rspub-gui Documentation

Release 1.0.0

h

Mar 08, 2018

Contents

E H R I
EUROPEAN HOLOCAUST
RESEARCH INFRASTRUCTURE

CHAPTER 1

About

Metadata Publishing Tool (MPT) is a desktop application that facilitates the publishing of *resource*s and *sitemap*s in conformance with the *ResourceSync Framework Specification*.

Metadata Publishing Tool, rspub-gui and rspub-core were developed by Data Archiving and Networked Services (*DANS*-KNAW) under auspices of the European Holocaust Research Infrastructure (*EHRI*).

The documentation in this rtd is intended for end users and system administrators.

- Download latest executables from the releases page of the rspub-gui project.
- Source location: https://github.com/EHRI/rspub-gui.
- The GUI is based on rspub-core. See https://github.com/EHRI/rspub-core.
- In case of questions contact the EHRI team.

CHAPTER 2

Introduction

Introduction to Metadata Publishing Tool

Note: Introducing the EHRI use case

"The mission of the European Holocaust Research Infrastructure (*EHRI*) is to support the Holocaust research community by building a digital infrastructure and facilitating human networks. *EHRI* provides online access to information about dispersed sources relating to the Holocaust through its Online Portal, and tools and methods that enable researchers and archivists to collaboratively work with such sources."

– The EHRI web site

In order to collect information on dispersed archives across Europe -and the rest of the world- that hold Holocaust-related collections the Encoded Archival Description (*EAD*) is used as the format for information dissemination; the *ResourceSync* Framework may be used as the transportation and *synchronization* mechanism.

Metadata Publishing Tool is the subject of this documentation. Examples in this introduction and the rest of the documentation will sometimes use the *EHRI* use case to describe the working of the tool.

2.1 ResourceSync

The *ResourceSync Framework Specification* describes a *synchronization* framework for the web consisting of various capabilities that allow third-party systems to remain *synchronize*d with a server's evolving *resource*s. More precisely the *ResourceSync* Framework describes the communication between *Source* and *Destination* aimed at synchronizing one or more *resource*s. Communication utilizes *http* and an extension on the *sitemap protocol*, an xml-based format for expressing metadata, relevant for *synchronization*.

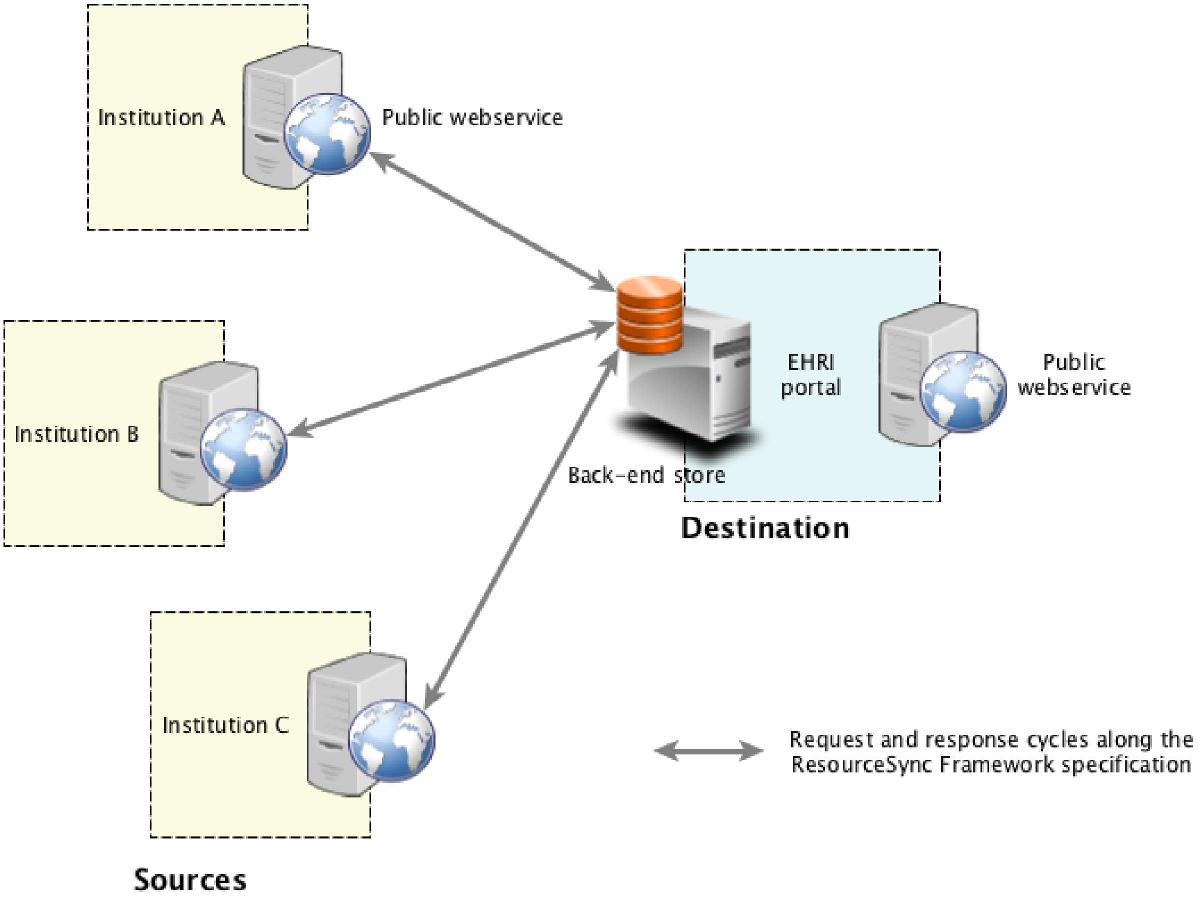


Fig. 2.1: *Fig. 1. External logistics. The ResourceSync Framework Specification at work. Collection Holding Institutions expose content and ResourceSync metadata on their web servers. The central hub (in this case the EHRI Portal) is actively collecting resources and keeping them in sync with the aid of published sitemaps.*

We can say that the *ResourceSync* Specification is a perfect fit for solving the **external logistics** when it comes to synchronizing *resource*s between a central *Destination* and various *Source*s. *Figure 1.* depicts the external logistics.

When the *resource*s we are trying to *synchronize* are not web-resources by them selves but instead stem from information systems, databases or other places within an organization, we are faced with other problems, which we can qualify as related to **internal logistics**.

2.2 Metadata Publishing Tool

Metadata Publishing Tool is an application that solves various problems related to the **internal logistics**:

- How do we collect and import *resource*s from various places within the organization;
- How do we select relevant *resource*s;
- How do we create *ResourceSync sitemap* metadata on relevant *resource*s;
- How do we export *resource*s and *sitemap*s to the web server;
- How do we verify that the exposed URL's are correct and our *ResourceSync* site ready to be harvested by a *Destination*.

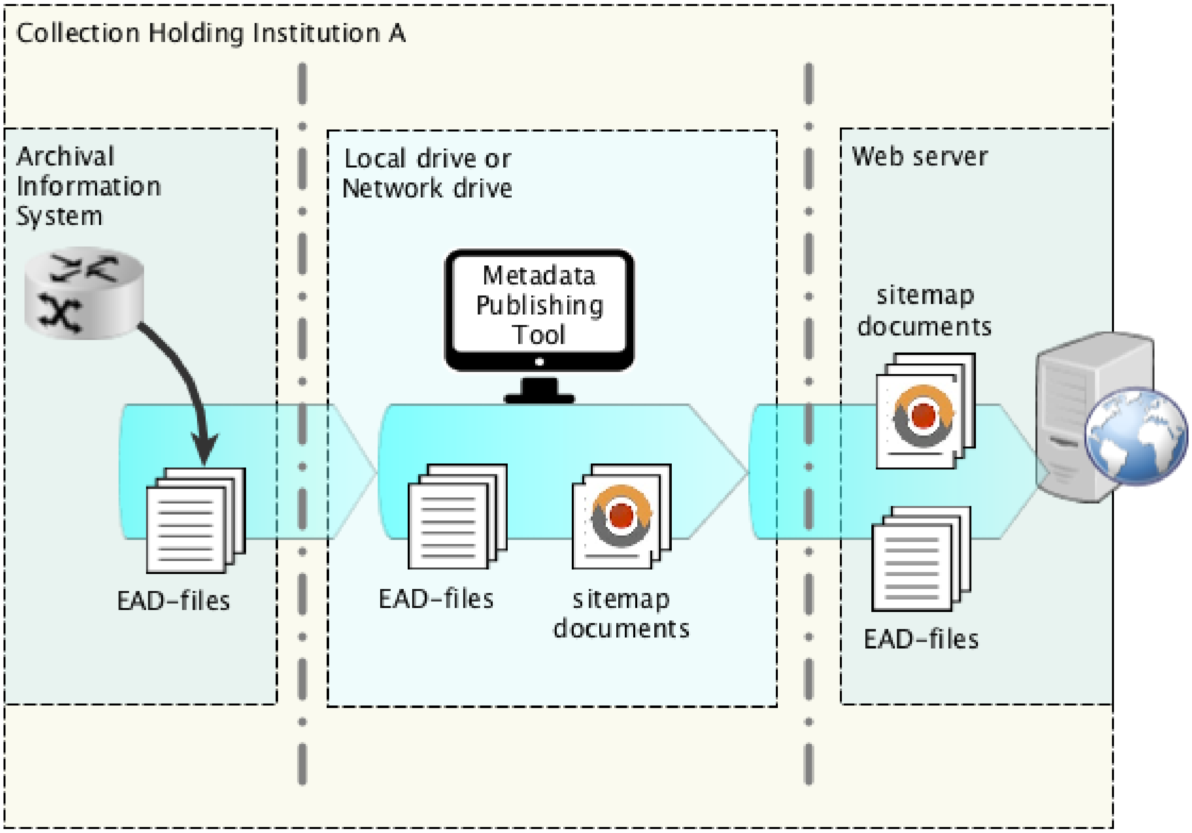


Fig. 2.2: *Fig. 2. Internal logistics. Metadata Publishing Tool at work.*

Figure 2. depicts **internal logistics** and the role of *Metadata Publishing Tool*. The situation described may be exemplary for Collection Holding Institutions (CHI's) within the EHRI infrastructure, although different situations may equally be applicable. *Metadata Publishing Tool* is an application that is deployed on your laptop or local work station. From there you collect and select *resource*s, create the *ResourceSync sitemap*s, export *resource*s and *sitemap*s to your web server and verify the exposed URL's.

Configuration of *Metadata Publishing Tool* may need the hand and insight of a technically skilled person. Once configured it can be managed by archivists and other content-savvy users that do not necessarily have technical skills.

2.3 This documentation

This documentation starts with describing how to install *Metadata Publishing Tool* on various operating Systems in the *Installation* pages. Each tab or wizard page of the application is the subject of the other chapters. There are *Help* buttons on each tab or wizard page that link to the relevant chapters in this documentation. Finally there is a *Glossary* on terms used throughout this documentation. .. and an appendix that handles various use cases that may be applicable to your situation on your institution.

CHAPTER 3

Installation

3.1 Windows installer

Install Metadata Publishing Tool on Windows using a wizard

Attention: If a previous release of *Metadata Publishing Tool* was installed on your machine, first uninstall this older release. See *Uninstall Metadata Publishing Tool*.

Download the latest `MPT_win_installer.exe` from the release page of the rspub-gui project on gitHub.

3.1.1 Circumvent security

After download, double-click the downloaded `MPT_win_installer.exe`. Because the software has not been code-signed by a trusted authority, security settings on your machine may object to executing it right away.

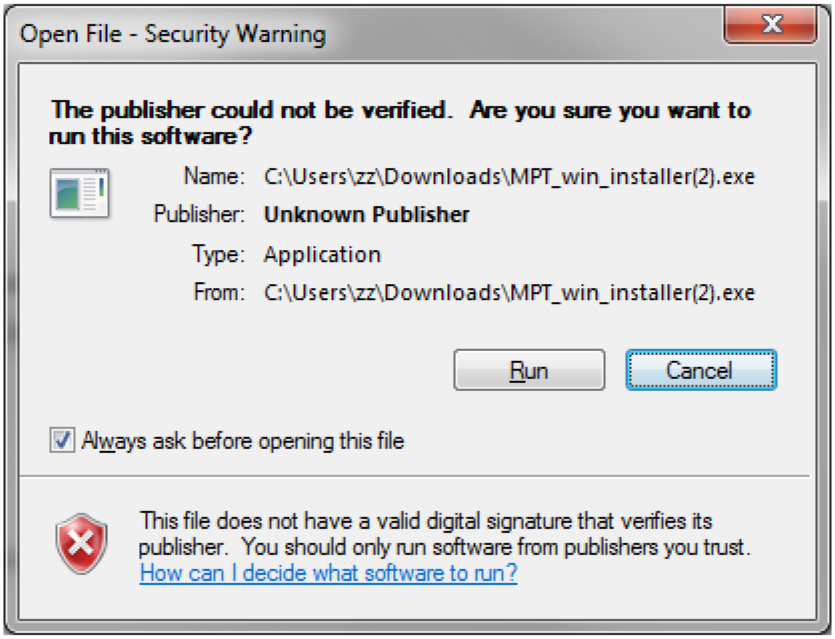
Windows 7

On Windows 7 you may see a sign similar to this.

Click *Run* and proceed to *Run the install wizard*.

Windows 10

On Windows 10 a blue warning may appear.



Open File - Security Warning

The publisher could not be verified. Are you sure you want to run this software?

Name: C:\Users\zz\Downloads\MPT_win_installer(2).exe

Publisher: Unknown Publisher

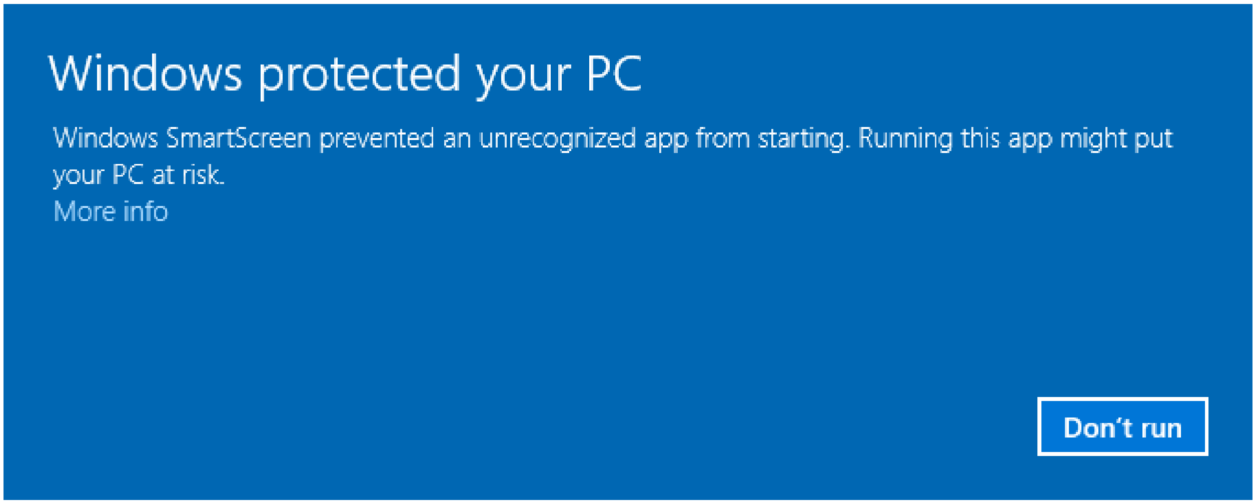
Type: Application

From: C:\Users\zz\Downloads\MPT_win_installer(2).exe

☑ Always ask before opening this file

This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust.
How can I decide what software to run?

Fig. 3.1: Security warning on Windows 7

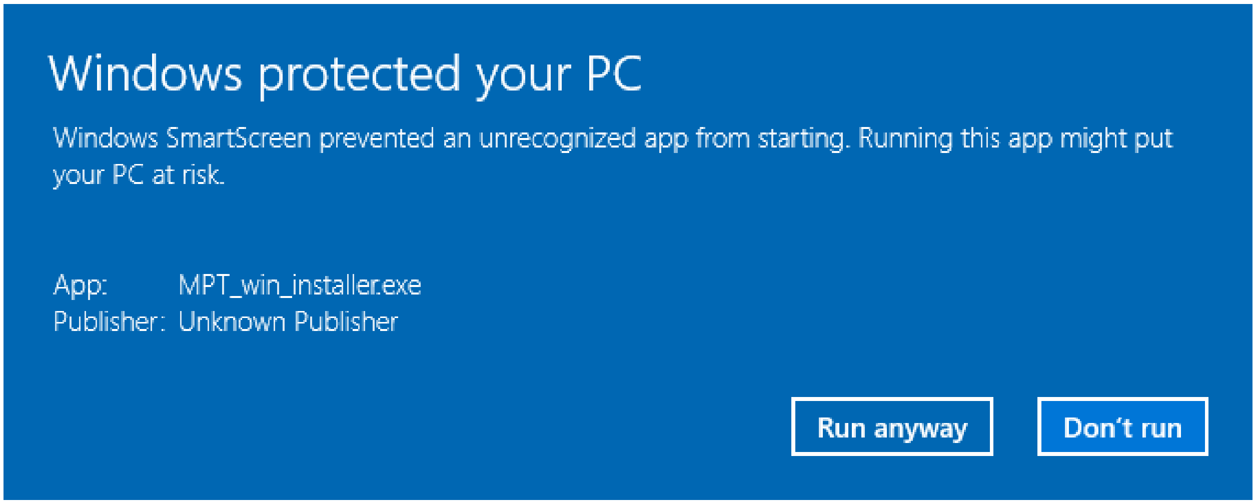


Windows protected your PC

Windows SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.
More info

Fig. 3.2: Security warning on Windows 10

Click *More info* on the left side of this blue screen. . .



Windows protected your PC

Windows SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

App: MPT_win_installer.exe
Publisher: Unknown Publisher

Fig. 3.3: Security warning on Windows 10, second screen

Click *Run anyway* and proceed to *Run the install wizard*.

3.1.2 Run the install wizard

The install wizard has several screens and will guide you through the installation process. On average, you can accept the predefined values.

Welcome message

The wizard opens with a welcome message. Click *Next*.

Installation folder

Choosing the installation folder. The default value is `C:\Program Files (x86)\EHRI\MetadataPublishingTool`. Choose *Browse. . .* if you want a different installation folder, choose *Next* when done.

Select components

Metadata Publishing Tool has only one component. So leave the component selected and click *Next*.

License agreement

Code and executables of the *Metadata Publishing Tool* are distributed under the Apache License 2.0. You must accept the license agreement before you may proceed with the install wizard. Click *Next* when done.

Start Menu shortcuts

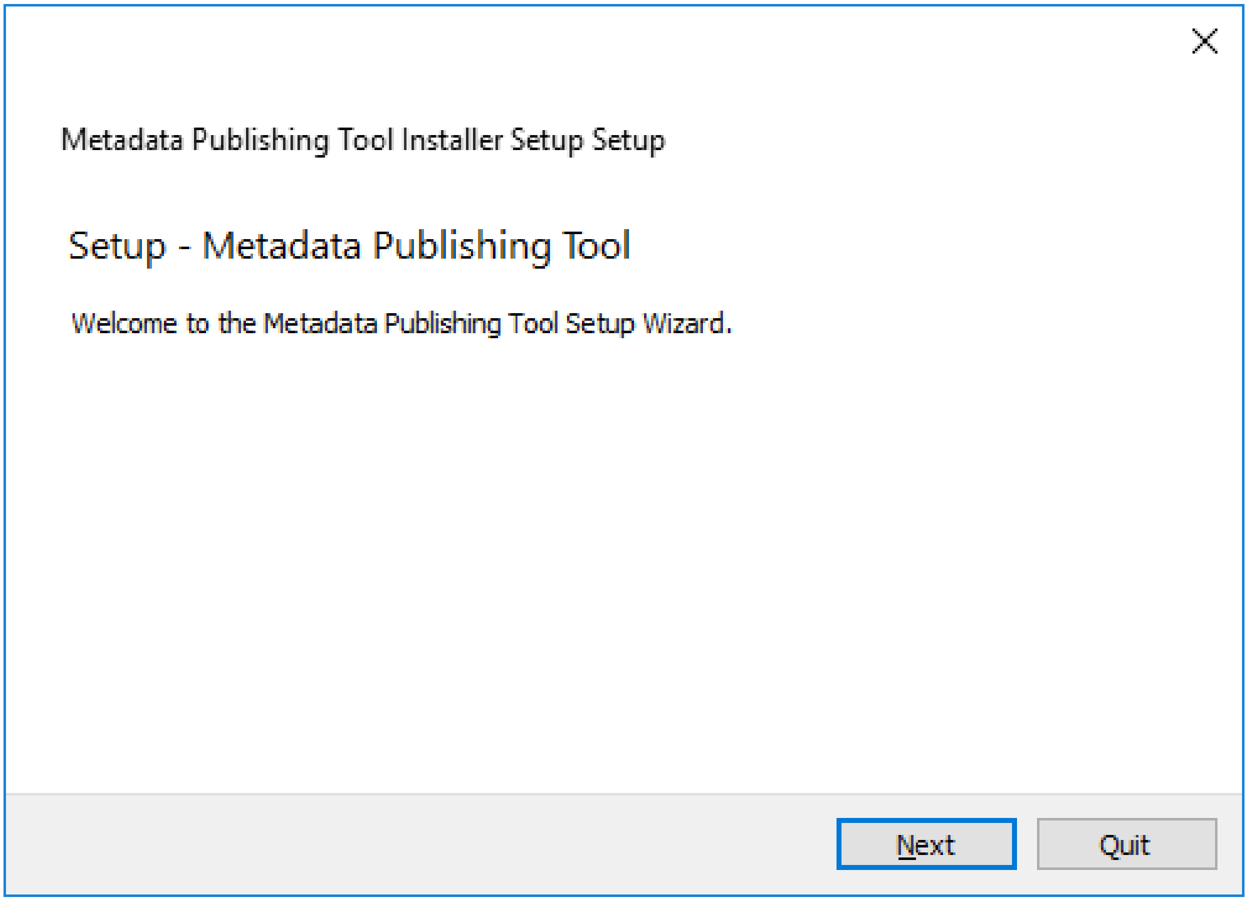
Default a shortcut to the `MetadataPublishingTool.exe` will be placed in the program group *EHRI*. Click *Next* when done.

Ready to install

Click *Next* to continue. A security warning may again appear, telling you are about to install software from an unknown publisher. Click *Yes* to proceed.

Installing Metadata Publishing Tool

The installation process will now begin. An indication of progress will be shown.

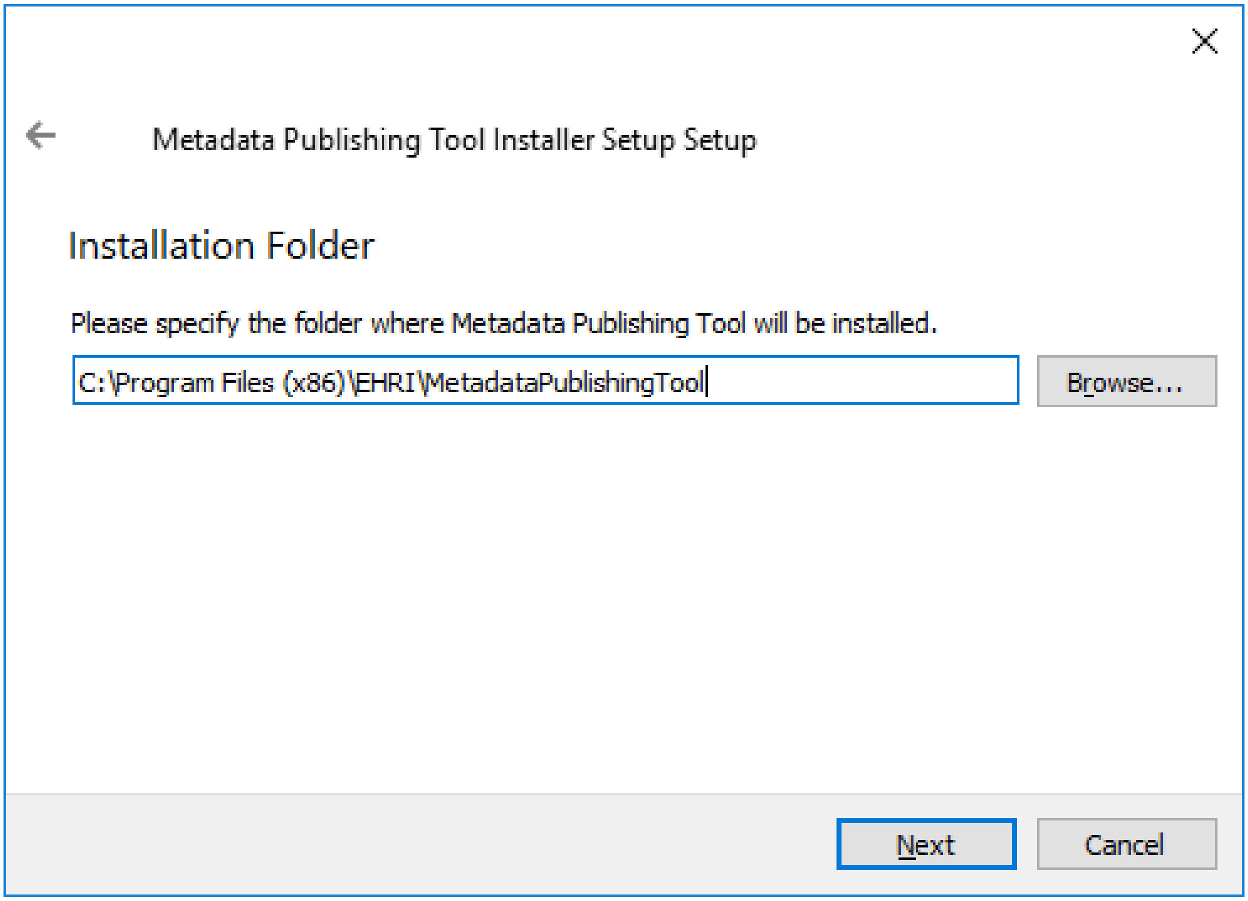


Metadata Publishing Tool Installer Setup Setup

Setup - Metadata Publishing Tool

Welcome to the Metadata Publishing Tool Setup Wizard.

Fig. 3.4: Welcome message of the install wizard

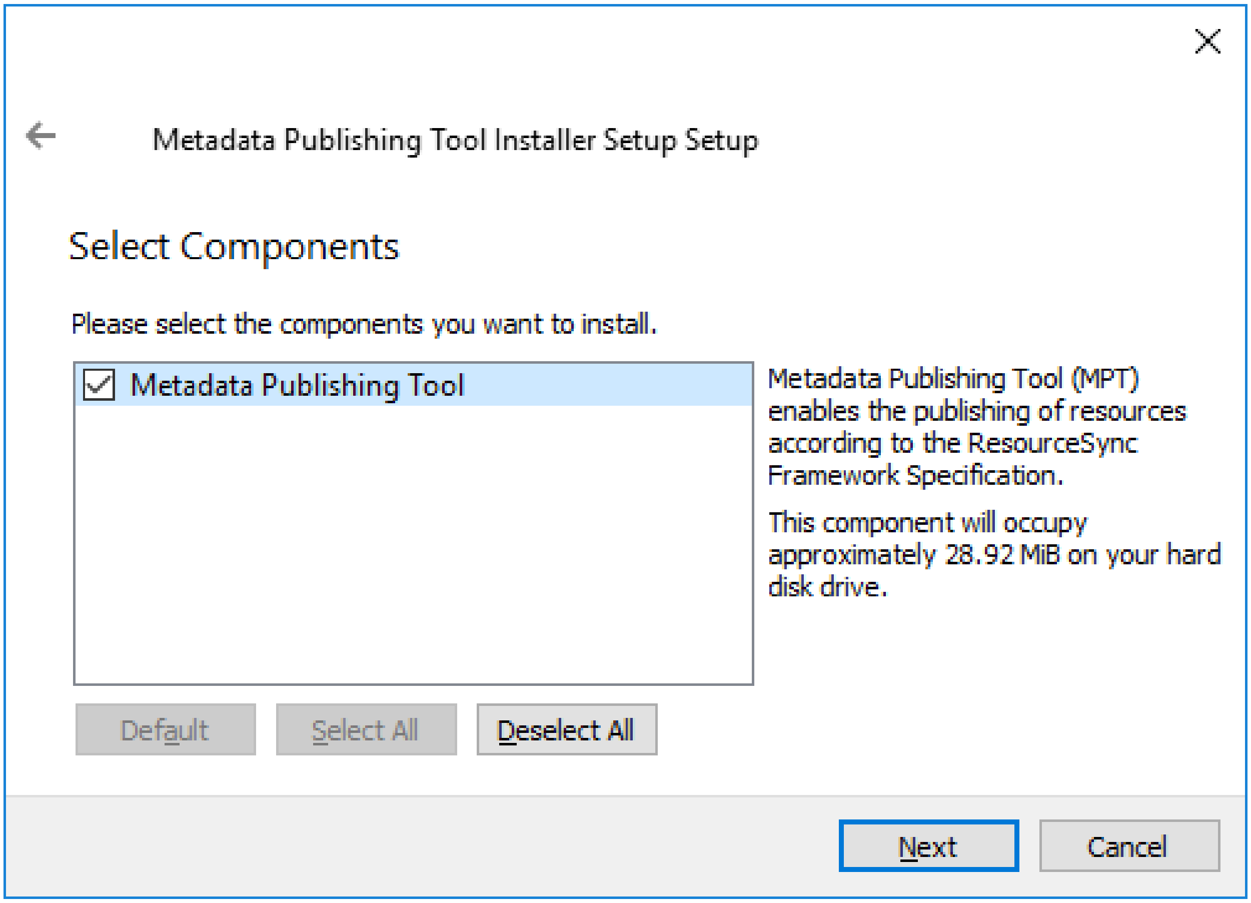


Installation Folder

Please specify the folder where Metadata Publishing Tool will be installed.

C:\Program Files (x86)\EHRI\MetadataPublishingTool

Fig. 3.5: Choosing the installation folder



Metadata Publishing Tool Installer Setup Setup

Select Components

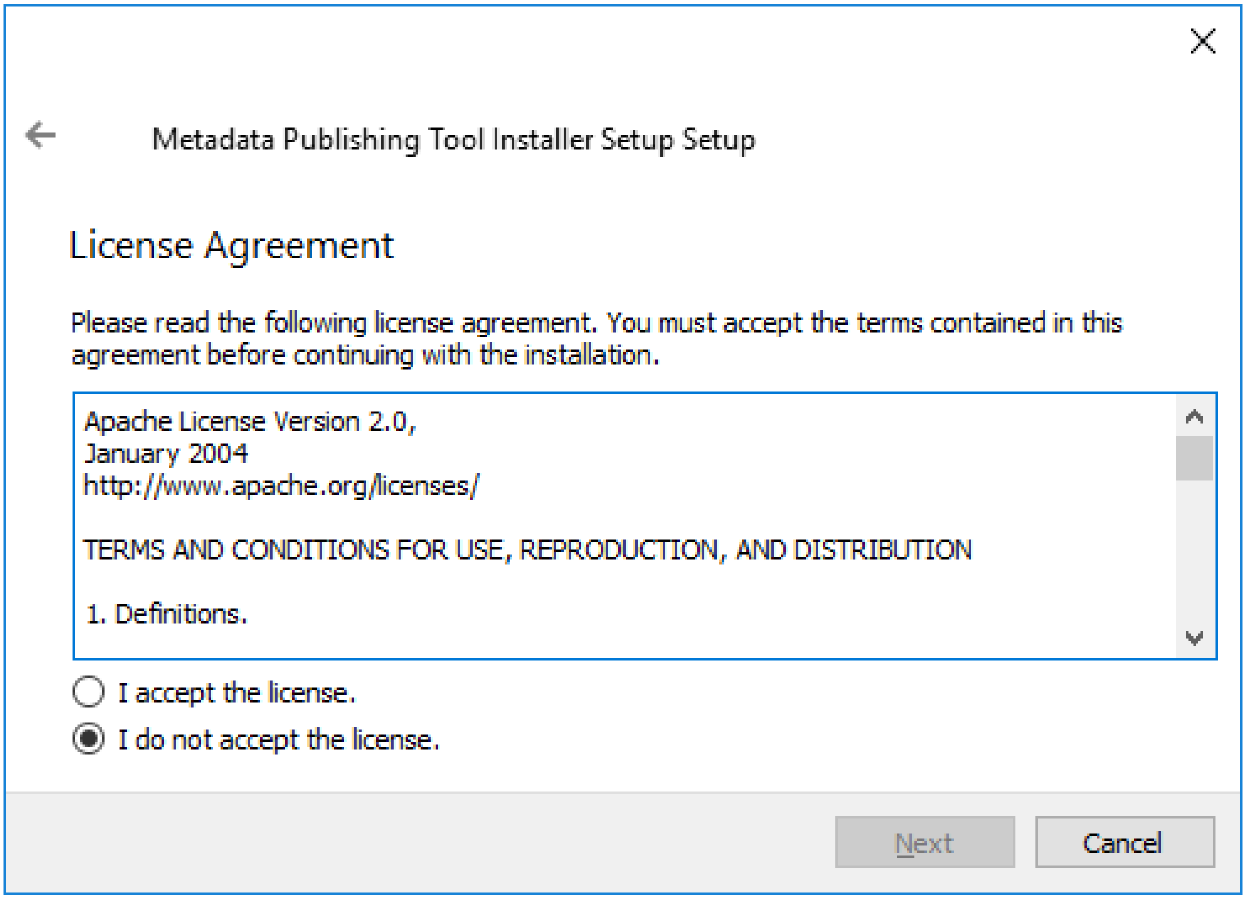
Please select the components you want to install.

☑ Metadata Publishing Tool

Metadata Publishing Tool (MPT) enables the publishing of resources according to the ResourceSync Framework Specification.

This component will occupy approximately 28.92 MiB on your hard disk drive.

Fig. 3.6: Selecting components



Metadata Publishing Tool Installer Setup Setup

License Agreement

Please read the following license agreement. You must accept the terms contained in this agreement before continuing with the installation.

Apache License Version 2.0,
January 2004
http://www.apache.org/licenses/

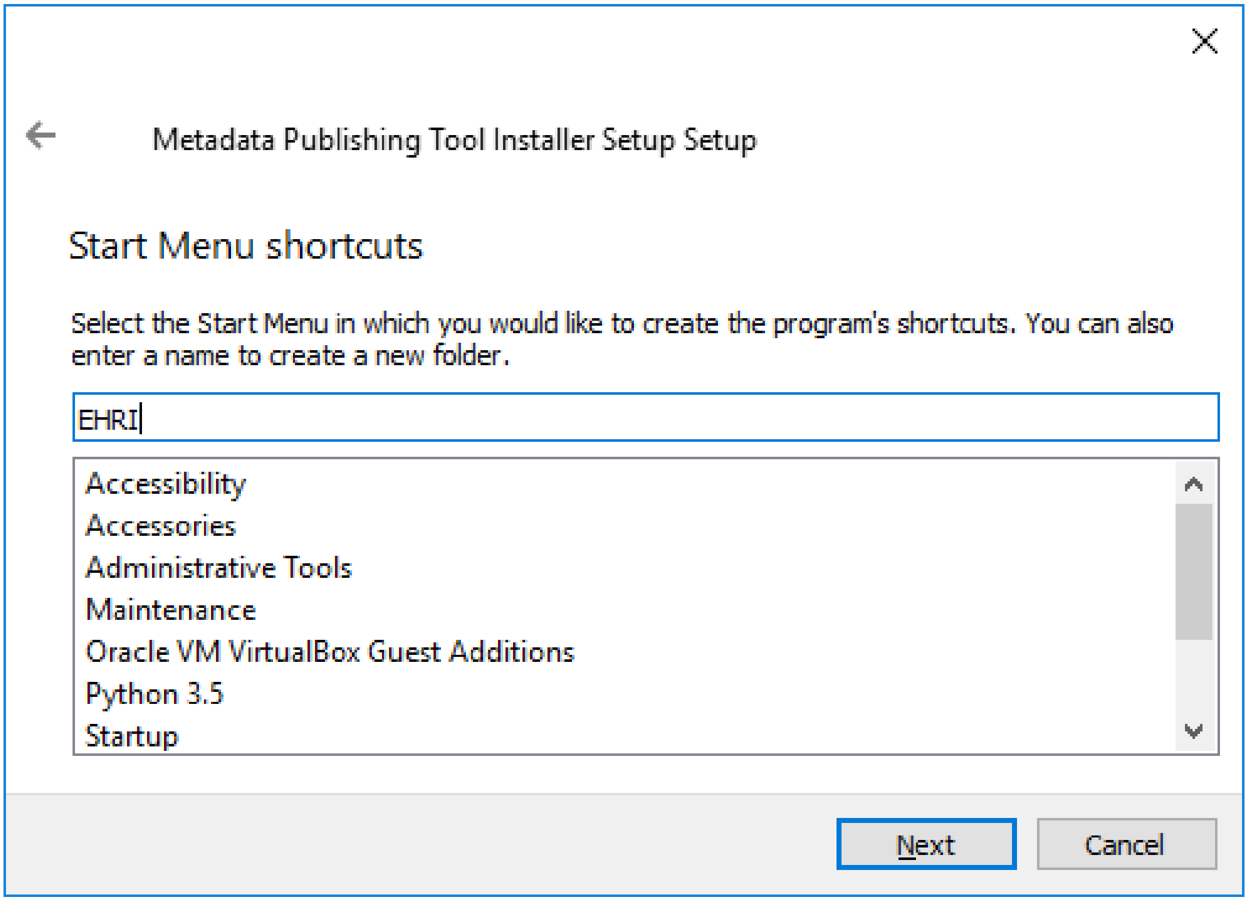
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

○ I accept the license.

● I do not accept the license.

Fig. 3.7: Inspecting the license



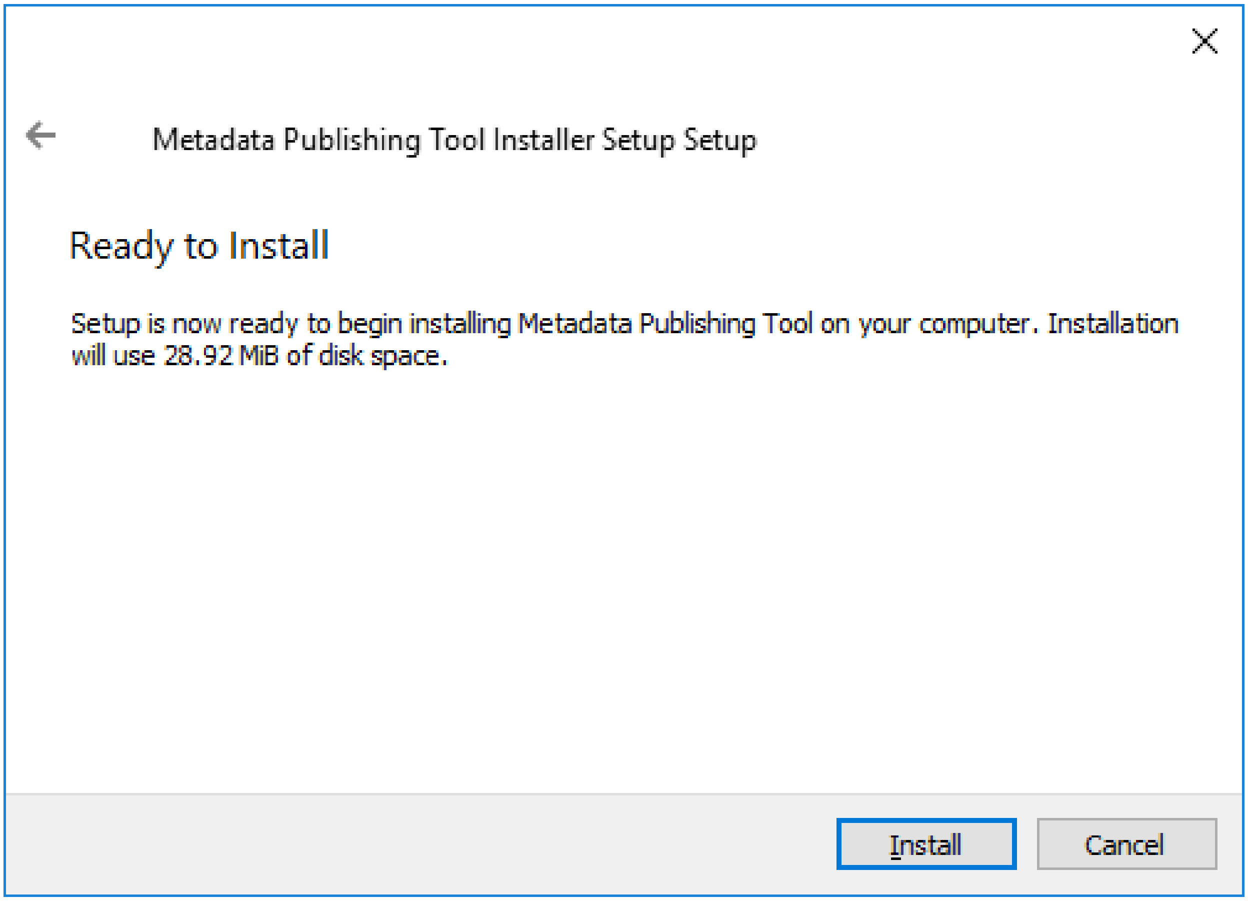
Metadata Publishing Tool Installer Setup Setup

Start Menu shortcuts

Select the Start Menu in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

EHRI

Fig. 3.8: Setting shortcuts

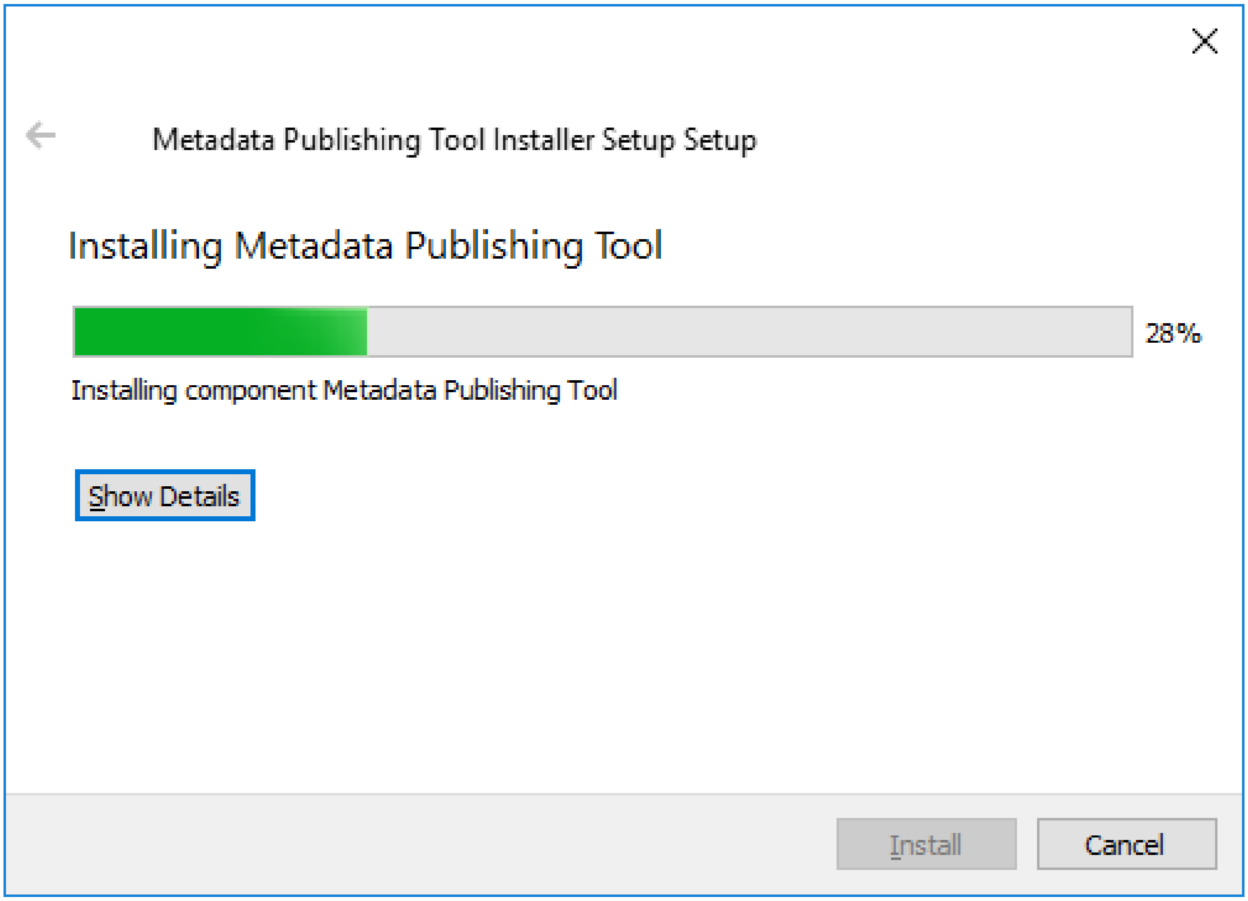


Metadata Publishing Tool Installer Setup Setup

Ready to Install

Setup is now ready to begin installing Metadata Publishing Tool on your computer. Installation will use 28.92 MiB of disk space.

Fig. 3.9: Ready to install


Metadata Publishing Tool Installer Setup Setup

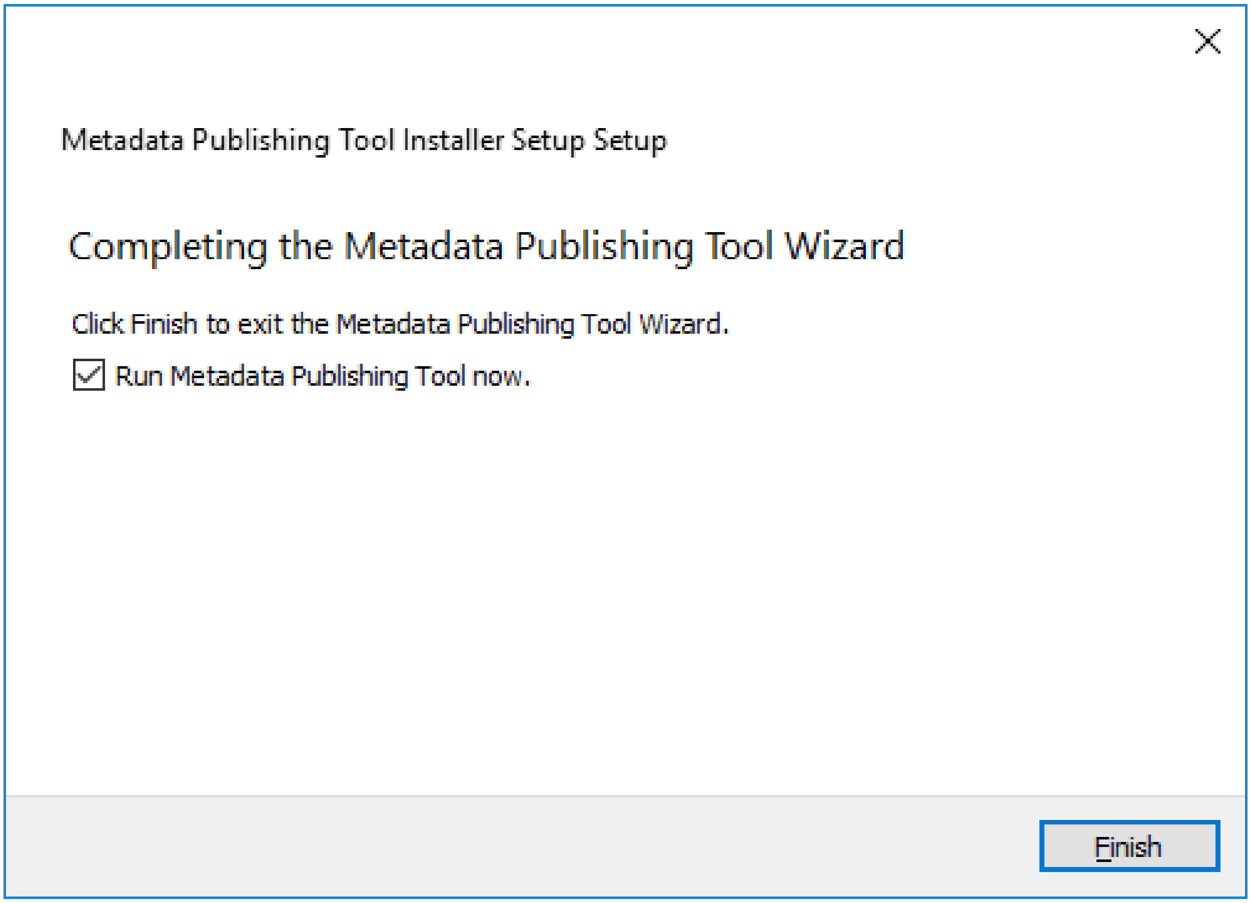
Installing Metadata Publishing Tool

28%

Installing component Metadata Publishing Tool

Fig. 3.10: Progress of installation

Completing the Metadata Publishing Tool Wizard



Metadata Publishing Tool Installer Setup Setup

Completing the Metadata Publishing Tool Wizard

Click Finish to exit the Metadata Publishing Tool Wizard.

☑ Run Metadata Publishing Tool now.

Fig. 3.11: Completing the installation

After installation has completed, the last screen of the installation wizard is shown. Click *Finish* to exit the wizard.

3.1.3 Start Metadata Publishing Tool

After completing the installation wizard you can find shortcuts to `MetadataPublishingTool.exe` and the `MPT_MaintenanceTool.exe` in the group *EHRI* of your Start Menu. Alternatively you may find the executables in the folder `C:\Program Files (x86)\EHRI\MetadataPublishingTool` (if you did not change the Installation folder during installation).

3.1.4 Uninstall Metadata Publishing Tool

The *Metadata Publishing Tool* and its components can be removed from your machine by the *MPT Maintenance Tool*. Click the shortcut *MPT_MaintenanceTool* in the *EHRI* group of your Start Menu or navigate to `C:\Program Files (x86)\EHRI\MetadataPublishingTool` and double-click `MPT_MaintenanceTool.exe`. The uninstall wizard will begin.

Click *Next* to continue.

Click *Uninstall*. After the uninstall process has run click *Finish*. The folder `MetadataPublishingTool` and its contents are now removed from your system. The folder `EHRI` may still remain with other executables or is empty if

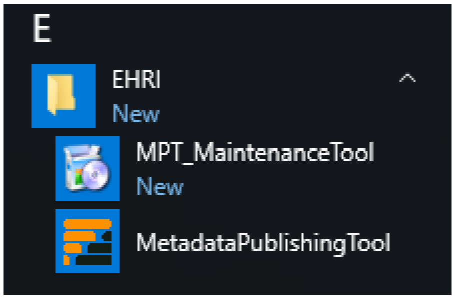
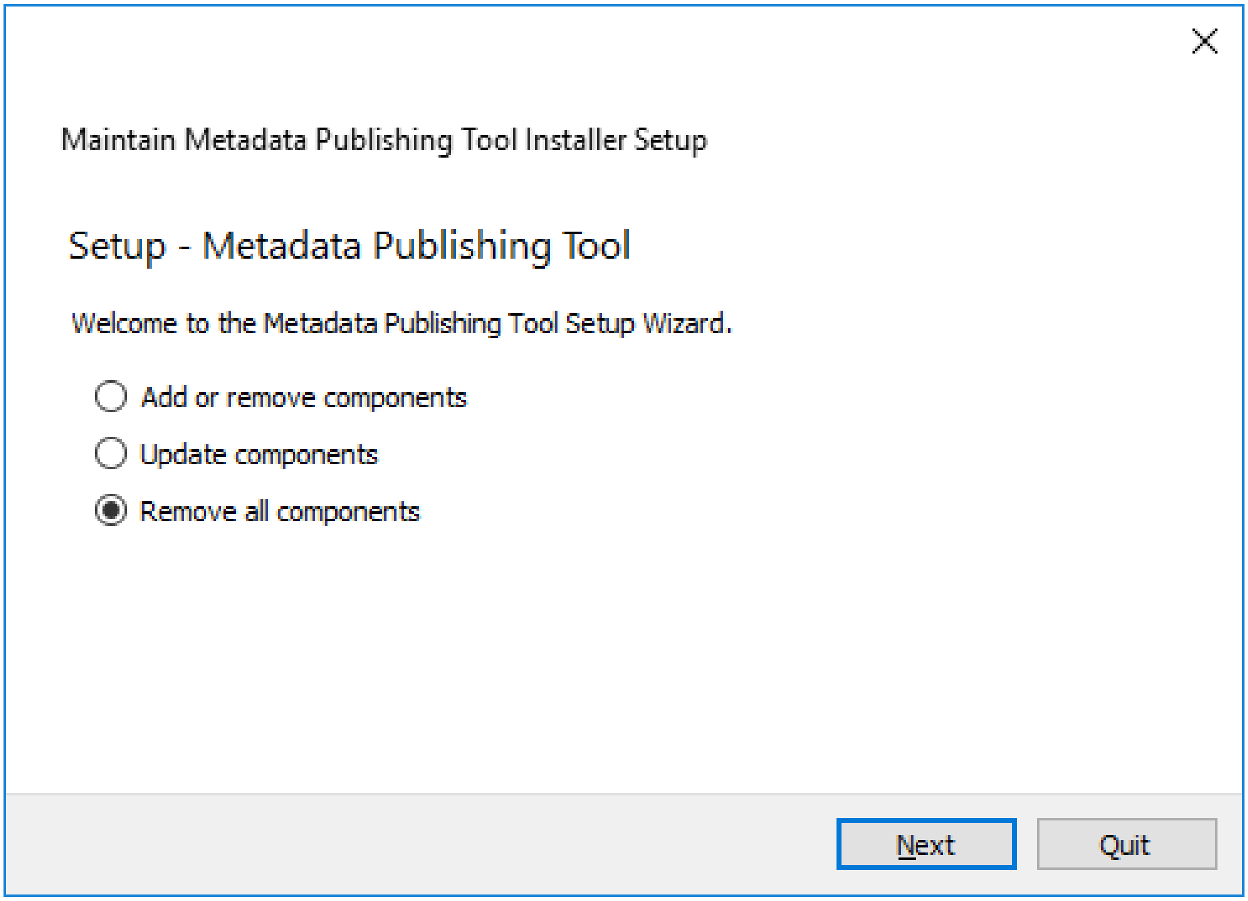


Fig. 3.12: Shortcuts to Metadata Publishing Tool and the MPT Maintenance Tool in the EHRI group



Maintain Metadata Publishing Tool Installer Setup

Setup - Metadata Publishing Tool

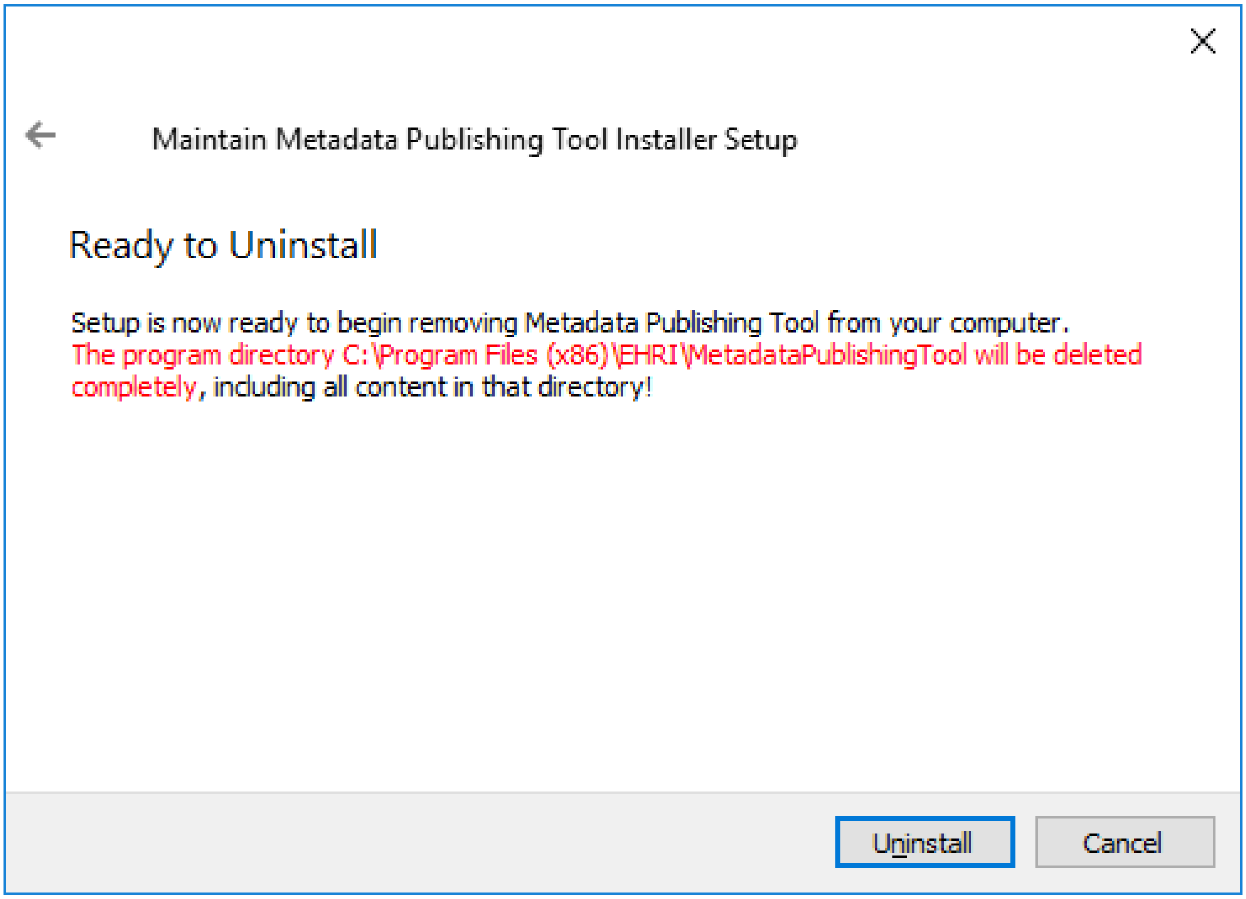
Welcome to the Metadata Publishing Tool Setup Wizard.

○ Add or remove components

○ Update components

● Remove all components

Fig. 3.13: Removing Metadata Publishing Tool



Maintain Metadata Publishing Tool Installer Setup

Ready to Uninstall

Setup is now ready to begin removing Metadata Publishing Tool from your computer.
The program directory C:\Program Files (x86)\EHRI\MetadataPublishingTool will be deleted completely, including all content in that directory!

Fig. 3.14: Ready to uninstall

you did not install other programs in there.

3.2 Windows folder installation

Install Metadata Publishing Tool and its dependencies as separate files in a folder

- *Download Metadata Publishing Tool*
- *Start Metadata Publishing Tool*

Install the *Metadata Publishing Tool* and its dependencies in a folder. Advantages of this kind of installation: faster start-up time.

3.2.1 Download Metadata Publishing Tool

Download the latest `MetadataPublishingTool.zip` from the release page of the rspub-gui project on gitHub. Unzip and place anywhere on your file system.

3.2.2 Start Metadata Publishing Tool

Open the folder `MetadataPublishingTool` and double-click the executable `MetadataPublishingTool.exe`.

The first time you double-click the downloaded executable you may get a warning: unknown app has been detected. To prevent this:

- right-click the executable
- on the pop-up menu click properties
- on the General tab under Security check the Unlock checkbox

3.3 Mac OS installer

Install Metadata Publishing Tool on Mac OS using a wizard

- *Circumvent security*
- *Run the install wizard*
- *Start Metadata Publishing Tool*
- *Uninstall Metadata Publishing Tool*

Attention: If a previous release of *Metadata Publishing Tool* was installed on your machine, first uninstall this older release. See *Uninstall Metadata Publishing Tool*.

Download the latest `MPT_mac_installer.dmg` from the release page of the rspub-gui project on gitHub. After download, double-click the downloaded `MPT_mac_installer.dmg`. This will mount the `MPT_mac_installer` image on your system.

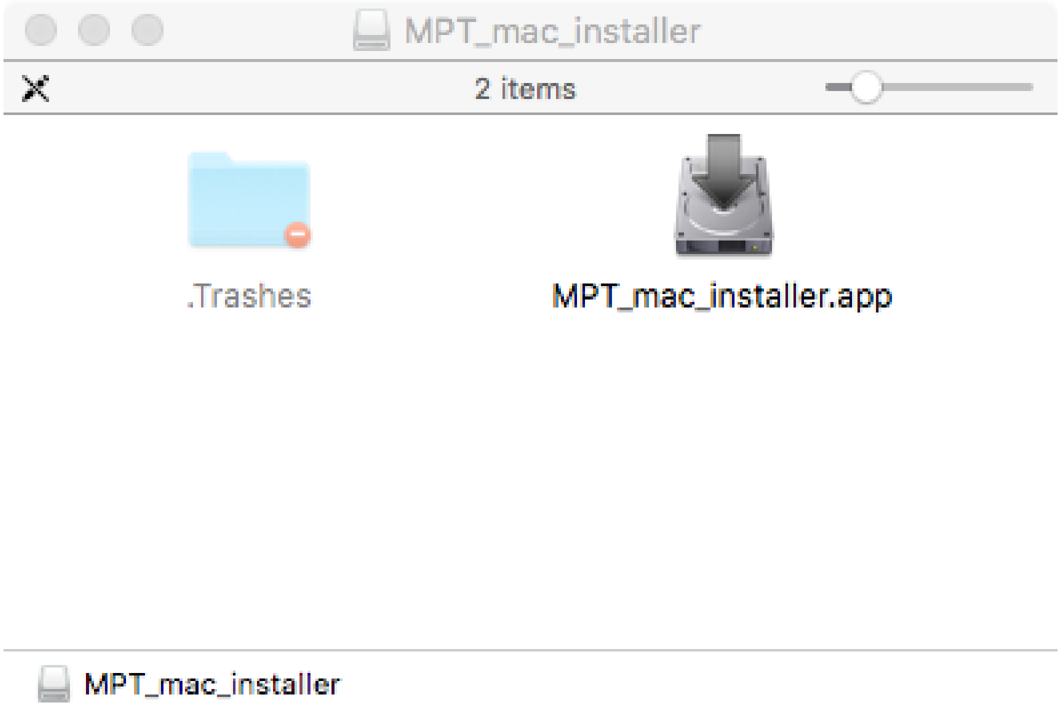


Fig. 3.15: Mounted MPT_mac_installer

Double-click the `MPT_mac_installer.app` in the mount window.

3.3.1 Circumvent security

Because the software has not been code-signed by a trusted authority, security settings on your machine may object to executing it right away.

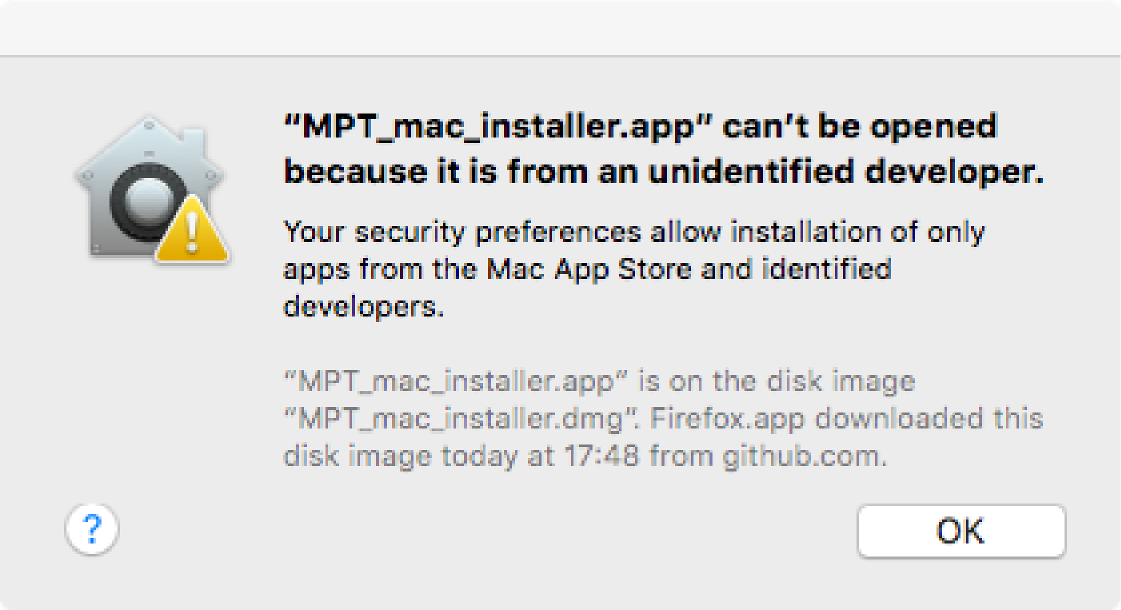
Click *OK*.

Open > *System Preferences* > *Security & Privacy*. You may see the following message.

Click *Open Anyway*.

Alternatively you may have to lower your security settings temporary. Under *Allow apps downloaded from:* click *Anywhere*. After the installation process has completed you may restore your security settings.

After starting the `MPT_mac_installer.app` you may receive a last warning. Click *Open* to start the install wizard.

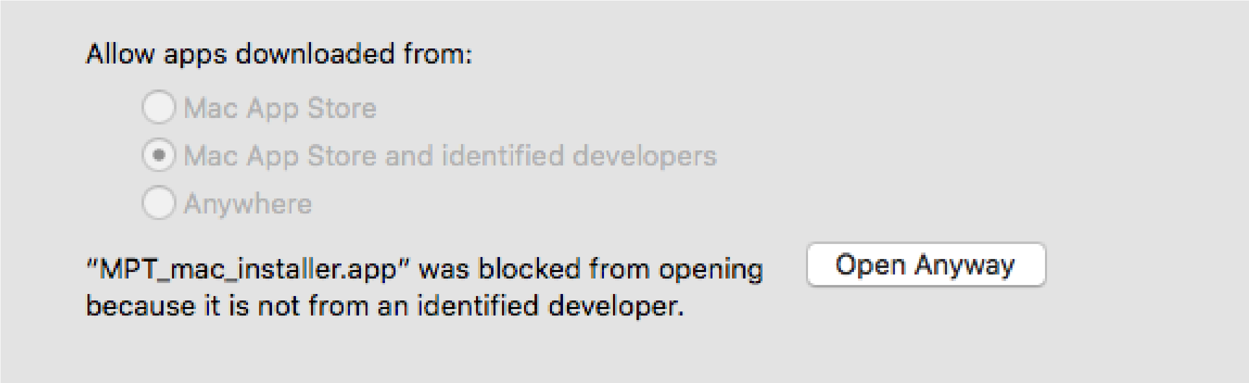


"MPT_mac_installer.app" can't be opened because it is from an unidentified developer.

Your security preferences allow installation of only apps from the Mac App Store and identified developers.

"MPT_mac_installer.app" is on the disk image "MPT_mac_installer.dmg". Firefox.app downloaded this disk image today at 17:48 from github.com.

Fig. 3.16: Security warning on Mac OS

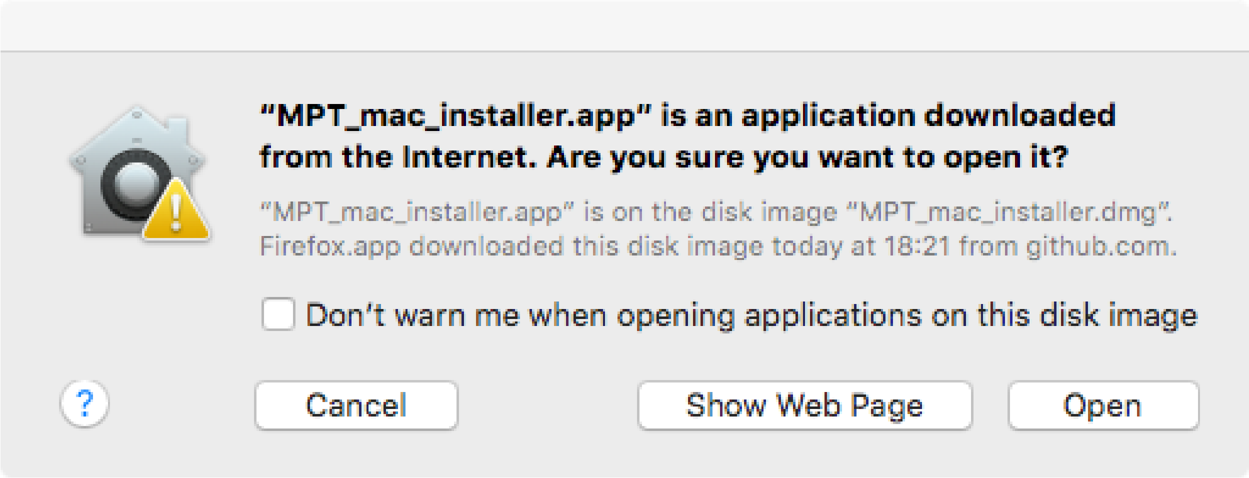


Allow apps downloaded from:

○ Mac App Store
● Mac App Store and identified developers
○ Anywhere

"MPT_mac_installer.app" was blocked from opening because it is not from an identified developer.

Fig. 3.17: Part of System Preferences, Security & Privacy



"MPT_mac_installer.app" is an application downloaded from the Internet. Are you sure you want to open it?

"MPT_mac_installer.app" is on the disk image "MPT_mac_installer.dmg". Firefox.app downloaded this disk image today at 18:21 from github.com.

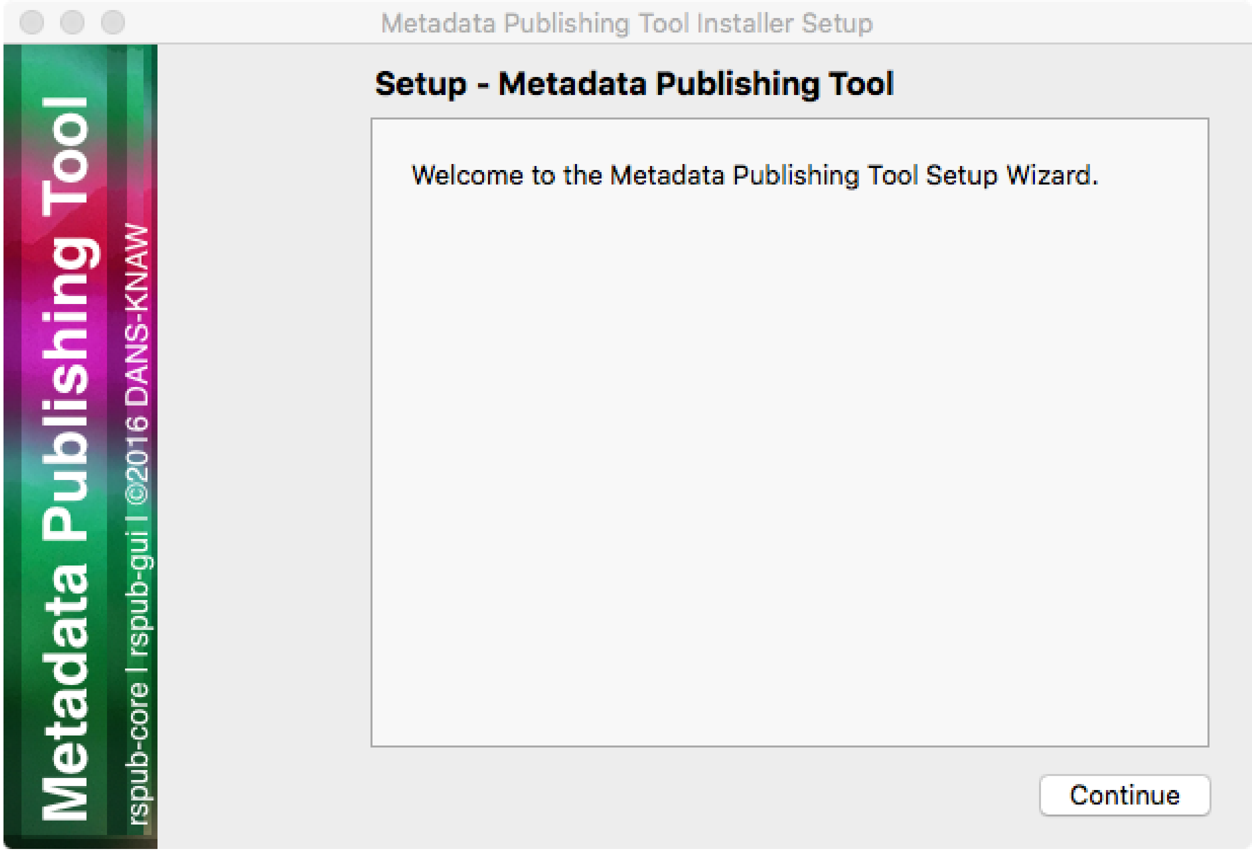
□ Don't warn me when opening applications on this disk image

Fig. 3.18: Last warning from security

3.3.2 Run the install wizard

The install wizard has several screens and will guide you through the installation process. On average, you can accept the predefined values.

Welcome message


Metadata Publishing Tool Installer Setup
Metadata Publishing Tool
rspub-core I rspub-gui I ©2016 DANS-KNAW
Continue

Setup - Metadata Publishing Tool

Welcome to the Metadata Publishing Tool Setup Wizard.

Fig. 3.19: Welcome message of the install wizard

The wizard opens with a welcome message. Click *Continue*.

Installation folder

Choosing the installation folder. The default value is `/Applications/MPT`. Choose *Browse...* if you want a different installation folder, choose *Continue* when done.

Select components

Metadata Publishing Tool has only one component. So leave the component selected and click *Continue*.

License agreement

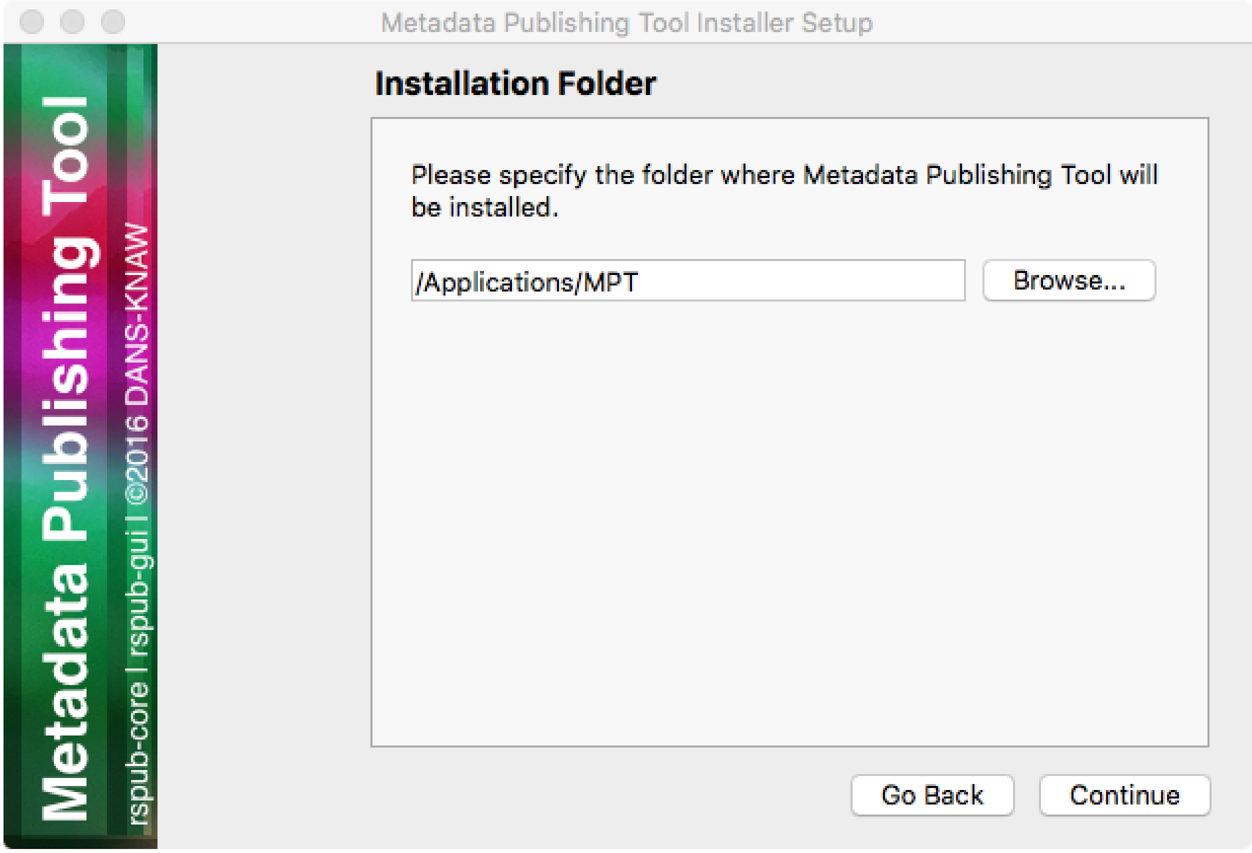
Code and executables of the *Metadata Publishing Tool* are distributed under the Apache License 2.0. You must accept the license agreement before you may proceed with the install wizard. Click *Continue* when done.

Ready to install

Click *Install*.

Installing Metadata Publishing Tool

The installation process will now begin. An indication of progress will be shown.



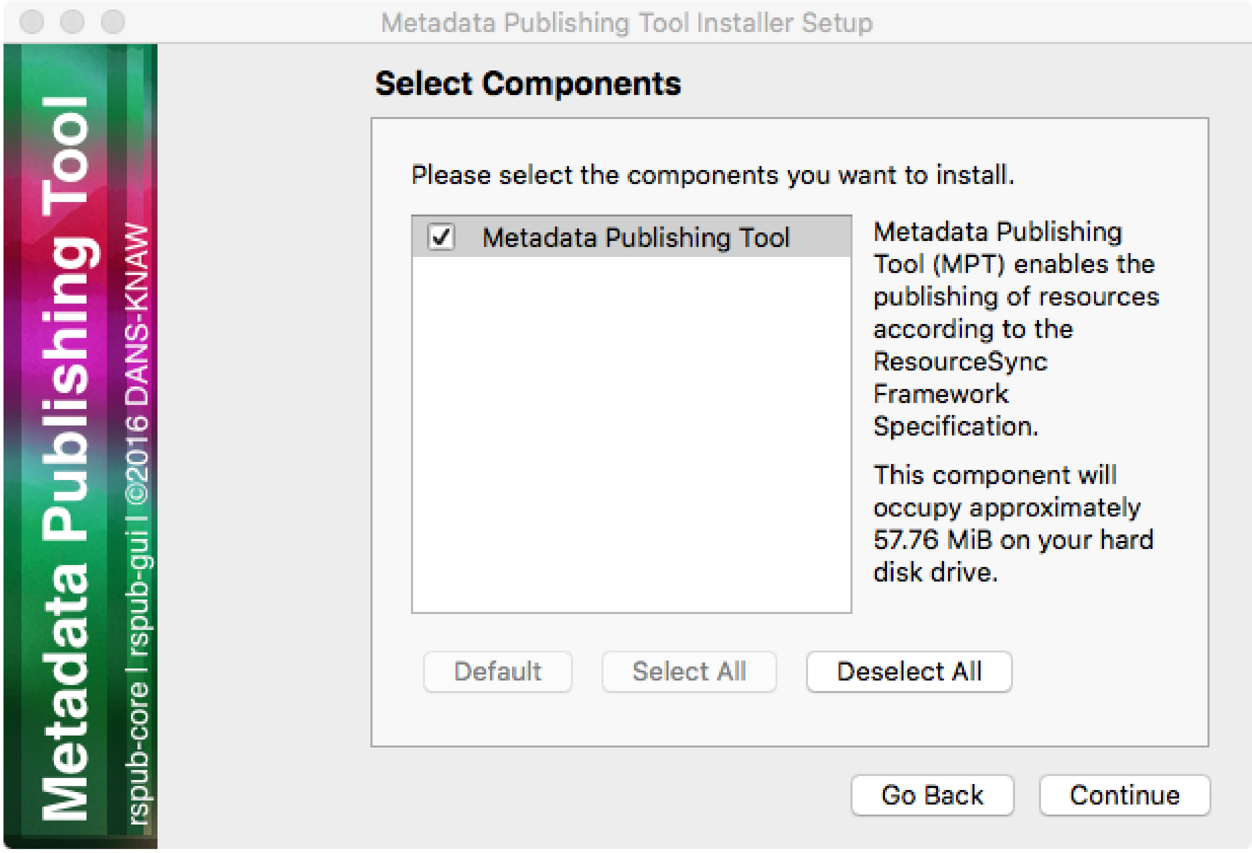
Metadata Publishing Tool Installer Setup

Installation Folder

Please specify the folder where Metadata Publishing Tool will be installed.

/Applications/MPT

Fig. 3.20: Choosing the installation folder



Metadata Publishing Tool Installer Setup

Select Components

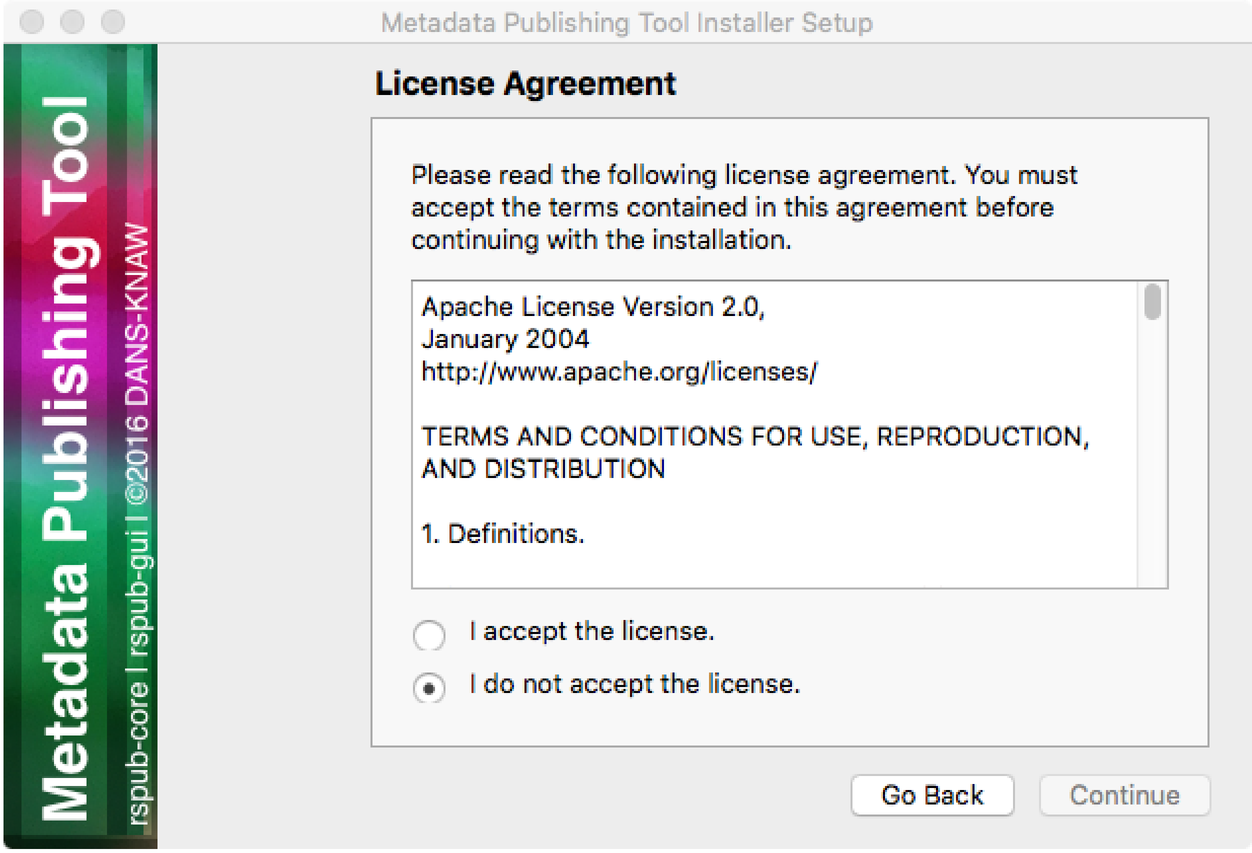
Please select the components you want to install.

☑ Metadata Publishing Tool

Metadata Publishing Tool (MPT) enables the publishing of resources according to the ResourceSync Framework Specification.

This component will occupy approximately 57.76 MiB on your hard disk drive.

Fig. 3.21: Selecting components


Metadata Publishing Tool Installer Setup
Metadata Publishing Tool
rspub-core I rspub-gui I ©2016 DANS-KNAW
Go Back
Continue

License Agreement

Please read the following license agreement. You must accept the terms contained in this agreement before continuing with the installation.

Apache License Version 2.0,
January 2004
http://www.apache.org/licenses/

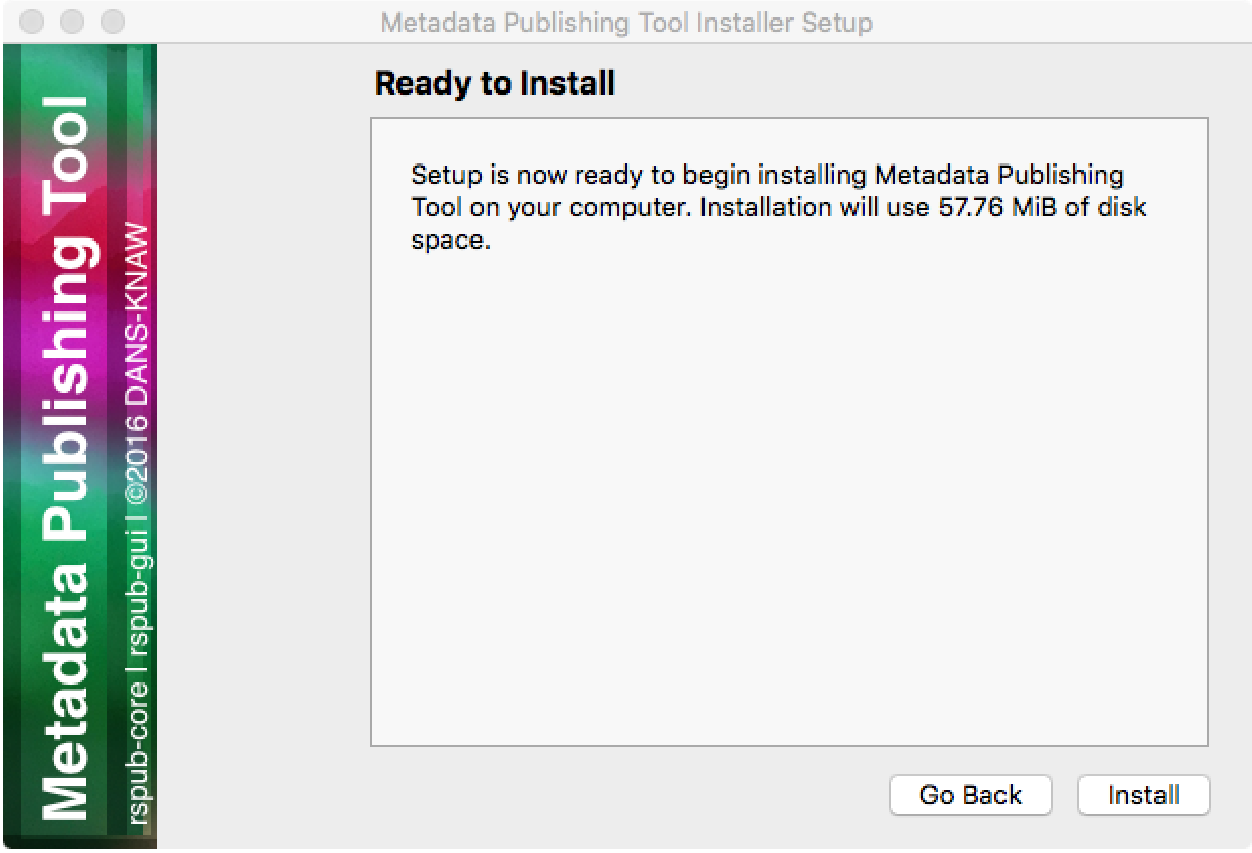
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

○ I accept the license.

● I do not accept the license.

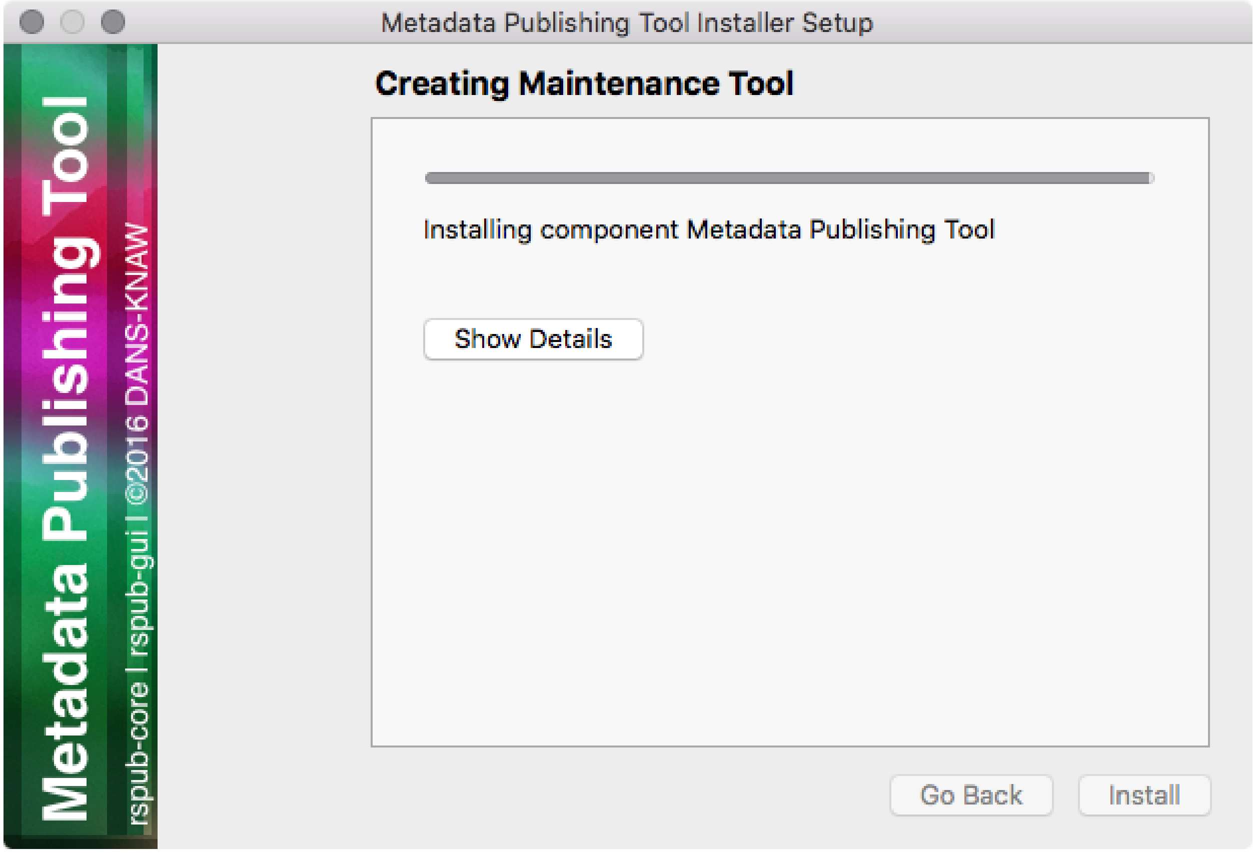
Fig. 3.22: Inspecting the license


Metadata Publishing Tool Installer Setup
Metadata Publishing Tool
rspub-core | rspub-gui | ©2016 DANS-KNAW
Go Back
Install

Ready to Install

Setup is now ready to begin installing Metadata Publishing Tool on your computer. Installation will use 57.76 MiB of disk space.

Fig. 3.23: Ready to install



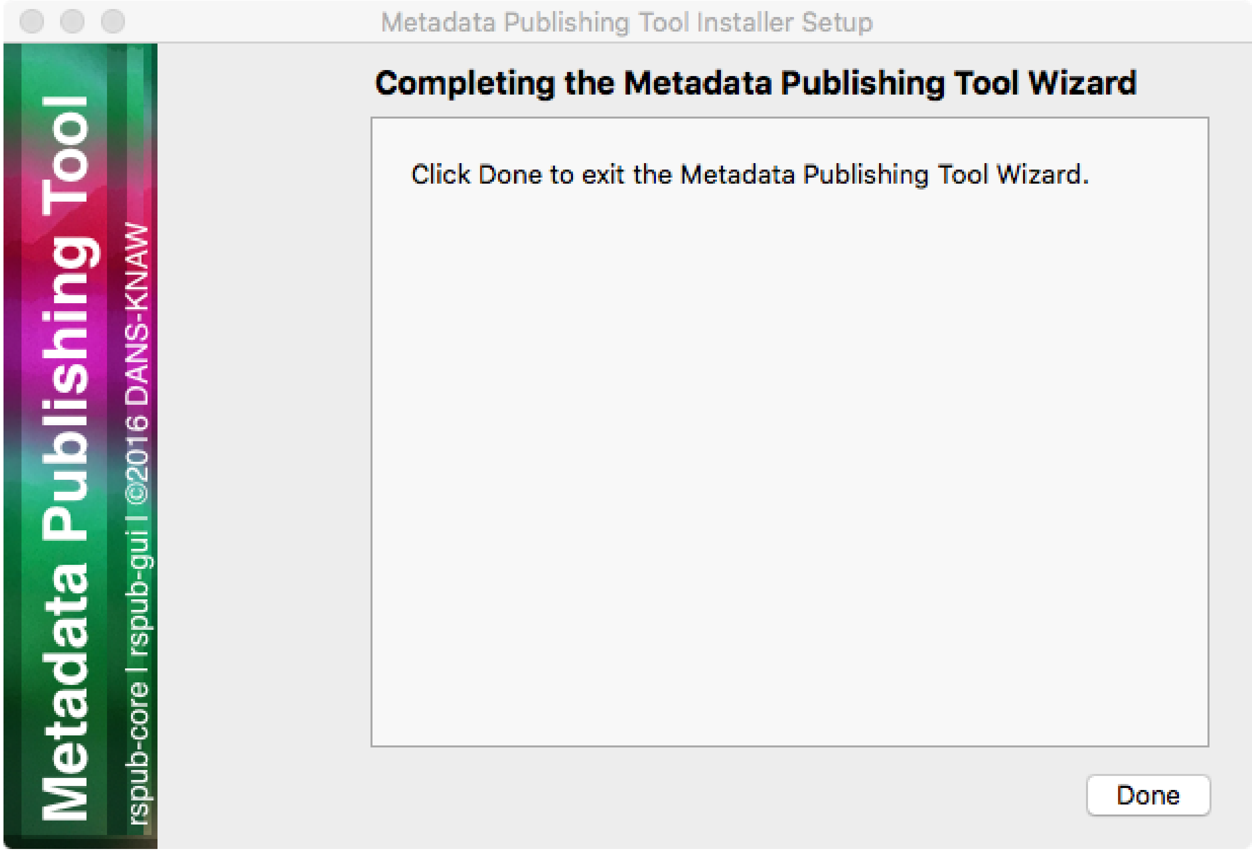
Metadata Publishing Tool Installer Setup

Creating Maintenance Tool

Installing component Metadata Publishing Tool

Fig. 3.24: Progress of installation

Completing the Metadata Publishing Tool Wizard



Metadata Publishing Tool Installer Setup
Metadata Publishing Tool
rspub-core I rspub-gui I ©2016 DANS-KNAW
Done

Completing the Metadata Publishing Tool Wizard

Click Done to exit the Metadata Publishing Tool Wizard.

Fig. 3.25: Completing the installation

After installation has completed, the last screen of the installation wizard is shown. Click *Done* to exit the wizard. You may now eject the mounted `MPT_mac_installer` image.

3.3.3 Start Metadata Publishing Tool

After completing the installation wizard you can find `MetadataPublishingTool.app` and the `MPT_MaintenanceTool.app` in the folder `MPT` of your `/Applications` folder. Alternatively you can click on the respective icon in your Launchpad.

3.3.4 Uninstall Metadata Publishing Tool

The *Metadata Publishing Tool* and its components can be removed from your machine by the *MPT Maintenance Tool*. Click the shortcut *MPT_MaintenanceTool* in your Launchpad or navigate to `/Applications/MPT` and double-click `MPT_MaintenanceTool.app`. The uninstall wizard will begin.

Click *Continue*.

Click *Uninstall*. After the uninstall process has run click *Finish*. The folder `MPT` and its contents are now removed from your system.

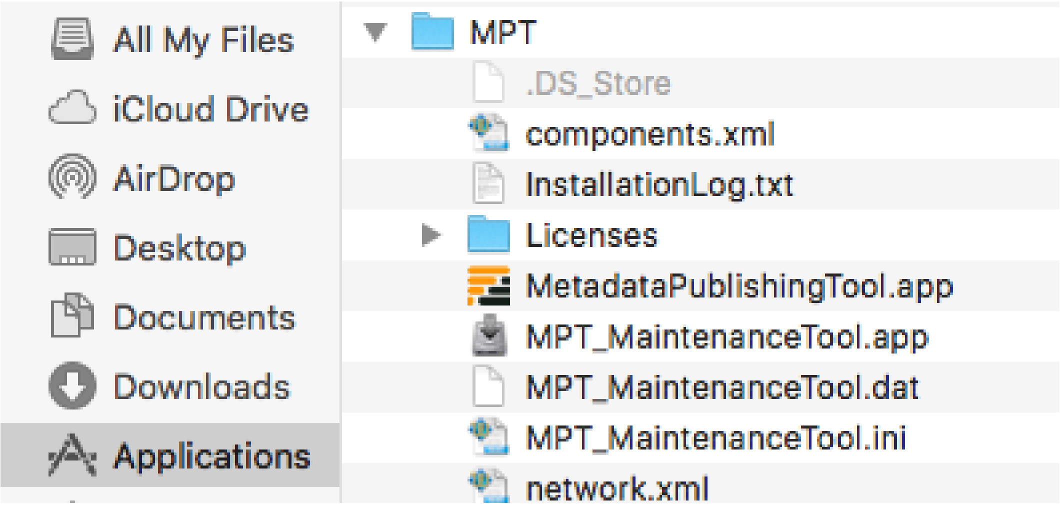
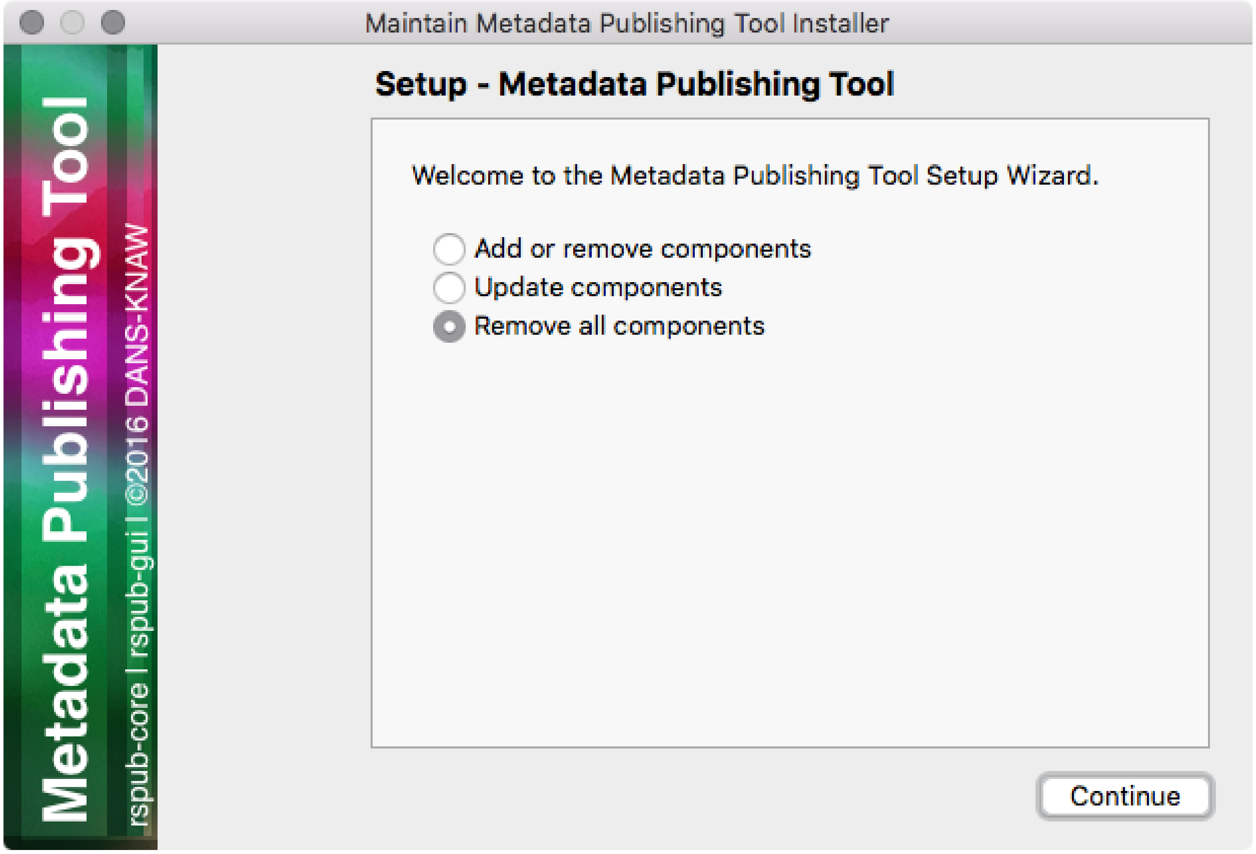


Fig. 3.26: *MetadataPublishingTool.app* and *MPT_MaintenanceTool.app* in the folder `MPT` in the Applications folder

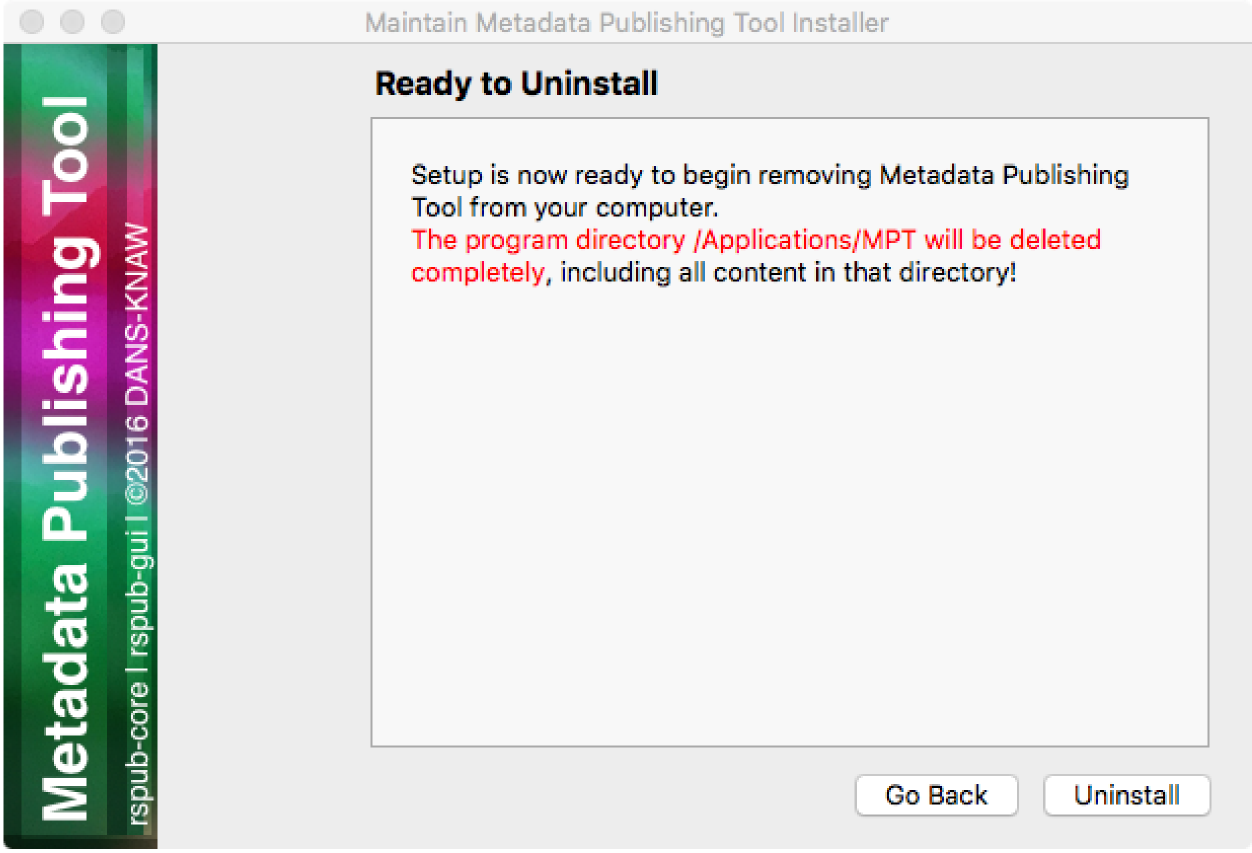

Maintain Metadata Publishing Tool Installer
Metadata Publishing Tool
rspub-core I rspub-gui I ©2016 DANS-KNAW
Continue

Setup - Metadata Publishing Tool

Welcome to the Metadata Publishing Tool Setup Wizard.

○ Add or remove components
○ Update components
● Remove all components

Fig. 3.27: Removing Metadata Publishing Tool


Maintain Metadata Publishing Tool Installer
Metadata Publishing Tool
rspub-core | rspub-gui | ©2016 DANS-KNAW
Go Back
Uninstall

Ready to Uninstall

Setup is now ready to begin removing Metadata Publishing Tool from your computer.
The program directory /Applications/MPT will be deleted completely, including all content in that directory!

Fig. 3.28: Ready to uninstall

CHAPTER 4

Application Menus

A quick look at the menus of Metadata Publishing Tool

4.1 The File menu

Important:

- The `File` menu gives options to manage multiple *configuration*s.
- The `File` menu gives an option to exit the application.

4.1.1 Load configuration >

Gives a list of previously saved configurations. Click on a named configuration in the list to close the current configuration and open the chosen one. The current configuration will automatically be saved.

4.1.2 Save configuration as…

Pops up a dialog to enter the name of the configuration. The current configuration will be saved under the given name. Any named configuration that was previously saved under the given name will be overwritten without warning.

4.1.3 Configurations. . .

Pops up a list with the names of saved configurations. You have the ability to delete configurations by selecting their names. Multiple configurations can be deleted by selecting multiple names from the list.

4.1.4 Exit

Exit the application.

4.2 The Preferences menu

Important:

- The `Preferences` menu gives options to change the overall appearance of the application.

4.2.1 Language >

Set the language of the application to the language of your choice.

Tip:

- Contribute your native language to the *MPT* tool. See *Miscellaneous*

4.3 The Windows menu

Important:

- The `Windows` menu gives options to select windows.

4.3.1 About. . .

Pops up a window that summarizes the origin and version of the application and presents links to interesting sites.

CHAPTER 5

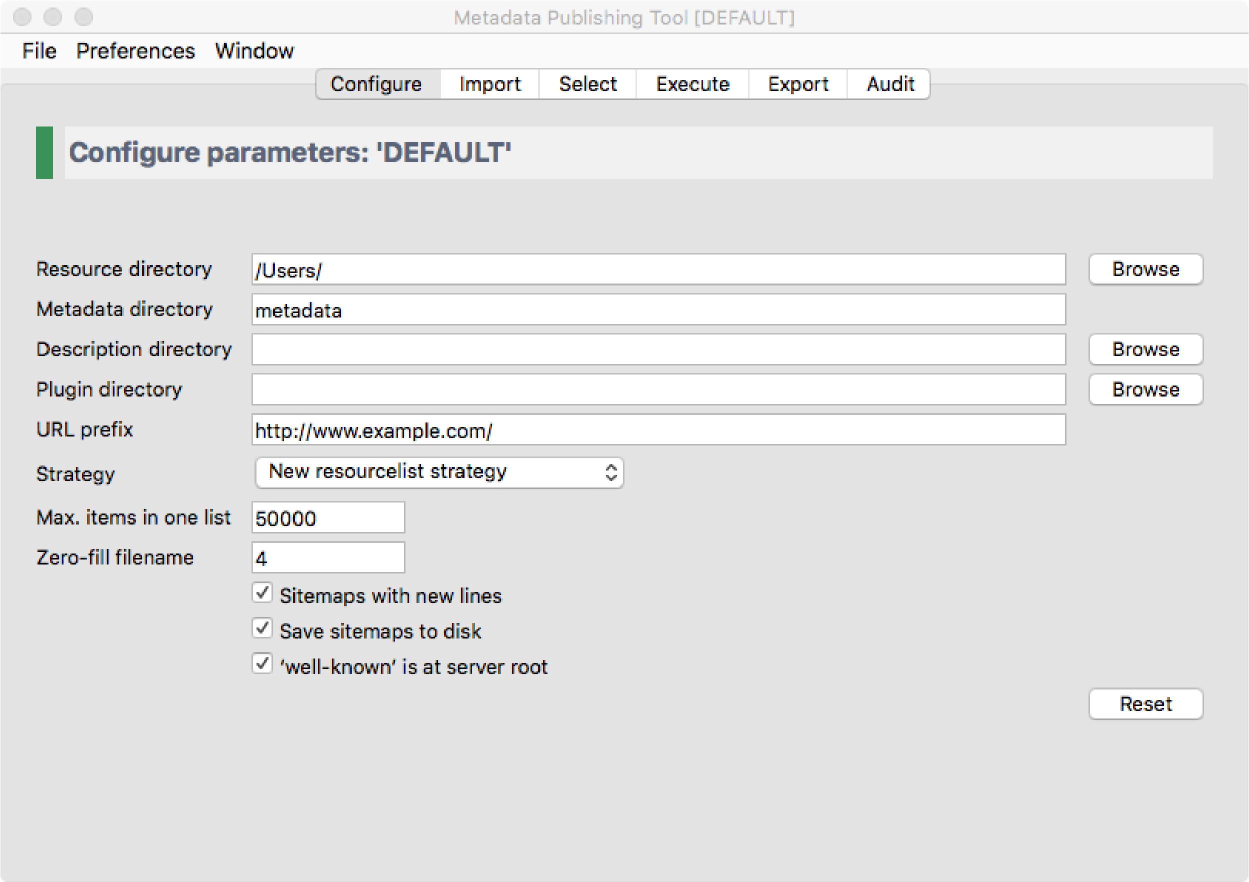
Configuration

Configure Metadata Publishing Tool

Important: The parameters on the configuration page constitute all variables needed to *synchronize* a *set of resources*. Variables are validated and, if valid, saved automatically. Multiple sets of parameters can be saved as named *configuration*s. See *The File menu* on how to save, load and delete configurations.

Variables on the configuration page are best set with the help of a technically skilled person. Once variables are set and tested to be correct, they can be reused over future *synchronization* runs.

In the following paragraphs we'll describe each parameter in detail.



Configure parameters: 'DEFAULT'

Resource directory /Users/

Metadata directory metadata

Description directory

Plugin directory

URL prefix http://www.example.com/

Strategy New resourcelist strategy

Max. items in one list 50000

Zero-fill filename 4

☑ Sitemaps with new lines

☑ Save sitemaps to disk

☑ 'well-known' is at server root

Fig. 5.1: *Screenshot of the configure page*

5.1 Resource directory

The *resource directory* should be an existing directory on the (local or networked) filesystem. In this base directory *resource*s to be *synchronize*d can be found. The resource directory should be chosen careful, because it influences the composition of the URL to the *resource*s that will be exposed in *sitemap*s:

```
local path to the resource: C:\path\to\resources\ehri\path\to\resource.ead
resource directory:         C:\path\to\resources\
URL prefix:                 http://www.example.com/
composed URL to resource:   http://www.example.com/ehri/path/to/resource.ead
```

Important:

The URL to a *resource* is composed as follows:

The value of *resource directory* is stripped of the local path to the *resource* and what remains is prefixed with the *URL prefix*.

The URL prefix may have a path segment:

```
local path to the resource: C:\path\to\resources\ehri\path\to\resource.ead
resource directory:         C:\path\to\resources\
URL prefix:                 http://www.example.com/rs/abc/
composed URL to resource:   http://www.example.com/rs/abc/ehri/path/to/resource.ead
```

The resource directory influences the composed URL:

```
local path to the resource: C:\path\to\resources\ehri\path\to\resource.ead
resource directory:         C:\path\to\resources\ehri\path\
URL prefix:                 http://www.example.com/rs/abc/
composed URL to resource:   http://www.example.com/rs/abc/to/resource.ead
```

The button *Browse* will open a file explorer that enables choosing the resource directory.

See also: *URL prefix*

Attention: When using the EHRI mapping tool, after conversion your *EAD* documents are in a directory that bears the conversion date as its name. Point the *resource directory* entry to this dated directory before execution of a *synchronization*.

5.2 Metadata directory

The *metadata directory* is the name of the directory where generated *sitemap*s are stored. The value of metadata directory may consist of multiple path segments. The metadata directory is always relative to the *resource directory*:

```
resource directory:         C:\path\to\resources\
metadata directory:         metadata
absolute path:              C:\path\to\resources\metadata
```

If the metadata directory does not exist, it will be created during first execution of a *synchronization*.

Hint: Executing a *synchronization* will produce one *Capability List* in the Metadata directory. Each *Capability List* defines a *set of resources*. When using multiple *configuration*s to define multiple sets of resources from the same *resource directory* make sure that each Metadata directory gets a different name. Like `md_01`, `md_02` etc.

5.3 Description directory

The *description directory* should be an existing directory on the (local or networked) filesystem. In this directory the document that describes the entire site, also known as `.well-known/resourcesync` or *Source Description* is expected or will be created. If the value of description directory is left blank, the document is expected or will be created in the *metadata directory*.

The button *Browse* will open a file explorer that enables choosing the description directory.

5.4 Plugin directory

The *plugin directory* is an existing directory on the (local or networked) filesystem. In this directory or its subdirectories a search for plugins will be conducted. At the moment there are slots for plugins of type ResourceGateBuilder. If plugin directory is left blank, no search will be conducted upon execution of a *synchronization*.

The button *Browse* will open a file explorer that enables choosing the plugin directory.

5.5 URL prefix

The *URL prefix* is the basename of the site, optionally followed by a path segment. It is the first part in the composition of URL's to *resource*s and *sitemap*s. The relative path to *resource*s and *sitemap*s calculated from *resource directory* is the second part:

```
local path to the resource: C:\path\to\resources\ehri\path\to\resource.ead
resource directory:         C:\path\to\resources\
URL prefix:                 http://www.example.com/
composed URL to resource:   http://www.example.com/ehri/path/to/resource.ead
```

The URL prefix may have a path segment:

```
local path to the resource: C:\path\to\resources\ehri\path\to\resource.ead
resource directory:         C:\path\to\resources\
URL prefix:                 http://www.example.com/rs/abc/
composed URL to resource:   http://www.example.com/rs/abc/ehri/path/to/resource.ead
```

See also: *Resource directory*

5.6 Strategy

The *strategy* defines what kind of *sitemap* documents will be generated when a *synchronization* is executed. At the moment you can choose between

- **New resourcelist strategy** - At each *synchronization* run a completely new *Resource List* will be generated. If previous Resource Lists or *Capability List*s exist in the *metadata directory* you will be asked if they can be deleted.
- **New changelist strategy** - Will create a new *Change List* at each *synchronization* run. At the start of *synchronization*, if no Resource List exists in the *metadata directory*, will conduct the *New resourcelist strategy* on first execution.
- **Incremental changelist strategy** - Will increment an existing *Change List* with the newly found changes. At the start of *synchronization*, if no Resource List exists in the *metadata directory*, will conduct the *New resourcelist strategy* on first execution.

Hint: The strategy can be changed before each successive execution of a synchronization run.

5.7 Max. items in one list

The maximum amount of items in one list. The *sitemap protocol* has an unofficial standard on how many items (links to *resource*s) can be in one list. This amount can vary between 1 and 50000.

5.8 Zero-fill filename

The amount of digits that generated *sitemap*s should have in their filenames. This value should be between 1 and 10. Example of filenames with zero-fill filename set at 4:

```
changelist_0000.xml
changelist_0001.xml
changelist_0002.xml
...
changelist_9999.xml
```

With zero-fill filename set to 4, the amount of *Change List*s can grow to 10000. This means you can execute 10000 *synchronization*s in *strategy* mode *new changelist strategy*.

5.9 Sitemaps with new lines

A completely 'pretty print' mechanism for the generated *sitemap*s is not (yet) available. Mark the checkbox for a more human readable version of *sitemap*s.

5.10 Save sitemaps to disk

You can do a *trial run* of a *synchronization*. A trial run will report an overview of how many *resource*s are affected, how many changes have been detected etc. but will not produce new *sitemap*s. Leave the checkbox unchecked for a trial run.

5.11 'well-known' at server root

This reflects the locality of the *Source Description* on the web server. The prescribed location of this document is in the *.well-known* directory at the root of the server:

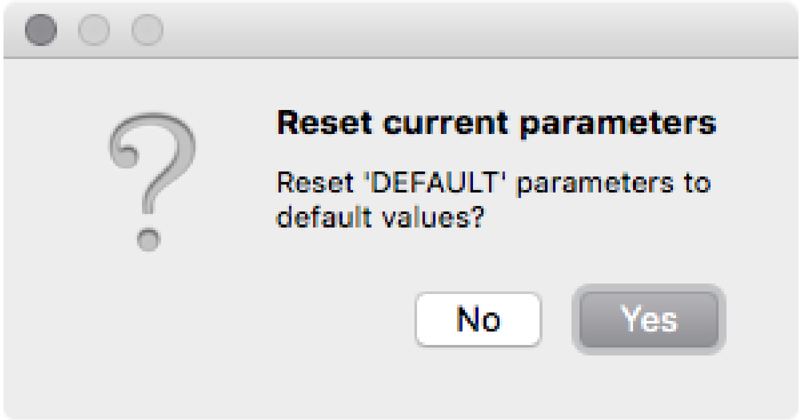
```
{server root}/.well-known/resourcesync
```

However, there may be situations where it is not allowed to access the server root directory. Since EHRI makes no use of discovering the site by means of the *well-known URI*, this document is not strictly necessary. The value for *'well-known' at server root* is used for computing the URL's that link *Capability List*s to their parent document.

Mark the checkbox if the *Source Description* is at the server root. If left unchecked the program assumes that the *Source Description* is in the *metadata directory*.

5.12 Reset

The Reset button will reset the current *configuration* to the default settings, after you confirm the warning dialog.



Reset current parameters

Reset 'DEFAULT' parameters to default values?

Fig. 5.2: *Warning dialog after pressing the Reset button*

Caution: Resetting the current configuration also affects the currently set values on wizard pages *Import*, *Select* and *Export*.

CHAPTER 6

Import resources

Import files from a server with the Metadata Publishing Tool

Hint: There are several ways to overcome problems with the exchange of files between servers and your local or network drive. One way is to use an *scp* client, which is described on this page.

Another way is to use DropBox-like solutions. See for instance *B2DROP* and *WebDAV*.

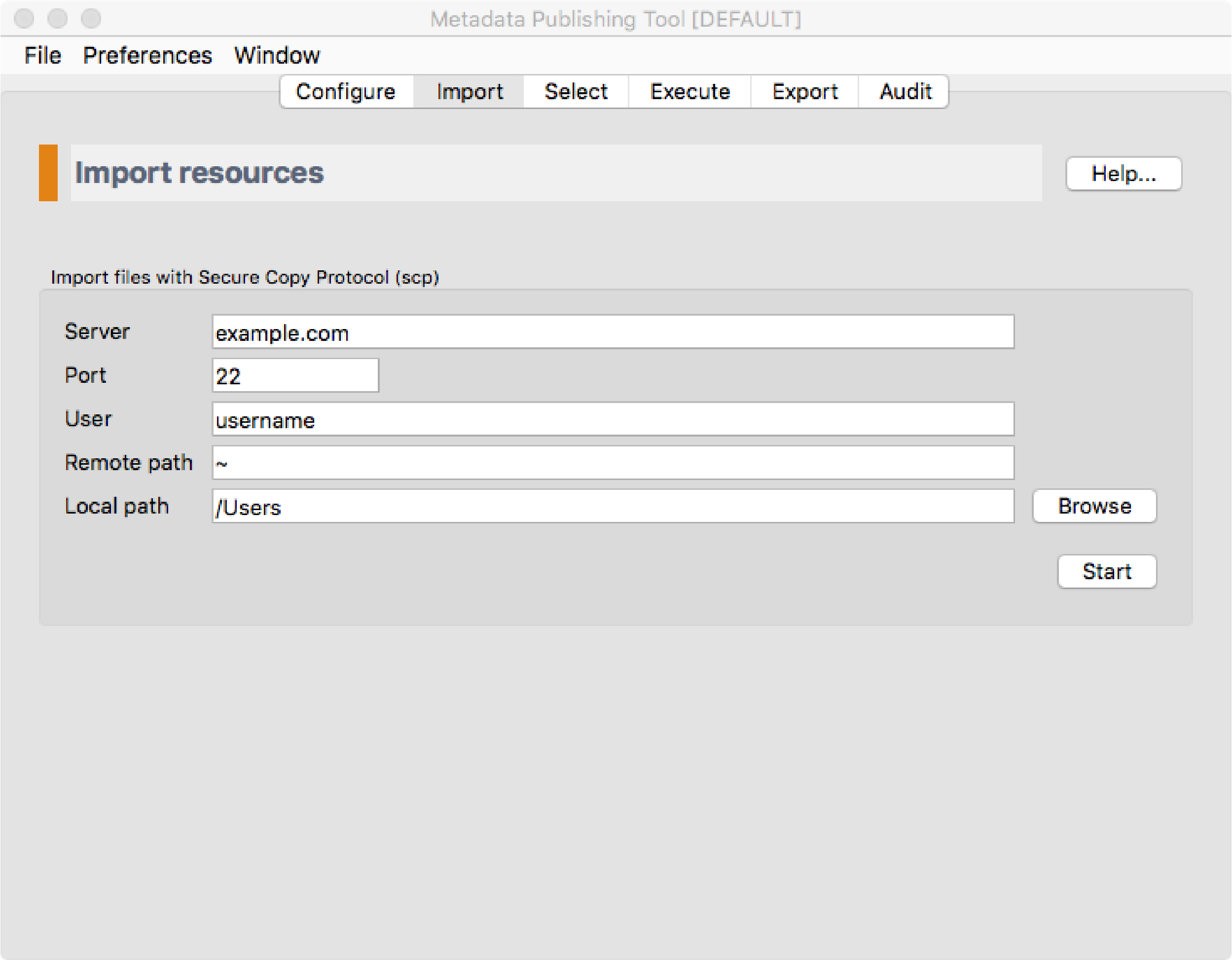
Important: The import functionality is only necessary if your files (*EAD* files in the case of *EHRI*) are on a server. If your files are on a local or networked drive you may skip this page of the wizard.

With the import page you can copy files from the remote server to a local or networked drive with the aid of the Secure Copy Protocol (*scp*).

Parameters on the import page are best set with the help of a technically skilled person. The variables on this page are part of the same named *configuration* as the one from the *Configuration* page and are saved automatically. Once parameters on this page are set, all you have to do each time you want to import your remote files to your local environment is press the *Start* button (and remember your password).

In the following paragraphs we'll describe the import process in detail.

6.1 SCP parameters



Import files with Secure Copy Protocol (scp)

Server example.com

Port 22

User username

Remote path ~

Local path /Users

Fig. 6.1: *Screenshot of the import page*

Hint: Different scp parameters can be set on each *configuration*. They are saved automatically.

Server The name or IP address of the server.

Port The port on the remote server. Default *scp* port is 22.

User The username on the remote server.

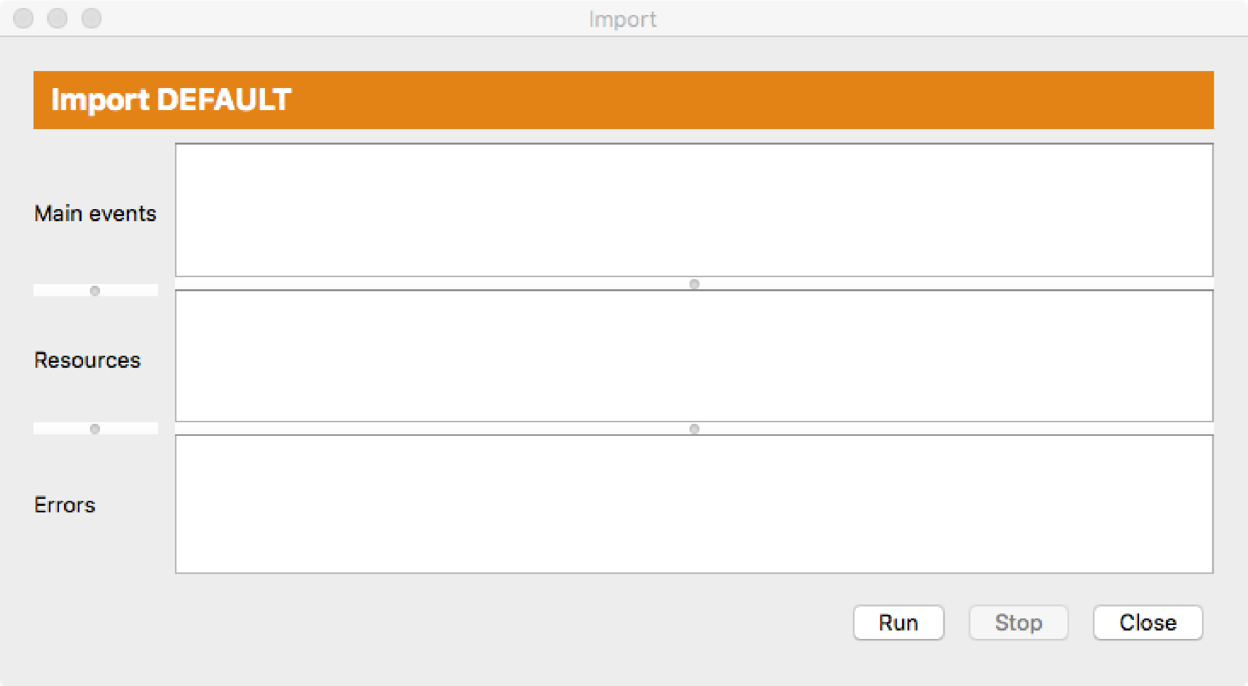
Remote path The path to the directory on the remote host. All files and folders in the remote directory will be copied recursively to the **Local path** (see below).

Local path The path to the directory on the local or networked drive that you wish to copy to. All files and folders in the directory on the **Remote path** (see above) will be copied recursively to this directory. Directories on the local path that do not exist will be created.

The button *Browse* will open a file explorer that enables choosing the local directory.

6.2 Running an import

After pressing the *Start* button, the import execution window will open.



Import

Import DEFAULT

Fig. 6.2: *Screenshot of the import execution window*

The import execution window has three areas for reporting events:

Main events In this area main events of the import process will be reported.

Resources In this area files that are imported are listed.

Errors In this area errors that took place during the import process are reported.

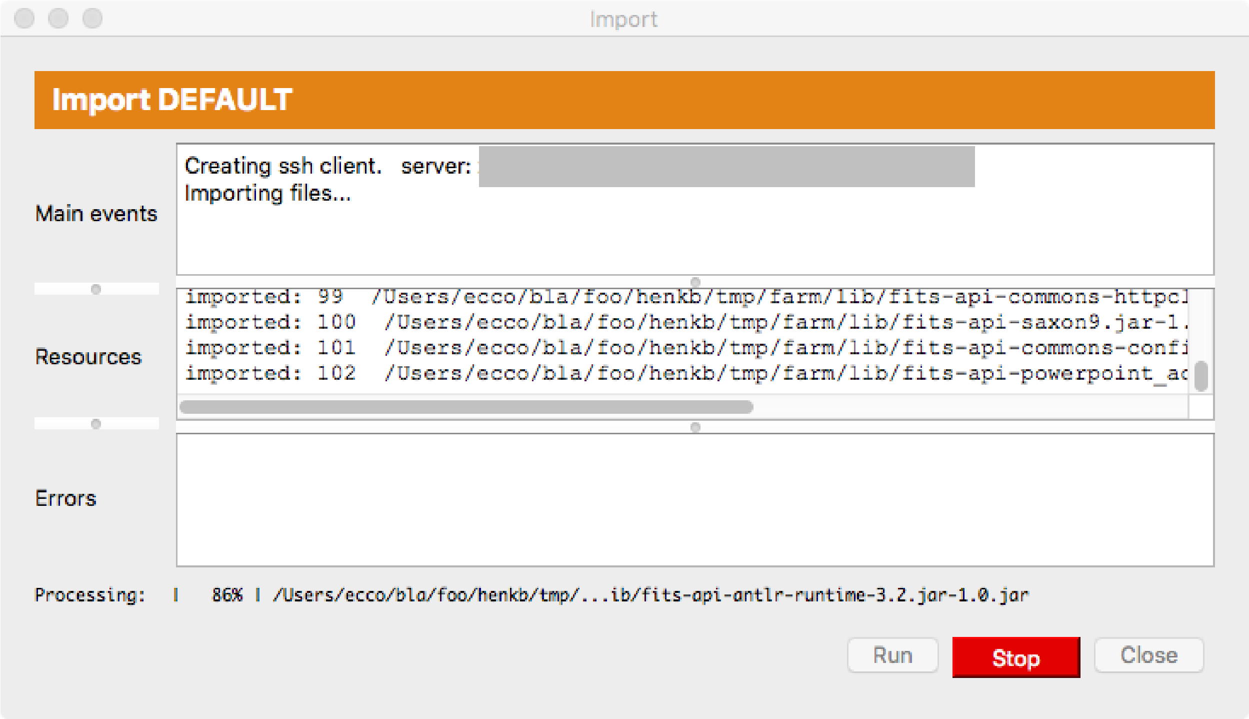
All areas can be enlarged or made smaller by grabbing the horizontal handle bars. Of course, the import execution window itself can also be reshaped.

Press the *Run* button to start the import process. A dialog appears in which you have to type your password for the remote server.

Tip:

A password may not be needed with key-based authentication.
See for instance: Configure SSH Key-Based Authentication

While the import process is running you may at any time press the `Stop` button to interrupt the process.



```
Creating ssh client.   server: [redacted]
Importing files...
```

```
imported: 99   /Users/ecco/bla/foo/henkb/tmp/farm/lib/fits-api-commons-httpcl
imported: 100   /Users/ecco/bla/foo/henkb/tmp/farm/lib/fits-api-saxon9.jar-1.
imported: 101   /Users/ecco/bla/foo/henkb/tmp/farm/lib/fits-api-commons-confi
imported: 102   /Users/ecco/bla/foo/henkb/tmp/farm/lib/fits-api-powerpoint_ac
```

```
Processing:  |   86% | /Users/ecco/bla/foo/henkb/tmp/...ib/fits-api-antlr-runtime-3.2.jar-1.0.jar
```

Fig. 6.3: *Screenshot of the import execution window while the import process is running*

After the process has ended without errors the complete file and folder tree of the remote path is now on the local path.

CHAPTER 7

Select resources

Select files to be synchronized with Metadata Publishing Tool

Important: On the select page you decide which files will be *synchronize*d. The page offers a choice on how to select:

- **Simple select** - All files in one directory will be *synchronize*d.
- **Advanced select** - Multiple files and folders can be included and multiple files and folders can be excluded. This gives the opportunity to precisely select the *resource*s that will be *synchronize*d.

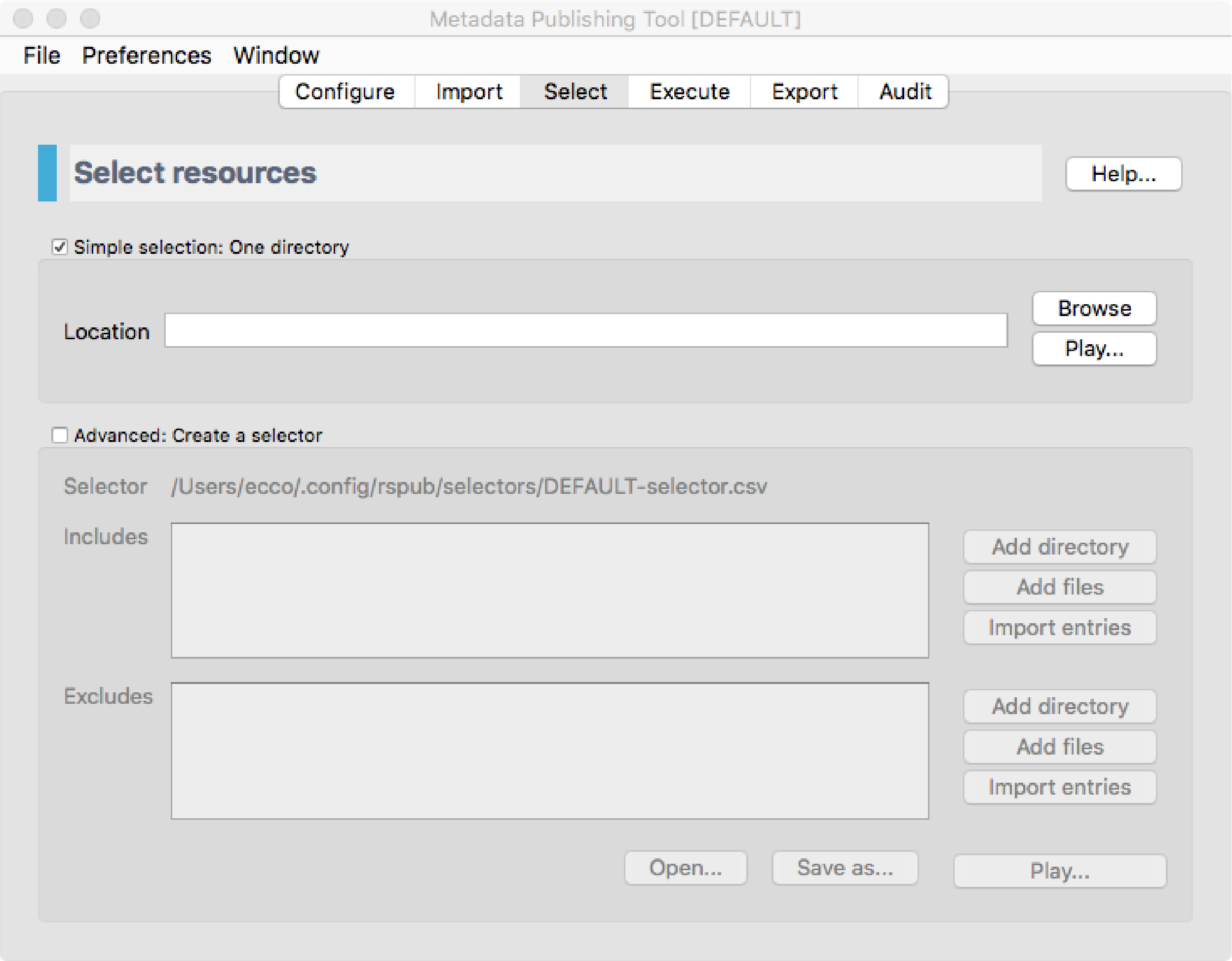
Both selection methods work recursive: files in subdirectories are included.

The variables on this page are part of the same named *configuration* as the one from the *Configuration* page and are saved automatically. Different selections can be set and saved for each *configuration*.

7.1 Simple selection

Select the checkbox for *Simple selection: one directory*. Press the *Browse* button to select the directory that contains the *resource*s that should be *synchronize*d.

Hint: The selected directory should be equal to or a subdirectory of the *resource directory* that you chose on the *Configure* page. Only *resource*s that are in the *resource directory* or one of its subdirectories are subject to *synchronization*.



☑ Simple selection: One directory

□ Advanced: Create a selector

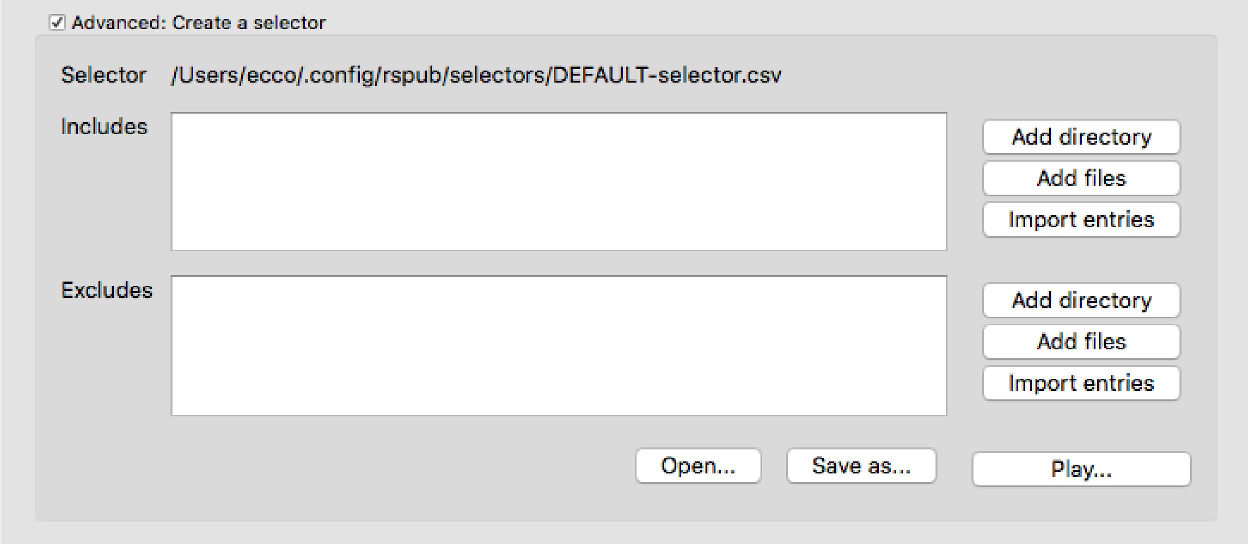
Selector /Users/ecco/.config/rspub/selectors/DEFAULT-selector.csv

Fig. 7.1: *Screenshot of the select page*

Press the *Play...* button if you want to get an impression of how many and which files are in the chosen directory and its subdirectories. The list and count displayed are estimates; automatically excluded files like hidden files are not excluded in this display, but will be excluded in the *synchronization*.

7.2 Advanced selection

Select the checkbox for *Advanced: create a selector*.



☑ Advanced: Create a selector

Selector /Users/ecco/.config/rspub/selectors/DEFAULT-selector.csv

Fig. 7.2: *The advanced part of the select page*

Important: With advanced selection you can choose multiple files and folders for inclusion and exclusion of *synchronization*. For instance if you include the directory `C:\my_files\ehri` but want to exclude its subdirectory `C:\my_files\ehri\other_files`.

Another feature is the ability to import lists of path names. This opens the possibility of doing the selection of what will be *synchronize*d in or by another application.

Press *Add directory* or *Add files* respectively if you want to browse to directories or files to be included or excluded. When adding files you can choose multiple files by pressing *Shift* and *Ctrl* or *Cmd* keys while choosing files with your mouse. You can also type or paste path names in the respective boxes. Remove path names by simply selecting and deleting them.

To import lists of path names press the *Import entries* button. Choose the file that contains the path names in the explorer that opens. Each entry in the file with path name entries should start on a new line.

Hint: Selected directories or files should be in the *resource directory* that you chose on the *Configure* page. Only *resource*s that are in the *resource directory* or in one of its subdirectories are subject to *synchronization*.

Press the *Play...* button if you want to get an impression of how many and which files are included and how many and which files are excluded. The lists and counts displayed are estimates; automatically excluded files like hidden files are not excluded in this display, but will be excluded in the *synchronization*.

The entries in *Includes* and *Excludes* are automatically saved with each *configuration*. Besides that, they can also be saved and loaded from the filesystem by pressing the *Save* and *Open* buttons respectively.

CHAPTER 8

Execute a synchronization

Synchronize resources with Metadata Publishing Tool

Important: On the execute page you can inspect the parameters of the current *configuration* and start a *synchronization*.

8.1 Inspect parameters

The execute page gives an overview of the value of all parameters that are involved with a *synchronization*. The parameters in boxes with a white background can directly be set on pages *Configure* and *Select*; the values in boxes with a blueish-grey background are derived or computed values.

8.2 Synchronize resources

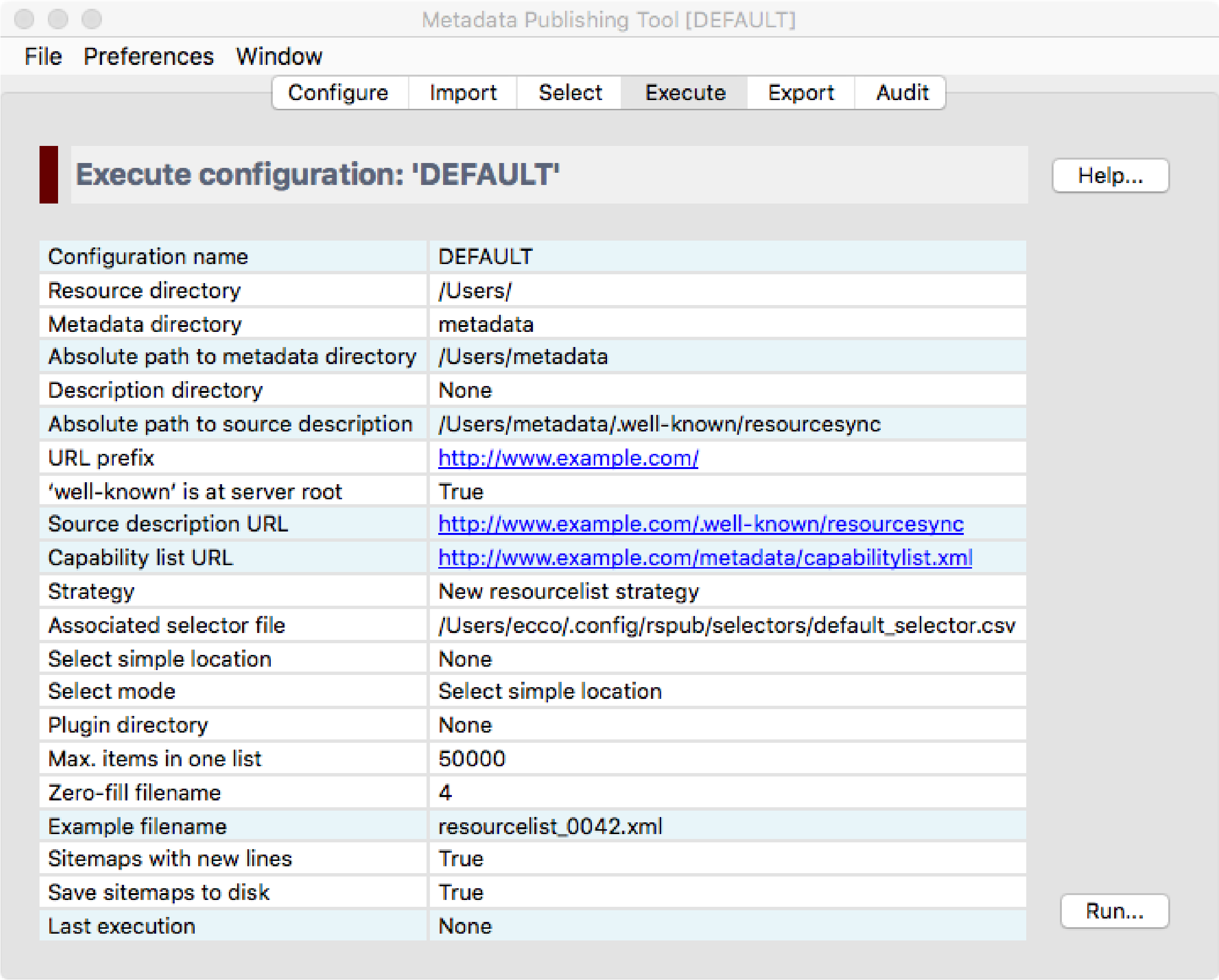
Press the *Run…* button on the execute page to start a *synchronization* run.

The synchronization window has three areas for reporting events:

Main events In this area main events of the *synchronization* process will be reported.

Resources In this area *resource*s that are *synchronize*d are listed.

Errors In this area errors that took place during the *synchronization* process are reported.



Execute configuration: 'DEFAULT'

Configuration name	DEFAULT
Resource directory	/Users/
Metadata directory	metadata
Absolute path to metadata directory	/Users/metadata
Description directory	None
Absolute path to source description	/Users/metadata/.well-known/resourcesync
URL prefix	http://www.example.com/
'well-known' is at server root	True
Source description URL	http://www.example.com/.well-known/resourcesync
Capability list URL	http://www.example.com/metadata/capabilitylist.xml
Strategy	New resourcelist strategy
Associated selector file	/Users/ecco/.config/rspub/selectors/default_selector.csv
Select simple location	None
Select mode	Select simple location
Plugin directory	None
Max. items in one list	50000
Zero-fill filename	4
Example filename	resourcelist_0042.xml
Sitemaps with new lines	True
Save sitemaps to disk	True
Last execution	None

Fig. 8.1: *Screenshot of the execute page*

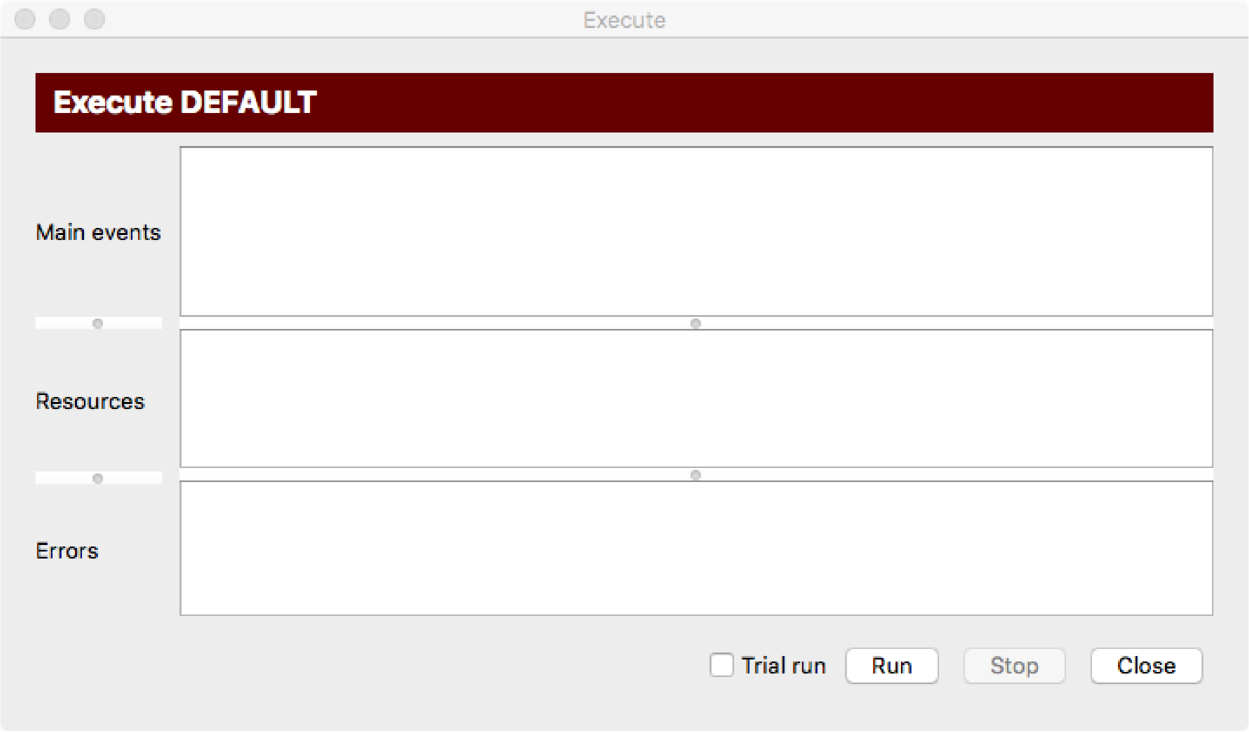


Fig. 8.2: *Screenshot of the synchronization window*

All areas can be enlarged or made smaller by grabbing the horizontal handle bars. Of course, the synchronization window itself can also be reshaped.

Press the *Run* button to start the *synchronization* process. If the chosen *strategy* is *new resourcelist strategy*, a confirmation window will appear asking if it is OK to delete existing *sitemap*s in de *metadata directory*.

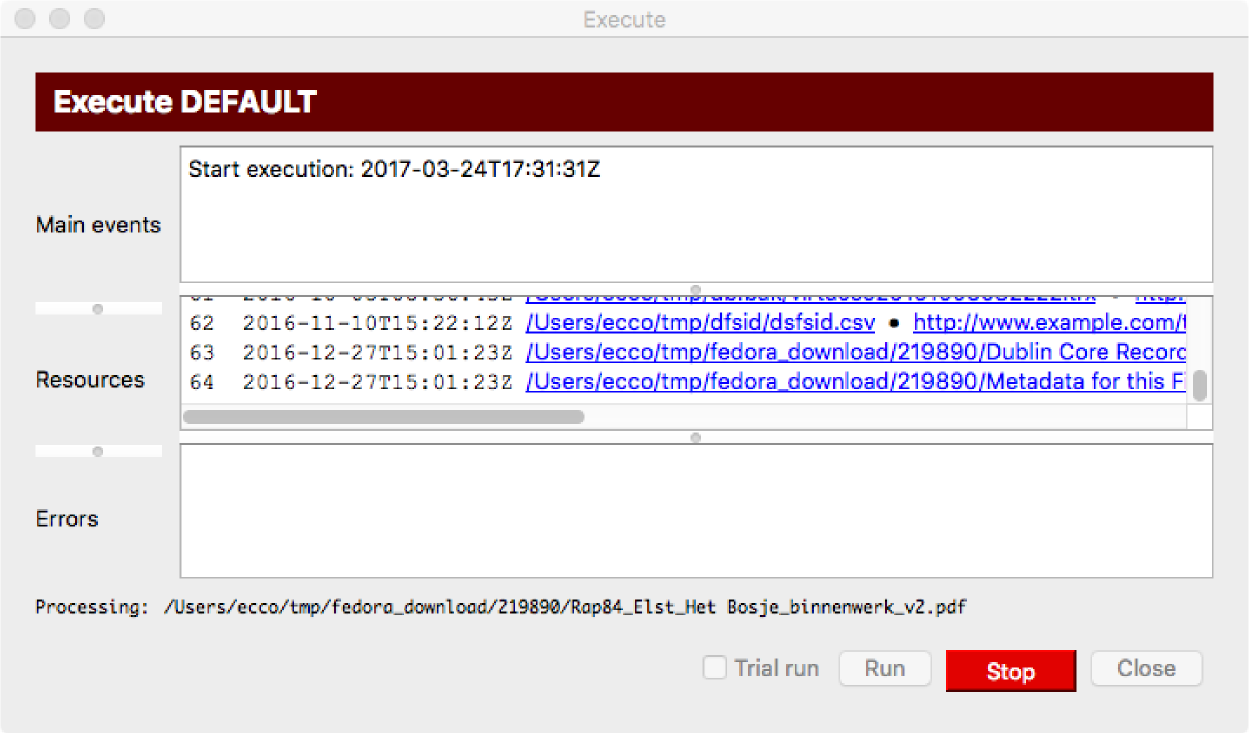
Tip: A *trial run* will report the number of *resource*s that are affected and all *sitemap*s that will be created, without writing them to disk.

While the *synchronization* process is running you may at any time press the `Stop` button to interrupt the process.

When the *synchronization* process has finished the main events of the *synchronization* process are reported.

Clicking on the paths to the *sitemap*s will open your local editor for xml-files with the contents of the *sitemap*. As long as the *sitemap*s are not exported to your web site the links to the URI's are of course stale or will not function.

The default resource gate will reject hidden files and files in the *metadata directory* in case this directory is on the select path.

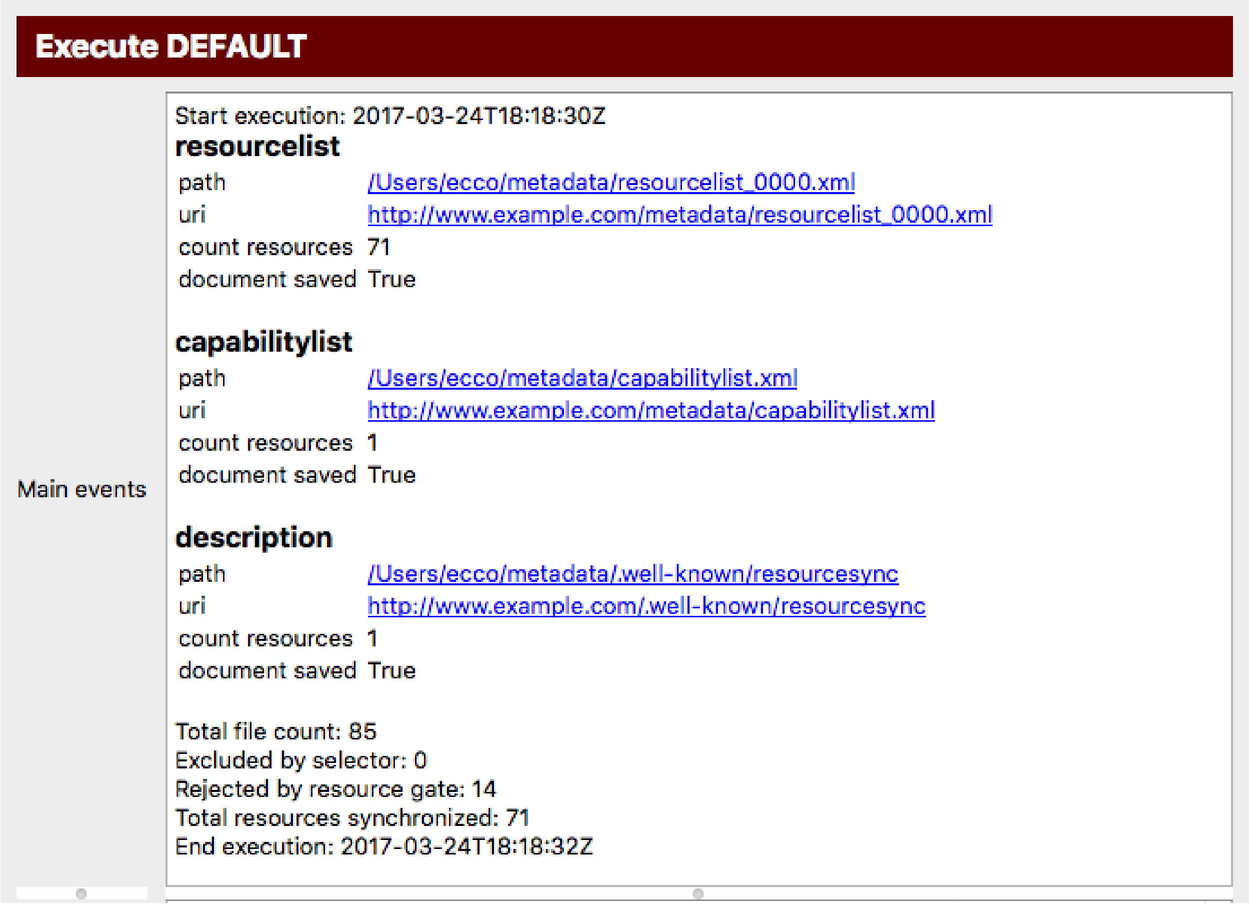


Start execution: 2017-03-24T17:31:31Z

```
62   2016-11-10T15:22:12Z /Users/ecco/tmp/dfsid/dfsid.csv • http://www.example.com/t
63   2016-12-27T15:01:23Z /Users/ecco/tmp/fedora_download/219890/Dublin Core Record
64   2016-12-27T15:01:23Z /Users/ecco/tmp/fedora_download/219890/Metadata for this F
```

```
Processing: /Users/ecco/tmp/fedora_download/219890/Rap84_Elst_Het Bosje_binnenwerk_v2.pdf
```

Fig. 8.3: *Screenshot of the synchronization window while the synchronization process is running*



Start execution: 2017-03-24T18:18:30Z

resourcelist

path	/Users/ecco/metadata/resourcelist_0000.xml
uri	http://www.example.com/metadata/resourcelist_0000.xml
count resources	71
document saved	True

capabilitylist

path	/Users/ecco/metadata/capabilitylist.xml
uri	http://www.example.com/metadata/capabilitylist.xml
count resources	1
document saved	True

description

path	/Users/ecco/metadata/.well-known/resourcesync
uri	http://www.example.com/.well-known/resourcesync
count resources	1
document saved	True

Total file count: 85
Excluded by selector: 0
Rejected by resource gate: 14
Total resources synchronized: 71
End execution: 2017-03-24T18:18:32Z

Fig. 8.4: *Part of the synchronization window after the synchronization process has finished*

CHAPTER 9

Export resources and sitemaps

Transfer resources and sitemaps to a web server

Hint: There are several ways to overcome problems with the exchange of files between servers and your local or network drive. One way is to use an *scp* client, which is described on this page.

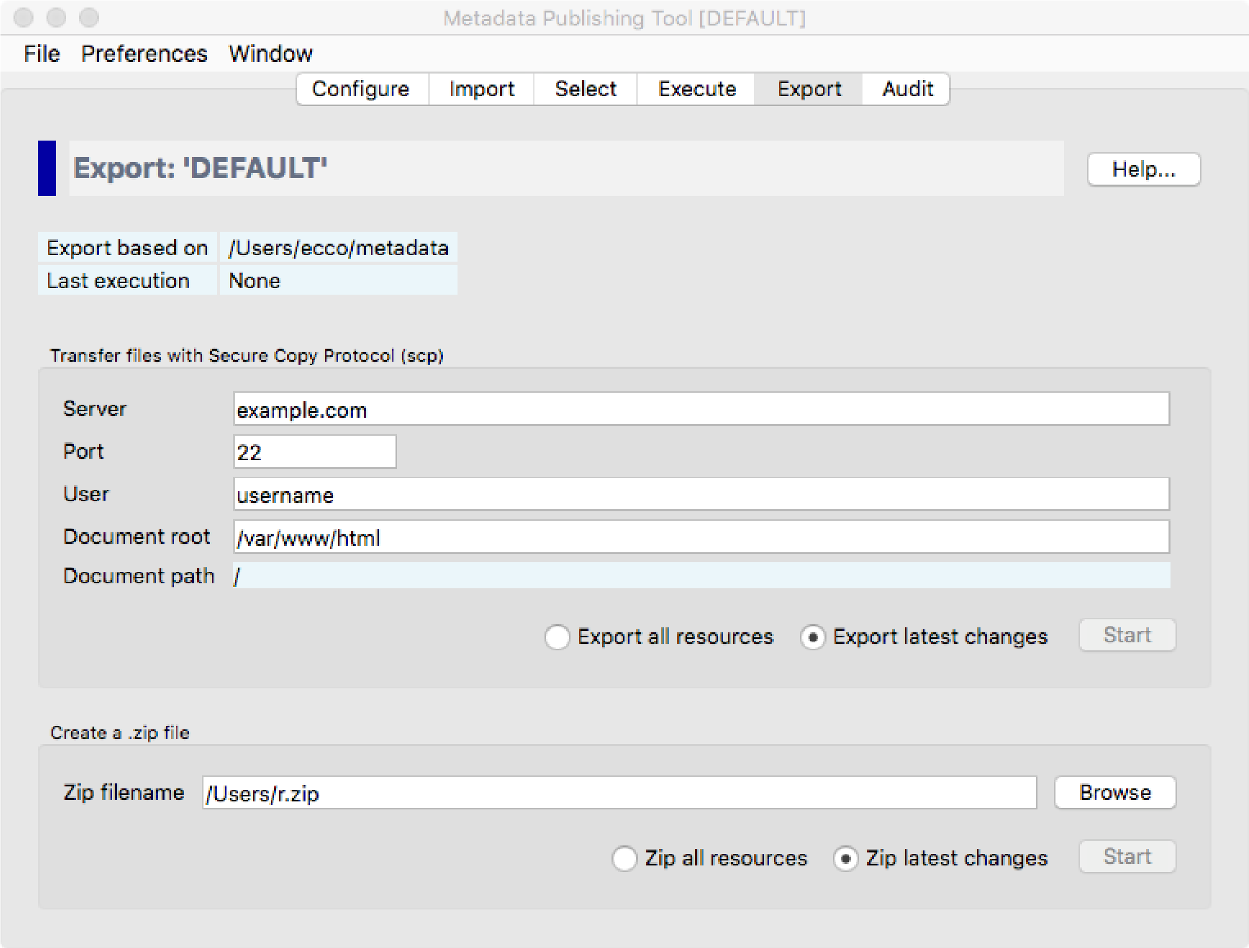
Another way is to use DropBox-like solutions. See for instance *B2DROP* and *WebDAV*.

Important: The local *synchronization* will not publish your *resource*s and *sitemap*s. For that, *resource*s and *sitemap*s must be made available on a web server. The export page offers two methods to export *resource*s and *sitemap*s from your local or networked drive to your web server:

- **Transfer files with SCP** - Uses the Secure Copy Protocol (*scp*) to transfer files directly to your web server.
- **Create a zip file** - This method creates a zip file of your *resource*s and *sitemap*s. You can hand over this zip file to your system administrator who should take care of publishing the contents on the web server.

The export page shows the current *configuration*, the location of the metadata the export will be based upon and at what date and time the last execution of the *synchronization* took place.

Attention: Do not change parameters on the *Configuration page* in between a *synchronization* run and the export. The outcome of the export may be undecided if you do so. Always export *resource*s and *sitemap*s right after a fresh *synchronization* run.



Export: 'DEFAULT'

Export based on	/Users/ecco/metadata
Last execution	None

Transfer files with Secure Copy Protocol (scp)

Server	example.com
Port	22
User	username
Document root	/var/www/html
Document path	/

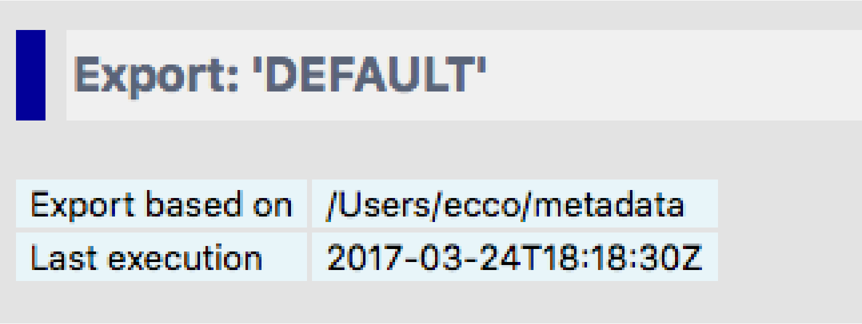
○ Export all resources ● Export latest changes

Create a .zip file

Zip filename /Users/r.zip

○ Zip all resources ● Zip latest changes

Fig. 9.1: *Screenshot of the export page*

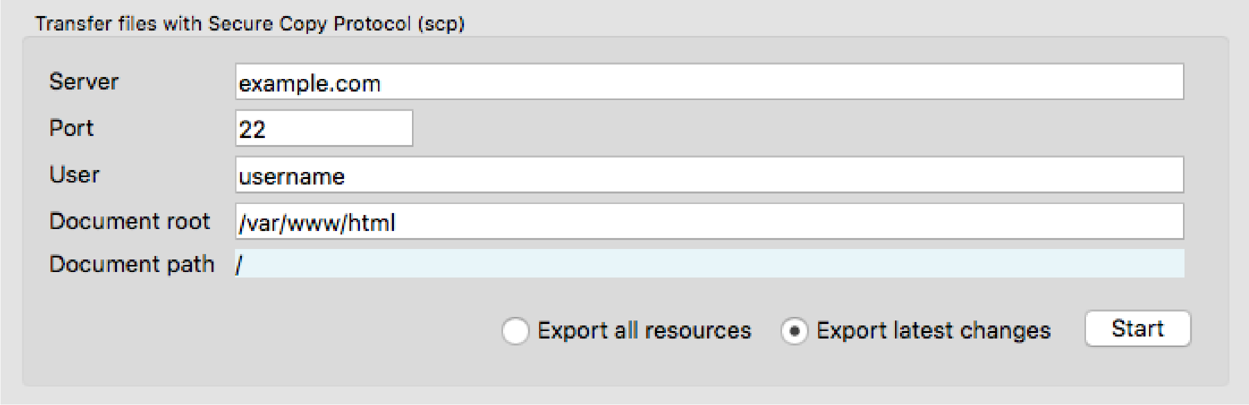


Export: 'DEFAULT'

Export based on	/Users/ecco/metadata
Last execution	2017-03-24T18:18:30Z

Fig. 9.2: *Detail of the export page*

9.1 Transfer files with SCP



Transfer files with Secure Copy Protocol (scp)

Server example.com

Port 22

User username

Document root /var/www/html

Document path /

○ Export all resources ● Export latest changes

Fig. 9.3: *Detail of the export page with the SCP parameters*

The parameters for export with *scp* can best be set with the help of a technically skilled person. These parameters are automatically saved with the current *configuration*.

Server The name or IP address of the web server.

Port The *scp* port on the web server. Default *scp* port is 22.

User The username on the web server.

Document root The document root is the folder where the website files for a domain name are stored. With the Apache HTTP Server for instance this defaults to `/var/www/html`.

The *Document path*, relative to the *Document root*, is derived from the *URL prefix* you set on the *Configuration page*. (The *Document path* will be equal to the path segment of the *URL prefix*.)

You have a choice between exporting all *resource*s and export the latest changes.

- **Export all resources** will include all resources mentioned in the *sitemap* documents currently in the *metadata directory*. Useful if you want to completely update the part of your site that hosts *resource*s and *sitemap*s of the current *configuration*.
- **Export latest changes** will only include resources that where affected according to the last *synchronization*. The default. This will suffice in most occasions.

Press *Start* to open the *scp* Transport execution window.

The scp Transport execution window has three areas for reporting events:

Main events In this area main events of the export process will be reported.

Resources In this area files that are exported are listed.

Errors In this area errors that took place during the export process are reported.

Press the *Run* button to start the export process. A popup dialog will ask for the password of the user at the web server.

Tip:

A password may not be needed with key-based authentication.
See for instance: Configure SSH Key-Based Authentication

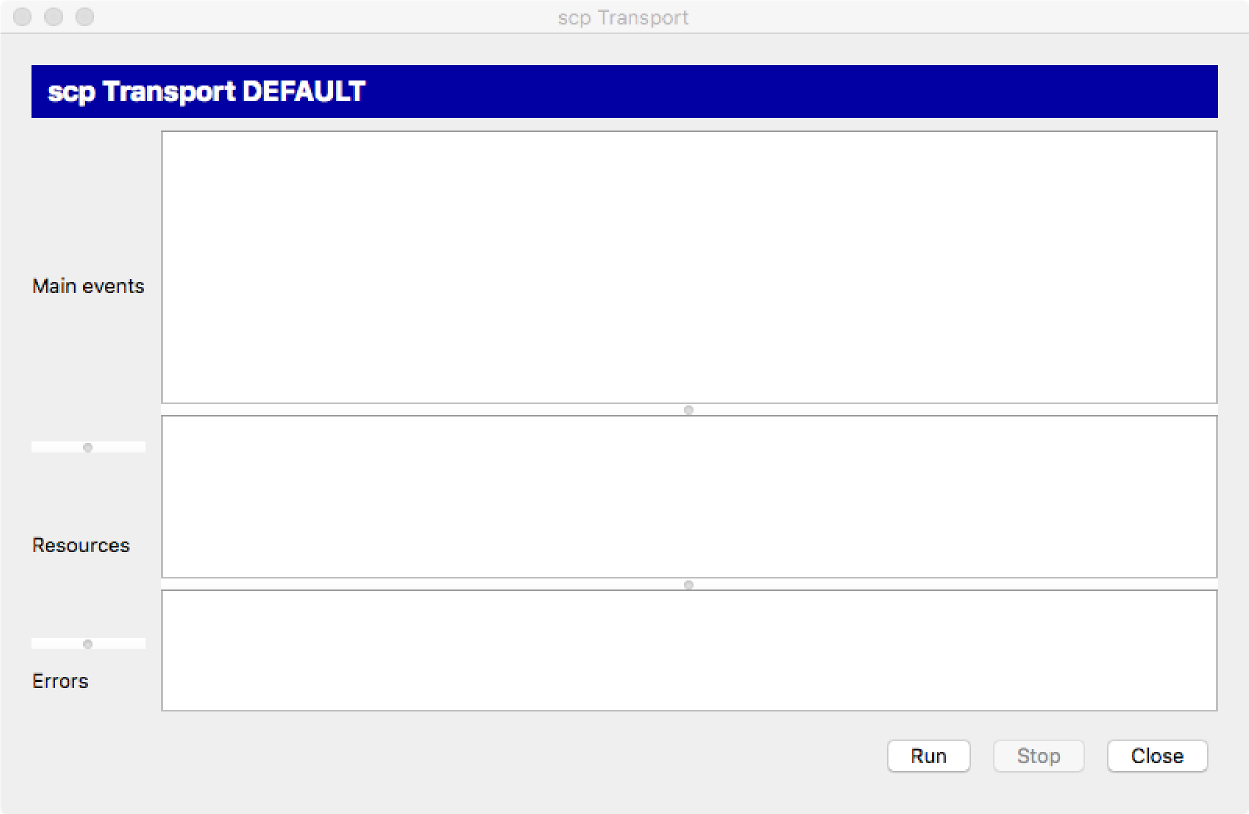
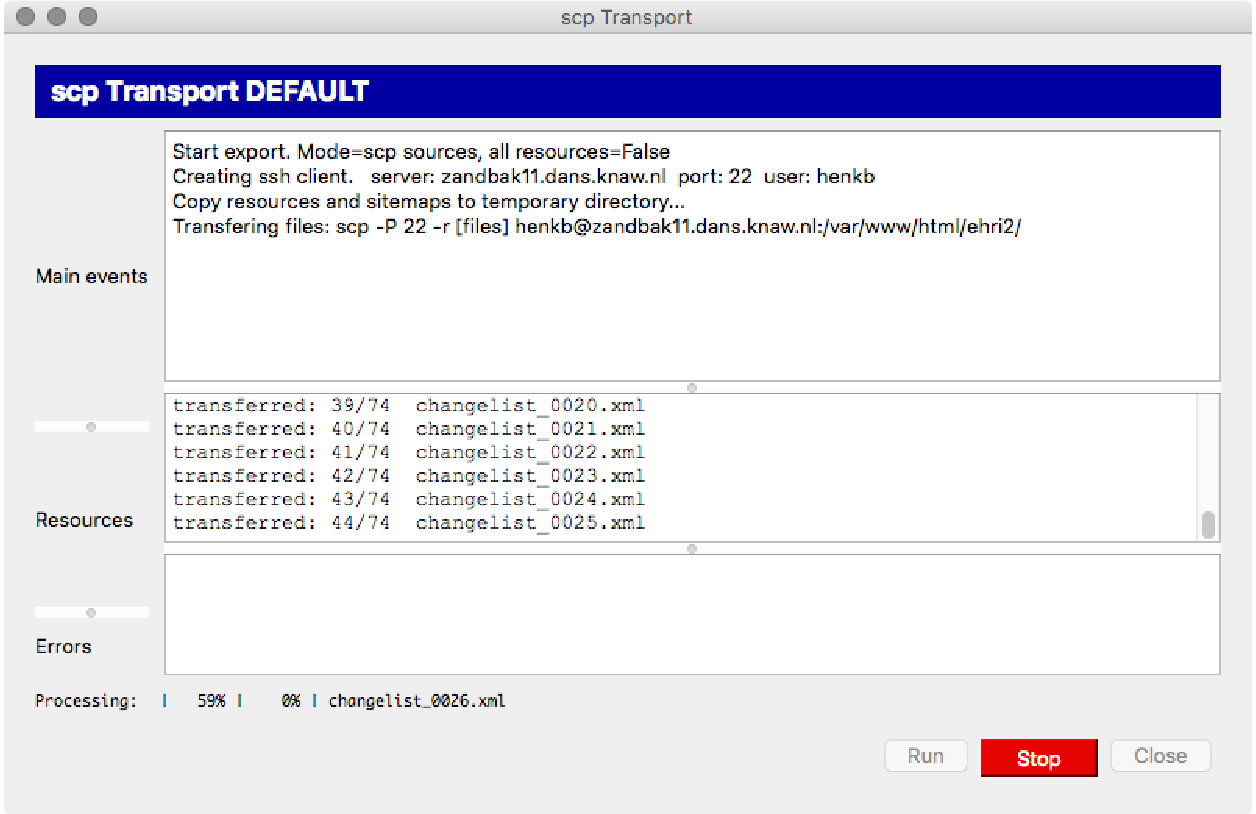


Fig. 9.4: *The scp Transport execution window*



```
Start export. Mode=scp sources, all resources=False
Creating ssh client.   server: zandbak11.dans.knaw.nl  port: 22  user: henkb
Copy resources and sitemaps to temporary directory...
Transfering files: scp -P 22 -r [files] henkb@zandbak11.dans.knaw.nl:/var/www/html/ehri2/
```

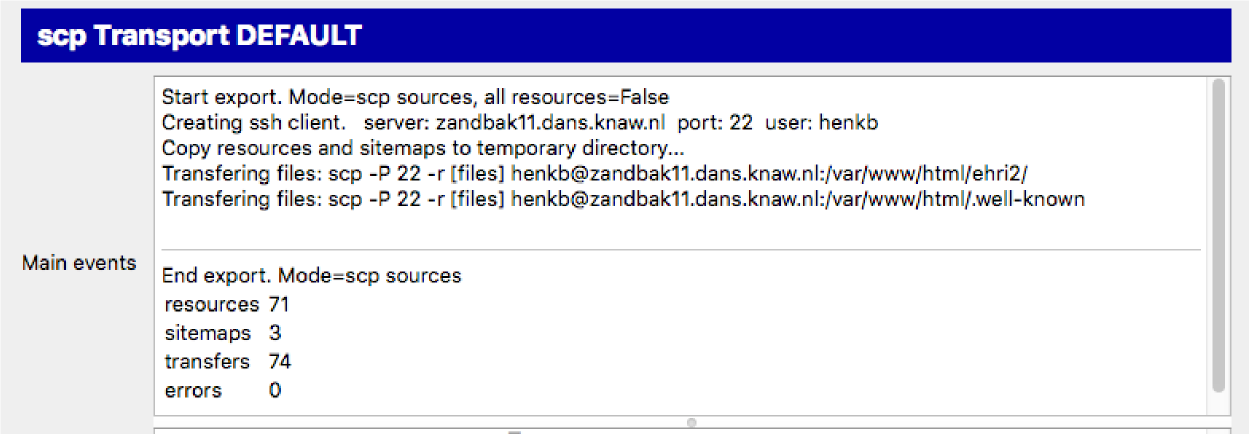
```
transferred:  39/74  changelist_0020.xml
transferred:  40/74  changelist_0021.xml
transferred:  41/74  changelist_0022.xml
transferred:  42/74  changelist_0023.xml
transferred:  43/74  changelist_0024.xml
transferred:  44/74  changelist_0025.xml
```

```
Processing:  |   59% |    0% | changelist_0026.xml
```

Fig. 9.5: *Screenshot of the scp Transport execution window while the export process is running*

While the import process is running you may at any time press the `Stop` button to interrupt the process.

After the export has finished without errors your *resource*s and *sitemap*s at the web server are now up to date with the local changes after your latest *synchronization*.



scp Transport DEFAULT

Main events

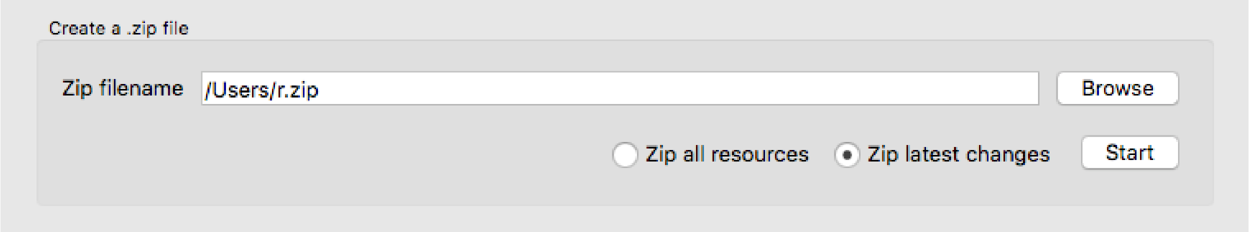
```
Start export. Mode=scp sources, all resources=False
Creating ssh client.   server: zandbak11.dans.knaw.nl  port: 22  user: henkb
Copy resources and sitemaps to temporary directory...
Transfering files: scp -P 22 -r [files] henkb@zandbak11.dans.knaw.nl:/var/www/html/ehri2/
Transfering files: scp -P 22 -r [files] henkb@zandbak11.dans.knaw.nl:/var/www/html/.well-known

End export. Mode=scp sources
resources  71
sitemaps   3
transfers  74
errors     0
```

Fig. 9.6: *Partial screenshot of the scp Transport execution window after the export process has finished*

Attention: You need to have write access to the *Document root/Document path* at the remote server. If not, you will have received an error message in the scp Transport execution window.

Also, if your *source description* is *at server root*, you need to have write access to the *.well-know* directory at the *Document root* of the web server.

9.2 Create a zip file



Create a .zip file

Zip filename: /Users/r.zip

○ Zip all resources ● Zip latest changes

Fig. 9.7: *Detail of the export page for creating zip files*

Zip filename Fill in the name of the zip file that will be created. Use the *Browse* button to open a file explorer that enables choosing the zip filename.

You have a choice between zipping all *resource*s and zipping only the latest changes.

- **Zip all resources** will include all resources mentioned in the *sitemap* documents currently in the *metadata directory*. Useful if you want to completely update the part of your site that hosts *resource*s and *sitemap*s of the current *configuration*.
- **Zip latest changes** will only include resources that where affected according to the last *synchronization*. The default. This will suffice in most occasions.

Press *Start* to open the zip Transport execution window.

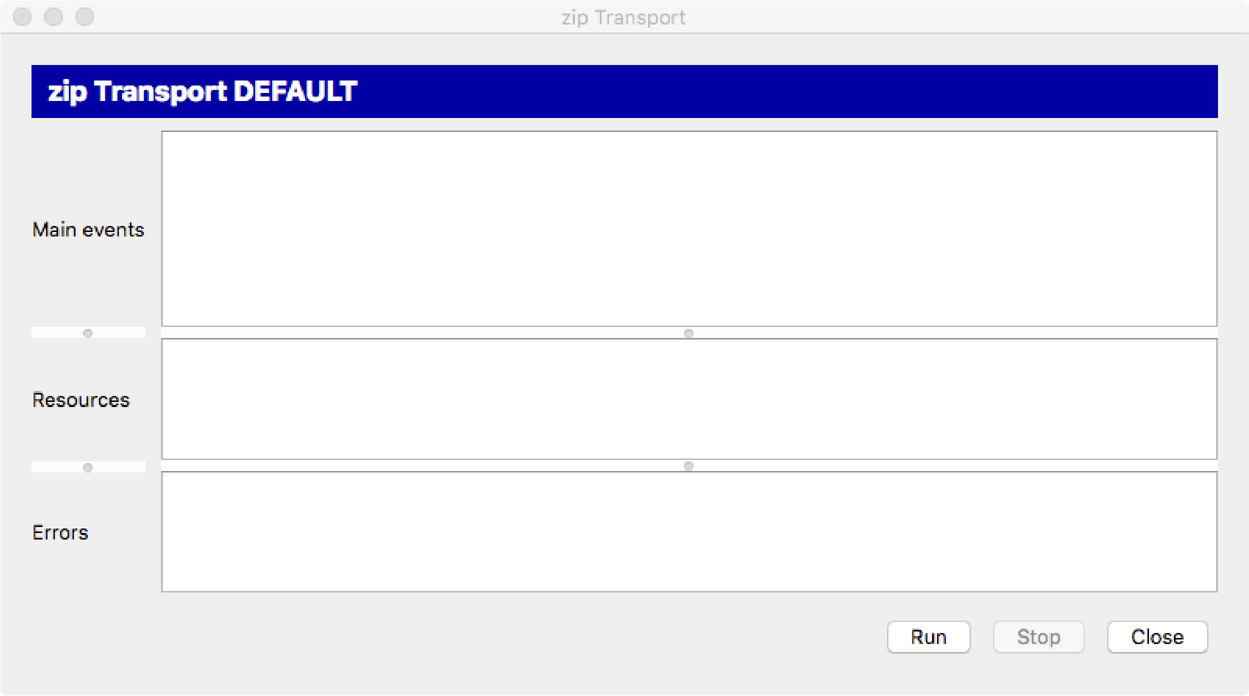


Fig. 9.8: *The zip Transport execution window*

The zip Transport execution window has three areas for reporting events:

Main events In this area main events of the zip process will be reported.

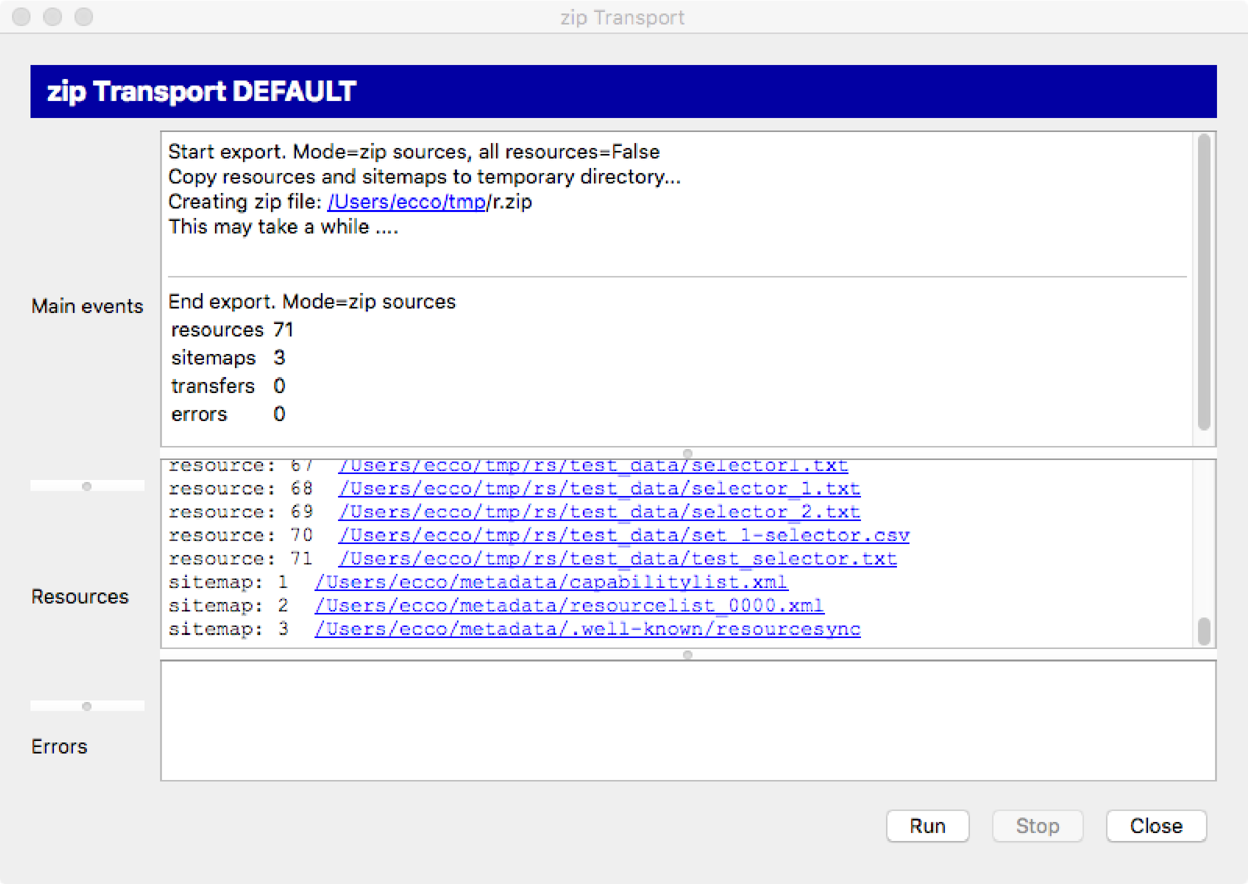
Resources In this area files that are zipped are listed.

Errors In this area errors that took place during the zip process are reported.

Press the *Run* button to start the zip process. While the zip process is running you may at any time press the `Stop` button to interrupt the process.

You now need to hand over the zip file to the system administrator of your web server. She should take care of copying *resource*s and *sitemap*s to the correct location on the web server.

> **Attention:** No matter what you chose at *'well-known' at server root* during *Configuration*, the *source description* will always be in the metadata directory in the zip file. Your system administrator should take care to place it in the correct location on the web server. Either keep it in the metadata directory (*'well-known' at server root* was set to **False**) or copy it to the `{Document root}/.well-known/resourcesync` (*'well-known' at server root* was set to **True**).



```
Start export. Mode=zip sources, all resources=False
Copy resources and sitemaps to temporary directory...
Creating zip file: /Users/ecco/tmp/r.zip
This may take a while ....

End export. Mode=zip sources
resources  71
sitemaps   3
transfers  0
errors     0
```

```
resource: 67  /Users/ecco/tmp/rs/test_data/selector1.txt
resource: 68  /Users/ecco/tmp/rs/test_data/selector_1.txt
resource: 69  /Users/ecco/tmp/rs/test_data/selector_2.txt
resource: 70  /Users/ecco/tmp/rs/test_data/set_1-selector.csv
resource: 71  /Users/ecco/tmp/rs/test_data/test_selector.txt
sitemap: 1  /Users/ecco/metadata/capabilitylist.xml
sitemap: 2  /Users/ecco/metadata/resourcelist_0000.xml
sitemap: 3  /Users/ecco/metadata/.well-known/resourcesync
```

Fig. 9.9: *The zip Transport execution window after the zip process has finished*

CHAPTER 10

Audit

Check sitemaps, resources and URL's on the web server

Important: The audit page enables verification of the *sitemap* and *resource* URL's on your web server. After a successful audit, you can rest assure that your *ResourceSync* site is healthy.

The audit page shows the current *configuration*, the location of the metadata the audit will be based upon, at what date and time the last execution of the *synchronization* took place, the web server that is being tested and the *Document path* for documents of the current *configuration*.

10.1 The audit process

You have a choice between auditing all *resource*s and auditing the latest changes.

- **Audit all resources** will include all resources mentioned in the *sitemap* documents currently in the *metadata directory*. Useful if you want to completely audit the part of your site that hosts *resource*s and *sitemap*s of the current *configuration*.
- **Audit latest changes** will only include resources that where affected according to the last *synchronization*.

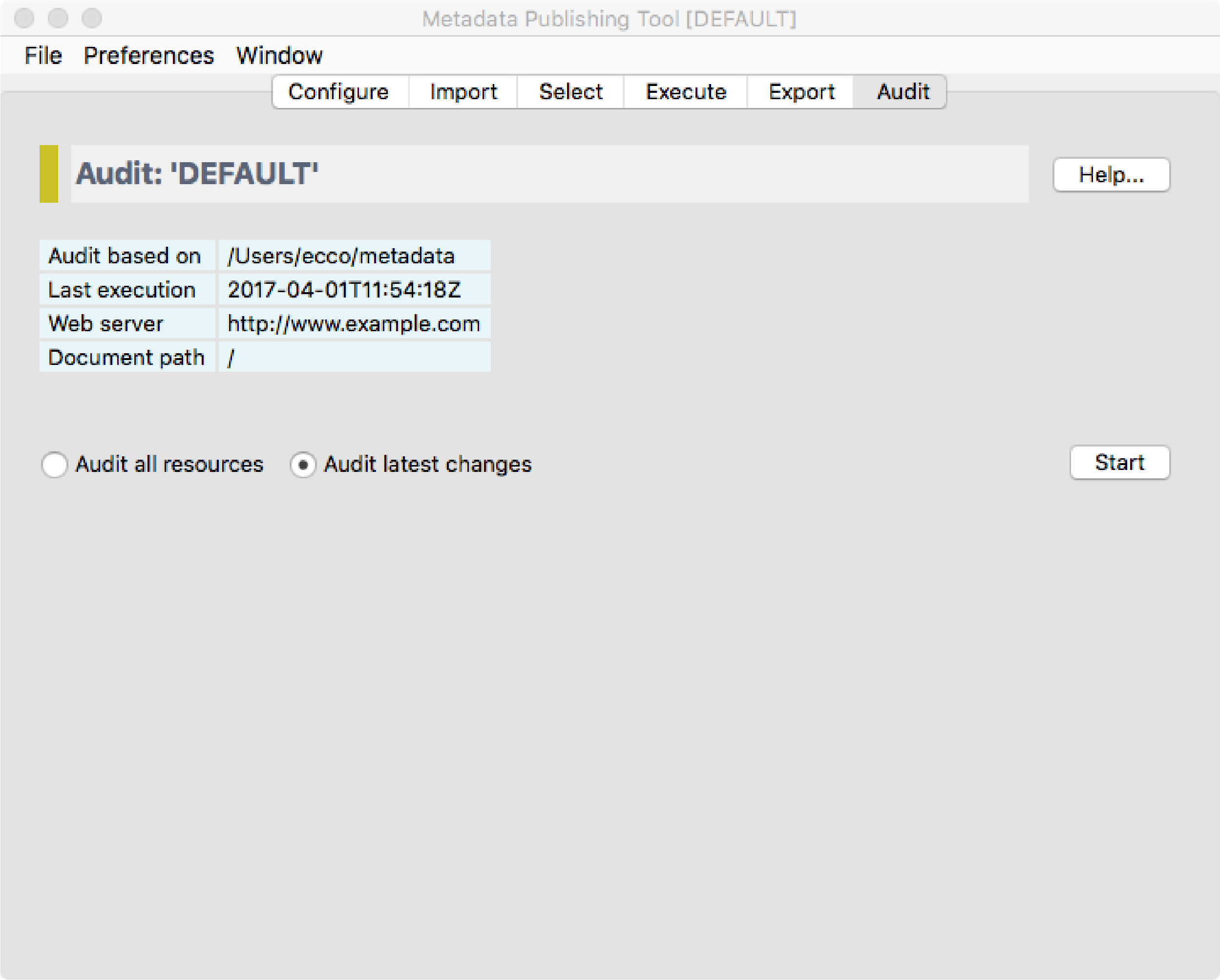
All *sitemap*s will be included in the audit, no matter what choice you made for included *resource*s.

Press *Start* to open the Audit execution window.

The Audit execution window has three areas for reporting events:

Main events In this area main events of the audit process will be reported.

Resources In this area URL's that are being checked are listed.



Audit: 'DEFAULT'

Audit based on	/Users/ecco/metadata
Last execution	2017-04-01T11:54:18Z
Web server	http://www.example.com
Document path	/

○ Audit all resources ● Audit latest changes

Fig. 10.1: *Screenshot of the audit page*

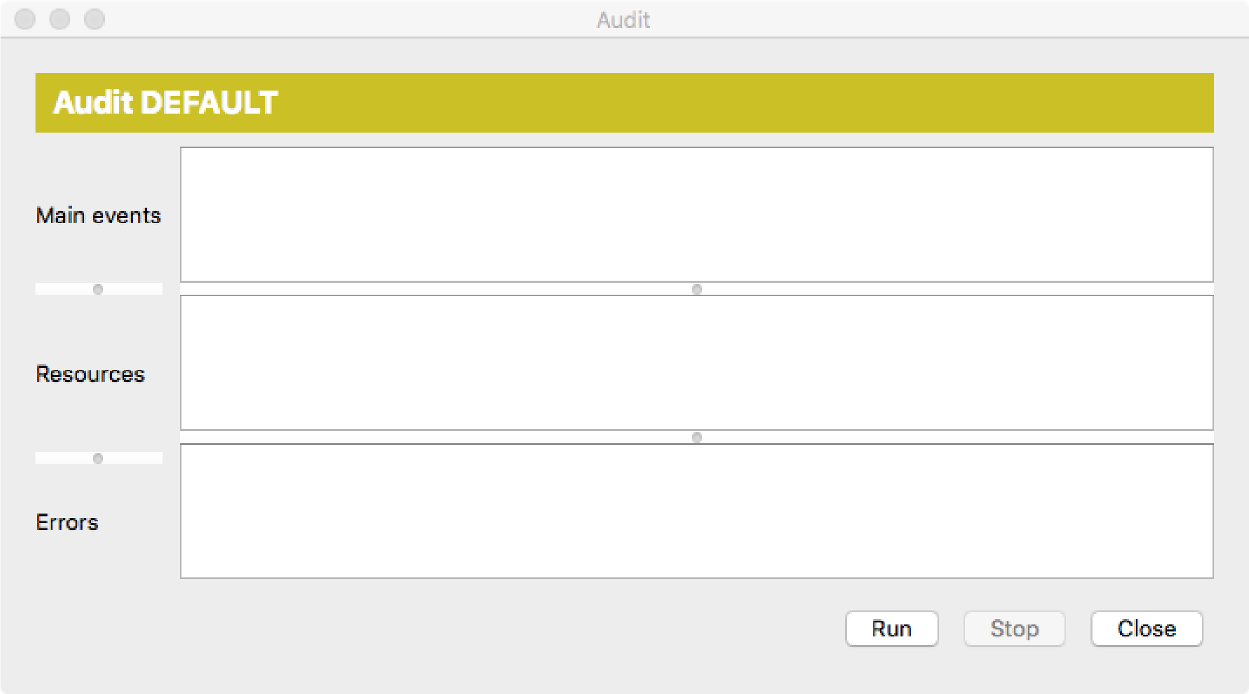


Fig. 10.2: *The Audit execution window*

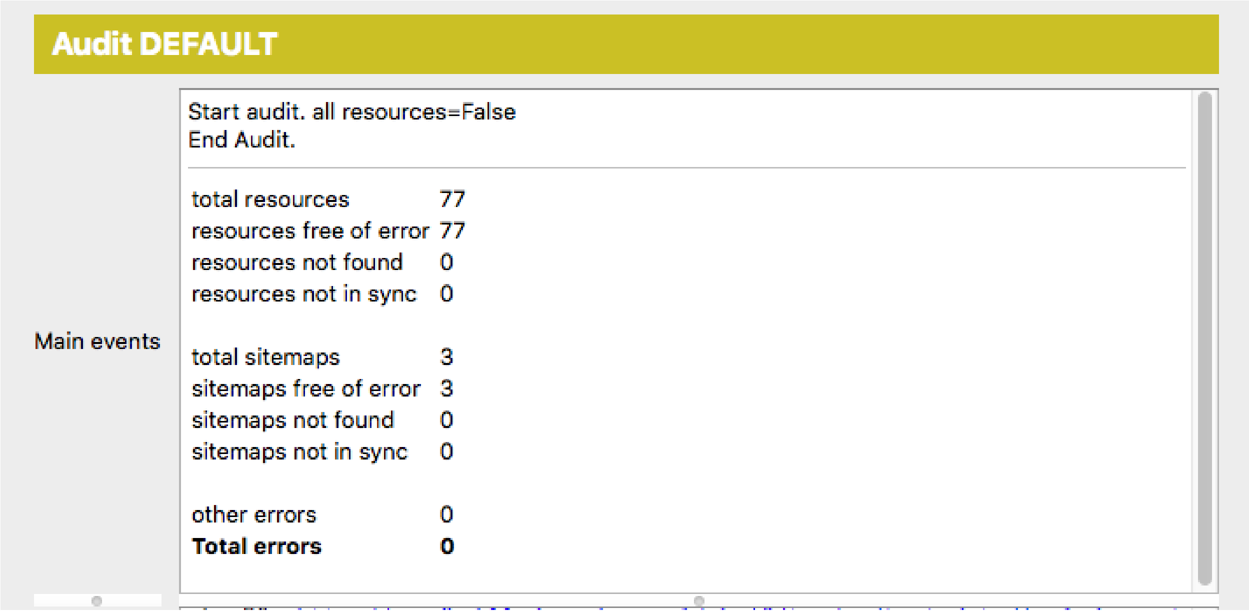
Errors In this area errors that took place during the audit process are reported.

Press the *Run* button to start the audit process.

While the audit process is running you may at any time press the `Stop` button to interrupt the process.

The audit was successful if the **Total errors** in the display reports 0 errors.

If the audit was not successful, try to analyze the reported errors and adjust *Configuration* and/or repeat the *Export* process.



```
Start audit. all resources=False
End Audit.

total resources          77
resources free of error  77
resources not found      0
resources not in sync    0

total sitemaps           3
sitemaps free of error   3
sitemaps not found       0
sitemaps not in sync     0

other errors             0
Total errors             0
```

Fig. 10.3: *Partial screenshot of the Audit execution window after the audit process has finished*

CHAPTER 11

Miscellaneous

Various other items concerning Metadata Publishing Tool

11.1 Reporting issues

Issues, bugs, crashes, unwanted or unexpected behavior of the *MPT* application can be reported at gitHub' rspub-gui issues. If relevant, please include the latest log file.

11.2 Log file location

The *MPT* application is creating log files. The log file location for various operating systems can be found in the table below.

Operating System	Log file location
Windows	`{user home}\AppData\Local\rspub\logs\rspub.log`
Mac OS and Linux	`{user home}/.config/rspub/logs/rspub.log`
Other	`{user home}/rspub/logs/rspub.log`

11.3 Contributing a language

The *MPT*-team appreciates if you can and will contribute a translation of the text on buttons, menus and descriptions in the *MPT* application. Here is how to.

1. Download and install the free Poedit application from https://poedit.net/download.
2. The `rspub.pot` template file contains the original entries in English. Download the `rspub.pot` template file.
   - Go to https://raw.githubusercontent.com/EHRI/rspub-gui/master/i18n/rspub.pot
   - Right-click on the page and choose `Save as` from the popup menu.
3. Start Poedit and on the Welcome page click `Create New Translation`.
4. Choose the `rspub.pot` template file you downloaded in step 2.
5. Choose language (and region) for the translation you want to contribute.

Start translating. Choose an entry in the top panel. Enter tour translation in the text area below. The Poedit application may give suggestions on the right-most side panel.

Save your translation as a **.po* file and contact the EHRI-team on how to send in your translation.

Alternatively you may clone the GitHub repository and issue a pull request for your translation.

If all goes well, your translation will be included in the next version release of the *MPT* application.

CHAPTER 12

Glossary

Glossary of terms used in context with Metadata Publishing Tool

B2DROP "B2DROP is a secure and trusted data exchange service for researchers and scientists to keep their research data synchronized and up-to-date and to exchange with other researchers."

– EUDAT web site

Together with a *WebDAV* client on a server it is an alternative for using *scp* for exchanging *resource*s and *sitemap*s between servers and local drives.

See also:

B2DROP EUDAT service

Mount B2DROP using the WebDAV client ownCloud Manual

Capability List A Capability List is an xml-document of type *sitemap* that enlists *Resource List*s and *Change List*s that contain links to a particular *set of resources*.

See also:

Capability List ResourceSync Framework Specification

Change List A Change List is an xml-document of type *sitemap* that enlists *resource*s that have changed since a previous *synchronization*.

See also:

Change List ResourceSync Framework Specification

configuration A named set of parameters that constitute all variables needed to *synchronize* a *set of resources*. The first configuration will be saved under the name 'DEFAULT'. Configurations can be loaded, saved, listed and deleted under the *File* menu. The parameters that are set under a configuration are automatically saved.

See also:

Configuration Configure page

DANS Data Archiving and Networked Services. Netherlands Institute for Permanent Access to Digital Research *Resource*s. DANS is an institute of the Dutch Academy KNAW and funding organisation NWO.

See also:

Data Archiving and Networked Services (DANS)

description directory The description directory is an existing directory on the (local or networked) filesystem. In this directory (a copy of) the document that describes the entire site, also known as `.well-known/resourcesync` or *Source Description* is expected or will be created. If no description directory is given, the document is expected or will be created in the *metadata directory*.

See also:

Configuration Configure page

Describing the Source ResourceSync Framework Specification

Destination A system that *synchronize*s itself with a *Source*'s *resource*s.

See also:

Definitions ResourceSync Framework Specification

EAD "Encoded Archival Description (..) is a non-proprietary de facto standard for the encoding of finding aids for use in a networked (online) environment. Finding aids are inventories, indexes, or guides that are created by archival and manuscript repositories to provide information about specific collections. While the finding aids may vary somewhat in style, their common purpose is to provide detailed description of the content and intellectual organization of collections of archival materials. EAD allows the standardization of collection information in finding aids within and across repositories."

– The Library of Congress, Official Site

See also:

Encoded Archival Description

EHRI European Holocaust Research Infrastructure.

"The mission of the European Holocaust Research Infrastructure (EHRI) is to support the Holocaust research community by building a digital infrastructure and facilitating human networks."

– About EHRI

See also:

European Holocaust Research Infrastructure

incremental changelist strategy Will increment an existing *Change List* with the newly found changes.

See also:

Configuration Configure page

metadata directory The name of the directory where generated *sitemap*s are stored. The value of metadata directory may consist of multiple path elements. The metadata directory is always relative to the *resource directory*.

See also:

Configuration Configure page

MPT

Metadata Publishing Tool Metadata Publishing Tool (MPT) is an application for publishing *resource*s in conformance with the *ResourceSync Framework Specification*. Metadata Publishing Tool was developed by Data Archiving and Networked Services (*DANS*-KNAW) under auspices of the European Holocaust Research Infrastructure (*EHRI*).

See also:

rspub-core at gitHub The base library under MPT

rspub-gui at gitHub The source code of the graphical user interface under MPT

new changelist strategy Will create a new *Change List* at each *synchronization* run.

See also:

Configuration Configure page

new resourcelist strategy At each *synchronization* run a completely new *Resource List* will be generated.

See also:

Configuration Configure page

plugin directory In this directory or its subdirectories a search for plugins will be conducted.

See also:

Configuration Configure page

ResourceGateBuilder Documentation on rspub-core

resource In web technology, a data resource is anything that can be accessed with a link. It can for instance be a file, (part of) a database or the output of a program.

resource directory The base directory on the (local or networked) filesystem where *resource*s that should be *synchronize*d are stored. The resource directory should be chosen careful, because it influences the composition of the URL to the *resource*.

See also:

Configuration Configure page

Resource List A Resource List is an xml-document of type *sitemap* that enlists available *resource*s on a particular site.

See also:

Resource List ResourceSync Framework Specification

ResourceSync

ResourceSync Framework Specification The ResourceSync specification describes a *synchronization* framework for the web consisting of various capabilities that allow third-party systems to remain *synchronize*d with a server's evolving *resource*s.

See also:

ResourceSync Framework Specification Open Archives Initiative ResourceSync Framework Specification

scp Secure Copy Protocol. SCP copies files over a secure, encrypted network connection.

See also:

Linux and Unix scp command

set of resources "A collection of *resource*s that is made available for *synchronization* by a *Source*. A *Source* may expose one or more such collections and support distinct ResourceSync capabilities for each. Individual *resource*s may be included in more than one set of *resource*s"

– ResourceSync Framework Specification

See also:

Definitions ResourceSync Framework Specification

sitemap

sitemap protocol An XML schema for xml-documents that describe the *resource*s of a site. The *ResourceSync Framework Specification* makes use of this protocol to express the location of *resource*s and to express changes that affected this *set of resources*. ResourceSync specifies these sitemaps:

- Source Description enumerates the *Capability List*s offered by a *Source*
- Capability List enumerates all capabilities supported by a *Source* for a specific *set of resources*
- Resource List lists and describes the *resource*s that a *Source* makes available for *synchronization*
- Resource List Index for grouping multiple *Resource List*s
- Resource Dump used to transfer *resource*s from the *Source* in bulk
- Resource Dump Manifest describes the bulk package's constituents
- Change List contains a description of changes to a *Source*'s *resource*s
- Change List Index for grouping multiple *Change List*s
- Change Dump used to transfer changed *resource*s from the *Source* in bulk
- Change Dump Manifest describes the bulk package's constituents

See also:

Sitemap protocol Official site

Source A server that hosts *resource*s subject to *synchronization*.

See also:

Definitions ResourceSync Framework Specification

Source Description In the context of *ResourceSync Framework Specification* the document at:

```
{server root}/.well-known/resourcesync
```

This document describes the site by listing all *Capability List*s that are available from the site. This practice is an extension on the *well-known URI* scheme, also known as RFC5785.

See also:

ResourceSync Well-Known URI ResourceSync Framework Specification

strategy The strategy defines what kind of *sitemap* documents will be generated when a *synchronization* is executed.

See also:

Configuration Configure page

synchronization

synchronize Keeping a *set of resources* at a *Destination* the same as the *set of resources* at a *Source*.

As seen from the *Destination* this includes copying the original *set of resources* from the *Source* and then keeping up to date with additions to, changes of and deletions from the *set of resources* at the *Source*.

As seen from a *Source* this involves providing the *resource*s and the metadata that describe the changes to this *set of resources*.

See also:

Source perspective ResourceSync Framework Specification

Destination perspective ResourceSync Framework Specification

Execute a synchronization Execute page

trial run The execution of a *synchronization* run that will not write *sitemap*s to disk.

See also:

Save sitemaps to disk

Synchronize resources

URL prefix The URL prefix is the basename of the site, optionally followed by a path segment.

See also:

Configuration Configure page

WebDAV "Web Distributed Authoring and Versioning (WebDAV) is an extension of the Hypertext Transfer Protocol (HTTP) that allows clients to perform remote Web content authoring operations. WebDAV is defined in RFC 4918 by a working group of the Internet Engineering Task Force."

> – Wikipedia

See also:

B2DROP EUDAT service

Mount B2DROP using the WebDAV client ownCloud Manual

well-known URI RFC5785. An internet standard that defines a path prefix for "well-known locations"

See also:

RFC5785 specification

Index