mzDBWizard User Guide

# Introduction

**mzDBWizard** was born in order to alter the labor intensive workflow that is followed when .raw files need to be converted and uploaded. It is a simple tool whose purpose is relieve the end user from the burden that chaperons the manual manipulation of .raw files. In order to achieve this, the application was built with a daemon-centric logic in mind requiring minimum interaction from the part of the user.

It works by automatically detecting file creations within a directory of the local file system. As long as a new file is detected, appropriate actions take place depending on the file type. If the latter one corresponds to a .raw file, a conversion mechanism is activated leading to the creation of an .mzdb file relying on the well established, raw2mzDB converter. On the other hand, when an .mzdb file is created within the monitored folder, **mzDBWizard** uploads it by using the existing functionality and infrastructure of the server.

# Installation

## System Requirements

The requirements that derive from software's local and network activity can be summarized into the three following prerequisites:

- 1. Java SE JRE of Version 7 or newer installed
- 2. An running Proline Server
- 3. A working copy of raw2mzDB converter

## Installation Using Archive Files

Given that all the aforementioned requirements are met, the installation of **mzDBWizard** software is a straightforward procedure as it only demands the respective **mzDBWizard** archive to be locally decompressed. From that moment and as long as all three requirements are respected the software is ready for launch and use.

# **Getting Started**

# Launching mzDBWizard

**mzDBWizard** can be launched executing the file start.bat which can be found inside the decompressed root folder as shown in the following screenshot.



# **Opening Dialog**

Start.bat's execution is always followed by an opening dialog that serves as a configurator for the current execution of **mzDBWizard**. The dialog encapsulates input fields for all the information that is required by the program. As seen on the respective screenshot, when the program is initiated, the user has to dictate the folder to be monitored, the **mzDBWizard** copy, the server's address or alias and the respective mounting point. In an effort to minimize the required workload, the filled in information is saved, eliminating the need to fill this information every time unless a modification is needed.

Monitored Directory D:\upload_test	
Converter's Path C:\Users\AK249877\Documents\NetBeansProjects\mzDB-wizard\target\.\converter\mzdb_x64_0.9.8d\raw2mzDB.exe	
Server gre046180	
Server Path Label (Mounting Point) mzdb_PROD	-
Start Kancel	

# Main User Interface

**mzDBWizard** has a built-in graphical user interface. The latter one, inextricably linked to the minimization of the user's workload serves a range of purposes, including among others:

- Conversion/upload tasks management
- Conversion/upload tasks extensive monitoring
- Configuration
- Debugging

Following this generic wireframe, the interface was organized into three main tabs, plus one fixed part that provides a quick overview of the current execution's attributes (Monitored directory, Converter in use, Server etc.).

<u>F</u> ile <u>H</u> elp							
MONITORED DIRECTORY							
USED CONVERTER C:\Users\AK249877\Documents	USED CONVERTER C:Users\AK249877\Documents\NetBeansProjects\mzDB-wizard\target\.converter\mzdb_x64_0.9.8d\raw2mzDB.exe						
USED SERVER gre046180							
MOUNTING PATH LABEL							
Task Manager 📄 Overview	Logfile Main Tabs			Fixed Part			
Tasks							
Туре	File	Dispatch Time	State	Errors/Warnings			
Converter	QEx2_007990b.raw	2017 01 11 17:21:06	FINISHED	21			
🖓 Uploader	QEx2_007990b.mzdb	2017 01 11 17:21:51	STOPPED	1			
🖓 Uploader	QEx2_007990b.mzdb	2017 01 11 17:22:21	ACTIVE	0			
Details							
🕕 Errors & Warnings 📄 Log							
File converter C:\Users\4K249877\Doc D:\upload_test\QEx2_007990b.mzdb.tr Converting for file: D:\upload_test\QEx2	umentsWetBeansProjects\mzDB-wizard\ mp size is 257232896 bytes 2_007990b.raw has come to its end.	targetl.\converterImzdb_x64_0.9.8dtraw2m	zDB.exefor file D∴upload_testiQEx2_	007990b raw has just been called.			

#### Task Manager Tab

The task manager dedicated tab, which can be seen at the previous screenshot is the cornerstone of the interface. Its main virtue is the fact that it provides very detailed information about all the launched tasks in an easily digestible way, using different elements (colors codes, image icons). Furthermore, apart from the provided situation awareness, it also provides the user with the means to manage tasks in the form of buttons and pop-up menus. In this context, the user has the ability to stop, restart or clear a task. It must be noted that a task that has entered its critical part cannot be

killed. A critical point can be seen as a "Rubicon" in the process which usually has no return. An example of a critical point in the process of uploading an .mzdb file is the establishment of the required connection with the server. Once the connection is active, killing the connection would lead to corrupted files at the server's side. As it is also seen in following screenshot, task manager also consists of two sub-tabs, one dedicated to logs and one that serves as a summary of all the encountered errors and warnings. Both sub-tabs regard only the selected task, highlighted with a black border.

<u>File</u> <u>H</u> elp						
MONITORED DIRECTORY						
D:\upload_test	D:upload_test					
USED CONVERTER						
C:\Users\AK249877\Documents\	NetBeansProjects\mzDB-v	vizard\targe	t\.\converter\mzdb_x64_0.9.8d\raw2mz	DB.exe		
USED SERVER						
gre046180						
MOUNTING PATH LABEL						
mzdb_PROD						
Tack Manager	Logfile					
Stop						
Туре	File		Dispatch Time		State	Errors/Warnings
Converter	QEx2_007990b.r	aw	2017 01 11 17:21:06		FINISHED	21
🖓 Uploader	QEx2_007990b.m	zdb	2017 01 11 17:21:51		STOPPED	1
🖓 Uploader	QEx2_007990b.m	zdb	2017 01 11 17:22:21		STOPPED	1
Details						
🕒 Errors & Warnings 📄 Log						
Importance			Title			Details
			JSON Error		Destination file already ex	istQEx2_007990b.mzdb in mzdb_PR

#### Overview Tab

As its name denotes, overview serves as a general overview focusing on the launched tasks on the contrary with the fixed part of the graphical user interface, as defined earlier, that although serves a very similar purpose, focuses on the current execution's attributes.

Task Manager 🔁 Overview 🖻 Logfile
Monitored Directory
D:\upload_test
Used Converter
$\label{eq:c:Users} \end{target} C: \end{target} ta$
Active Tasks
0
Finished Tasks
3
Total Tasks
3
First Dispatch Time
2017 01 11 17:21:06
Last Dispatch Time
2017 01 11 17:22:21

#### Logfile Tab

The logfile tab can be compared with a centralized logfile regarding the launched tasks as the well as the application itself. As we can see, the logs themselves are organized using different color codes:

- **Bold Red**, for error messages
- Blue, for debugging messages
- Black, for information messages

Although it is available for the user, it is highly unlike that the end user will use it as it was mainly built for debugging reasons. In case that he wants to access logs, the relevant task manager subtab can be used. An advantage for the latter one is that it corresponds to a decluttered version.

🔚 Task Manager 🛛 🗁 Overview 📄 Logfile	
💞 Clear	
D:\upload_test\QEx2_007990b.mzdb size is 257232896 bytes	
11 janv. 2017 17:22:48.533 [pool-5-thread-2] INFO_c.p.m.r.workers.FileUploaderWorker - It took: 27271 milli to send the file!	seconds
11 janv. 2017 17:22:48.543 [pool-5-thread-2] DEBUG org.hornetq.core.client - client ack messageID = 2164	363162
11 janv. 2017 17:22:48.543 [pool-5-thread-2] DEBUG c.p.m.r.workers.FileUploaderWorker - JSON Respons ID:20cfd7d9-d81a-11e6-b14b-61960a0e0f77	e Id:
11 janv. 2017 17:22:48.543 [pool-5-thread-2] ERROR c.p.m.r.workers.FileUploaderWorker - Error code -32002, message : "Destination file already existQEx2_007990b.mzdb in mzdb_PROD\mzDB-wizard\\"	JSON
11 janv. 2017 17:22:51.543 [pool-5-thread-2] DEBUG org.hornetq.core.client - Calling close on session ClientSessionImpl [name=20ccca98-d81a-11e6-b14b-61960a0e0f77, username=null, closed=false, factory ClientSessionFactoryImpl [serverLocator=ServerLocatorImpl	=
[initialConnectors=[TransportConfiguration(name=20c9bd53-d81a-11e6-b14b-61960a0e0f77,	

# Use Case Example

In order to familiarize ourselves with **mzDBWizard** we will proceed with a Use Case Example, demonstrating the basic functionality of the software. The goal is to launch, configure and then process a user folder that contains both .raw and .mzdb files. For the sake of completeness we will also deliberately provoke an error so that the user becomes familiar with error handling. The example's workflow is organized into eight (8) steps as follows:

- 1. Launch mzDBWizard
- 2. Initiate the application using opening dialog
- 3. Modify preferences
- 4. Trigger a detection
- 5. Visualize progress
- 6. Provoke/handle an abnormal situation

## 1. Launch mzDBWizard

As mentioned before, in order to launch **mzDBWizard** it suffices to execute the file start.bat, located inside **mzDBWizard** root directory.



## 2. Initiate the application using the opening dialog

Once start.bat file is executed, the opening dialog of **mzDBWizard** will appear. Based on the assumption that this is the first execution of the application we will have to verify or modify the respective form accordingly starting from the selection of the directory to be monitored. We select the on purpose created "monitored\_directory" and press ok as shown in the following screenshot.

Welcome to mzDB-wizard 0.1.0	실 Ouvrir			<b>x</b>	
Monitored Directory	Rechercher dans :	Données Non Sauvegat	rdées (D:) 🔻		1
C:\Users\AK249877\Documents\N	Andreas	📑 marjolaine	Contraction of the second seco	src 📑	
Converter's Path	Backup_ETNA	Matt	Pdf	Temp	
C:\Users\AK249877\Documents\N	Data	mzdb_x64_0.9.8d	Save.zip	rest	
Server	datasets	i output	Scripts	📑 Username	
gre046180	Dev	output4	Servers		
Server Path Label (Mounting Point)		output5	small		
mzdb_PROD					-
L	Nom du dossier :	D:\monitored_directory			
	Type de fichier :	Tous les fichiers		<b></b>	
			2 0	Ouvrir Annuler	Cancel

The next item to our initial configuration is converter that will be used for this execution cycle. As we can see, if not changed, the selected "Converter's Path" points to the converter executable file that is shipped/packed with **mzDBWizard**. In case that a different version is desired due to the special nature of a file in question, another converter executable can be selected following the same logic with the selection of the "Monitored Directory". The last two items to check on the opening dialog are the server's address and the respective mounting point. For the purposes of this tutorial we will leave them as they are, as they correspond to the laboratory's server. Finishing the configuration we press ok, event that automatically starts the monitoring activity and shows the main dialog of the

application. Note that as mentioned before, we can validate the configuration that just took place by looking to the fixed part at the top of the dialog.

<u>F</u> ile <u>H</u> elp						
MONITORED DIRECTORY						
USED CONVERTER	USED CONVERTER					
C.IUSEISIAR2490771DUCUITE	Its weldeansprojects in 200	-wizarutargett.converterinizub_x04_0.9.80	adw21112DD.exe			
USED SERVER gre046180						
MOUNTING PATH LABEL						
Task Manager 🔂 Overview	E Logfile					
Tasks						
Stop						
Туре	File	Dispatch Time	State	Errors/Warnings		
Details						
🕒 Errors & Warnings 📄 Log	]					
Importance		Title		Details		

We observe that, "D:\monitored\_directory" is indeed the folder that is currently under the diligent "eye" of the application. Since the newly created directory is currently empty, no task exist yet.

### 3. Modify Preferences

Before triggering our first file detection, it is appropriate to have a first contact with the preferences dialog within the application. The latter can be accessed using the menu-bar at the top and gives us access to a short list of options regarding the behavior of the application. For the sake of

completeness we must state that more options do exist in a respective configuration file. As their more technical nature surpasses the boundaries of this Use Case Example so we will refrain from referring to them for now. The two most significant items on the dialog are the two ones that regard the deletion of a file after an action has taken place. The features were introduced as an answer to the awareness that most of the time the application will be

User Preferences
Preferences
✓ Delete .raw file after its convertion.
✓ Delete .mzdb file after its upload.
✓ Open mzDB-wizard in fullscreen.
Save X Cancel

executed on shared laboratory computers with limited local space for which different users compete. At this point it is important to add to the self-explanatory labels that a deletion takes place only when a task was successfully finished. In cases of critical events the deletion is aborted preventing any possible loss of data.

## 4. Trigger a detection

As previously explained, from the moment the main dialog appears, the selected directory will be monitored in a recursive way. The latter feature can be exploited in order to have a more organized destination directory. It is thus recommended that you firstly create a new folder, inside which you will deposit the files that are of your interest and that would like to convert and upload. At the destination, files will be saved respecting your structure, regardless of its depth, creating any not already existing participating directories. One good practice could thus be to create a folder named after a name or username and then create a number of subfolders named after an attribute like the project name. For the purposes of our example we consider as suffice to create one root folder named "username" which contains only one subfolder named "project-x" consisting of two files, "testfile1.raw" and "testfile2.mzdb". In order to trigger the detection it is sufficient to just copy "username" folder into "D:\monitored\_directory". This will dispatch the required procedures that can be seen at the task manager at the following screenshot.

### 5. Visualize Progress

As we can see, both files are successfully detected and are processed by appropriate procedures. At this point it is important to remember that we can either drop a .raw file, expecting it to be converted and then uploaded or directly an .mzdb file. The latter one does not need conversion, it will thus be directly uploaded. The following screenshot is taken a few seconds later showing two finished tasks plus one active one. Being observant we can distinguish that the conversion task finished with twenty one (21) warnings but no critical errors leading to the creation of "testfile1.mzdb" which triggered one more upload task shown as active at the moment the screenshot was taken.

File Help					
MONITORED DIRECTORY	MONITORED DIRECTORY				
D:\monitored_directory	D:Imonitored_directory				
USED CONVERTER					
C:\Users\AK249877\Documents\NetB	eansProjects\mzDB-wizard\target\.\conver	ter\mzdb x64 0.9.8d\raw2mzDB.exe			
	, ,				
USED SERVER					
gre046180					
MOUNTING PATH LABEL					
mzdb_PROD					
¥					
Task Manager Coverview	nfile				
	gino				
Tasks					
Stop					
Туре	File	Dispatch Time	State	Errors/Warnings	
Converter	testfile1.raw	2017 01 12 17:05:20	ACTIVE	0	
Uploader	testfile2.mzdb	2017 01 12 17:05:20	ACTIVE	0	
Details					
BErrors & Warnings					
Importance		Title		Details	
L					

D.unointored_unectory				
USED CONVERTER				
C:\Users\AK249877\Documents\NetE	BeansProjects\mzDB-wizard\target\.\converte	er\mzdb_x64_0.9.8d\raw2mzDB.exe		
USED SERVER				
gre046180				
MOUNTING PATH LABEL				
mzdb PROD				
Task Manager 🛛 🧮 Overview 🗍 🖃 Lo	ogfile			
asks				
ダ Clear 💥 Stop				
Туре	File	Dispatch Time	State	Errors/Warnings
Converter	testfile1.raw	2017 01 12 17:05:20	FINISHED	21
🖓 Uploader	testfile2.mzdb	2017 01 12 17:05:20	FINISHED	0
🖓 Uploader	testfile1.mzdb	2017 01 12 17:06:18	ACTIVE	0
බ Uploader Details	testfile1.mzdb	2017 01 12 17:06:18	ACTIVE	0
<ul> <li>Gal Uploader</li> <li>Betails</li> <li>              € Errors &amp; Warnings</li></ul>	testfile1.mzdb	2017 01 12 17:06:18	ACTIVE	0
Uploader      Letails     Errors & Warnings     Log     Importance	testfile1.mzdb	2017 01 12 17:06:18 Title	ACTIVE	0 Details
C Uploader  etails  Errors & Warnings  Log  Importance	testfile1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream	ACTIVE	0 Defails GoogleLogging() is written to STDERR
Uploader  etails  Errors & Warnings Log Importance	testfiie1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream Converter ErrorStream	ACTIVE WARNING: Logging before Ini 10112 17:05:32.783951 4748 m	0 Details GoogleLogging() is written to STDERR Zdb_writter htp:215) Thermo metadata
etails etails  Errors & Warnings  Log Importance  e  e  e  e  e  e  e  e  e  e  e  e	testfile1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream Converter ErrorStream Converter ErrorStream	ACTIVE WARNING: Logging before Ini 10112 17:05:32.783951 4748 m 10112 17:05:33.013957 4748 m 10112 17:05:33.013957 4748 m	0 Details GoogleLogging() is written to STDERR Zdb_writer.tpp:215] Thermo metadata Zdb_writer.cpp:411] updating dataProce
Uploader  etails  Errors & Warnings Log  Importance	testfile1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream Converter ErrorStream Converter ErrorStream Converter ErrorStream Converter ErrorStream	ACTIVE WARNING: Logging before ini 10112 17:05:32.783951 4748 m 10112 17:05:33.013957 4748 m 10112 17:05:33.013957 4748 m 10112 17:05:33.043957 4748 m	0 Details GoogleLogging() is written to STDERR 2db_writer.tpp.215] Thermo metadata zdb_writer.cpp.211] Thermo metadata zdb_writer.cpp.211] DhAI ded eded.e.
C Uploader  etails  Errors & Warnings Log  Importance    C  C  C  C  C  C  C  C  C  C  C  C	testfile1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream Converter ErrorStream Converter ErrorStream Converter ErrorStream Converter ErrorStream Converter ErrorStream	ACTIVE	0 Details GoogleLogging() is written to STDERR Zdb_writer.pp:215] Thermo metadata Zdb_writer.cpp:1014] DDA/ Swath test Zdb_writer.cpp:1014] DDA/ Swath test Zdb_writer.cpp:2015] DDA Mode detecte
Carlot Uploader  Carlot Control Contr	testfiie1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream	ACTIVE	0 Details GoogleLogging() is written to STDERR zdb_writer.top.215] Thermo metadata zdb_writer.cop.1014] DDA / Swath test zdb_writer.cop.1014] DDA / Swath test zdb_writer.top.332] SOLITE VERSION zdb_writer.top.333] ProteoWizard relea
Setalis  Errors & Warnings  Errors & Warnings  Errors & Warnings  Buportance  e  e  e  e  e  e  e  e  e  e  e  e	testfile1.mzdb	2017 01 12 17:06:18 Title Converter ErrorStream	ACTIVE  ACTIVE  WARNING: Logging before Ini I0112 17:05:32.783951 4748 m I0112 17:05:33.013957 4748 m I0112 17:05:33.043957 4748 m I0112 17:05:30.043957 478 m I0112 17:05:30.043957 478 m I0112 17:05:30.043957 478 m I0112 17:05:30.043957 478 m I0112 17:05:3058 m I0012 17:05:30058 m I0012 17:05 m I0012 17:05:30058 m I0012 17:05 m I0	0 Details GoggleLogging() is written to STDERR Zdb_writer.tpp.215) Thermo metadata Zdb_writer.tpp.1011 DDA Mode detecte Zdb_writer.tpp.332] SOLITE VERSION Zdb_writer.tpp.333] ProteoWizard relea Zdb_writer.tpp.333] ProteoWizard relea

# 6. Provoke/handle an abnormal situation

Once more, for the sake of completeness we will derive from the normal workflow in order to replicate an abnormal situation, or what is considered as a critical error within our application. One simple error that can occur is a duplicate file in destination. It occurs when an upload takes for a file that already exists in destination at the exact same path. As we can see in the relevant screenshot this will provoke a critical error. During the latter one, the state of the task in question will initially change to not-responding and will then be automatically be killed by the application itself. Moreover, given the fact that it failed to upload, even if in preferences, "Delete .mzdb file after upload" is selected file will not be deleted. Regarding handling, the application follows a try-once-more approach according to which it will try to restart once more a critically failed task. In case of a second failure the task is not restarted automatically demanding the explicit restarting order from the user. The order is given through by right clicking on the killed task and selecting the restarting option.

<u>File</u> <u>H</u> elp				
MONITORED DIRECTORY				
D.anontored_unectory				
USED CONVERTER				
C:\Users\AK249877\Documents\NetB	eansProjects\mzDB-wizard\target\.\converte	r\mzdb_x64_0.9.8d\raw2mzDB.exe		
USED SERVER				
are046180				
MOUNTING PATH LABEL				
mzdb_PROD				
Task Manager 🛛 🗁 Overview 🗍 🖻 Log	gfile			
Tasks				
Stop				
Туре	File	Dispatch Time	State	Errors/Warnings
@ Converter	testfile1.raw	2017 01 12 17:05:20	FINISHED	21
🖓 Uploader	testfile2.mzdb	2017 01 12 17:05:20	FINISHED	0
🖓 Uploader	testfile1.mzdb	2017 01 12 17:06:18	FINISHED	0
Converter	testfile1.raw	2017 01 12 17:11:16		21
🖓 Uploader	testfile1.mzdb	2017 01 12 17:12:51	STOPPED	1
Detelle				
Errors & Warnings				
Initiating uploading procedure for file: D:\mon	itored_directory/username\project-n\testfile1.r	nzdb		
D.\monitored_directorylusernamelproject-ntlestfile1.mzdb size is 707952640 bytes				
It took: 72723 milliseconds to send the file!				
JSON Response Id: ID:77(37eb9-d8e1-11e6-8d3-593b783bc67d				
ISON Error code ( 22002) Destination file als	andy existing the trian and be a set of the trian and tr			
Unloading for file: D'Imonitored, director/und	amama/anoiect-altestfile1 mzdb has come to it	s and		
oproduing for me. D. monitored_directorylase	smanne groje of miestine i mizuo nas come lo n	a unu.		-
(L				