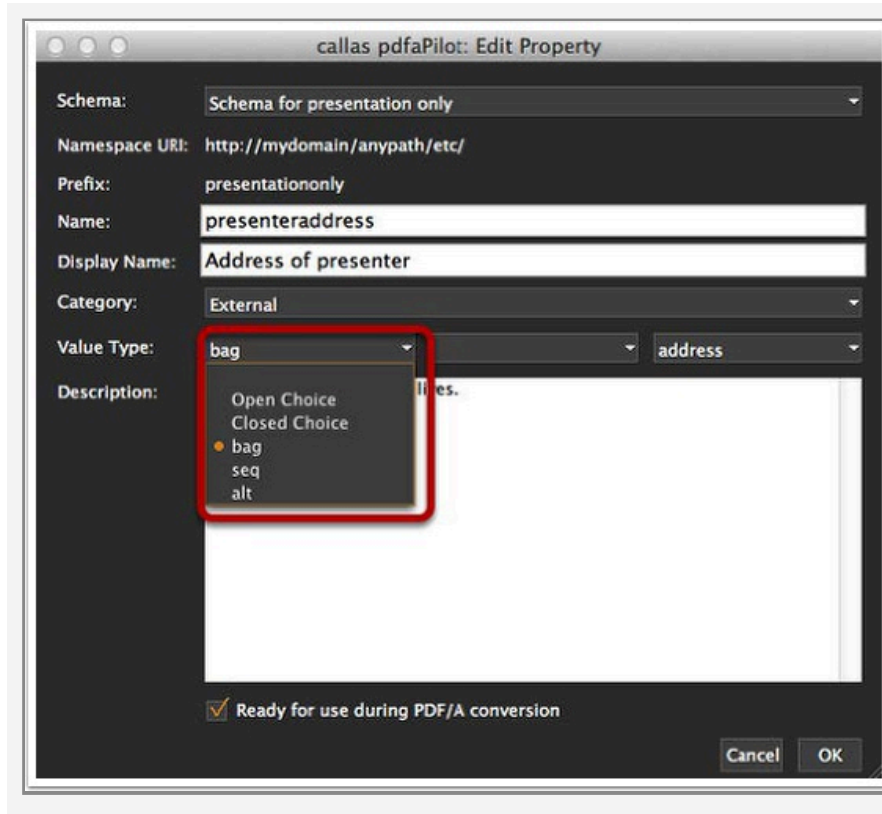
Now go back to the properties view and add a new property by clicking on +:



You may now use the new value type for a new property in the existing namespace:



You may even use a structure e.g. if a presenter could have more than just one address, in this case you should use bag:



Finding Extension Schemas

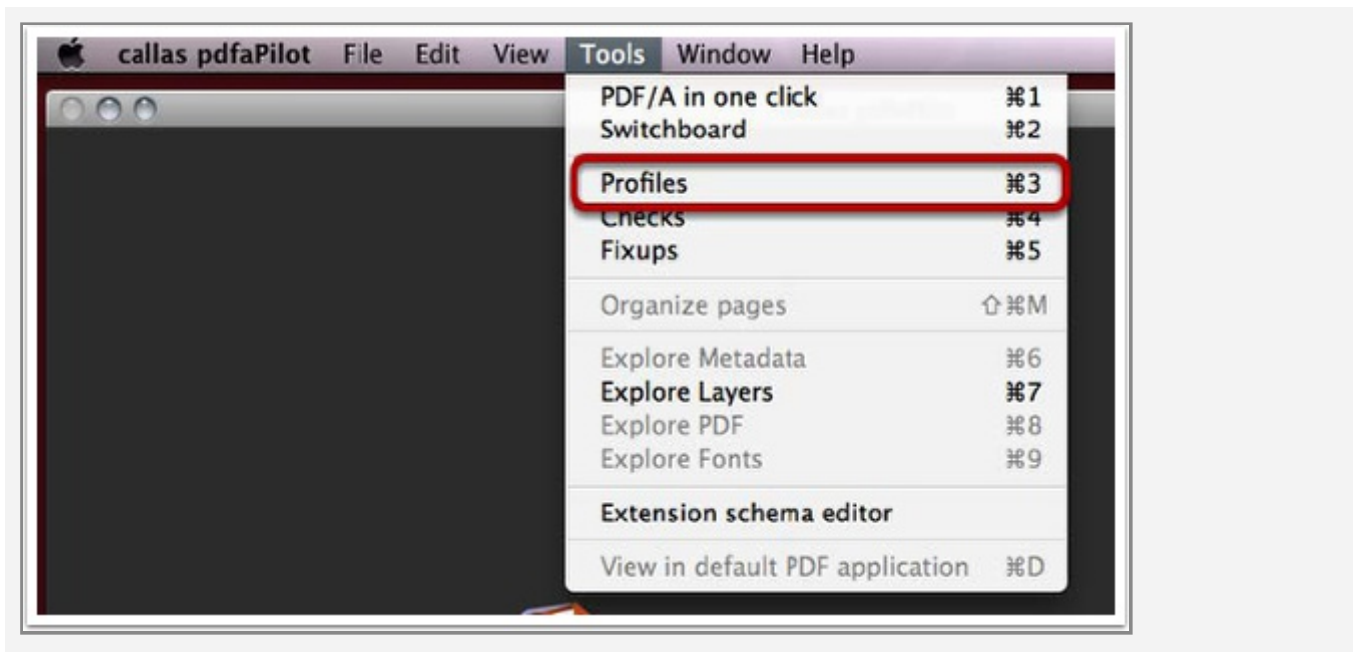
Created extension schemas are stored in your pdfaPilot user folder within "PDFExtSchema/Custom.pdfaschema". These can be used in the CLI/Server version by copying the file to the appropriate folder (e.g. ".callas software gmbh/callas pdfaPilot <version>/PDFExtSchema" on Unix).

5.2 Inserting XMP Metadata into PDFs using callas pdfaPilot Desktop

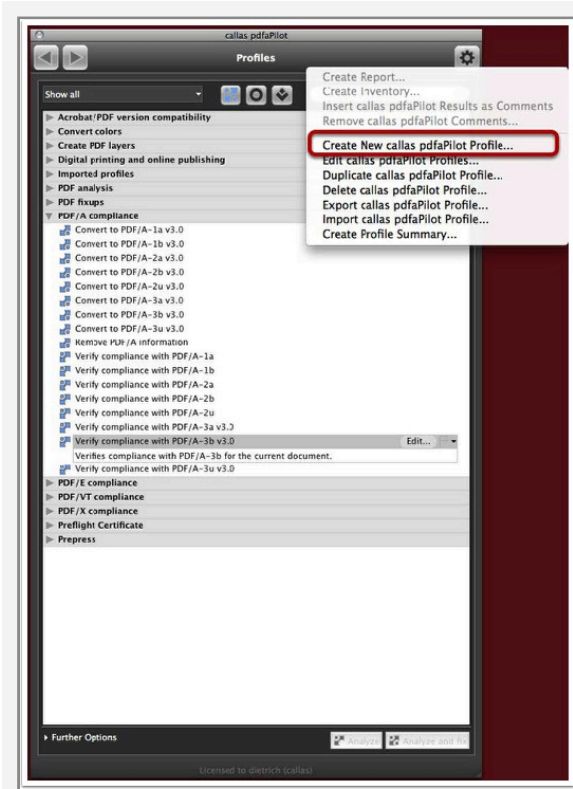
This page describes how callas pdfaPilot can be used in order to insert XMP Metadata into PDF files. It will make use of variables in order to dynamically adapt the values.

Insert XMP Metadata into PDF files

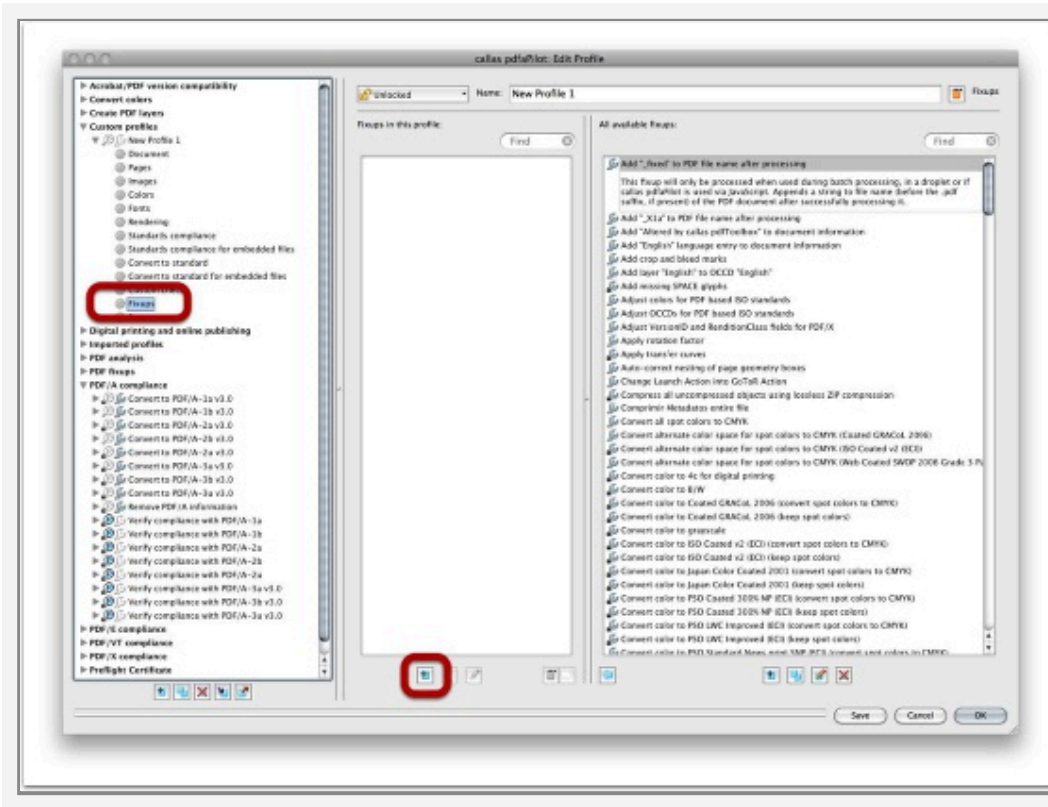
Open the profiles window in the callas pdfaPilot main menu:



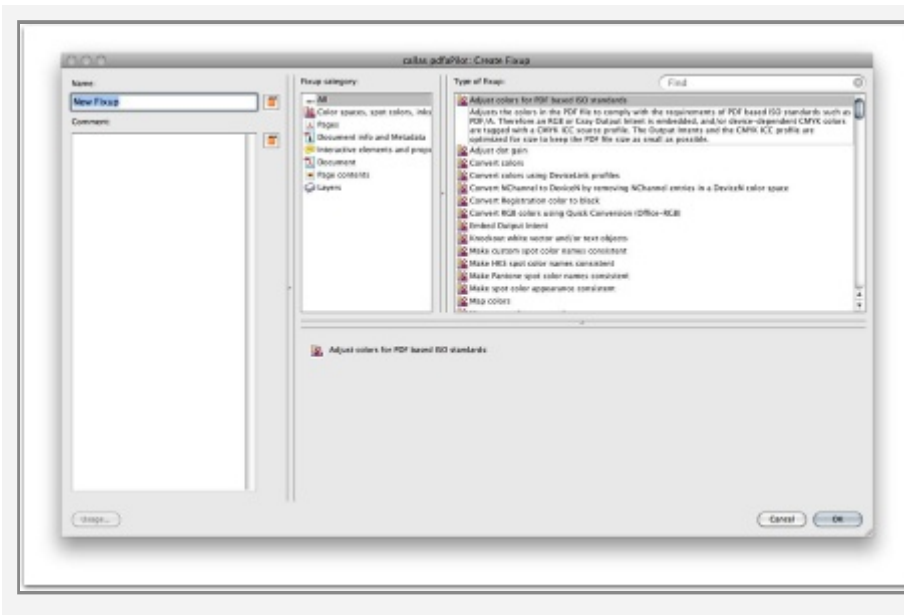
In the new window create a new profile:



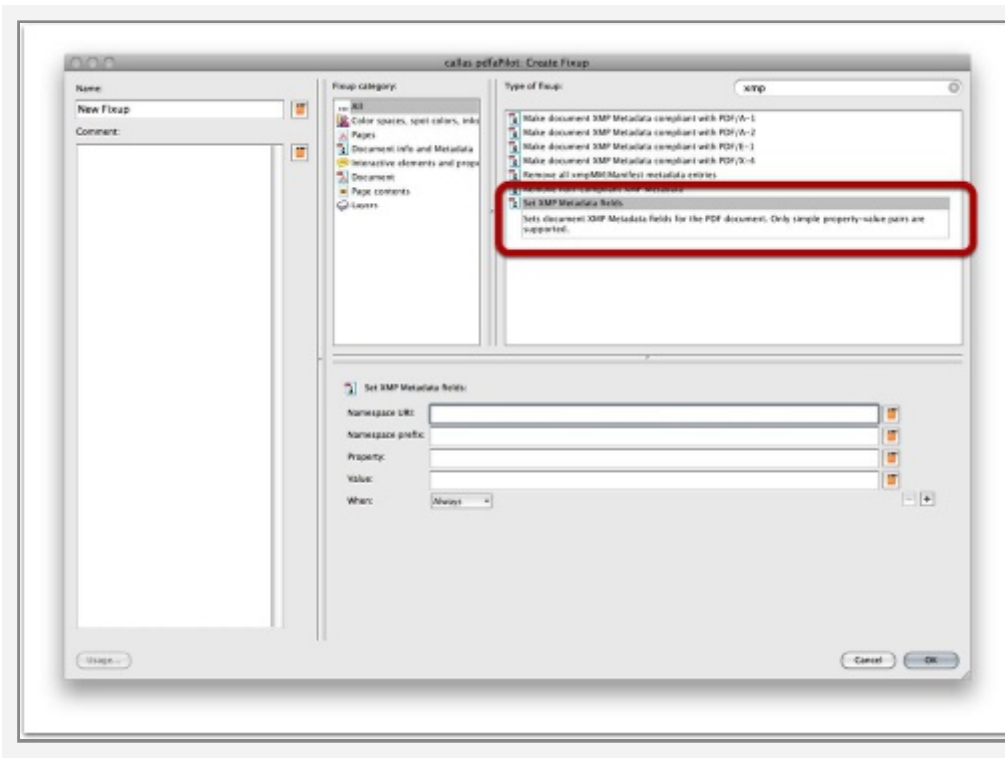
A new window opens. In the left column select Fixups and click on the blue icon in order to create a new fixup:



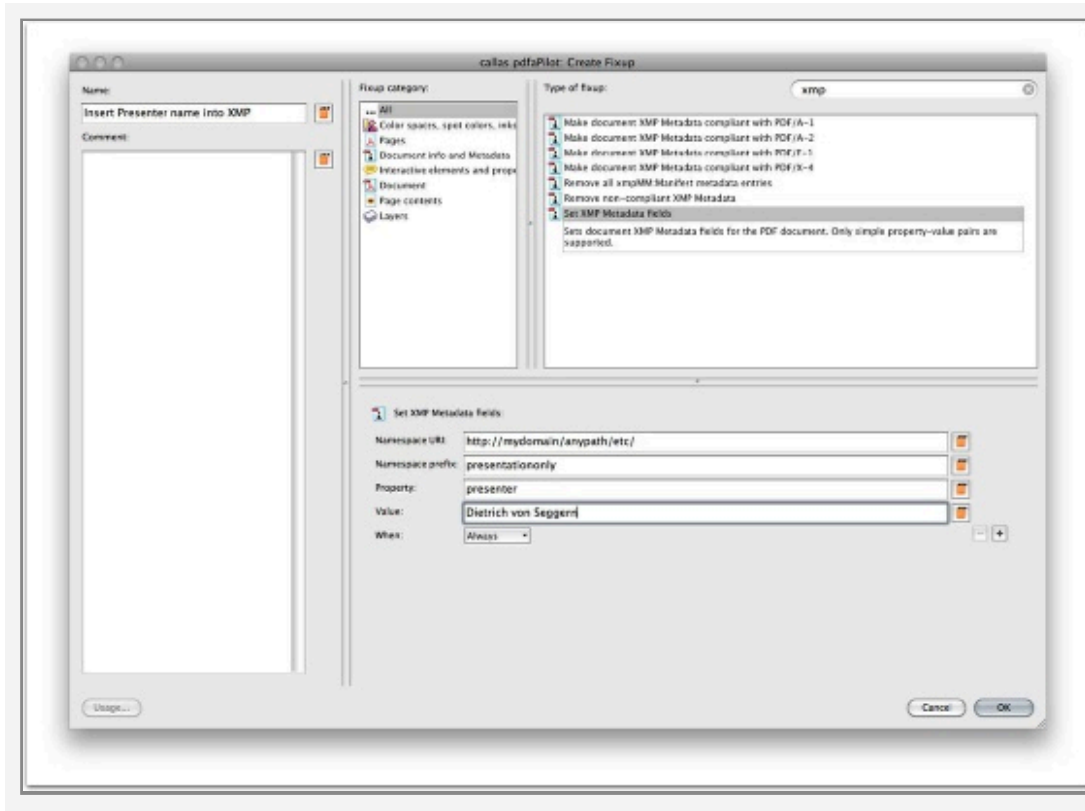
The fixup editor window opens:



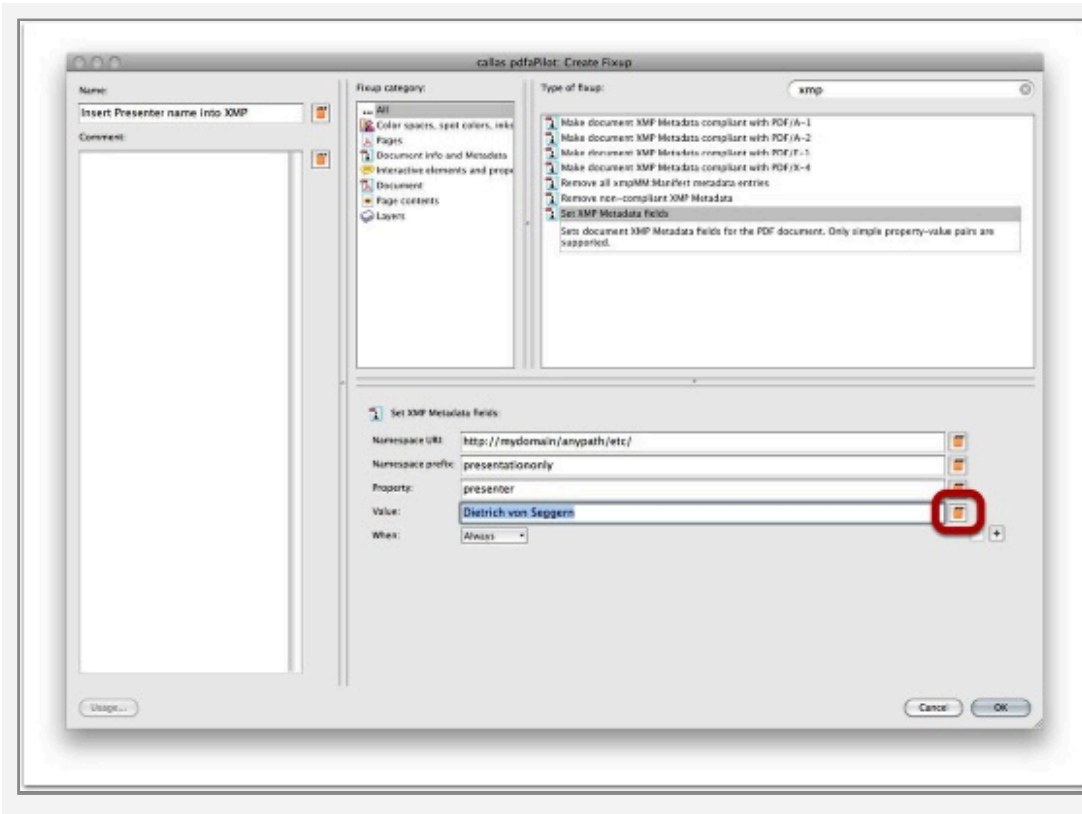
Enter "xmp" into the filter field, and select "Set XMP Metadata fields" from the filtered list:



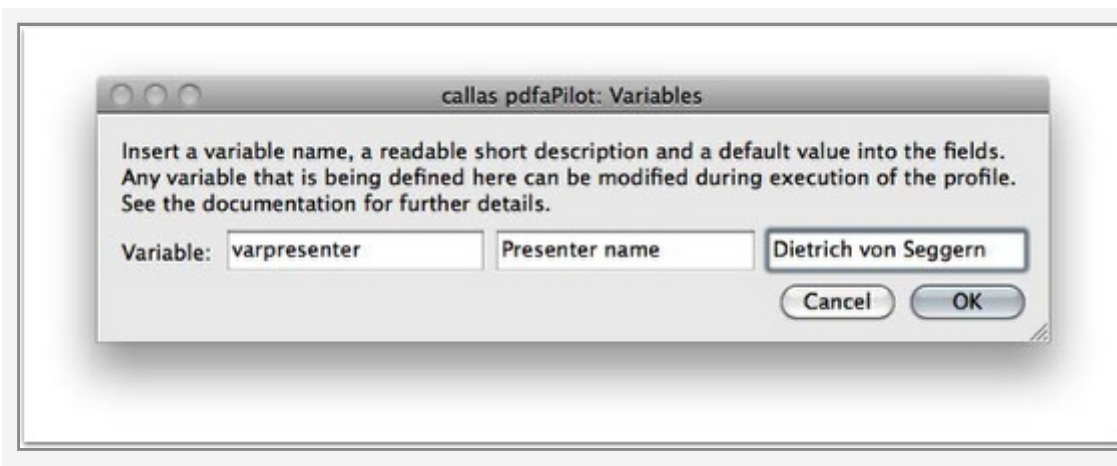
Enter a proper namespace URI, an appropriate namespace prefix, name and value for the XMP Metadata property:



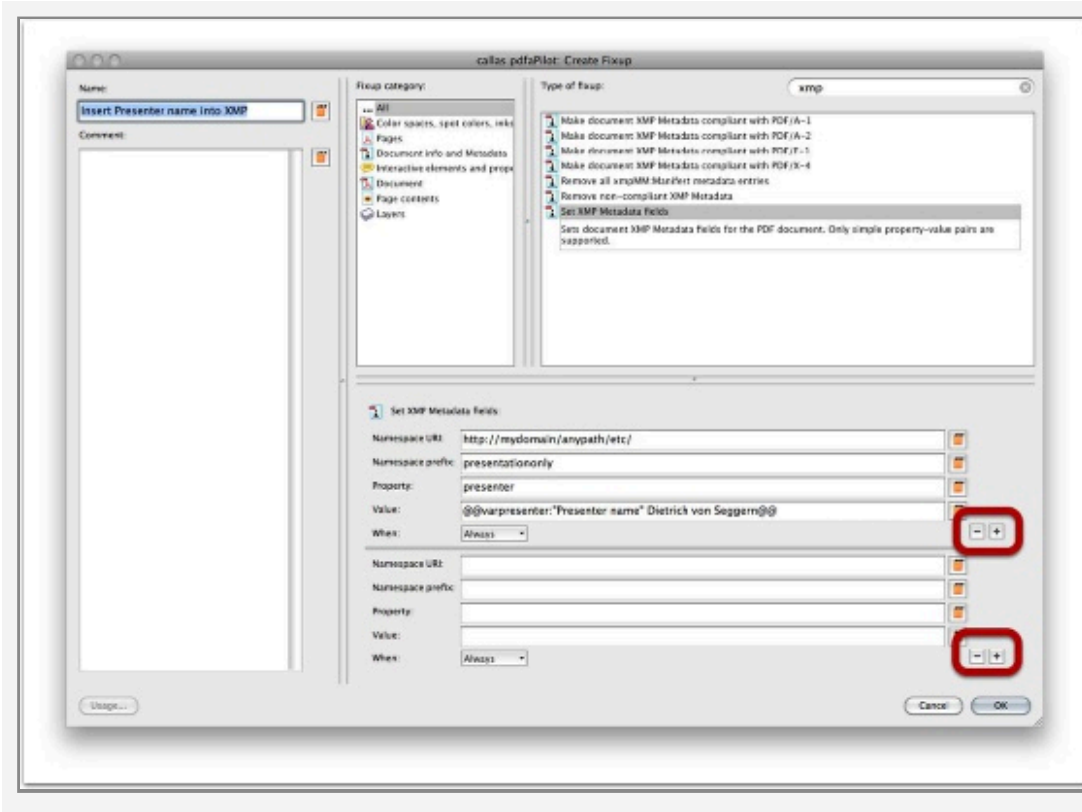
If you want to adapt the value whenever you are using the fix-up, use a variable instead of a fixed value. Click on the orange icon on the right hand side:



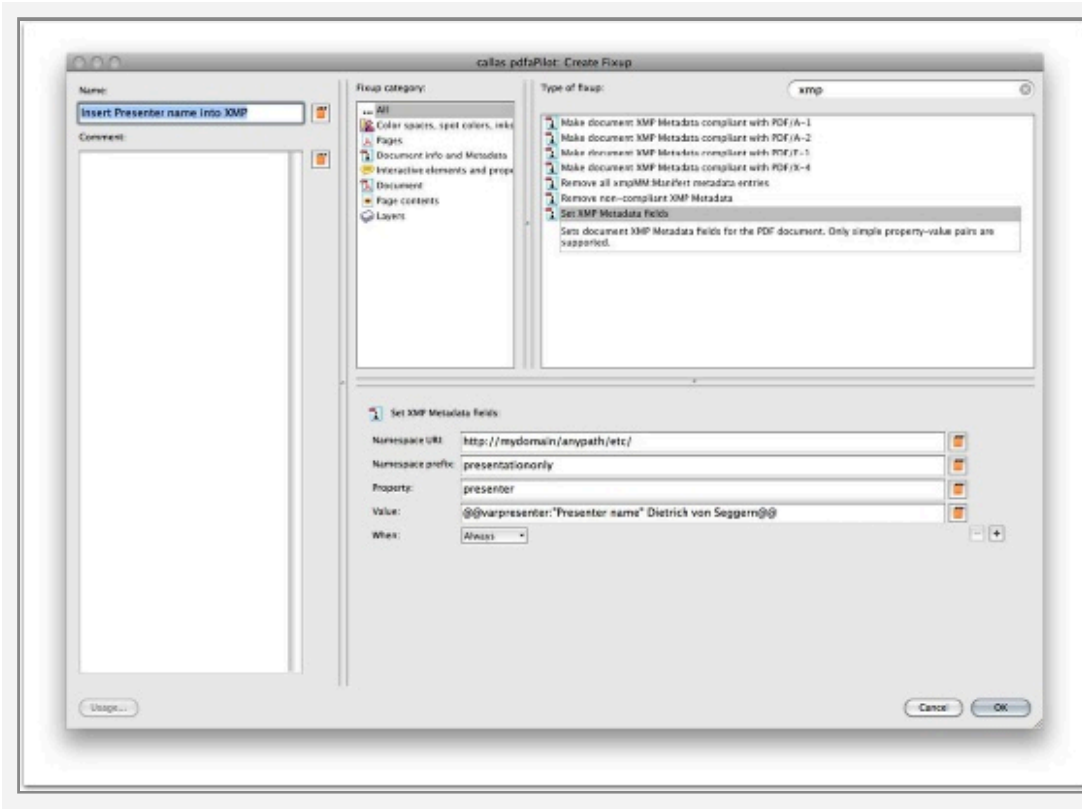
A new window opens that lets you define a variable. Give the variable a name, a display name and a default value for the variable and save by clicking OK. This name can be used in pdfaPilot CLI or pdfaPilot SDK in order to modify the values during runtime:



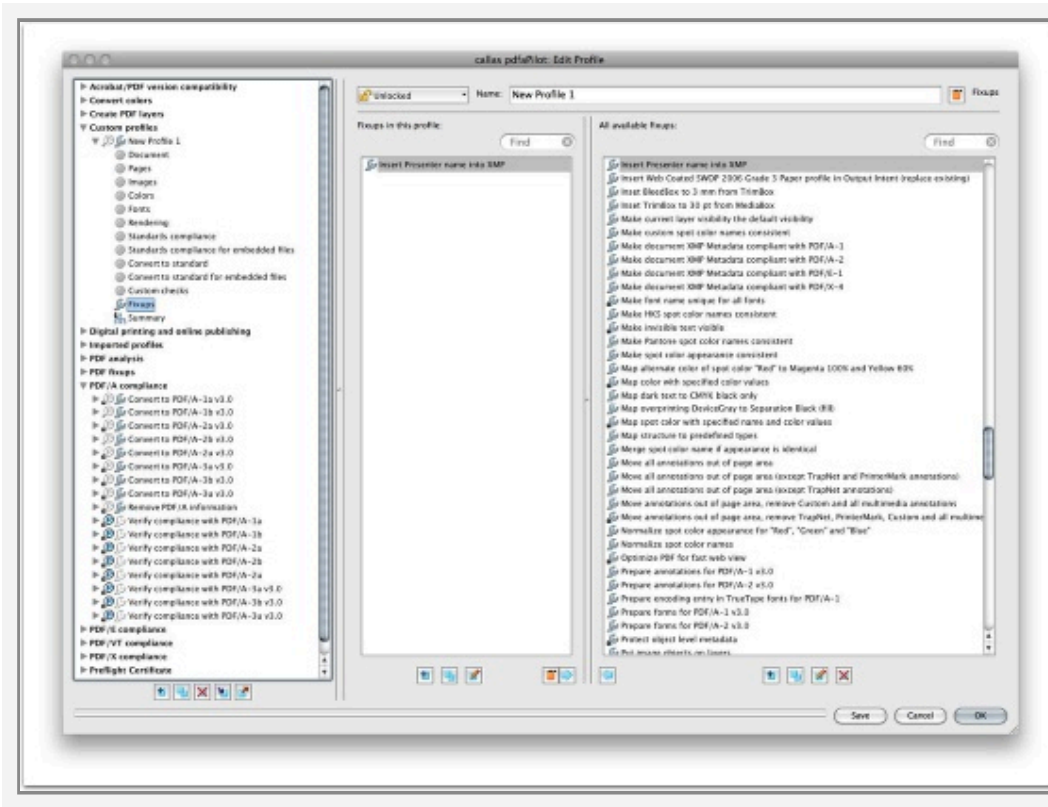
Now the variable data is inserted instead of a fixed value. You may use the [+] and [-] icons in order to define more or fewer properties:



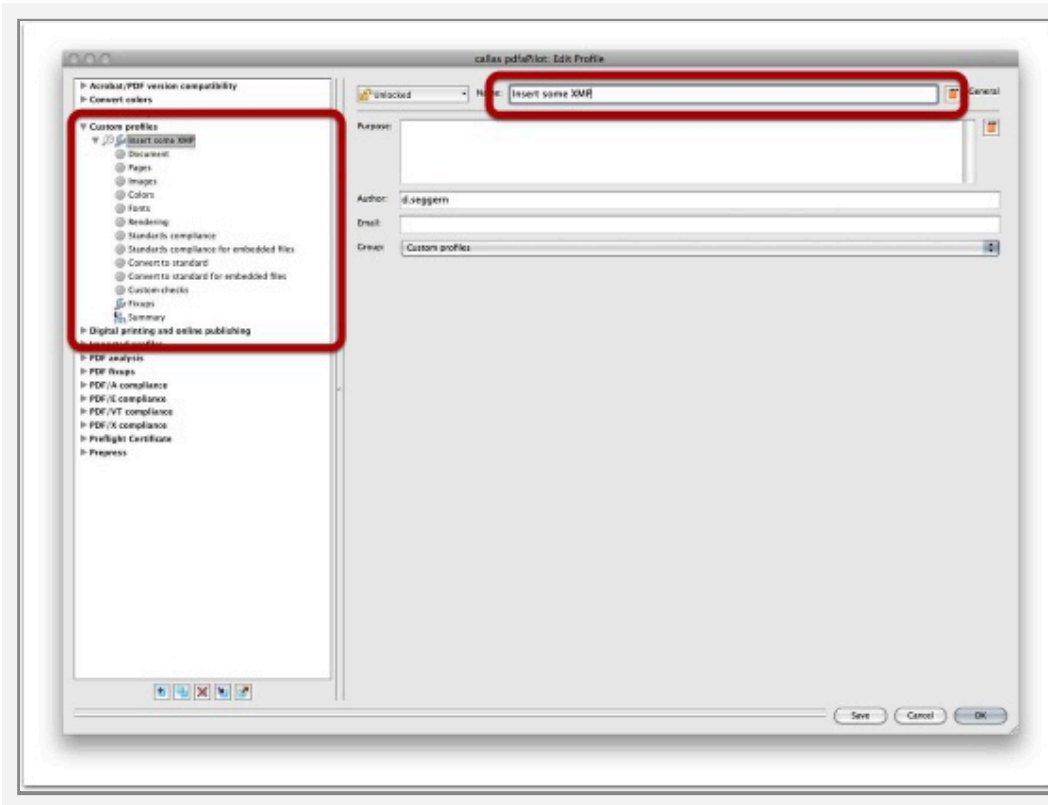
Define a name for the fixup and save by clicking OK:



The new fixup shows up in the profile editor:

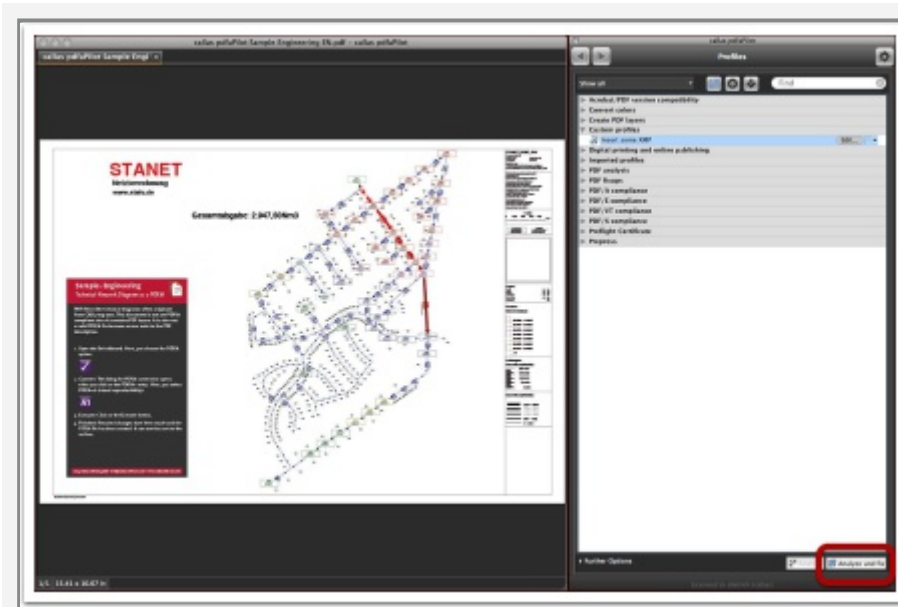


Define a name for the profile and save by clicking OK:

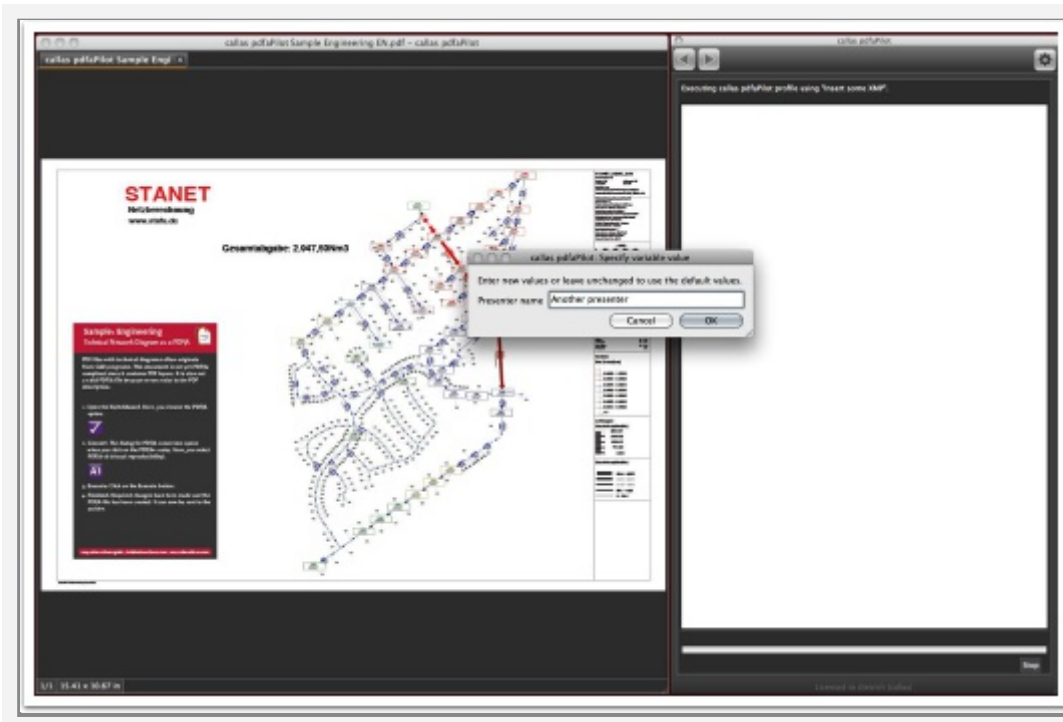


Executing the profile

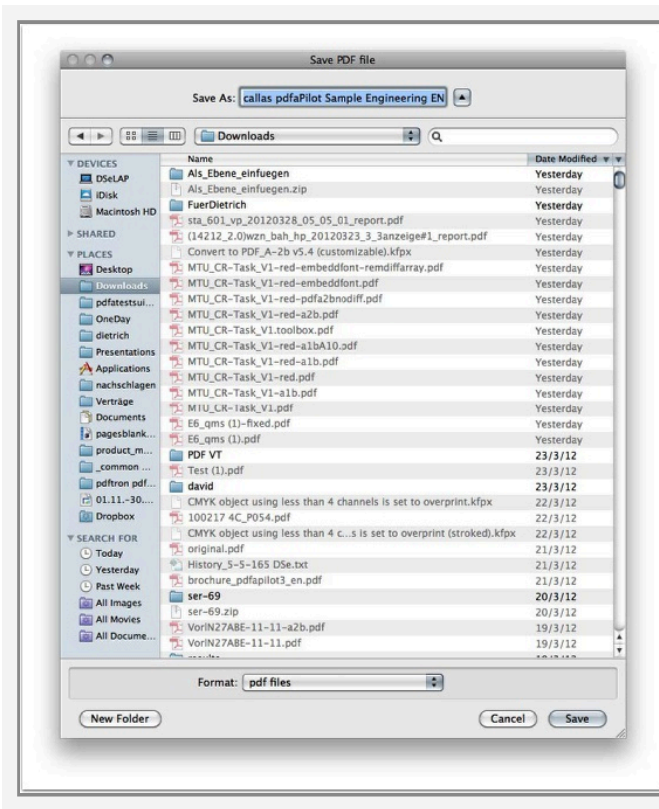
In order to apply the new fixup open any PDF and execute the profile:



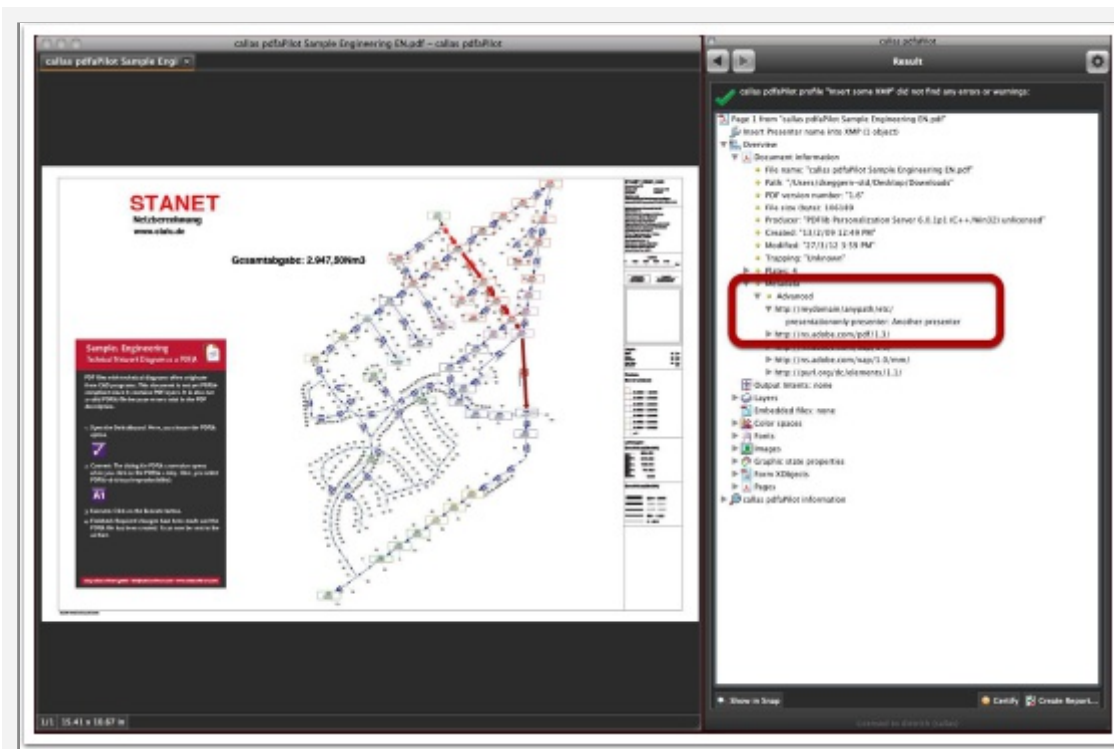
A dialog opens that lets you define the value for the property. You can overwrite the default value:



Save the new PDF:



In the result view you can see that the new property and the adapted value has been inserted:



6. callas pdfaPilot CLI

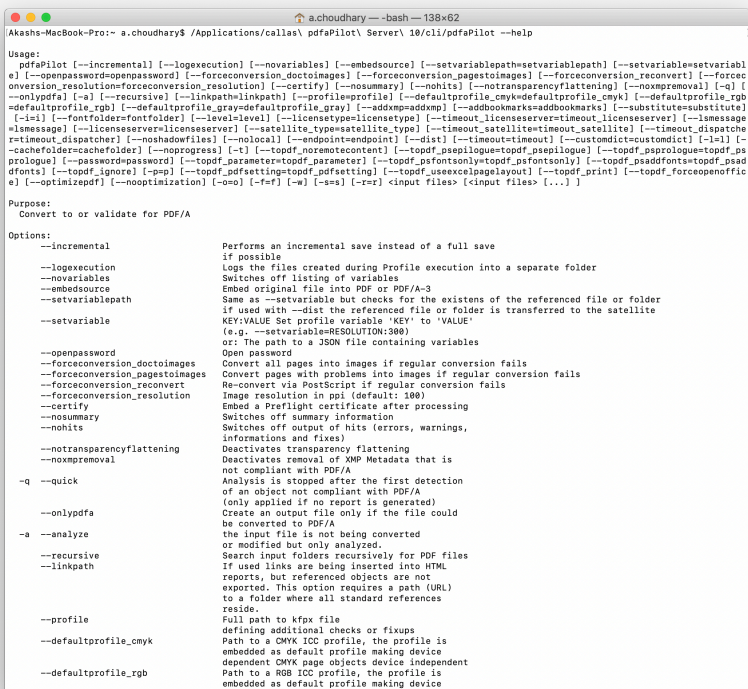
6.1 Introduction to pdfaPilot CLI

Install and activate

To get going with pdfaPilot CLI, the first thing needed is installation and activation. This is explained step by step in the [next article](#).

Usage

The easiest way to find out what you can do with the command-line is to run the `--help` command.



```

Usage:
  pdfaPilot [--incremental] [--logexecution] [--novariables] [--embedsource] [--setvariablepath=setvariablepath] [--setvariable=setvariable]
  a) [--openpassword=openpassword] [--forceconversion_doctoimages] [--forceconversion_pagestoimages] [--forceconversion_reconvert] [--force
  onversion_resolution=forceconversion_resolution] [--certify] [--nosummary] [--nohits] [--notransparencyflattening] [--nompremoval] [-q] [
  --onlypdfa] [-a] [--recursive] [--linkpath=linkpath] [--profile=profile] [--defaultprofile_cmyk=defaultprofile_cmyk] [--defaultprofile_rgb
  =defaultprofile_rgb] [--defaultprofile_gray=defaultprofile_gray] [--addbookmark] [--addbookmark=bookmark] [--substitute=substitute]
  [-i=1] [--fontfolder=fontfolder] [--level=level] [--licensetype=licensetype] [--timeout_licenseserver=timeout_licenseserver] [--lsmessag
  e=lsmessag] [--licenseserver=licenseserver] [--satellite_type=satellite_type] [--timeout_satellite=timeout_satellite] [--timeout_dispatche
  restimout_dispatcher] [--noshadowfiles] [--nocall] [--endpoint=endpoint] [--dist] [--timeout=timeout] [--customdict=customdict] [-i=1] [
  --cachefolder=cachefolder] [--noprogess] [-t] [--topdf_noremotecontent] [--topdf_psepiogue=topdf_psepiogue] [--topdf_pspilogue=topdf_p
  spilogue] [--password=password] [--topdf_parameter=topdf_parameter] [--topdf_psfontsonly=topdf_psfontsonly] [--topdf_psdadddfontsonly=topdf_psdad
  dfontsonly] [--topdf_ignore] [-e=1] [--topdf_pdfsetting=topdf_pdfsetting] [--topdf_useonlylayout] [--topdf_print] [--topdf_forceopenoffic
  e] [--optimizepdf] [--nooptimization] [-o=0] [-f=f] [-w] [-s=s] [-r=r] <input files> [<input files> [...]]

Purpose:
  Convert to or validate for PDF/A

Options:
  --incremental          Performs an incremental save instead of a full save
                        if possible
  --logexecution         Logs the files created during Profile execution into a separate folder
  --novariables          Switches off listing of variables
  --embedsource          Embed original file into PDF or PDF/A-3
  --setvariablepath      Same as --setvariable but checks for the existence of the referenced file or folder
  --setvariable          KEY=VALUE Set profile variable 'KEY' to 'VALUE'
                        (e.g. --setvariable=RESOLUTION:300)
                        or: The path to a JSON file containing variables
  --openpassword         Open password
  --forceconversion_doctoimages  Convert all pages into images if regular conversion fails
  --forceconversion_pagestoimages  Convert pages with problems into images if regular conversion fails
  --forceconversion_reconvert  Re-convert via PostScript if regular conversion fails
  --forceconversion_resolution  Image resolution in ppi (default: 300)
  --certify              Embed a Preflight certificate after processing
  --nosummary           Switches off summary information
  --nohits              Switches off output of hits (errors, warnings,
                        informations and fixes)
  --notransparencyflattening  Deactivates transparency flattening
  --nompremoval         Deactivates removal of XMP Metadata that is
                        not compliant with PDF/A
  -q --quick            Analysis is stopped after the first detection
                        of an object not compliant with PDF/A
                        (only applied if no report is generated)
  --onlypdfa            Create an output file only if the file could
                        be converted to PDF/A
  -a --analyze          the input file is not being converted
                        or modified but only analyzed.
  --recursive           Search input folders recursively for PDF files
  --linkpath            If used links are being inserted into HTML
                        reports, but referenced objects are not
                        exported, this option requires a path (URL)
                        to a folder where all standard references
                        reside.
  --profile             Full path to kfx file
                        defining additional checks or fixups
  --defaultprofile_cmyk  Path to a CMYK ICC profile, the profile is
                        embedded as default profile making device
                        dependent CMYK page objects device independent
  --defaultprofile_rgb  Path to a RGB ICC profile, the profile is
                        embedded as default profile making device
  
```

This will provide a long help text with

1. Available parameters for validating with OR converting to PDF/A (as shown in the screenshot above)
2. Various "commands" usable with pdfaPilot CLI (not visible in the screenshot above but explained below)

1. Validating with or converting to PDF/A using pdfaPilot CLI

In the most basic form pdfaPilot needs only input file or files to be validated or converted to PDF/A. Additional to this, there are other parameters as shown in the screenshot above. For example:

```
--analyze
```

which would only analyze the input file for validation but not convert the file to PDF/A. Or multiple parameters like (this would analyse the input PDF for PDF/A-3u conformance level):

```
./pdfaPilot --analyze --level=3u <PDF>
```

2. Run commands directly (without Profiles)

Another way to process PDF files using pdfaPilot CLI is **running commands**. pdfaPilot --help will also furnish the list of commands that you can execute (as shown below):

```

Commands:
pdfaPilot --command [options]

-h --help show help
--version show version information
--status Displays status information (Serialization, Error Codes)
-k --keycode Request activation code
--activate Activate license
--deactivate Deactivate license
--emptyfontcache Removes all font files from font cache folder
--emptyprofilecache Removes all profile files from profile cache folder
--uncertify Remove a Preflight certificate if present
--booklet Creates booklet for printing
--quickfix Performs a QuickFix
--nup Imposes by N-Up
--fillpage Imposes by filling up a page
--impose Imposes the PDF based on rules defined in run list
and sheet setup
--mergeimpose Imposes based on sheet template and runlist
--slice Slices in two files by object type
--readerspreads Combines two pages to one spread
--splithalf Creates single pages from spread
--steprepeat Imposes by Step and Repeat
--splitpdf Splits multipage documents into smaller packages
--mergepdf Merges PDF files
--splitmergepdf Splits multipage documents into smaller packages and merge them to one document
--duplicatepage Duplicate pages of the PDF
--splitatmark Splits multipage documents into smaller packages based on a Profile with a single, page-based Check
--presentation Prepares for presentation in Acrobat
--handout Creates a handout from a PDF presentation
--passpartout Creates a passe partout
--lighttable Positions pages on a virtual light table
--overlay Places the chosen content on top of the input PDF
--createeps Converts the PDF into EPS
--createps Converts the PDF into PostScript
--saveasimg Renders an image per page preserving the page
aspect ratio
--extracttext Extract text from PDF
--extractcontent Extract content of PDF
--extractimages Extract images from PDF
--redistill Recreates the PDF via PostScript, prepares for use with older equipment (RIPs)
--convertcolors Performs a color conversion as defined in
the configuration files
--extracticcprofiles Extracts ICC profiles
--enumeratelayers Creates layers with respect to names of fonts,
spot colors or ICC profiles
--importslayer Imports a PDF file and puts the content on a layer
--splitlayers Splits layers/layer views into single PDFs
with just the content of the layers/layer views
--extractxmpmetadata Extracts XMP Metadata from the PDF into a XML file
--list Lists all installed sheetconfigs and runlists
--listfonts Lists all font names that can be used for the 'Set TextFont' command of imposition
--quickpdfinfo Writes some basic PDF information to output
--createcustomdict Creates a XML template for translation
--visualizer Create a visualizer report
--compare Compare two documents and creates a report
--topdf Converts supported non-PDF files to PDF
--collection Creates a collection from the files within a folder
--securepdf Secures the PDF with a password
--createepub Converts a tagged PDF into EPUB
--createhtml Converts a tagged PDF into HTML

```

Once you find the command you want to work with, drill-down; for example, to know how to save images from pdfaPilot, issue the command:

```
./pdfaPilot --help saveasimg
```

Get in touch

If some necessary information is not provided by this manual or if there are any questions or feedback please contact our support line by using the "[Contact Support](#)" form.

Alternatively, you can also send an e-mail to support@callas-software.com.

6.2 Installation and activation of pdfaPilot Server/CLI

pdfaPilot CLI (command line interface) may be used to analyse files or to analyse and convert to PDF/A (if possible). For activating the SDK (software developer kit), please visit [this page](#).

System requirements

The command line version of pdfaPilot is available for Windows, Mac and Linux.

You can check the currently supported version of these operating systems on the following website:

- [System requirements](#)

You can easily test if pdfaPilot CLI is working on your system: Just type pdfaPilot --help in the terminal.

There are 64 bit versions of pdfaPilot CLI available for MacOS, Windows and Linux. pdfaPilot CLI does also run on 64 bit systems if the required 32 bit compatibility packages are available.

at least 2 GB RAM

at least 10 GB of free Disk Space

Hardware recommendations

pdfaPilot does not have extensive hardware requirements, so a decent computer should usually satisfy for processing PDF files. Concrete requirements depend very much on the complexity of the files and the operations to be performed. Some general recommendations are:

- For any PDF processes (checking and/or fixing) information about all PDF objects has to be available to the engine. The size of the RAM memory needs to be big enough for all such objects. If possible the size of the RAM should be not too small, especially for complex PDF files with many pages.

- Any operations will include reading and writing to "disc". Faster storage devices therefore speed up the process and processing will take advantage of using SSDs or similar.
- pdfaPilot allows for parallel processing. This will, however, only result in faster processing if the number of cores is higher than the number of parallel processes. It is recommended to have at least one additional core for system processes.

Examples:

Entry-Level:

Dual-Core Processor
4GB RAM
160 GB SATA hard disc

Mid-Range:

Quad-Core Processor
8GB RAM
500 GB SATA SSD

Installing the software

Installer

You can download the latest version of pdfaPilot Server/CLI from our website by requesting a trial:

1. pdfaPilot Server:
[pdfapilotserver](#)
2. pdfaPilot CLI:
[pdfapilotcli](#)

You can also [register on our website](#) for easy access to all pdfaPilot download links. Once registered and logged in, you can access the list of download links:

1. pdfaPilot Server:
[pdfapilotserver](#)
2. pdfaPilot CLI:
[pdfapilotcli](#)

MacOS/Windows

To install the software start the pdfaPilot Server installer. The

installation program will then take you through the necessary steps.

Linux

Extract all files from the archive to a destination folder of your choice.

Additional information is provided in <pdfaPilot CLI directory>/ReadMe.txt

Request activation

Before callas pdfaPilot CLI can be used, the software has to be activated. Open a terminal window and change to your pdfaPilot CLI installation directory.

Request an activation code for full version

If you have bought a license for callas pdfaPilot Server/CLI you can activate the full version with your license key that can be found on your License.pdf:

```
pdfaPilot --keycode [--aws] <name> <company> <licenseCode>
```

Request an activation code for trial version

If you just want to try the product, you can request an activation code for a free 14-day trial:

```
pdfaPilot --keycode [--aws] <name> <company> trial
```

Example:


```
pdfaPilot --keycode "Demo Name" "callas software" trial
```

Parameters

name	Name of licensee (e.g. "Registered User")
company	Name of company (e.g. "User's company")

licencecode	Licence key obtained from the registration card or the License.pdf provided by callas or the reseller.
trial	To make a request for a trial version, please use the keyword " <i>trial</i> " (for a pdfaPilot Server/CLI trial version)
aws	For installation on Amazon Web Services (using Windows, Linux 32bit and 64bit)

The textual output of `--keycode` has to be send via e-mail to the e-mail address named in the text in order to receive an activation code from the registration server.

-  To make a request for a trial version, please use the keyword "*trial*" (for a pdfaPilot trial version) for this parameter

Activating pdfaPilot CLI


After receiving the automatic e-mail reply from the activation server, save the attached 'activation file' to the file system. Then use the following command:

```
pdfaPilot --activate <activation file>
```

If no response is received or in the event of an error, please contact support@callassoftware.com to determine the exact cause.

Parameters

activation file	Full path to Activation PDF
-----------------	-----------------------------

-  It is necessary to activate the received license file to get a permanent valid license file.

The Activation.pdf (or the content of the e-mail) can only be used for activation **for 48 hours**.

After this timeframe, a new Activation.pdf has to be requested from the activation server.

The activated license file will be stored in the user-preferences when the normal activation (command above) is used.

To create an activated license file at a custom location, just use the following command:

```
pdfaPilot --activate <licence file> -o=<path to result folder/License.txt>
```

pdfaPilot CLI is searching for the license file at various folders:

- user-preferences-folder of actual user
- next to the pdfaPilot CLI binary
- cachefolder (if set)
- user-preferences-folder for all users (shared)

When using UNIX-based-systems the environment variable `CALLAS_SYSTEM_PREFERENCES` the path of the standard `/usr/share/callas software/callas pdfaPilot CLI` can be changed:

```
CALLAS_SYSTEM_PREFERENCES=tmp
```

would result in the searchpath: `/tmp/callas software/callas pdfaPilot CLI`

It is highly recommended to use the option `--cachefolder` instead.

Time-limited trial version

After requesting and entering a trial activation code, pdfaPilot CLI can be tested without any restrictions. When the evaluation period has expired, processing PDF files will no longer be possible until you request and enter a new activation code.

Activation using the Standalone application

Using Windows and MacOS, also the activation dialog of the Standalone Application can be used for requesting an activation as well as using a Keycode or just for a trial version. Also the activation itself can be done using that Interface.

All activations (for Desktop and Server/CLI) can be done using this dialog.

Deactivate pdfaPilot using the CLI

As the activation (and the resulting license file) is bound to the hardware. It is necessary to deactivate a license on one machine before an activation takes place on the new machine.

```
pdfaPilot --deactivate <activation code>
```

activation code	Unique identifier for each license
-----------------	------------------------------------

The respective license will be removed from the system

To complete the deactivation, the output of the command has to be sent manually to the activation server by e-mail.

The activation code for all license are listed using the status command:

```
pdfaPilot --status
```

Deactivation using the Standalone application

Similar to the deactivation using the CLI, the Desktop on Windows and MacOS can be used for deactivation.

The selected license will be removed from the system as well and the necessary e-mail to the activation server will be sent automatically.

6.3 Hints and trouble shooting & Displaying program information

Displaying program information

Display program version

```
pdfaPilot --version
```

will display the currently used version of pdfaPilot CLI.

Display usage information

```
pdfaPilot --help
```

will give you a complete overview about all available commands for processing.

```
pdfaPilot --help <command>
```

will give you an overview about all available options for the command.

Display status

```
pdfaPilot --status
```

will inform you about the current license state as well as the possible return and reason codes (see "Results").

Hints and troubleshooting

Ensure sufficient free disk space

To ensure stable processing, it is recommended to have at least 4 times of the input file size of processed files available for intermediate file system storage (e.g. /tmp on Unix and similar on other systems).

Avoid stopping workflows on Windows

On Windows, you can prevent your workflow from stopping in case of a pdfaPilot CLI crash by setting the following registry entry:

```
HKEY_LOCAL_MACHINE\  
SYSTEM\  
CurrentControlSet\  
Control\  
Windows\  
ErrorMode
```

If **ErrorMode** is set to "2", crash dialogs will be suppressed. For further details, see: <http://support.microsoft.com/kb/128642/en-us?fr=1>

Limiting the maximum memory used by pdfaPilot

Using Linux, you can limit the amount of memory used by a single process by an additional parameter:

```
--maxmemory=<max. memory in MB>
```

Processing will stop and result in an error if memory is exceeded.

Limitation of concurrent processes

If the StdOut of the command line indicates "Maximum number of parallel processes allowed by license already running; waiting until one of the current processes terminates" (before pdfaPilot 8: "Waiting for CPU"), the maximum number of parallel processes is reached. A default Server/CLI license allows up to 8 parallel processes. To run more than 8 processes simultaneously, you should look at callas License Server as the better option. You can find more info [here](#).

Performance enhancement

If you want to enhance the performance of your pdfaPilot CLI processes, please keep in mind the following rules:

For embedding missing fonts your system font folder will be scanned unless defined otherwise (see "*Font Embedding*"). pdfaPilot will create a font cache to improve the performance time, but still it might be useful to remove fonts that are not needed from this directory.

If you are using any font embedding fixups, your system font folder will be scanned unless defined otherwise in the fixup configuration. A font cache will be created to improve the performance time, but still it might be useful to remove fonts that are not needed from this directory.

Creation of XML or PDF reports takes less time than the XSLT option (see "*Report types*").

Creation of reports takes additional time even if a profile contains only fixups, an analysis will be executed for gathering report information.

Optimization of needed installation space

To reduce the space needed by the installation of pdfsPilot, it is possible to delete some subfolders of the CLI component (in subfolder /cli) if their respective functions are not needed in the individual use case.

To avoid processing errors or unexpected behaviour of pdfaPilot any modification should be done well-considered.

etc/Actions	If no Arrange action is used
etc/APDFL	If no font embedding or PDF/A conversion is used (or if font situation is clear)
etc/Backgrounds	If no layer/image mask report is used
etc/Certify	If no preflight certification is used
etc/ColorConversion	If no color conversion is used
etc/HtmlConverter	If no PDF report based on HTML template is used
etc/Inventory	If no inventory report is used
etc/MailConverter	If no e-mails are processed
etc/PDFOfficeTool	If no Office-files are processed
etc/PDFPSTool	If no PostScript-files are processed
etc/Reports	If no PDF/A-HTML Report or ZUGFeRD is used
etc/TPex	If no tagged PDF to HTML/EPUB export is used
etc/UnpackTool	If no archive files are processed
etc/Visualizer	If no Comparison is used

Install basic fonts on Linux

As most Linux distributions are delivered without or with very few fonts, it is advisable to install at least a basic set of fonts. We recommend installing at least the the fontconfig package and the MS core fonts, which is e.g. done on Debian-based distributions by the following commands:

```
sudo apt -y install fontconfig
```

```
sudo apt install ttf-mscorefonts-installer
```

For other distributions, please consult the respective documentation.

Possible errors within the Server-UI communication

The communication between the Server-UI and the underlying Server process (which observes the folders) takes place by network communication ports using SOAP.

When the Server-UI shows a warning for a connected Server like "Error 28", it indicates that there were errors either during the connection to the Server process itself or that the Server process has stopped working.

You can try to reconnect to the Server in the UI or try to start the local Server process again by the "Start Server" option.

Get in touch

If some necessary information is not provided by this manual or if there are any questions or feedback please contact the product management by using the "Contact Support" form on www.callassoftware.com.

You can also send an e-mail to support@callassoftware.com.

If you file a bug report please make sure your inquiry contains the following information:

- operating system
- pdfaPilot version (call pdfaPilot --version)
- command line call
- original PDF (please delete unnecessary pages to avoid long file transfers), used profiles or configuration files
- converted PDF (if available)

You can also visit the support section on www.callassoftware.com to get answers to common questions or find a reseller near you. The latter might be useful if you want to send a support request that is neither in English nor German.

6.4 Converting office documents to PDF or PDF/A

Input files from Office applications

The CLI component is able to convert common file formats from Office applications directly to PDF/A. For more information and a list of supported applications and files have a look at: http://www.callassoftware.com/goto/apl_ENU_topdf (for pdfaPilot) http://www.callassoftware.com/goto/tbx_ENU_topdf (for pdfToolbox)

- Note: *Since pdfToolbox11/pdfaPilot 9/pdfEngine 11 onwards in the 64bit variant on Windows, the Libre/OpenOffice has to be 64bit as well (and 32bit when the 32bit variant is installed).*

Forced use of OpenOffice

```
--topdf_forceopenoffice
```

When defined, Microsoft Office files are processed with OpenOffice (or LibreOffice if installed).

Create PDF for screen display

```
--topdf_screen
```

Images from Office documents will have lower quality, result file will be smaller.
Default in callas pdfaPilot.

Create PDF for print

```
--topdf_print
```

The PDF will be created with image resolution sufficient for printing, thus leading to larger files.
Default in callas pdfToolbox.

Excel-Sheets without removing white space

```
--topdf_useexcelpagelayout
```

Use Excel page layout, white space will not be removed.

Special handling for MS Office files

```
--topdf_parameter=[ShowHiddenColumns|ShrinkToFit|PrintQualityAndComments|Update-  
ChangedFields|DoNotHideOffice|NoMemoryOptimization]
```

Special parameters to achieve some special layouts for MS Office files.

Parameters

ShowHiddenColumns	Show columns which are not visible due to small width or other settings. (for Office files processed with MS Excel only). Note: When <code>--topdf_useexcelpagelayout</code> is used, this parameter will not be respected.
ShrinkToFit	Shrinks the content of a cell so that the content fits inside. (for Office files processed with MS Excel only). Note: When <code>--topdf_useexcelpagelayout</code> is used, this parameter will not be respected.
PrintQualityAndComments	Images will have bigger resolutions and comments/annotations (e.g. from tracking changes) will be included as well (for Office files processed with MS Word only; can not be combined with <code>--topdf_print</code>).
UpdateChangedFields	Updates the changed fields like automatically generated Table of contents. By default, such fields are not updated (for Office files processed with MS Word only)

DoNotHideOffice	Parameter to avoid hiding Office application for a successful conversion to PDF
NoMemoryOptimization	Parameter to to disable internal memory optimization during processing for improved performance

Logging of dialogs in defined log file

```
--topdf_guiactionslog=<path>
```

Parameters

path Path to folder or logfile.

All dialogs occurring during processing the office file will be logged within this file.

- Note: See the "to PDF" internet page (listed above) for further information about handling of dialogs from MS Office applications.

6.5 General command line options

Usually pdfaPilot CLI is started with:

```
pdfaPilot <PDF file>
```

Input file

May be one or a number of input files (PDF or Office files) to be analysed and converted.

If an input file spec is pointing to an existing folder, all files inside this folder are processed:

Process folders recursively

```
--recursive
```

If the file spec for the input file is pointing to an existing folder all PDF files inside the folder on all levels are processed

Empty the font cache

```
--emptyfontcache
```

Removes all font files from the font cache folder of pdfaPilot CLI.

Incremental saving

```
--incremental
```

Allows to modify the input file, only writing the changes to the original PDF. This can increase the speed significantly since pdfaPilot CLI does not need to create a new copy of the file.

PDF structure and font optimization

```
--nooptimization
```

The internal PDF structure and fonts are not optimized when saving the PDF file. PDF structure and font optimization

Analyze only

```
-a
```

```
-analyze
```

The input file is not being converted but is analyzed whether it is PDF/A compliant.

Both parameters can be used.

Analyze only certain pagerange

```
-p --pagerange=<firstpage>[-<lastpage>]
```

Only applied when analyzing not when converting PDF files.

- Note: When converting non-PDF documents the page range of the original document can be specified.

Parameters:

firstpage	Page where analysis should start
lastpage	optional, page where analysis should end

Example:

```
--pagerange=5-33
```

Setting the cache folder

```
--cacheFolder=<path>
```

Sets the cache folder path. This is set by default to:

Windows:

```
%AppData%\callas software\callas pdfaPilot CLI <version>
```

MacOS:

```
/Users/<USERNAME>/Library/Preferences/callas software/  
callas pdfaPilot CLI <version>
```

Linux:

```
home directory as defined in /etc/passwd/.callas software/  
callas pdfaPilot CLI
```

- Note: This option is mandatory when running the CLI as a user without a home directory.

Parameters

path	absolute path to custom cache folder
------	--------------------------------------

Font folders

If a font is not embedded and an embedding is required by a profile, pdfaPilot CLI will search the system font directories in order to find the needed font file, which are:

System	Folder
Windows	<ul style="list-style-type: none"> • C:\Windows\Fonts
MacOS	<ol style="list-style-type: none"> 1. /Users/<user>/Library/Fonts 2. /Library/Fonts 3. /System/Library/Fonts

System	Folder
Linux	<ul style="list-style-type: none"> • /usr/share/fonts • /usr/lib/X11/fonts • /usr/local/X11R6/lib/X11/fonts • /<user home>/.fonts

Additionally the font folder installed together with pdfaPilot CLI will be searched.

This folder lies next to the executable in "<callas pdfaPilot CLI directory>\etc\APDFL\Resource\Font".

Enable processing PDF with password protection for editing and printing

```
--password=<password>
```

To enable Profile-processing of a password-protected PDF. This option can be used for PDFs with restrictions for editing and printing, which will become unsecured. The resulting PDF will have no security setting.

The entered password will be visible and may be grabbed or logged by other processes on the machine.

Parameters

password	Password of the PDF (avoiding editing or printing of the PDF)
----------	---

Enable processing PDF with password protection for opening

```
--openpassword=<password to open PDF>
```

To enable Profile-processing of an "open"-password-protected PDF. With this option only PDFs with restrictions for opening can be unsecured. The resulting PDF will have no security setting.

Available since pdfaPilot 8.2.

The entered password will be visible and may be grabbed or logged by other processes on the machine.

Parameters

openpassword	"Open" password of the PDF
--------------	----------------------------

Set a processing timeout

```
--timeout=<seconds>
```

Sets the maximum processing time in seconds. If the process exceeds this duration, the execution process will be killed and the processing will result in an error.

6.6 Additional command line options and response files

Quick processing

```
-q --quick
```

Processing is stopped after the first detection of an error in the corrected PDF (only applied if no report is generated).

Define the overwrite mode

```
--overwrite
```

New files override existing files with the same name (applies to report files and to created PDF files).

Create output files only for successful conversion

```
--onlypdfa
```

Create an output file only if the file could be converted to PDF/A.

Set the result path

- Note: If neither an output path nor an output folder is defined, any result will be created next to the input file (default: input file name with suffix `_PDF/A` or `_NOPDF/A`, will be indexed if necessary).
- Note: The use of `--outputfile` together with `--outputfolder` is not supported within one CLI call.

Path to output file

```
-o --outputfile=<path>
```

Defines the absolute path of the destination file. The parent folder must exist.

- Note: Consult section ["Results"](#) to see if a new file was created.

Parameters

path	absolute path to output file
------	------------------------------

Path to output folder

```
-f --outputfolder=<path>
```

Defines an absolute path to a folder where the resulting files of an execution are stored.

If neither an output path nor an output folder is defined any result will be created next to the input file (filename will be indexed if necessary).

The use of `--outputfile` together with `--outputfolder` is not supported within one CLI call.

Parameters

path	absolute path to output folder Output file
------	--

Use an additional profile

```
--profile
```

Run additional checks and fixups by defining a full path to a kfp or kfpX file (exported from pdfToolbox / pdfaPilot Desktop or Preflight in Adobe Acrobat). The input file is converted if no warnings or errors occur.

Using response files

To keep the command line call structured and straightforward, pdfaPilot CLI supports the usage of response files.

These offer the possibility to define each command line switch line by line and also add some comments.

Please make sure the response file is saved as UTF-8 (without BOM).

Example

Response file analyze.rsp:

```
#####
# PDF/A analysis
# with additional Profile for checking the image resolution, where the resolution
# and the name of the Check can be adjusted dynamically
#
--analyze
--profile=<Path to Profile file>
--setvariable=RESOLUTION:300
--setvariable=CHECKNAME:Image resolution for images
#
#####
# EOF
```

Using different responsefiles enables the easy definition of own, localized sets of strings for names of Custom Fixups and Custom Checks as well as different settings for processing PDFs for different output environments.

Command line call:

```
pdfaPilot @<absolute path to "analyze.rsp"> <PDF file>
```

If command line arguments or options with space characters are used in the response file, they shall not be escaped or set in quotes as normally used in CLI commands, where it would e.g. look like: `--setvariable=CHECKNAME:"Image resolution for images"`

`"var/Profiles/PDF analysis/List page objects, grouped by type of object.kfpx"`

or


```
var/Profiles/PDF\ analysis/List\ page\ ob-  
jects\,\ grouped\ by\ type\ of\ object.kf-  
px
```

You can also reference files, e.g. Profiles directly from the response file (you have to use the correct path to the correct location or course).

Like explained above, the correct syntax in a response file has to look like:

```
#####  
# PDF/A analysis  
# with additional Profile for checking the image resolution, where the resolution  
# and the name of the Check can be adjusted dynamically  
#  
--analyze  
--profile=Resolution of color and grayscale images is less than specified value.kf-  
px  
--setvariable=RESOLUTION:300  
--setvariable=CHECKNAME:Image resolution for images  
#  
#####  
# EOF
```

You'll find a sample Profile with the Variables used above here:



Resolution_of_color_and_grayscale_images_is_less_than_specified_value.kfpx

Additional notes

- It is possible to use multiple response files within one CLI call.
- You can also add other command line parameters to the response file.
- The order of the content and the order of the response files will define the final order of all options and commands for the CLI call.
- Limitations with regard to the maximal length of a command line call can thus be avoided.

6.7 PDF/A specific command line options

PDF/A Compliancy level

```
--level=<level>
```

You can define which PDF/A level you need (default is 1b).

Parameters

level	4, 4e, 4f, 3b, 3u, 3a, 2b, 2u, 2a, 1b or 1a
-------	---

Deactivate removal of non-compliant metadata

```
--noxmpremoval
```

Normally, XMP Metadata which is not compliant with PDF/A is removed during conversion. This switch prevents the removal.

Force Conversion to PDF/A

Due to several reasons a regular conversion may not result in a valid PDF/A document. To ensure conversion 3 additional steps can be performed after a normal conversion if a file can not be converted to PDF/A within this first step. The order they will be performed is as listed below. After each step the resulting file is checked for compatibility with the chosen standard. Each step is optional.

```
--forceconversion_reconvert
```

Performs a re-conversion of the PDF via PostScript.

```
--forceconversion_pagestoimages
```

Convert pages with problems into images, while converting the text is transmuted into invisible text, which is correctly positioned to keep the text available for marking and copying.

```
--forceconversion_doctoimages
```

Convert all pages into images, while converting the text is transmuted into invisible text, which is correctly positioned to keep the text available for marking and copying.

```
--forceconversion_resolution=<resolution in ppi>
```

Image resolution in ppi used for the rendered content. (Default = 100 ppi).

Font folders

If a font is not embedded and an embedding is required by a PDF/A-con-version or a profile, pdfaPilot CLI will search the system font directories in order to find the needed font file, which are:

Windows	<ul style="list-style-type: none"> • C:\Windows\Fonts
MacOS	<ul style="list-style-type: none"> • /Users/<user>/Library/Fonts • /Library/Fonts • /System/Library/Fonts
Linux	<ul style="list-style-type: none"> • /usr/share/fonts • /usr/lib/X11/fonts • /usr/local/X11R6/lib/X11/fonts • /<user home>/fonts

Additionally the font folder installed together with pdfaPilot CLI will be searched. This folder lies next to the executable in "**<callas pdfaPilot CLI directory>\etc\APDFL\Resource\Font**".

ICC-profiles folders

The following folders are searched for required ICC-profiles, unless they are already contained in the .kfx-profile already. These folders lies next to the executable in:

- "<callas pdfaPilot CLI directory>\etc\ICC profiles"
- "<callas pdfaPilot CLI directory>\etc\APDFL\Resource\Color\Profiles"

Some system folders for colors are searched additionally:

- MacOS:
"\Library\Application Support\Adobe\Color"
- Windows:
"\Windows\system32\spool\drivers\color"

Set a processing timeout

```
--timeout=<seconds>
```

Sets the maximum processing time in seconds. If the process exceeds this duration, the execution process will be killed and the processing will result in an error.

6.8 File content command line options

Deactivate transparency flattening

```
--notransparencyflattening
```

This switches off both flattening of any contained transparency and setting the blend color space to sRGB.

Add XMP metadata

```
--addxmp=<path>
```

The XMP metadata is merged into any existing XMP metadata.

Parameters

path	Path to an XMP file
------	---------------------

Add bookmarks

```
--addbookmarks=<path>
```

The bookmarks are embedded into the PDF as defined in the XML file.

Parameters

path	Path to an XML file containing
------	--------------------------------

	book-- marks
--	-----------------

Bookmark structure to be used in the XML file

The structure for bookmarks has to be defined in the XML file.

It is also possible to use various formatting styles for the bookmarks ("PLAIN", "BOLD", "ITALIC", "BOLD_ITALIC" and colors "R,G,B"). This formatting will only be shown if the respective PDF viewer supports such styling information.

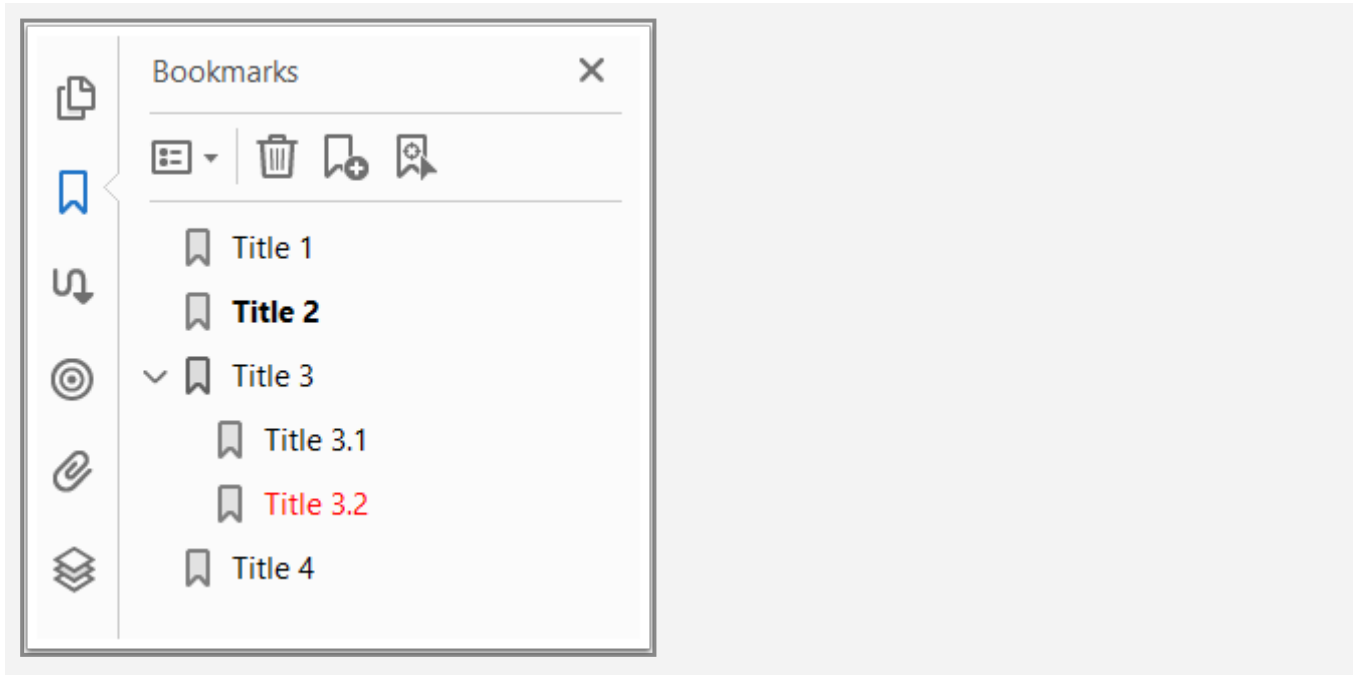
Also a subcategorization for the bookmark tree is possible.

Only page numbers as destinations can be defined, but e.g. no coordinates on a specific page.

Sample for an XML file

```
<?xml version="1.0" encoding="utf-8"?>
<BOOKMARKS xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noName-
spaceSchemaLocation="bookmarks.xsd">
  <BOOKMARK ORDER="100" TITLE="Title 1" PAGE_NUMBER="2" />
  <BOOKMARK ORDER="200" TITLE="Title 2" PAGE_NUMBER="3" STYLE="BOLD" />
  <BOOKMARK ORDER="300" TITLE="Title 3" PAGE_NUMBER="1" />
  <BOOKMARK ORDER="301" TITLE="Title 3.1" PAGE_NUMBER="4" />
  <BOOKMARK ORDER="302" TITLE="Title 3.2" PAGE_NUMBER="5" COLOR="255,0,0"/>
  <BOOKMARK ORDER="400" TITLE="Title 4" PAGE_NUMBER="7" />
</BOOKMARKS>
```

This XML would result into the following bookmark tree:



Set the OutputIntent

```
-i --OutputIntent=<path>
```

Path to a PDF file with an OutputIntent – forces use of this OutputIntent.

Parameters

path	Path to an OutputIntent
------	-------------------------

Define ICC profiles

CMYK

```
--defaultprofile_cmyk=<path>
```

The given profile is embedded as the default profile making device dependent CMYK page objects device independent.

Parameters

path	Path to a CMYK ICC profile
------	----------------------------

RGB

```
--defaultprofile_rgb=<path>
```

The given profile is embedded as the default profile making device dependent RGB page objects device independent.

Parameters

path	Path to an RGB ICC profile
------	----------------------------

Gray

```
--defaultprofile_gray=<path>
```

The given profile is embedded as the default profile making device dependent Gray page objects device independent.

Parameters

path	Path to a Gray ICC profile
------	----------------------------

6.9 Font embedding command line options

If a font is not embedded pdfaPilot CLI will search the system's font directories in order to find the needed font file, which are:

Windows	<ul style="list-style-type: none">• C:\Windows\Fonts
MacOS	<ul style="list-style-type: none">• /Users/<user>/Library/Fonts• /Library/Fonts• /System/Library/Fonts
Linux	<ul style="list-style-type: none">• /usr/share/fonts• /usr/lib/X11/fonts• /usr/local/X11R6/lib/X11/fonts• /<user home>/fonts

Additionally the font folder installed together with pdfaPilot CLI will be searched. This folder lies next to the executable in "<callas pdfaPilot CLI directory>\etc\APDFL\Resource\Font".

Define an additional font folder

```
--fontfolder=<path>
```

Additional folder to look up fonts for embedding.

Parameters

path	Path to config file
------	---------------------

Example:

```
--fontfolder="C:\AdditionalFonts"
```

- Note: You can force pdfaPilot CLI to only scan the folder defined by `--fontfolder` (and not search the system's font folders) by using the option `--fontonly`.

Substitute fonts

```
--substitute[=<path>]
```

Font substitution can be used when the original fonts used in a PDF file are not available. By default, the font substitution file `pdfa.cfg` stored in `"etc/FontSubstitution/"` provides the basis for substitution.

Alternatively, you can also enter a custom path to a fontsubstitution file, e.g.:

```
--substitute=C:\fontsubstitution.cfg
```

If you want to switch off font substitution completely, just hand over an invalid value, e.g.:

```
--substitute=no
```

Parameters

path	Optional, path to config file
------	-------------------------------------

The following notations are allowed in the `fontsubstitution.cfg`:

SubstituteAll

Each font can be substituted by every other font of the entry.

```
SubstituteAll<tab>fontname<tab>fontname<tab>fontname...
```

SubstituteFirst

Only the first font of the entry can be substituted by the following fonts.

```
SubstituteFirst<tab>font to be substituted<tag>fontname <tab>...
```

More examples

Please see [Embedding fonts: Font substitution file](#) for further details regarding font and glyph substitution

6.10 Creating file packages

Some PDF standards allows the embedding of PDF- and also non-PDF-files into another PDF file. Sometime these filepackages are also called collections. Using pdfaPilot CLI it is possible to create such file packages from a complete folder or to define different ways how a file which shall be embedded is handled. In general a file package is created with `--collection` This will create an index document, which lists all embedded files from the given folder. Also an existing folder structure will be respected

```
--collection <folder>
```

In general a file package is created with `--collection` This will create an index document, which lists all embedded files.

```
--collection <file> [<file>]
```

Settings for file embedding

```
--collection [--embedinto=[target],<file>] [--embedfile= [target,[relationship],<file>] [--embedwithlink= [area,<file>]
```

--embedinto

It is possible to use own templates or normal PDF for embedding files. The standard for the file where other files will be embedded can be defined using the conversion target (see below). If no file is defined, an index file is created. Instead of using a destination file, also a folder containing a HTML-based Template with an index.html can be used as destination. The overview page of all embedded files will be created with the layout defined within this template then.

Parameters

target	A3b, A3u, A3a, A2b, A2u, A2a,
--------	-------------------------------------

	A1b, A1a or No (Default)
--	-----------------------------

--embedfile

Also for files to embed a conversion target can be defined using the con-version target. For PDF/A-3 standards also a relationship entry for each embedded file can be set.

Parameters

target	A3b, A3u, A3a, A2b, A2u, A2a, A1b, A1a, PDF or No (Default)
--------	--

Using the target "**No**", no conversion to PDF is done. (Only available for embedded files.)

relationship	Source, Data, Alternative, Supplement, Unspecified (Default)
--------------	---

--embedwithlink

Alternatively, files can be embedded with defining an area in the contain-ing document, where a link to the contained file is created. No conversion will take place with the file to embed.

Parameters

area	X1,X2,Y1,Y2[pt, in, cm, mm]
------	--------------------------------

Defines a rectangular area, based on the lower left corner of the page, where a link to the embedded file is inserted. Default unit is pt.

- Note: The "**--embedwithlink**" function is not supported when using an HTML-based template for the creation of the overview page.

Example:

```
--collection --embedinto=A3b,<PDF file> --embedfile=A3b, Alternative,<file> --embedfile=A2b,Source,<Office file> --embedfile=No,Data,<file>

--collection --embedinto=A3b,<path to HTML template folder> <folder with files to create a collection from> -o=<path to result PDF>
--collection --embedwithlink=10,10,100,100,<file> --embedwithlink=10mm,100mm,100mm,200mm,<file>
```

6.11 Converting email files to PDF

Some common email file formats (MSG, EML, EMLX) are supported for conversion to PDF. Some PDF standards do not allow embedding files (like PDF/A-1) but other allows the embedding of PDF- (PDF/A-2) and also non-PDF-files (PDF/A-3) into another PDF file. To match these different targets, using pdfaPilot CLI it is possible to define variety of different settings for handling attachments of emails. In general a PDF is created with `--emailtopdf` This will create a PDF containing all attachments, those who can be converted as PDF (like Office files or images), all other in their native format.

```
--emailtopdf <email file>
```

Settings for email conversion

```
--emailtopdf
  [--level=<level>]
  [--attachments=<parameter>,<parameter>,...]
  [--onerror=<parameter>]
  [--embedsource]
  [--noembed=<file type>,<file type>,...]
  [--noconvert=<file type>,<file type>,...]
  --template=<path to folder>
  <email file>
```

--level

You can define the required PDF/A level (default is no PDF/A conversion).

Parameters

level	4, 4e, 4f, 3b, 3u, 3a, 2b, 2u, 2a, 1b, 1a or No (default: No)
-------	--

--attachments

Depending on the defined parameter, attachments will not be embedded or embedded in their original file format, converted to PDF or as pages. The parameters can be combined comma separated.

Parameters

IGNORE	Attachments will be ignored and not converted or attached
ORIGINAL	Attachments will be embedded in their original file format
PDF	Attachments will be converted to PDF (if possible) (Default)
PAGES	Attachments will be converted (if possible) and attached as additional pages to the email content

--onerror

If an attachment can not be converted to PDF, this setting allows the definition of handling such email files.

Parameters

ABORT	Aborts processing if an attachment can not be converted to PDF
SKIP	Attachment will be skipped and reported on the CLI.
FALLBACK	Attachment will be embedded in original file format, if it can not be converted to PDF. If a PDF/A-level has been selected, PDF/A-3 with the respective sublevel will be used. (Default)

--embedsource

The email-file itself will also be embedded in the resulting PDF.

--noembed

RegEx expression for file type extensions for attachments, which will not be embedded or converted to PDF although is attached to the email file. Samples for valid RegEx (RegEx is case-sensitive, "(?i)" will make it case-insensitive. --noembed="*.vir" matches extension "vir" in small letters --noembed="*. (VIR|EXE)" matches the listed extensions in big letters --noembed="(?i) *.VIR" matches all possible variations of "VIR"

--noconvert

RegEx expression for file type extensions for attachments, which will not be embedded or converted to PDF although is attached to the email file. See **--noembed** for RegEx samples.

--template

It is possible to create own HTML-Template and Cascading Style Sheets (CSS) to adjust the visual appearance of the created PDF.

Creating custom templates for email conversion

In order to get a "customizable" template, go to:

/var/Mail/Templates/Basic/manifest.xml
and temporarily set:

```
<x:settings>
  <x:keeptemp>true</x:keeptemp>
</x:settings>
```

A folder "Email.html" is created next to the email when you now execute any email conversion, like:

```
pdfaPilot.exe --emailtopdf Email.msg
```

You can now set:

```
<x:keeptemp>false</x:keeptemp>
```

in Email.html/manifest.xml and again run your conversion command, like:

```
pdfaPilot --emailtopdf Email.msg --template=Email.html
```

- ⚠ Please note that the files and folders mentioned above should not be edited because all changes will be ignored/overwritten. The static data can be used only to verify the current visual appearance without any execution with pdfaPilot again.

6.12 Creating HTML or EPUB

Creating HTML from tagged PDF

Converts a tagged PDF file into HTML, which can be opened in a browser to check the tagging structure or viewing the content in various styles.

```
--createhtml <PDF file>
```

It is possible to create own Cascading Style Sheets (CSS) to adjust the visual appearance of the created EPUB.

```
--stylefolder=<path to folder>
```

By adjusting the `<tag-and-attr-processing.cfg>` in the `/config` folder of the predefined templates, it is possible to suppress tags or attributes or their content as well as replacing tags or attributes by other tags or attributes.

Creating EPUB from tagged PDF

Converts a tagged PDF file into EPUB, which is the standard format for eBooks.

```
--createepub <PDF file>
```

Defining the EPUB level will result in the respective standard conformance.

```
--level=[2|3]
```

When one or more tags are defined, new chapters will be created at these contents.

```
--chapter=[TAG,][TAG,]...
```

It is possible to create own Cascading Style Sheets (CSS) to adjust the visual appearance of the created EPUB.

```
--stylefolder=<path to folder>
```

The visual appearance of the created EPUB files can easily be adjusted by modifying a CSS-file. Predefined templates are located in the folder `../var/Actions/TaggedPDF/ExportEPUB`. By changing the values in the CSS, the resulting EPUB can be adapted to individual needs. By adjusting the `<tag-and-attribute-processing.cfg>` in the `/config` folder of the predefined templates, it is possible to suppress tags or attributes or their content as well as replacing tags or attributes by other tags or attributes.

6.13 Creating ZUGFeRD / Factur-X invoices

The German e-invoicing standard ZUGFeRD (Central User Guide Forum electronic Invoice Germany) defines a document, which contains an invoice in a human readable (PDF/A-3) and a machine readable format (XML). By using the PDF/A-3 standard the created file satisfies long-term archiving regulations. With pdfaPilot it is possible to embed such a ZUGFeRD-XML file (created by an electronic billing or accounting system) into a PDF, setting the required ZUGFeRD-XMP metadata and making the PDF a valid PDF/A-3 in one run.

Factur-X is the corresponding French standard to ZUGFeRD 2.0 for electronic invoicing in EU countries.

Create a ZUGFeRD invoice using an invoice PDF and an invoice XML:

```
--zugferd --create=<ZUGFeRD XML file> <PDF file>
```

Adding a GiroCode during creation with a ZUGFeRD invoice using an invoice PDF and an invoice XML (GiroCode data will be determined from XML file):

```
--zugferd --create=<ZUGFeRD XML file>  
--girocode=<lower left x>,<lower left y>,<size>,<unit>,<pagenumber>  
<PDF file>
```

Parameters for GiroCode:

lower left x	Position from the lower left on x axis
lower left y	Position from the lower left on y axis

size	Size of resulting GiroCode (always quadratic)
unit	Optional unit for coordinates and size: mm, inch, pt (default: mm)
pagenumber	Page, where the GiroCode will be positioned

Validate a PDF and the containing XML against the ZUGFeRD specification:

```
--zugferd <PDF file>
```

Extract the embedded XML from a ZUGFeRD PDF file:

```
--zugferd --extract <PDF file>
```

Create several reports from a ZUGFeRD invoice:

```
--zugferd --export=[<option>,<option>,...]
  [--resolution=<resolution in ppi>]
  [--imgformat=<image format>]
  <PDF file>
```

Options for ZUGFeRD reports:

NOPDFIMAGE	No images from the PDF pages
------------	------------------------------

NOXML	XML invoice is not extracted
NOHTML	No HTML view of the XML
NOHTMLSYNOPSIS	No synopsis of PDF images and XML
NOVALIDATIONREPORT	No ZUGFeRD validation report

Additional parameters for the export of PDFIMAGES:

RESOLUTION IN PPI	optional, resolution in ppi or width x height in pixel, e.g. 1024x800 (default: 72)
IMGFORMAT	optional, JPEG, PNG, TIFF, PDF (default: JPEG)

6.14 Report creation and command line output

Report creation

For each pdfaPilot CLI run several reports may be generated by inserting `--report` or `-r` switches into the pdfaPilot call. Following all options and reporttypes are listed. The options of `--report` are treated case insensitive and have to be separated by commas.

```
--report=[<type>],[<trigger>],[<options>],[<PATH=path>
```

Parameters

type	Optional, see "Report types and their options"
trigger	Optional, see "Report triggers"
options	Optional, see "Report types and their options"
path	Optional, see "Report path"

Report types and their options

HTML (default)

A html file is created. The format may be modified by combining this option with other options. It can be opened by any webbrowser.

Additional options for HTML reports

All following switches may be simultaneously used in order to add different types of additional content.

NOICONS	Create a report
---------	-----------------

	without any images
NOCORRECTIONS	Do not log corrections
NODETAILS	Suppress details for the occurrences
OPENRESULT	All entries in the report are opened in Initial view (closed by default if JavaScript is enabled)

Set the path for referenced objects

```
--linkpath
```

HTML reports are generated without exporting referenced objects. This option requires a path (URL) to a folder where the referenced objects reside (Default: folder "\etc\report-template" in pdfaPilot CLI directory)

Example:

```
--linkpath="file:\\Programme\pdfaPilot\etc\reporttemplate"
```

Customize your report

If you create a HTML report, you can completely customize it by adapting the CSS-File to your needs. You can even exchange the pictures.

You can find all material used for the HTML reports within the folder "`\etc\reporttemplate`" in your pdfaPilot CLI directory.

XML

XML file which is intended to be processed with software (parsers). The scheme of the XML report structure can be found within the xsd-file stored in "`\var\XMLV2 report schema\`".

MHT

A mht file is created. MHT is a data format created by Microsoft which offers the possibility to include all resources (like style sheets, images and JavaScripts) to be included in one file. This file type can only be read by Internet Explorer.

TEMPLATE

```
TEMPLATE=<path>
```

Path to template folder (or respective index.html directly)
PDF overview reports are created based on a style defined in a HTML/CSS template. A predefined template can be found in the Server/CLI path: `../var/Reports/Templates`. [Click here](#) for further information.

Report triggers

ALWAYS	Always create a report (default)
IFNOPDFA	Create a report only if the file cannot be converted to PDF/A

IFPDFA	Create a report only if the file can be converted to PDF/A
--------	--

Report path

```
--report=<Report type>,<Report options>,PATH=<Path>
```

Full path of report file – the report path must not contain any commas.

- Note: **PATH=** must always be the last element.

Example:

```
--report=HTML,NOICONS,OPENRESULT,PATH=C:\Sample.html
```

Set the report language

```
-l --language=<language>
```

Defines the language in which the report is generated. Default language is English (en). The language is specified by using a two digit abbreviation. The following values can be used in the standard configuration:

Parameters

EN	English
DE	German
FR	French
IT	Italian

ES	Spanish
JA	Japanese

Command line output

Display progress

```
--noprogess
```

Switch off progress information.

Display hits

```
--nohits
```

Switch off output of hits (errors, warnings, information).

Display summary

```
--nosummary
```

Switch off summary information.

Display timestamp

```
--timestamp
```

Show a timestamp for each line of command line output.

Structure of command line output

Checks

Hit	Type of hit	Name of check	[Name of font]	[Name of glyph]
-----	-------------	---------------	----------------	-----------------

1. Hit

- always visible
- keyword

2. Type of hit

- always visible
- possible values: "PDFA", "Error", "Info", "Warning" (only for additional profile)

3. Name of check

- always visible
- never empty

4. Name of font

- only visible for hits that belong to a font check
- can be empty

5. Name of glyph

- only visible for hits that belong to a font check
- can be empty

Corrections

Fix	Name of fixup	[Name of font]
-----	---------------	----------------

1. Fix

- always visible
- keyword

2. Name of fixup

- always visible
- never empty

3. Name of font

- only visible for fixups that belong to a font correction
- can be empty

Failed corrections

```
FixFailure  Name of fixup  [Name of font]
```

1. FixFailure

- always visible
- keyword

2. Name of fixup

- always visible
- never empty

3. Name of font

- only visible for fixups that belong to a font correction
- can be empty

Progress

```
Progress Value %
```

6.15 Results (Return codes - Error codes - Reason codes)

Return codes and their usage

All return codes below 100 indicate a successful operation.

When executing kfpX Profiles

0	No hit, no Fix-ups executed
1	At least one hit with severity 'info', no Fix-ups executed
2	At least one hit with severity 'warning', no Fixups executed
3	At least one hit with severity 'error', no Fix-ups executed
5	No hit, Fixups have been executed
6	At least one hit with severity 'info', Fixups have been executed
7	At least one hit with severity 'warning', Fix-

	ups have been executed
8	At least one hit with severity 'error', Fixups have been executed; Fixup failed

When executing in pdfa mode

0	PDF is valid PDF/A-file, additional Checks without problems
1	PDF is valid PDF/A-file, but additional Checks with problems – severity info
2	PDF is valid PDF/A-file, but additional Checks with problems – severity warning
3	PDF is valid PDF/A-file, but additional Checks with problems – severity error
4	PDF is not a valid PDF/A-file

When executing an e-mail conversion

0	Successful conversion to PDF/A
2	PDF is valid PDF/A file, but some attachments of the converted e-mail are missing
4	E-mail was converted to PDF, but is not a valid PDF/A-file

When executing the compare action

0	Compared PDFs are equal
1	Compared PDFs have differences

Other commands or actions

0	Successful operation
---	----------------------

Errors

100	Not serialized (no
-----	--------------------

	valid serial-ization found or keycode expired)
101	Command line parameter error
102	Command line syntax error (illegal command)
103	Unknown error (internal error)
104	A file could not be opened
105	An encrypted PDF file could not be opened for writing
106	A file could not be saved
107	File is damaged and needs repair
Error codes above 128 are NOT Windows (only Linux or Mac)	
130	Processing is aborted due to user cancel
132	Illegal instruction
134	Abnormal termination (e.g. out-of-memory)
136	Floating point exception (e.g. div-by-zero)
137	Kill (e.g. explicitly sent by user or by the system)
138	BUS error (e.g. bad memory access)

139	Segmentation violation (e.g. invalid memory reference)
Cells from 134-139 on Mac/Linux indicate an error (crash) of the environmental system. Please report such cases together with the details and files to support@callassoftware.com	
141	Broken pipe (e.g. used for interprocess communication)
142	Execution is cancelled after timeout
143	Termination request (e.g. explicitly sent by user or by the system)
145	Child process died (e.g. used for interprocess communication)
159	Bad system call (e.g. a system call that is not supported by the kernel)

- Note: On a Windows computer the return code may be accessed by using the system variable `errorlevel`, on a Linux/MacOS computer `$?` can be used.

Errors for distributed processing

110	Action is not distributable
111	No Dispatcher was found
112	No Satellite was found or is ready for execution

Reason codes

1000	Unknown reason
------	----------------

1001	A parameter is wrong
1002	A requested file could not be found
1003	A requested folder could not be found
1004	A requested folder is a file
1005	A requested file is a folder
1006	30 days trial period expired
1007	Time limited keycode expired
1008	Invalid activation
1009	PDF does not contain ICC profiles
1010	A file could not be opened
1011	An encrypted PDF file could not be opened for writing

1012	A file could not be saved
1013	File is damaged and needs repair

Listing of all possible codes on the command line

All Return codes and Reason codes can also be listed on the CLI, using the following command:

```
pdfaPilot --status
```

This command will also list the current licensing status of the software.

6.16 Run as a server

callas pdfaPilot Server/CLI can also be used for processing hotfolders on platforms, where no user interface for the configuration of the require set-tings is available (like Linux). This possibility to start a Server without a user interface is available on MacOS and Windows also of course. First, the pdfaPilot CLI has to be started in server mode:

```
--server [--quiet] [--accesskey=accesskey] [--port=port] [--cachefolder=cachefolder]
```

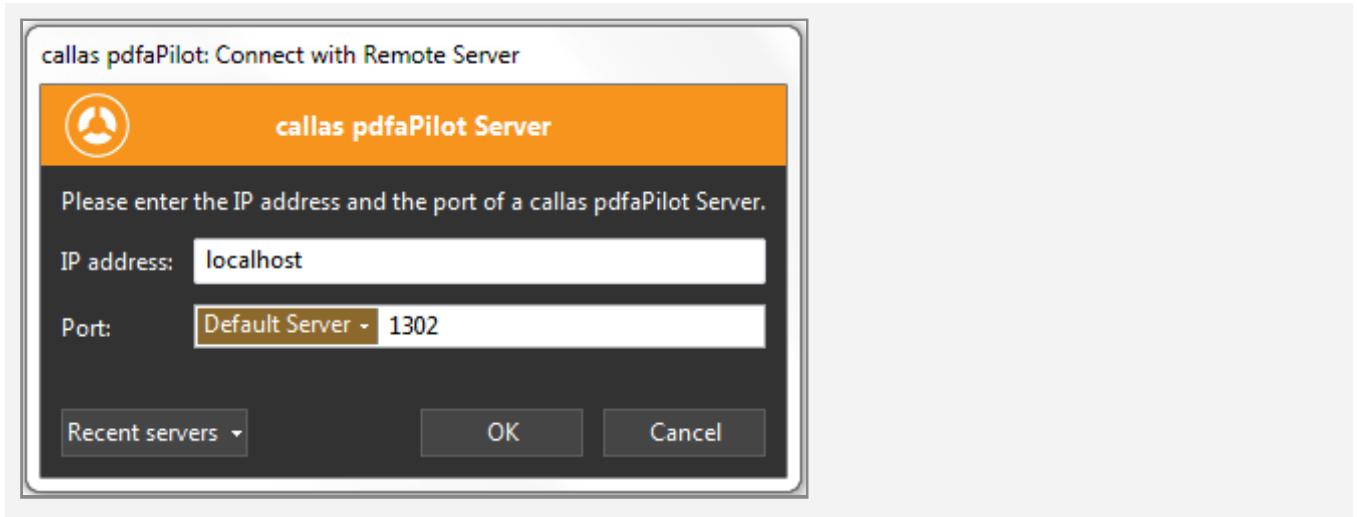
Options

--quiet	optional, suppresses output
--accesskey	optional, sets a accesskey for restricting the possibility to change configuration using the server user interface
--port	optional, defines the port for communication between CLI and server user interface via the network (Default: 1302)
--cachefolder	optional, defines the path to cachefolder

Example:

```
--server --port=1302 --accesskey=123456
```

For setting up a job for hotfolder processing, connect to the remote server using any pdfaPilot Desktop installation in the same network:



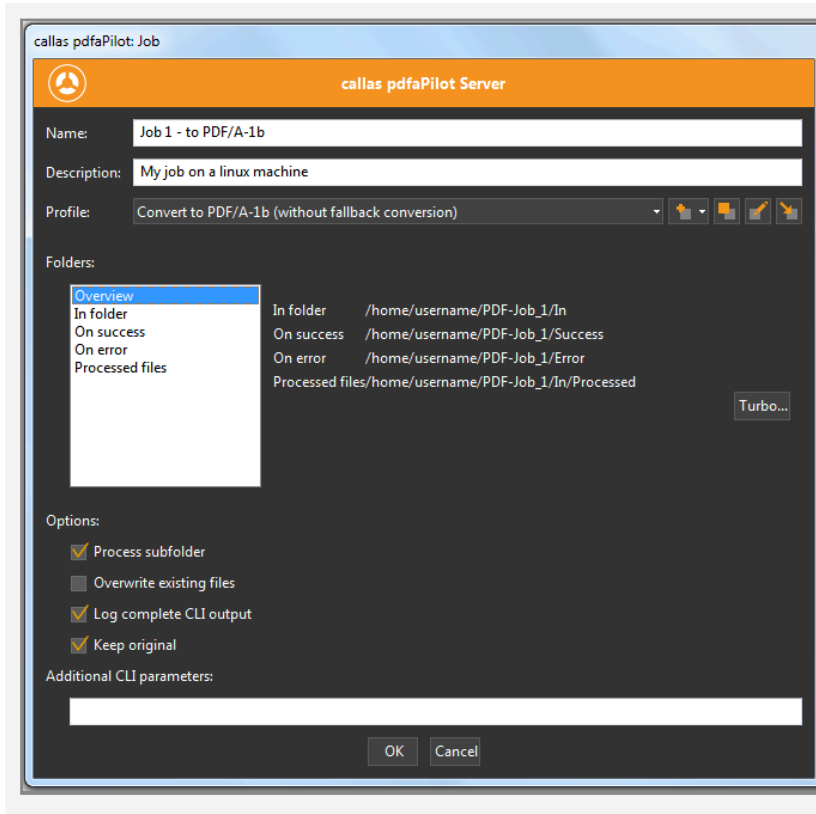
After entering the accesskey, new jobs can be configured, started or stopped. Profiles and Settings will be transferred to the remote server, where they are stored at `/usr/share/callas software` (path must be writable).

If this location can not be used caused by limitations on the respective environment, the additional option `--cachefolder` can be used for defining a custom path when starting the server:

Example:

```
--server --port=1302 --cachefolder=PATH
```

All paths defined for hotfolder processing need to be entered manually and have to be valid path specifications. (Hotfolder paths of any remote server jobs (IN, OUT, etc.) have to be configured so that they are valid from the service's perspective (the system where the service is running) - and not from the perspective of the controlling standalone application.) The server can also be stopped by remote, but not started.



Preferences files of the Server

In general, there is no need to touch the preferences files of the Server.

Using MacOS and Windows, the Server settings from a previous version can be imported when starting a new major version the first time. Within updates in a generation of a major version (e.g. 8.0 to 8.1), there is no change regarding the configuration files in the preferences.

Nonetheless, if there is a need for a manual change to these files, these are the common locations for the preferences folder of the Server:

- Windows:
C:\ProgramData\callas software\callas pdfaPilot CLI <version>
(can easily be reached by enterin "%allusersprofile%" into the address bar of the Windows Explorer)
- MacOS:
/Library/Application Support/callas software/callas pdfaPilot CLI <version>

- **Linux:**
 /usr/share/callas software/callas pdfaPilot CLI <version>

Please contact our support team if you have any questions about the files and there usage before changing them.

6.17 Distributed processing

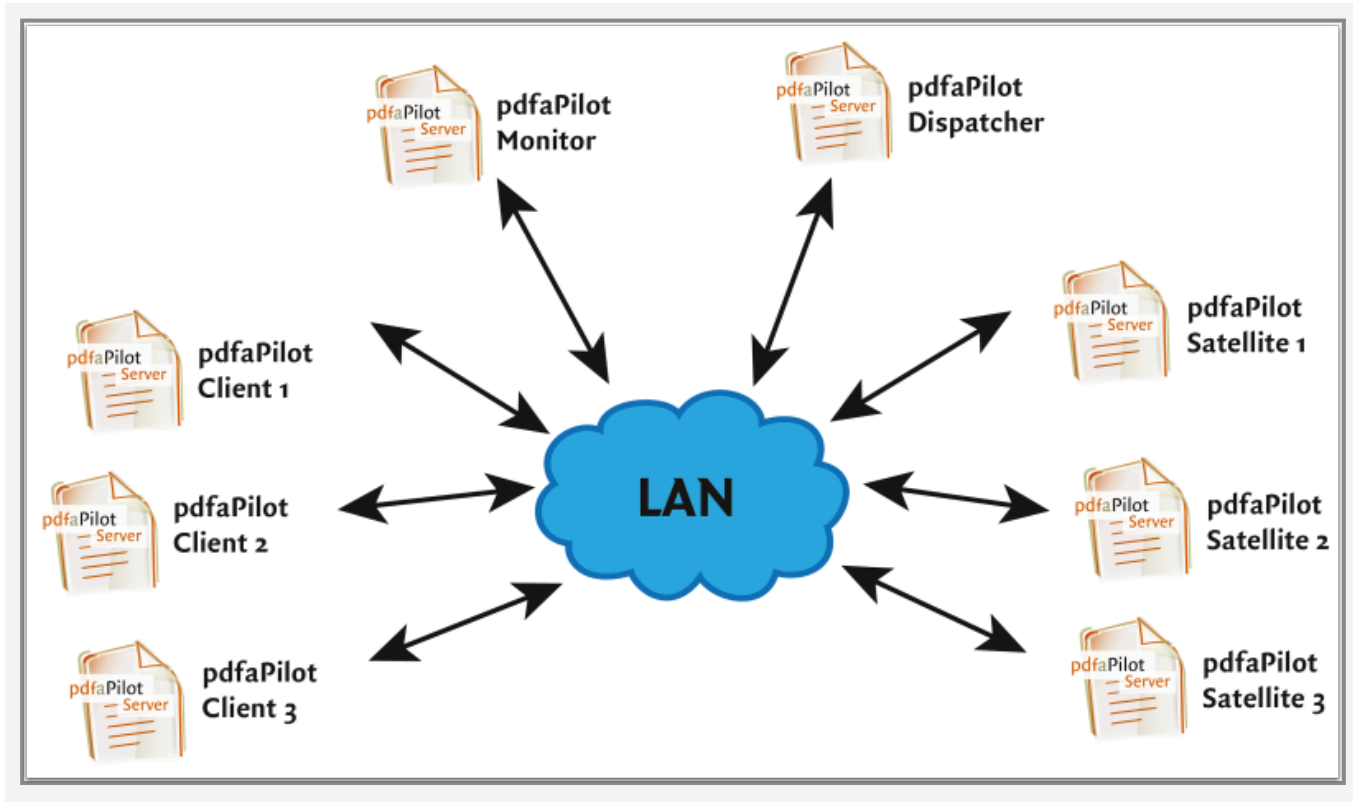
Distributed Processing mode

callas pdfaPilot Server/CLI can be used in distributed processing mode in which all tasks are distributed over the network to as many "satellites" as present and results are sent back to the originator. Therefore pdfaPilot Server/CLI may be started in different modes:

- "Dispatcher" must be present at least one time in the network. This node controls which tasks are to be processed by which machines: the "Satellites".
- "Satellite" receives tasks from the "Clients" or directly from the Dispatcher (if the Dispatcher is run with hotfolders), processes them and sends them back to the Clients.
- "Client" asks the Dispatcher for Satellites and after receiving an available Satellite it sends tasks to the Satellites and receives the results after processing.
- "Monitor" monitors the dispatcher and displays the current situation.

All of these modules can run on the same or on different machines. There needs to be **at least one Dispatcher** and **at least one Satellite**. In order to submit tasks, at least one Client is required.

Distributed processing is supported for Windows, MacOS, Linux.



Starting a Dispatcher

```
--dispatcher [--port=<port number>] [--noserver]
```

Example:

```
--dispatcher [--port=1300] --noserver
```

Port is the port number on which the Dispatcher can be called over the network. This port is set to 1300 as default.

- Note: By the setting the `--noserver` option, the Dispatcher will not observe existing hotfolders, but only distribute tasks to Satellites sent in by Clients. This option is only available using the CLI.

Starting a Dispatcher using the ServerUI

There is also the possibility to start a server as a dispatcher on Windows and MacOS using the user interface. Also hot-

folder-processing can be set up here. In this mode, the Dispatcher will also distribute tasks which are send by other Clients.

Starting a Satellite

```
--satellite --endpoint=<dispatcher ip number>[:<dispatcher port>] [--port=<port number>] [--connections=<number of concurrent connections>]
```

Example:

```
--satellite --endpoint=10.0.0.100:1300 --port=1301
```

In order to process tasks, at least one Satellite is required. Endpoint is the IP number and the port of the Dispatcher. Default is 1300, but it can be changed at the start of the Dispatcher (see above).

Port is the port that the Satellite is using in order to communicate with the Clients. The port of the Satellite is 1301 as default and can be defined optionally to another one port at the startup.

It is highly recommended to use separate port numbers for the communication between Satellite and Dispatcher than for Satellite and Client.

Starting a Satellite using the ServerUI

There is also the possibility to start a Server as a Satellite on Windows and MacOS using the user interface. In this mode, the Satellite will not process any hotfolder jobs on the computer.

- Note: A Satellite will always use the number of CPUs on the respective machine as the number of concurrent connections/processes. To limit this number, the Satellite has to be started by CLI with the `--connections` parameter. The number of connections should not exceed the number of CPUs, as this might reduce the performance per process and could result in some system stability problems.

Distribute a process using a Client

The client is called using any regular pdfaPilot command line command. In order to distribute the call over the network the command line parameters `--dist` and `--endpoint` are added. The client will then first ask the Dispatcher to receive a Satellite connection and then send the command to the Satellite and wait until the result is sent back from the Satellite.

```
pdfaPilot --dist --endpoint=<dispatcher ip number>[:<dispatcher port>] <any regular pdfaPilot call>
```

Examples:

```
pdfaPilot --dist --endpoint=10.0.0.100:1300 <myPDF.pdf>
```

```
pdfaPilot --dist --endpoint=10.0.0.100:1300 --level=2b --analyze <myPDF.pdf>
```

Variables and resources with Distributed processing

When using normal Profiles, nothing has to be considered when processing a file. All needed resources (like ICC profiles or "Place content"-Templates) are included in the Profile kfp-file.

But sometimes, some enhanced scripting of e.g. a Template requires external resources, which are defined/referenced by a Variable.

To ensure that these resources are transferred to the Satellite during Distributed processing, a variant of the `--set-variable=` option can be used:

```
--setvariablepath=<path to resources file or folder>
```

Set type of satellite

As some kinds of tasks shall only be processed on a defined type of Satellite, it is possible to start a Satellite with one or more types set. Every CLI call can also be amended with one or more typification of allowed types of Satellites the task shall be processed by.

Set typification for Satellite:

```
pdfaPilot --satellite --endpoint=<dispatcher IP number> --satellite_type=<type> [--
satellite_type=<type>]
```

for example:

```
pdfaPilot --satellite --endpoint=10.0.0.100 --satellite_type=A
```

```
pdfaPilot --satellite --endpoint=10.0.0.100 --satellite_type=A --satellite_type=B
```

Set typification for Client:

```
pdfaPilot --dist --endpoint=<dispatcher IP number> --satellite_type=<type> [--satellite_type=<type>] <any regular pdfaPilot call>
```

for example:

```
pdfaPilot --dist --endpoint=10.0.0.100 --satellite_type=A <any regular pdfaPilot call>
```

Implementation details:

- If a Satellite has been started with a typification, only Client calls with the same type set will be send to this satellite.
- If a Client call contains a number of typifications, all typifications must match with those set for a satellite.

- If a Client call has no typification set, it can be processed on all satellites, even they have been started with a typification.
- The <type>-string has to be alpha-numeric and is case sensitive.

Avoid local processing

As a fallback, processing can be performed locally if either the action can not be distributed, a Satellite can not be assigned within a timeframe or if no Dispatcher is available. This type of local processing might be not desired for several reasons. To avoid such local processing, the Client call can be amended as well as the start of a Dispatcher (if run as a server with hotfolders) with the option `--nolocal`.

Example for Client:

```
pdfaPilot --dist --endpoint=<dispatcher IP number> --nolocal <any regular pdfaPilot call>
```

Local processing will be disabled and tasks will fail if no Satellite is ready for processing.

Example for Dispatcher:

```
pdfaPilot --dispatcher --nolocal
```

Here `--nolocal` is forwarded to child processes for hotfolder jobs. It has no effect on the processing of non-hotfolder files from a Client distributed by the Dispatcher. If a Client wants to disable local processing, the `--nolocal` setting has to be set in each CLI call of the Client.

Fallback for Dispatcher

In some workflow systems, a fallback for a Dispatcher might be required to ensure production stability. To cover this, a number of Dispatcher can be set up, which will run indi--

vidually. One or multiple Dispatcher can be assigned to a Satellite.

Define multiple Dispatcher to a Satellite

Connects a satellite to two (or more) Dispatcher.

```
pdfaPilot --satellite --endpoint=<dispatcher 1 IP> [--endpoint=<dispatcher 2 IP> [-
endpoint=<dispatcher IP>]
```

Set multiple Dispatcher in a Client call

Distributes a Client call via two (or more) Dispatcher. First reachable Dispatcher with free satellite will process the task.

```
pdfaPilot --dist --endpoint=<dispatcher 1 IP> --endpoint=<dispatcher 2 IP> [--end-
point=<dispatcher IP>] <any regular pdfaPilot call>
```

Define a timeout for processing

In some workflow systems, long running processes might not be allowed and shall be cancelled if a give timeframe is reached. Due to the flexibility of distributed processing, a variety of timeouts for the individual parts can be set:

- for the Client call
- for the Satellite
- for the Dispatcher

Timeout for processing on a Satellite

When defining a timeout for the Client call, the execution will be cancelled after the given period. When defining a timeout when starting a Satellite, all tasks processed by this Satellite will be cancelled after the given period. If both are defined, the shorter timeframe will be used.

Example for Client:

```
pdfaPilot --dist --endpoint=<dispatcher IP> --timeout_satellite=<seconds> <any reg-
```

```
ular pdfaPilot call>
```

Example for Satellite:

```
pdfaPilot --satellite --endpoint=<dispatcher IP> --timeout=<seconds>
```

Timeout for local processing of Dispatcher or Client

A processing timeout (if no satellite is available or if the type of task can not be distributed) for the fallback to local processing on the Client or the Dispatcher (when used as a server for hotfolders) can also be defined. If both are defined, the shorter timeframe will be used.

Example for Client:

```
pdfaPilot --dist --endpoint=<dispatcher IP> --timeout=<seconds> <any regular pdfaPilot call>
```

Example for Dispatcher:

```
pdfaPilot --dispatcher --timeout=<seconds>
```

Timeout for Dispatcher to search for Satellites

Additionally, also a timeout for the Dispatcher can be set, which will define the timeframe in which is searched for Satellites. This can also be set individually for every Client call or when starting the Dispatcher (will have effect on all distributed files then). If both are defined, the shorter timeframe will be used.

Example for Client:

```
pdfaPilot --dist --endpoint=<dispatcher IP> --timeout_dispatcher=<seconds> <any regular pdfaPilot call>
```

Example for Dispatcher:

```
pdfaPilot --dispatcher --timeout_dispatcher=<seconds>
```

- Note: If a timeout for satellites or dispatcher is set and the `--nolocal` option has been defined, it will not be tried to process the task locally. Processing will end up in an error.
- Note: Setting `--timeout...` or `--nolocal` parameters in the "Additional CLI parameter" area of the Server UI is not supported at the moment.

Using the CLI-Monitor

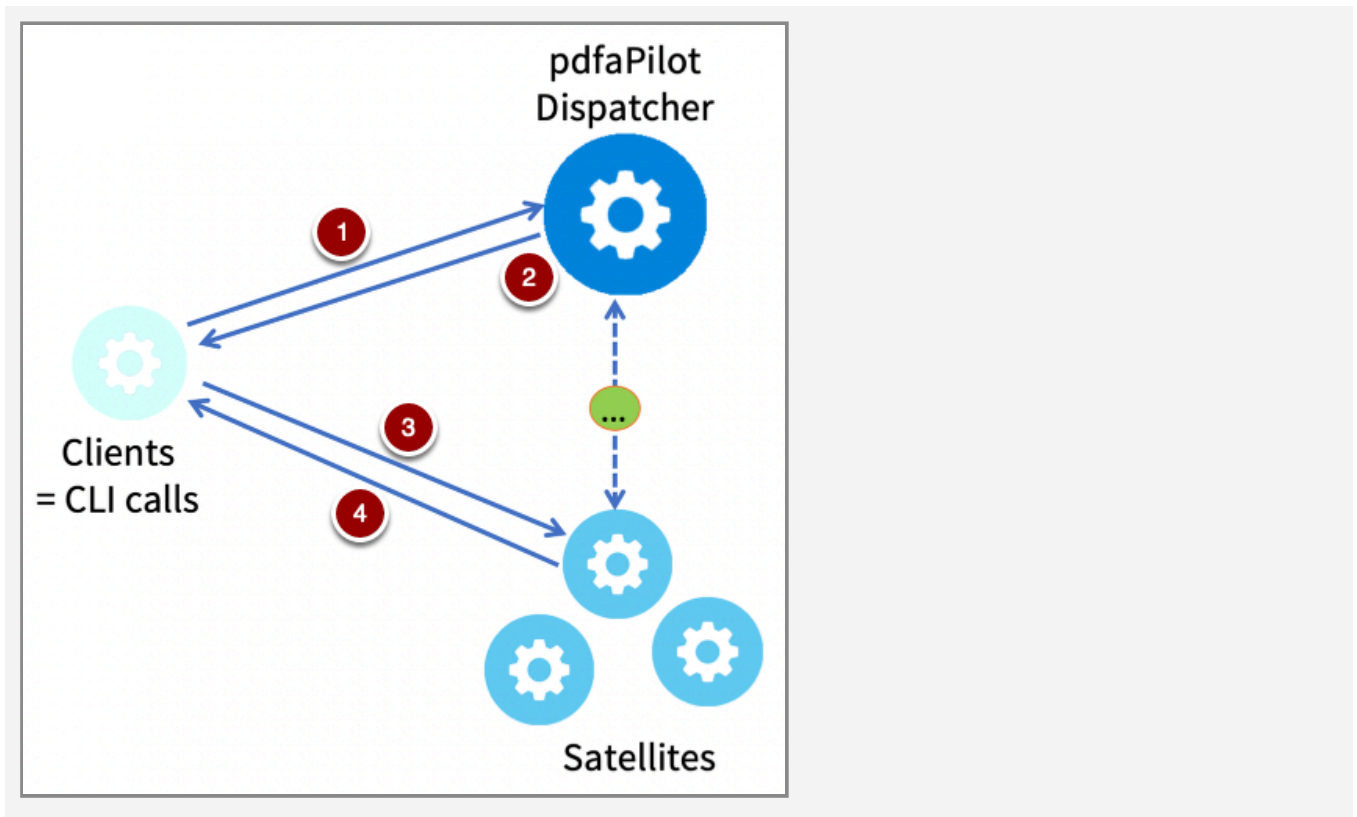
```
pdfaPilot --monitor --endpoint=<dispatcher ip number>:<dispatcher port> [--end-  
point=<dispatcher IP>:<dispatcher port>]
```

Example:

```
--monitor --endpoint=10.0.0.100:1300
```

Monitor is optional and mirrors the command line output of the dispatcher to another computer. Endpoint is the IP number and the port of the dispatcher. When using more than one Dispatcher, also multiple Dispatcher IPs can be entered and observed.

Communication



- 1) Clients sends a request for Satellite to Dispatcher
- 2) Dispatcher assigns a Satellite and sends the address to the Client
- 3) Client sends the task to the Satellite
- 4) Satellite sends the result back to the Client

Licensing requirements for Distributed Processing setups

- Server: Regular pdfaPilot Server/CLI license required
- Dispatcher: Dispatcher pdfaPilot Server/CLI license required
- Satellite: Regular pdfaPilot Server/CLI license required
- Monitor: No license required
- Client: No license required

Distributed Processing in Enfocus Switch

Distributed Processing can also be used within Enfocus Switch.

Just configure the respective settings within the configurator for steps which shall be distributed. If all tasks shall be processed on other machines (Satellites), no local Server license is needed.

Some installations made better experiences, when the setting "Concurrent transfers to the same site" in Switch was set to "Automatic". Also the "Default number of slots for concurrent elements" should not be "0" (zero).

Known limitations until pdfaPilot v. 9.1

Due to technical limitations of the used communication method (SOAP), it is not possible to process files greater than 2 GB using distributed processing until pdfaPilot v.9.1.

callas has eliminated this technical limitation since version 9.1. Until this version, files will be processed locally on the Client.

6.18 Running pdfaPilot via Webservices (SOAP)

It is possible to submit jobs to a pdfaPilot instance and retrieve the results via SOAP requests.

In order to do so you first have to start the pdfaPilot in a "listening mode":

```
./pdfaPilot --satellite --port=<any free port number, e.g. 1201>
```

Or on Windows:

```
.\pdfPilot.exe --satellite --port=<any free port number, e.g. 1201>
```

It is currently not possible to submit PDF files or kfx Profiles via such SOAP requests, so both have to be available to pdfaPilot on mounted volumes.

The SOAP requests wrap any command line parameters in `<ns:args>` tags. The simple requests below asks pdfaPilot to send usage information back by submitting the `--help` parameter.

Beginning from pdfaPilot 7:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ns="http://callassoftware.com/cws.xsd">
  <SOAP-ENV:Body>
    <ns:extExecute>
      <ns:args>
        <ns:userID></ns:userID>
        <ns:args>--noprogress</ns:args>
        <ns:args>--nosummary</ns:args>
        <ns:args>--nohits</ns:args>
        <ns:args>-o=/AppData/pdfToolbox/WebService/sample_result.pdf</ns:args>
        <ns:args>-r=XML,ALWAYS,ALL,PATH=/AppData/pdfaPilot/WebService/sample_result.
xml</ns:args>
```

```

    <ns:args>/Applications/callas pdfaPilot Server 8/cli/var/Profiles/PDFX compli-
ance/Convert to PDFX-1a (ISO Coated v2 (ECI)).kfp</ns:args>
    <ns:args>/AppData/pdfToolbox/WebService/sample.pdf</ns:args>
  </ns:args>
</ns:extExecute>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Beginning from pdfaPilot 8:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ns="http://callassoftware.com/cws.xsd">
  <SOAP-ENV:Body>
    <ns:extExecute>
      <args>
        <userID></userID>
        <args>--noprogress</args>
        <args>--nosummary</args>
        <args>--nohits</args>
        <args>-o=/AppData/pdfaPilot/WebService/sample_result.pdf</args>
        <args>-r=XML,ALWAYS,ALL,PATH=/AppData/pdfaPilot/WebService/sample_result.
xml</args>
        <args>/Applications/callas pdfaPilot Server 8/cli/var/Profiles/PDFX compliance/
Convert to PDFX-1a (ISO Coated v2 (ECI)).kfp</args>
        <args>/AppData/pdfaPilot/WebService/sample.pdf</args>
      </args>
    </ns:extExecute>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

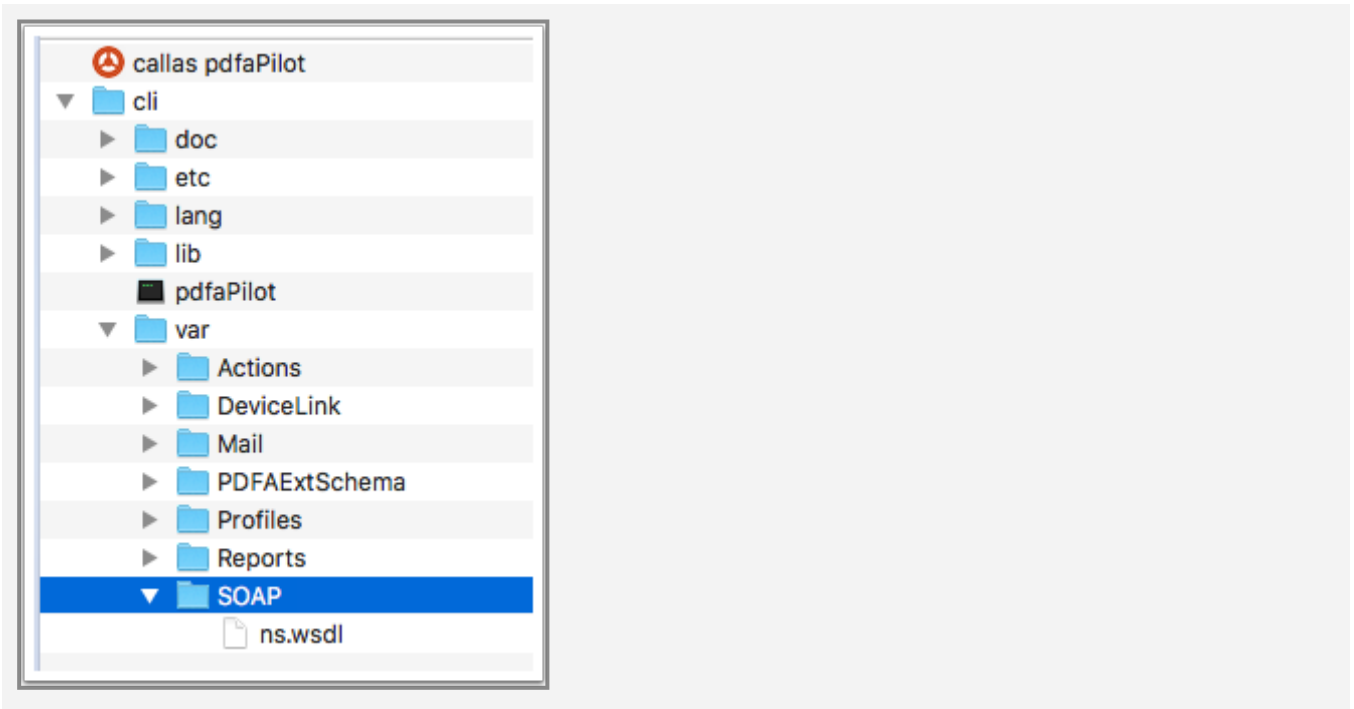
You may include as many parameters as required, there is no limitation.

Any results will be sent back in XML back to the program that you have used to send the request.


You may use the curl tool on command line to send requests and retrieve results:

```
curl -H "Content-Type: text/xml; charset=utf8" -d@.\ns.extExecute_help.req.xml  
http://<pdfaPilot' IP>:<Port>
```

A Web Service Defintion Language file can be found in the pdfaPilot program folder:



The example below is a simple SOAP request that asks pdfaPilot to create single pages from the original PDF file. Path and name of the PDF need of course to be adapted.

 ns.extExecute_makesinglepages.req.xml

6.19 Handling Licensing through the License Server

In order to avoid having to activate a callas software product, you can use a reference to a running License Server in the call to that product. The `--licenseserver` option points the callas software product to the License Server so it can get permission to run your call. There are [two License Server-based models](#):

Using the License Server

All commands related to the License Server are explained in the License Server manual: [Using the License Server](#).

Using the License Server on-premise

All commands related to the License Server on-premise are explained in the License Server manual: [Using the License Server on-premise](#).

6.20 Modifying structure of bookmarks and DPart

pdfaPilot can extract existing DPart Metadata or bookmark structures in a JSON format so that you can modify them and inject them back into the PDF document. This functionality is available as [part of Process Plans](#) where you extract the structures using Quick Check and import them via the “Apply structures” action and directly on the command-line with pdfaPilot CLI using the following commands:

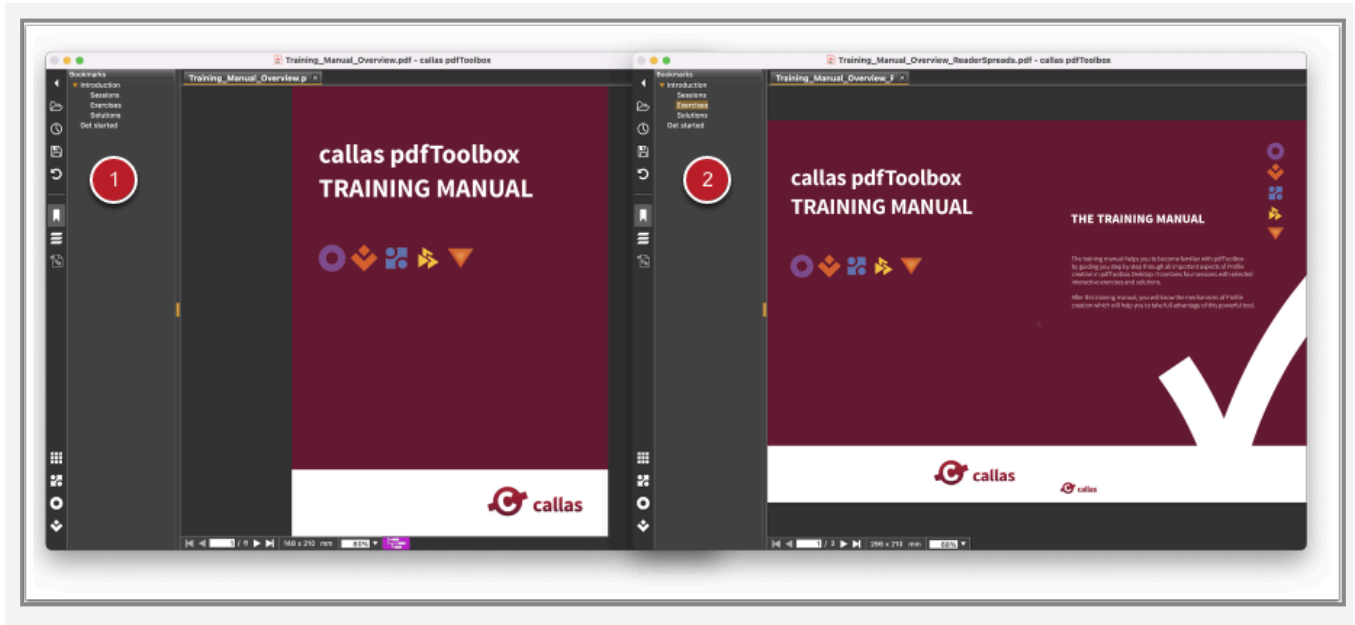
```
--modifystructures --extract  
--modifystructures --apply
```

This article provides two examples of how to update the structures:

- [Example 1: Modifying bookmark structure](#)
- [Example 2: Modifying DPart Metadata structure](#)


Modifying bookmark structure

If you modify a document, such as converting a single-page booklet to reader spreads, adding new pages, or deleting existing pages, most of the time the bookmark structure is not updated and can no longer be used. Here is an example:



1. Document with single pages and existing bookmark structure
2. After conversion to reader spreads, the existing bookmarks won't work anymore because the bookmark structure is not updated accordingly.

To update the bookmark structure, follow the next steps. The modified testfile with the "old" bookmark structure can be downloaded here:

 [Training_Manual_Overview_ReaderSpreads.pdf](#)

1. Extract current bookmark structure to JSON

Use the following command on the CLI to extract the existing bookmark structure to JSON:

```
pdfaPilot <sample.pdf> --modifystructures --extract=bookmarks</sample.pdf>
```

To modify the extracted bookmark structure, it is important to understand the different properties of the JSON objects:

"bookmarks": Array that contains a list of all bookmarks found in the PDF file.

"level": Indicates the nesting level of each bookmark as typically displayed in a PDF viewing program.

"name": Indicates the name of the bookmark.

"page": Specifies the page number that defines the destination of the bookmark.

"open": True if the bookmark is initially open. This only has an effect for a bookmark hierarchy.

```
1  {
2    "bookmarks" :
3    [
4      {
5        "level" : 1,
6        "name" : "Introduction",
7        "open" : true,
8        "page" : 0
9      },
10     {
11       "level" : 2,
12       "name" : "Sessions",
13       "page" : 0
14     },
15     {
16       "level" : 2,
17       "name" : "Exercises",
18       "page" : 0
19     },
20     {
21       "level" : 2,
22       "name" : "Solutions",
23       "page" : 0
24     },
25     {
26       "level" : 1,
27       "name" : "Get started",
28       "page" : 0
29     }
30   ]
31 }
```

2. Adjust bookmark structure

The bookmark structure should now be adjusted using algorithms for the specific use case. In our example, the page numbers have to be divided by 2 to so that the bookmarks refer to the correct pages in the reader spreads document. Also, the name of the first bookmark was changed from "Introduction" to "Manual content".

```
1  {
2    "bookmarks" :
3    [
4      {
5        "level" : 1,
6        "name" : "Manual Content",
7        "open" : true,
8        "page" : 1
9      },
10     {
11       "level" : 2,
12       "name" : "Sessions",
13       "page" : 2
14     },
15     {
16       "level" : 2,
17       "name" : "Exercises",
18       "page" : 2
19     },
20     {
21       "level" : 2,
22       "name" : "Solutions",
23       "page" : 3
24     },
25     {
26       "level" : 1,
27       "name" : "Get started",
28       "page" : 3
29     }
30   ]
31 }
```

3. Apply new bookmark structure to PDF

Use the following command on the CLI to inject the new bookmark structure into the PDF:

```
pdfaPilot --modifystructures --apply=<json file> <pdf file></pdf></json>
```

Modifying DPart structure

DPart is page based metadata in a PDF file, stored in a hierarchy to group pages into “document parts” (which can group pages to “records”). It is organized in a tree so that it is easier for any PDF processors to identify pages or page ranges that have certain properties. pdfaPilot can display such metadata.

When the page structure in a document is modified by any application, such as adding new pages or deleting existing pages, quite often the DPart structure is not updated.


In the example below (taken from the PDF/VT sample files at <https://pdfa.org/resource/cal-poly-pdfvt-test-suite/>) there is

a PDF file that has 10 "document parts" / "records". Each "record" consists of 4 pages: the first two pages are a brochure and the last two pages are luggage tags.

Let us assume that we need to delete the luggage tags from each record which means that the DPart metadata will not fit anymore.



To update the bookmark structure, follow the next steps. The modified testfile with the "old" DPart metadata structure can be downloaded here:

 [Travel V1.0.1 - 10 small - Brochure.pdf](#)

1. Extract current DPart structure to JSON

Use the following command on the CLI to extract the existing DPart structure to JSON:

```
pdfaPilot <sample.pdf> --modifystructures --extract=dpart</sample.pdf>
```

To modify the extracted DPart structure, it is important to understand the different entries:

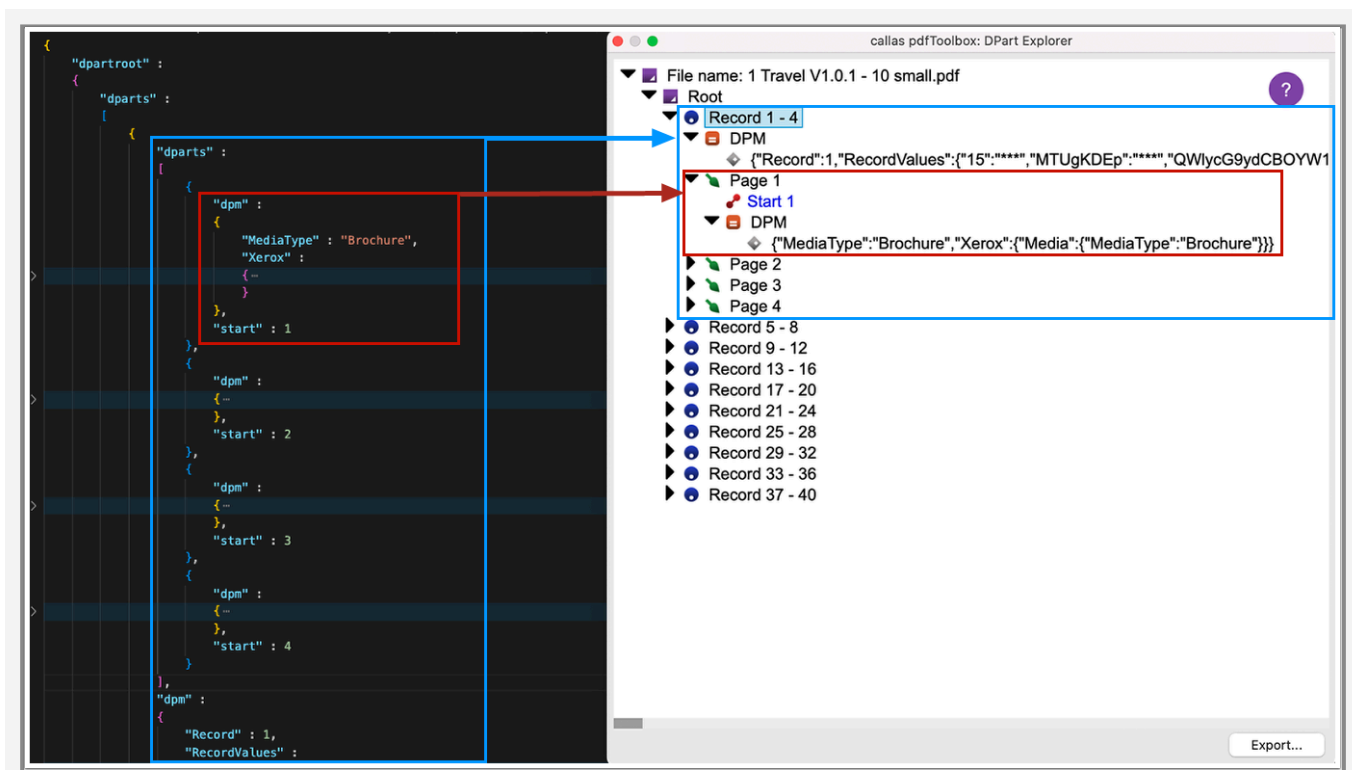
"dpartroot" : The root node of the hierarchy.

"dparts" : Can refer to a specific range of one or more PDF page objects, identified by their start and end keys.

"dpm" : The Document Part Metadata Dictionary contains the actual metadata related to the different parts of the document. The DPM refers to the DPart node in which it is defined and its pages.

"start" : The start key is the number of the first PDF page to which the DPart node refers.

"end" (optional): If a DPart node refers to not just one page but to a range of pages there is an end property (in addition to the start property) indicating the last page to which the node refers.



At the end there are two more entries:

"nodenamelist" (optional): Specifies the names/meaning of the DPart node levels in the tree hierarchy. These names can provide some explanation and could be used by software to display information. In our example we have three levels: The root, the Record level and the pages.

"recordlevel" (optional): Indicates on which level of the DPart hierarchy the record boundaries are, in our exam-

ple RecordLevel is 1. Since the root is zero this indicates that the first level of the DPart hierarchy refers to the records. In our example that are ranges of always four pages specified via their start and page entries.

```
"nodenamelist" :
[
  "Root",
  "Record",
  "Page"
],
"recordlevel" : 1
```

2. Adjust DPart structure

The DPart metadata for the luggage tags has been manually removed from the records. Now the records consist only of the two brochure entries. The page numbers (`"start"` keys) have also been adjusted:



Travel V1.0.1 - 10 small - Brochure - new structure.json

```
{
  "dpartroot" :
  {
    "dparts" :
    [
      {
        "dparts" :
        [
          {
            "dpm" :
            {
              "MediaType" : "Brochure",
              "Xerox" :
              {
                "Media" :
                {
                  "MediaType" : "Brochure"
                }
              }
            },
            "start" : 1
          },
          {
            "dpm" :
            {
              "MediaType" : "Brochure",
              "Xerox" :
              {
                "Media" :
                {
                  "MediaType" : "Brochure"
                }
              }
            },
            "start" : 2
          }
        ],
        "dpm" :
        {
          "Record" : 1,
          "RecordValues" :

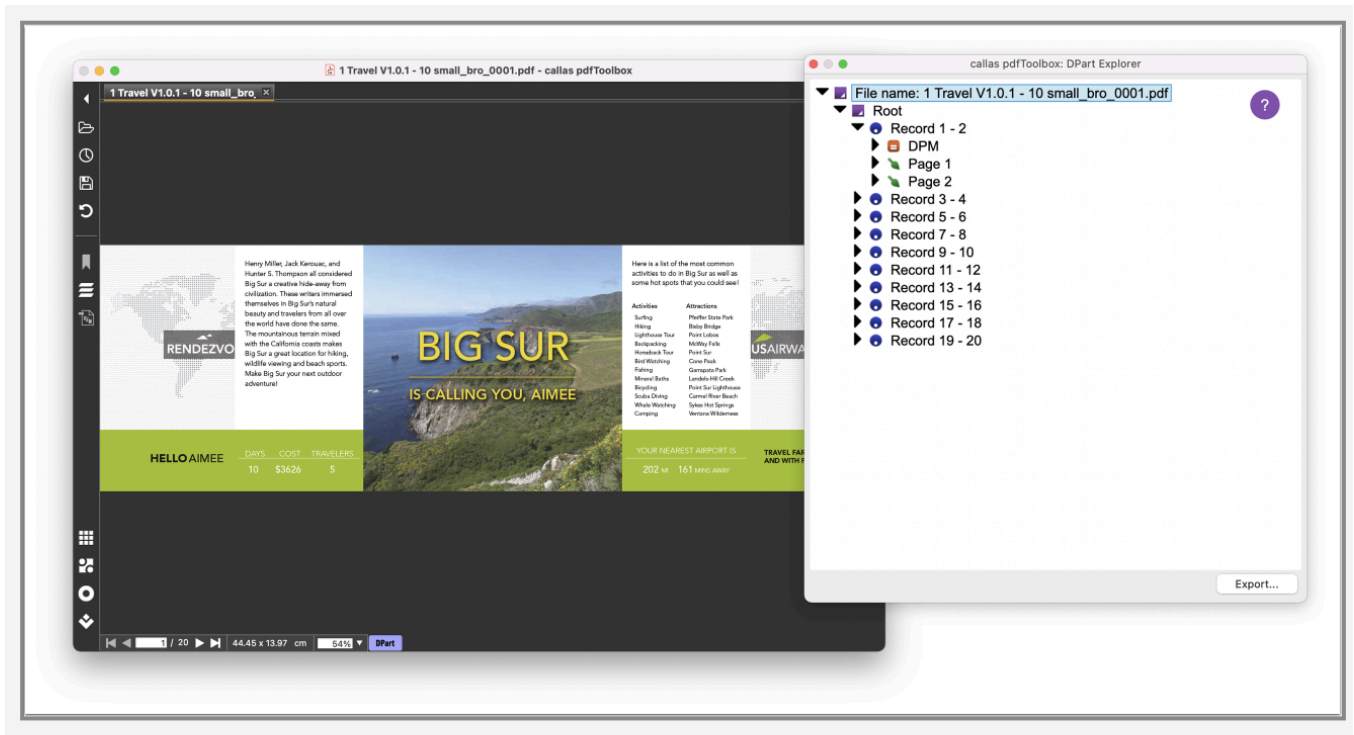
```


3. Apply new DPart structure to PDF

Use the following command on the CLI to inject the DPart structure into the PDF:

```
pdfaPilot --modifystructures --apply=<json file> <pdf file></pdf></json>
```

Now each record has only two pages, and the DPart metadata for all luggage tags in the DPart structure is deleted:



7. callas pdfaPilot Server

7.1 Working with hot folders in pdfaPilot Server

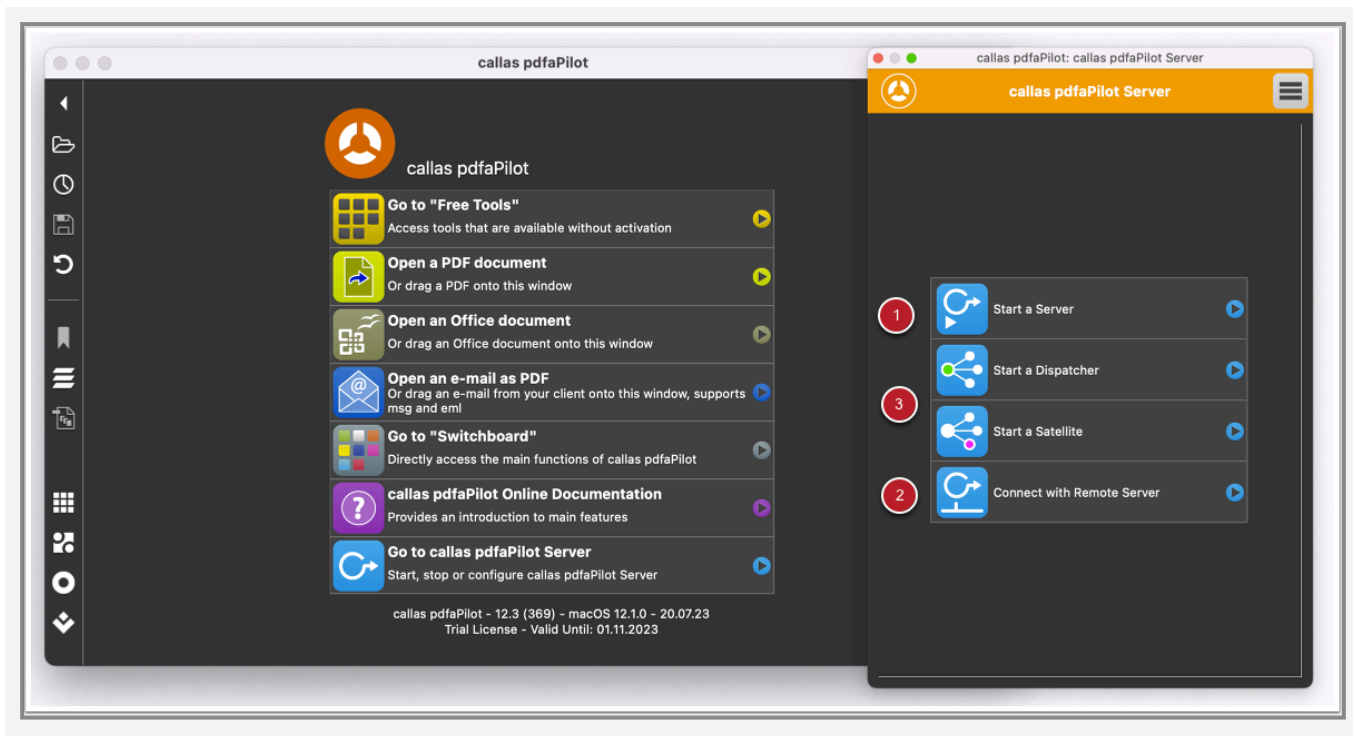
pdfaPilot Server is one of the automatic flavors of the pdfaPilot product family. It is capable of handling any number of *jobs*; each job is a hot folder setup with an input folder and one or more output folders. PDF documents dropped into an input folder are picked up and processed automatically.

Launching the pdfaPilot Server user interface

pdfaPilot Server doesn't have its own user interface; instead it can be controlled through pdfaPilot Desktop. To launch it:

1. Launch pdfaPilot Desktop.
2. Click the "Tools" > "Server" menu item.

The pdfaPilot Server control panel appears:



pdfaPilot Server can be started in different modes:

1. **Start a Server**
Use "Start a Server" to launch pdfaPilot Server *on the same machine* as pdfToolbox Desktop.

2. Connect with Remote Server

Use "Connect with Remote Server" to connect to a pdfPilot Server instance that is already running either on the same machine or on a different machine on the network. This allows remote monitoring and configuration over the network. You will need to know the IP address of the machine running pdfaPilot Server.

3. Start a Dispatcher and Start a Satellite

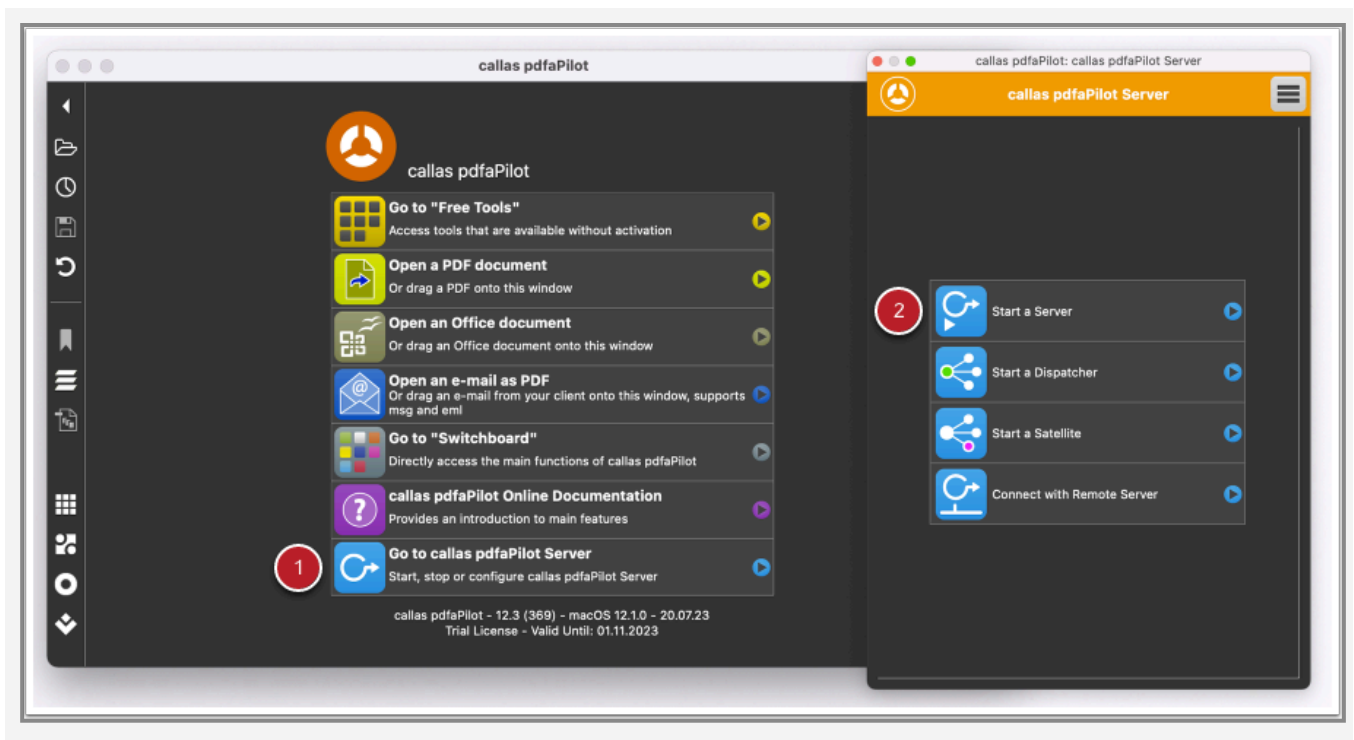
With pdfaPilot Server, tasks can be distributed via a dispatcher in the network to one or more satellites, which receive the tasks, process them and send them back to the dispatcher. The dispatcher controls which tasks are to be processed by which satellite.

7.2 Start a Server / connect with Remote Server

pdfaPilot Server combines the user interface of pdfaPilot Desktop Version in interaction with the CLI component, which is automatically called in the background as soon as you start pdfaPilot Server.

Starting pdfaPilot Server

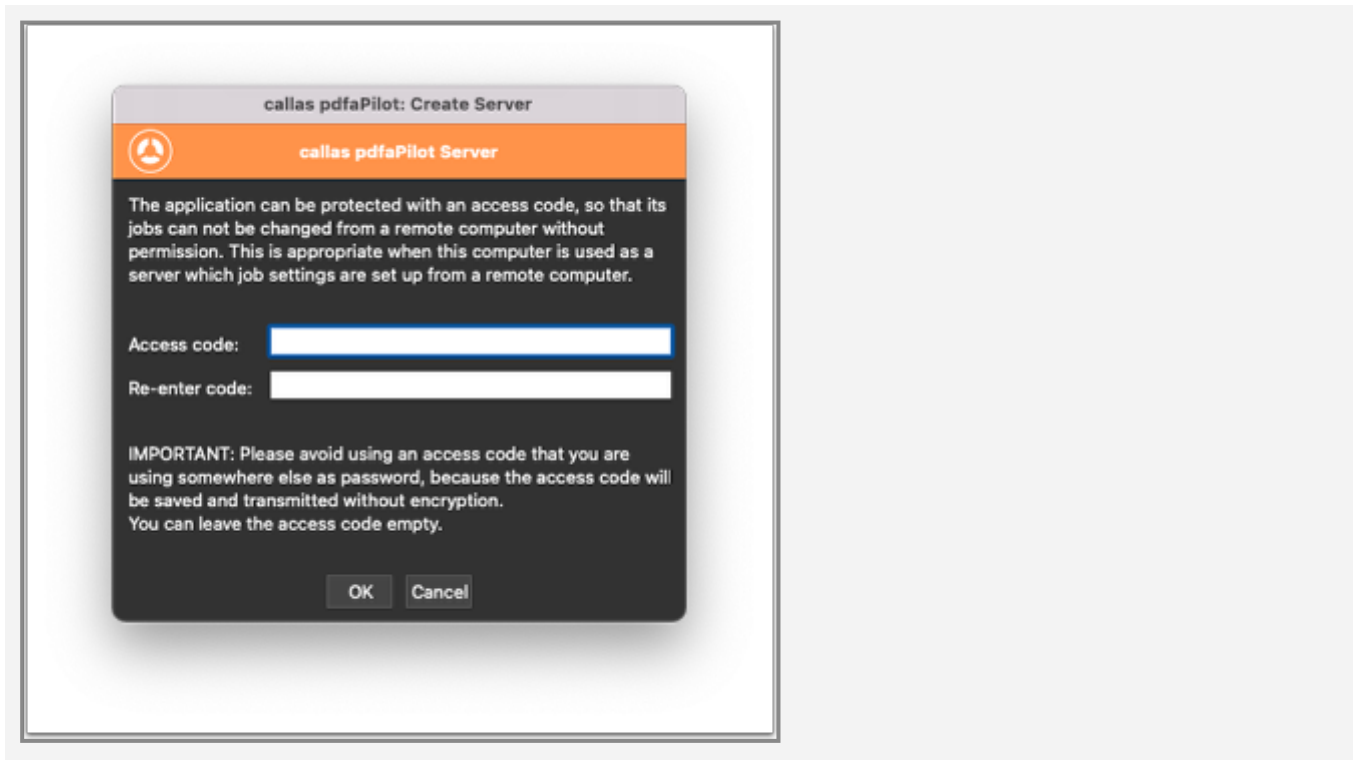
Follow the two steps below to start pdfaPilot Server.



By clicking this button the CLI is started in the background (similar to the manual call with the `--server` command). As soon as the process is running, jobs can already be processed.

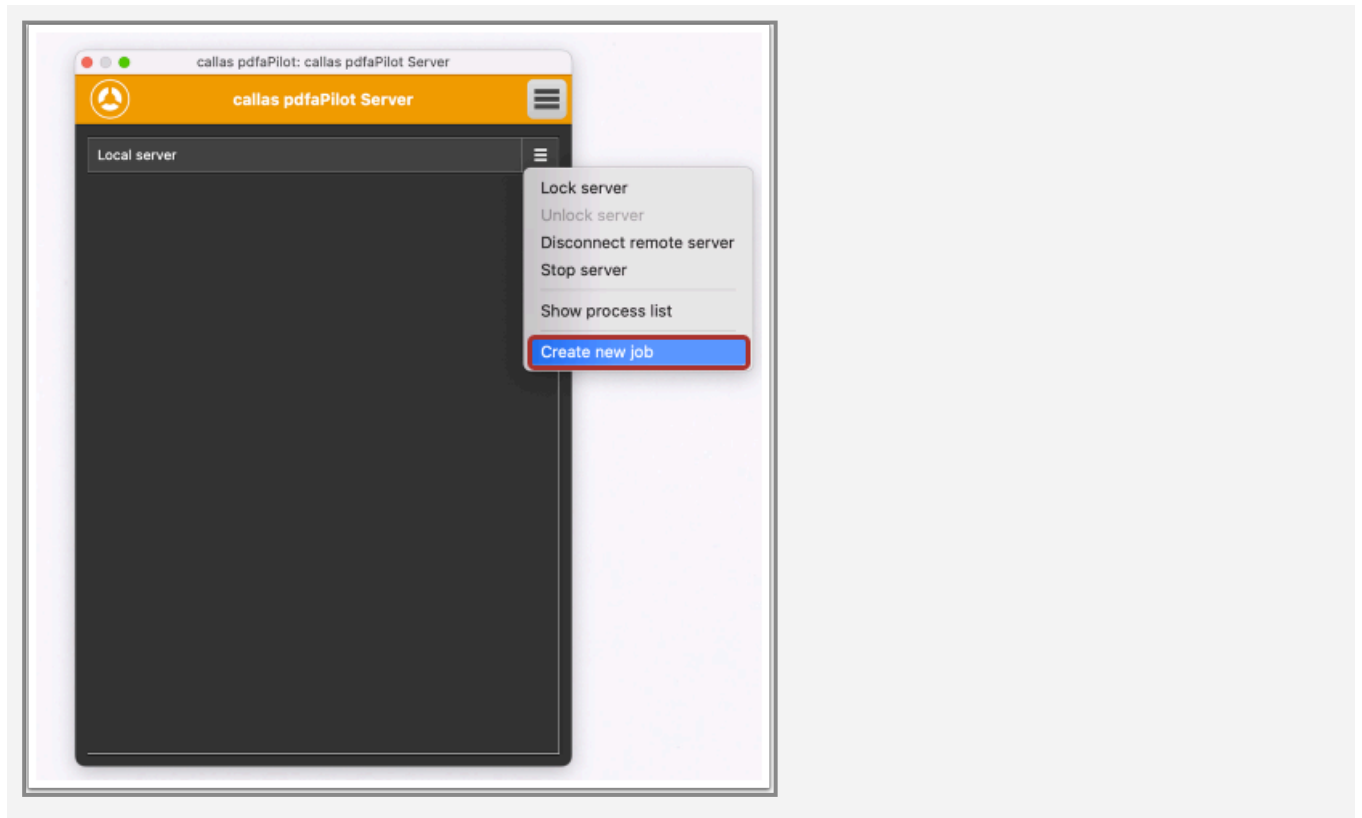
Set access code

To protect the server and the server jobs from unauthorized access and changes, an access code can be assigned.



Create a new Server job

Once the server is running, a server job can be created. See the [next article](#) how to configure a new job.



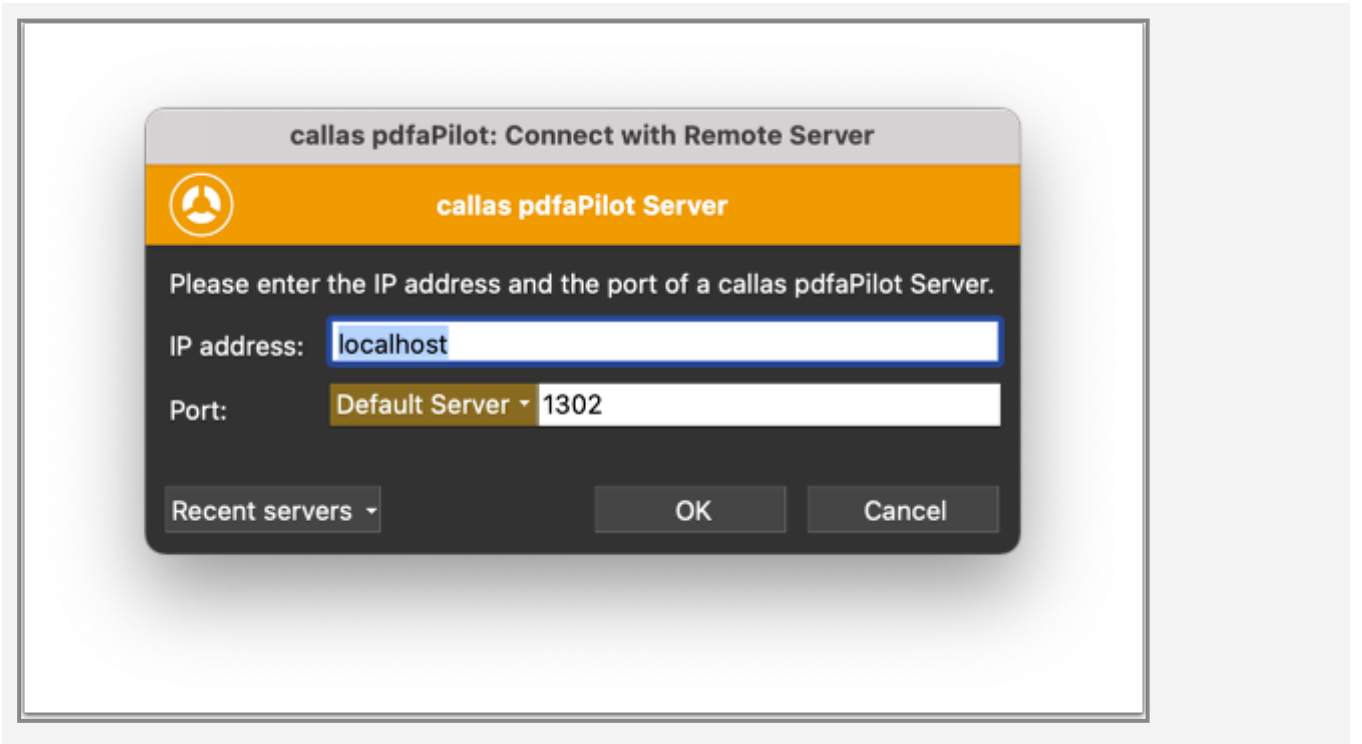
Connect with Remote Server

If the server on which the processing is to take place is located on another computer, you can also connect to the system remotely.



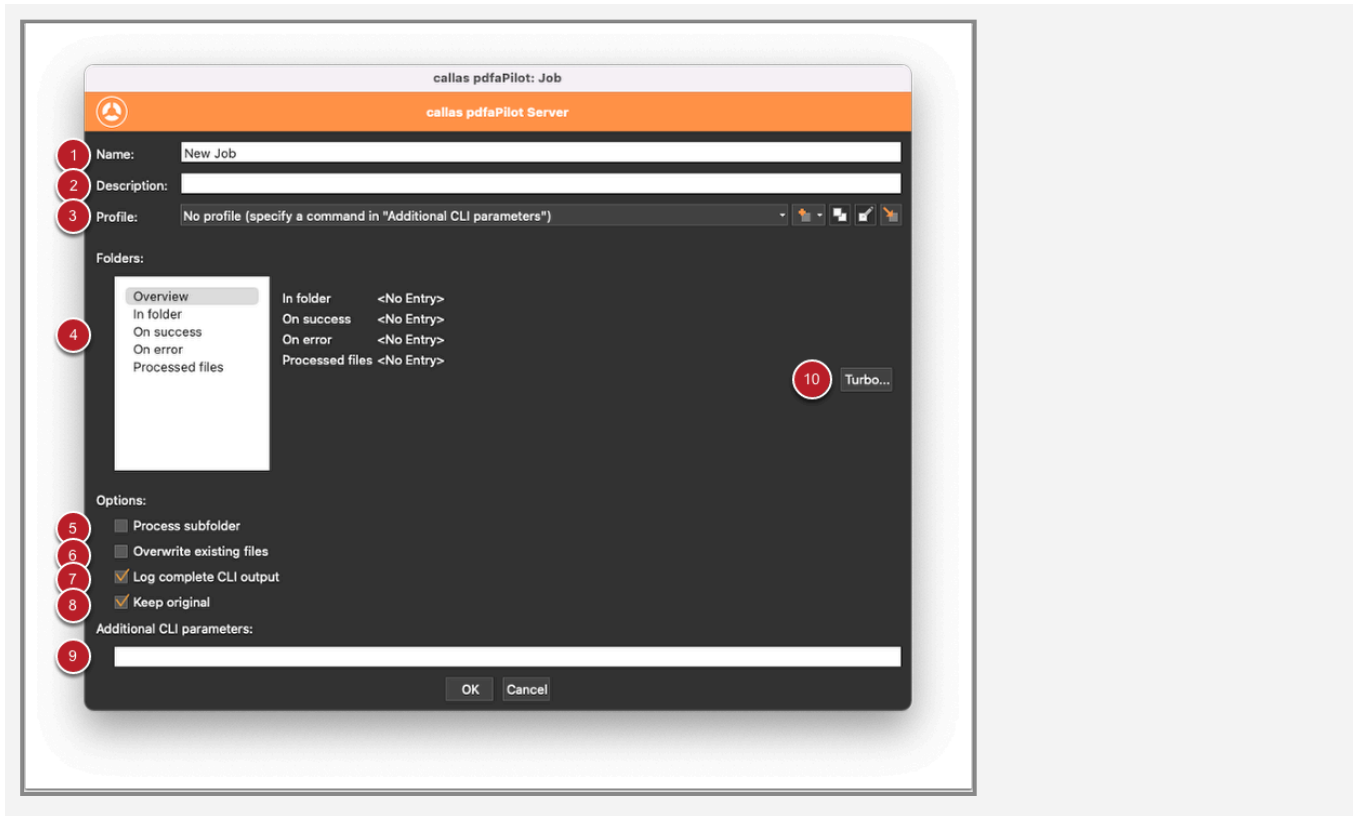
IP and Port

To connect to the target system the IP and the respective port must be specified. Please note: If a firewall is used, the port must also be released accordingly.



7.3 Job settings

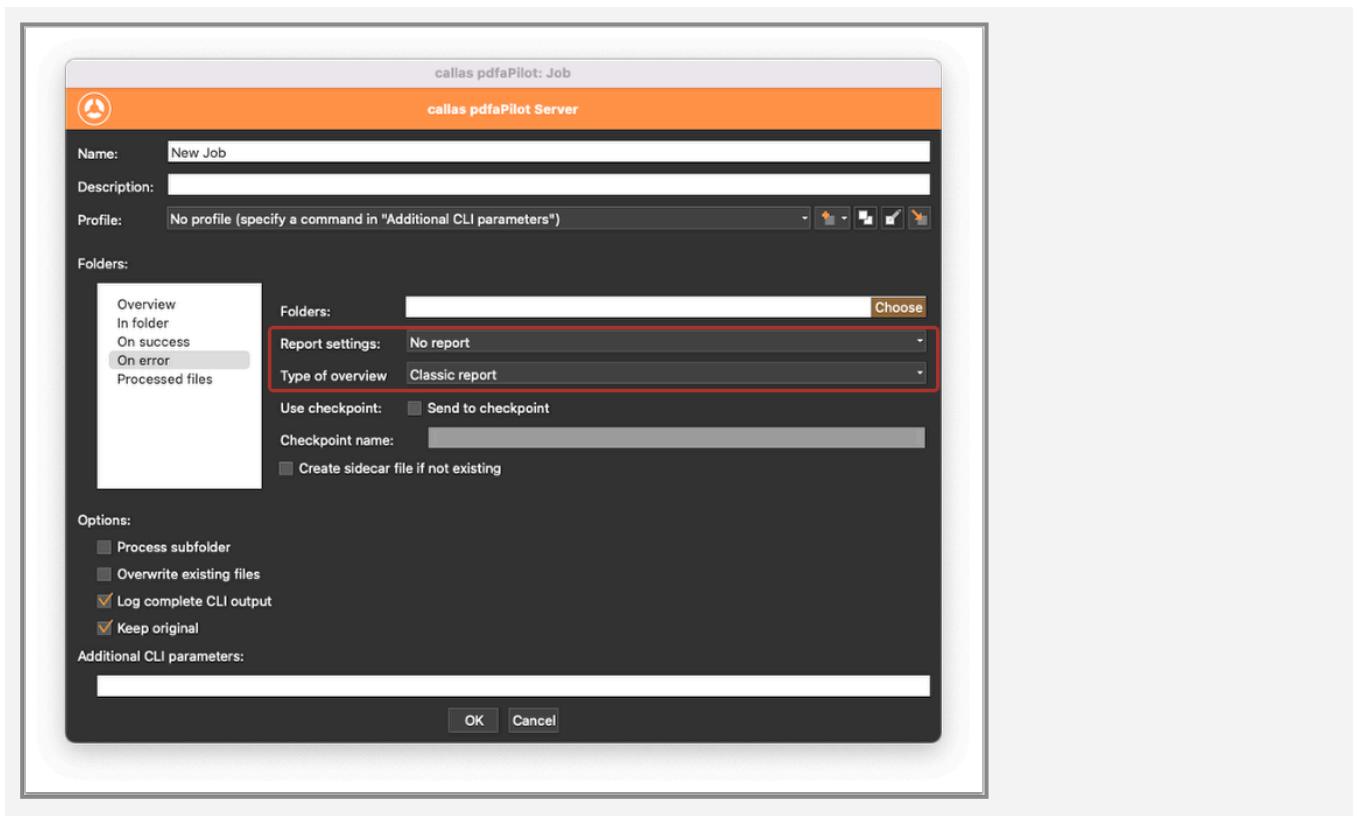
The server job has a variety of setting options, which are described below.



1. Name: Here you can specify a name for the job, this will then be displayed in the list of jobs.
2. Description: In addition to the name, an additional description can also be stored.
3. Profile: Predefined Profiles can be selected, new created, duplicated, or imported from an external path.
4. Hotfolders: The Hotfolders are folders through which files are imported and output after processing, depending on the result of a profile.
5. Process subfolder: By activating this option, nested directories can also be processed, otherwise only files located directly in the folder are processed.
6. Overwrite existing files: In the output folders (Success, Error, Processed) new files are indexed with _0001, _0002 ... and stored next to them. If this option is activated, existing files with the same name are always overwritten.

7. Log complete CLI output: Under "Processed Files", a log file with the CLI output in text form is also created for each processed file.
8. Keep original: The original input file will be stored under "Processed files" as well.
9. Additional CLI parameters: As an alternative to selecting a profile, commands for the CLI can be specified here as well as other parameters that exist for a CLI command. Similar to the command call (via the command prompt/terminal), the parameters must always be separated by a space.
10. Turbo button: When clicking this button, only one path must be specified. Within this folder, the subfolders (Inbox, Success, Error, Processed) are automatically created at the same time.

Create reports



Overview of possible report types

- No report
- HTML report
- HTML report without images
- XML report
- MHT report
- JSON report

Use "non-PDF files" as input files

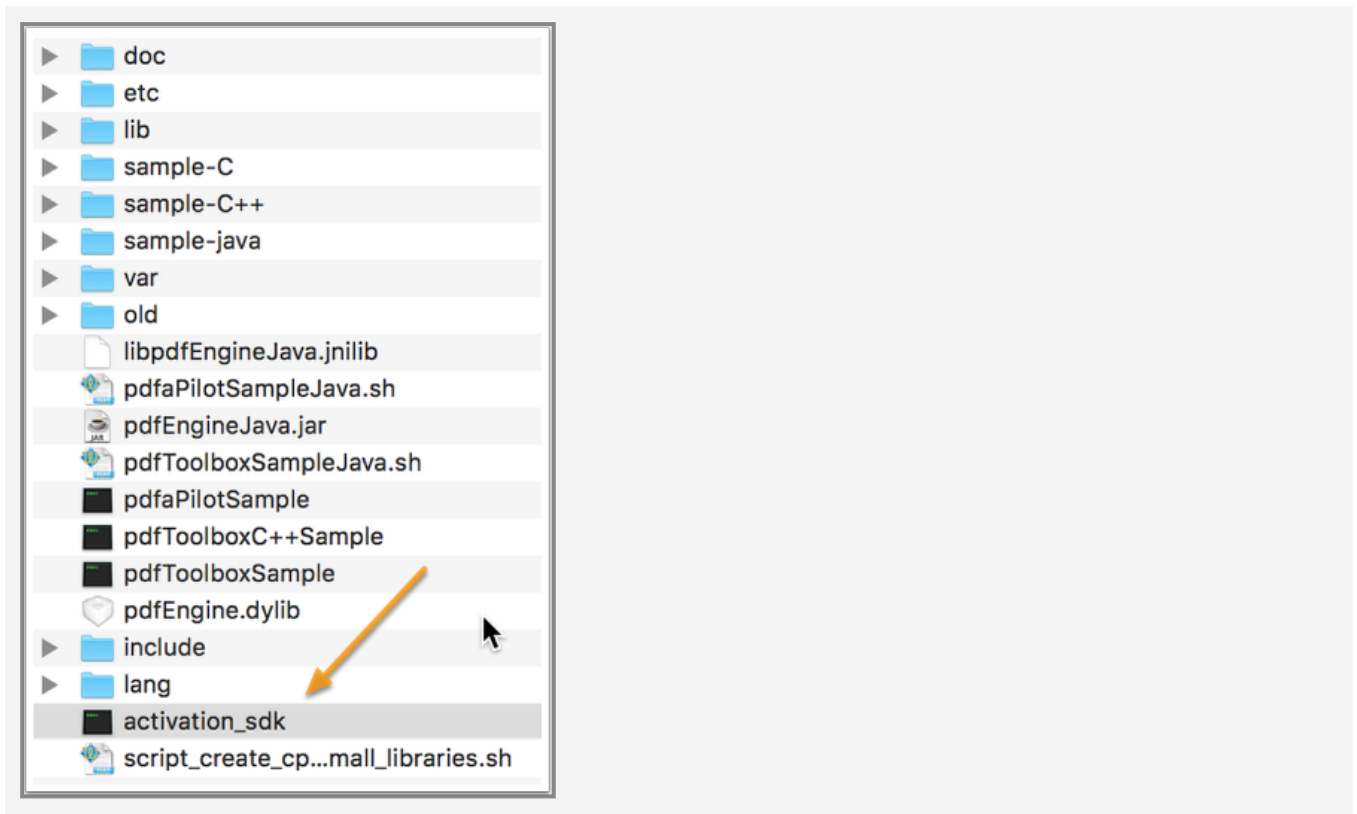
If files, that are not already a PDF file, are to be processed as input files (such as images or even Office files), the program tries to convert them automatically, if the format is supported.

Please also note the following article: [Requirements for conversion to PDF](#)

8. callas pdfaPilot SDK

8.1 Activation and Deactivation of pdfaPilot SDK

After downloading and unarchiving the pdfaPilot SDK folder you will find a command line tool "activation_sdk" in the library folder.




You should also have received a License PDF together with the download path for the library. (If you do not have a License PDF, please send an email to sales@callasoft-ware.com.) The License PDF contains your License key as well as some other information.

tivation server. After you have followed these instructions, the activation server will send an **Activation PDF** back to you. This you should then install using the `--activation` parameter and the **Activation PDF**:

```
./activation_sdk --activate <Path to Activation PDF>
```

Example:

```
./activation_sdk --activate /path/to/file/Activation.pdf
```

-  The License Key (as printed on the License PDF) is needed in each initialization call to the library as described in the SDK documentation. See first steps in the next article...

Similarly, in order to deactivate the license, use the `--deactivate` parameter and the **Activation code**:

```
activation_sdk --deactivate <activation code>
```


8.2 callas pdfEngine SDK: First steps

callas pdfEngine SDK is the name of callas' software development kit package name. The product related features (pdfToolbox or pdfaPilot) are activated using a license. This package includes full documentation and sample codes and contains executables and source code for all supported environments (C/C++, C# and Java).

To know more about the file components inside the package, go to [File components and their use in pdfaPilot SDK](#) article.

Executing a sample using callas pdfEngine SDK

Please note: This article refers to pdfToolboxSample but pdfaPilotSample functions in the same way.

Once you have [activated](#) your SDK using a license, you can run a sample, in this example 'pdfToolboxSample'. You can see below all the pre defined modes, along with the arguments that can be executed.

For example, pdfToolboxSample <keycode> --listlanguages

```

Akashs-MBP:~ a.choudhary$ /Users/a.choudhary/Downloads/callas_pdfEngine_SDK/2/pdfToolboxSample
callas pdfEngine SDK 9.4.445 (x64)
x64: yes
ThreadSafe: no
argv[0] = /Users/a.choudhary/Downloads/callas_pdfEngine_SDK/2/pdfToolboxSample
USAGE:
pdfToolboxSample <keycode> [-DL addon keycode] <kfpx> <pdf> <dest file> [-report file] <flags>]
flags:
 1: Deactivates optimization of the internal structure when saving the PDF
 4: Analyzes image pixels for plate count
 8: Analyze only; Fixups will not be applied
16: Embed a Preflight Certificate after processing
pdfToolboxSample <keycode> --importacrobatsprofiles <dest folder>
pdfToolboxSample <keycode> --listlanguages
pdfToolboxSample <keycode> --listfontnames
pdfToolboxSample <keycode> --listvariables <kfpx>
pdfToolboxSample <keycode> --impose <runlist file> <sheetconfig file> <pdf> <dest file>
pdfToolboxSample <keycode> --processconversion <cfg_source> <cfg_spot> <cfg_dest> <pdf> <dest file>
pdfToolboxSample <keycode> --booklet <pdf> <bOutmarks> <dest file>
pdfToolboxSample <keycode> --rup <pdf> <times> <htimes> <distance> <bOutmarks> <dest file>
pdfToolboxSample <keycode> --fillpage <pdf> <pagewidth> <pageheight> <distance> <bOutmarks> <dest file>
pdfToolboxSample <keycode> --readerspreads <pdf> <dest file>
pdfToolboxSample <keycode> --splithalf <pdf> <dest file>
pdfToolboxSample <keycode> --stereopair <pdf> <vtimes> <htimes> <distance> <bOutmarks> <dest file>
pdfToolboxSample <keycode> --slice <kfpx> <pdf> <destUpper> <destLower>
pdfToolboxSample <keycode> --presentation <pdf> <transition> <selfrunning> <bFullscreen> <bBlackpage> <bProgress> <dest file>
transition:
 0: use blinds
 1: use box
 2: use comb
 3: use cover
 4: use dissolve
 5: use fade
 6: use glitter
 7: use push
 8: use random
 9: use replace
10: use split
11: use uncover
12: use wipe
13: use zoom in
14: use zoom out
pdfToolboxSample <keycode> --handout <pdf> <slotconfig> <bNotes> <bSlidefirstpage> <pageformat> <dest file>
slotconfig:
 0: 2
 1: 3
 2: 2x2
 3: 2x3 (3x2 with notes)
pageformat:
 0: NMA
 1: letter
pdfToolboxSample <keycode> --passepartout <pdf> <background> <borderwidth> <dest file>
pdfToolboxSample <keycode> --lighttable <pdf> <pagewidth> <pageheight> <columns> <background> <dest file>
pdfToolboxSample <keycode> --overlay <pdf> <overlay-pdf> <placement> <h-offset> <v-offset> <dest file>
placement:
 0: top left
 1: top center
 2: top right
 3: center left
 4: center
 5: center right
 6: bottom left
 7: bottom center
 8: bottom right
pdfToolboxSample <keycode> --extractmpmetadata <pdf> <cfg> <xml>
pdfToolboxSample <keycode> --extracticprofile <pdf>

```


8.3 Help? Displaying program information for pdfaPilot SDK

If you use the command line tool on command line with `--help`, you see the overview of options about all available commands for processing:

```
./activation_sdk --help
```

To get an overview about all the available options for a specific command, run:

```
./activation_sdk --help <command>
```

Commands like:

- `--version` show version information
- `--activate` Activate license
- `--deactivate` Deactivate license

8.4 File components and their use in the SDK

Package too big? You can remove some components from the SDK and reduce the size of the software. Here is what you can delete without hampering the outcome of the software.



What to keep

For a smoother usage of the SDK, we recommend that you keep the following components

All			
Platform	Dependencies	Configuration	Language
Windows	all *.dll, .exe and *.ppi	etc	lang
OS X	lib	etc	lang
Unix	lib	etc	lang

Java Wrapper	
Windows	pdfEngineJava.jar and pdfEngineJava.dll
OS X	pdfEngineJava.jar and libpdfEngineJava.jnilib
Unix	pdfEngineJava.jar and libpdfEngineJava.so

.NET Wrapper	
Windows	pdfEngineDotNet.dll

.NET Core Wrapper (starting pdfEngine 12)	
Windows	EngineDotNetCore.*
MacOS	EngineDotNetCore.*
Linux	lib/EngineDotNetCore.*

What can be removed

The following components could be safely removed to reduce the delivery size.

Component	Functionality
doc	Documentation
include	C/C++ API Header
sample-C	C API Sample
sample-C++	C++ API Sample
sample-DotNet	.NET Sample
sample-java	Java Sample
var	Predefined configuration files (e.g. Profiles, Templates)

Functionality based

To further reduce the package size, the following subfolders can be removed from the 'etc' folder- depending on your requirements.

Sub-folder	Functionality
etc/Actions/Imposition	If no Arrange action is used
etc/Actions/LFP	If no Tiling or Grommet action is used
etc/Actions/PlaceContents	If no PlaceContent Fixups OR If no Place Barcode or Text Fixups are used
etc/APDFL	If no font embedding or no PDF/A conversion is used (or font situation is clear)
etc/Backgrounds	If no layer/image mask report is used
etc/Certify	If no Preflight certificate should be embedded
etc/ColorConversion	If no color conversion is used
etc/FontSubstitution	If no font substitution is used (for PDF/A)
etc/HtmlConverter	If no PDF report based on HTML template is used
etc/Inventory	If no inventory report is used
etc/MailConverter	If no emails are processed
etc/OCRTool	If no OCR needs to be done to text (Fixup: Create invisible text via OCR)
etc/PDFAExtSchema	If not PDF/A metadata entry needs to be embedded
etc/PDFOfficeTool	If no Office-files are processed
etc/PDFPSTool	If no PostScript-files are processed
etc/pmime	If no unknown files or wrong extensions are processed
etc/Reports	If no PDF/A-HTML Report or ZUGFeRD or CxF is used
etc/UnpackTool	If no archives (.zip) are processed or Profiles with Fixup

Sub-folder	Functionality
	"Place content on page" or Protected Profiles are used
etc/Variables	If no JavaScript post mortem debugger is used
etc/Visualizer	If no Comparison is used
etc/QuickTools	If no QuickFix or QuickCheck is required