

Proline Zero User Guide

Table of Content

Overview of Proline	2
Proline Zero	2
Architecture	3
HOW TO	3
Memory Panel	5
Error management	6
Mounting Points Panel	6
Error management	9
Parsing Rules Tab	10
Add A Parsing Rule	11
Edit Parsing Rules	12
Testing one parsing rule (local test)	13
Global test	14
Error handling:	14
Server Tab	15
Error management	16
Conclusion	16

Overview of Proline

Proline is a suite of software and components dedicated to mass spectrometry proteomics. Proline lets you extract data from raw files, import results from MS/MS identification engines, organize and store your data in a relational database, process and analyze this data to finally visualize and extract knowledge from MS based proteomics results.

More information on Proline Suite could be found on profiproteomics website: <u>http://www.profiproteomics.fr/proline</u>



Proline Zero

Proline Zero is an all-in-one solution running on a workstation or a laptop for a single user. This distribution integrates Proline Server (the server Core component, Cortex), Proline Admin (for system administration purpose) and Sequence Repository (to gather protein sequence from fasta files). It also integrates the GUI, Proline Studio.

This documentation describes specifically the functionalities of the GUI of Proline-Zero launcher. All concepts, features and documentation of Proline Suite are accessible on profiproteomics support page. It is also possible to download Proline Zero and it's documentation from the download page.

Note that Proline is initially designed to be executed on powerful servers therefore your PC should have at least 8GB of ram memory.



Architecture

HOW TO

When Proline Zero is launched, a window is displayed, allowing users to configure settings for all proline modules before launching them.

Allocated memory, server, mounting points, sequence repository parsing rules and other options could be customized. Each of these elements has its own panel. This GUI will also help solving errors before launching the application.

🗫 Proline zero config window	×
Modules Start Proline Server Start Sequence Repository Start Proline Studio	
Memory Folders Server Parsing rules panels	
Parsing Rules Specify parsing rules used by Sequence Repository to retrieve Protein accession from fasta entry. A complete description is given in Proline Admin documentation or on Forum "Parsing rules for Sequence Repository" topic	
Add parsing rule:	
Default protein accession rule: >\w{2}\\\w+\\(\w+)	
List of parsing rules Label: Uniprot Fasta name rule: COD (COD (COD (COD (COD (COD (COD (COD (AUP_; \AS_cerevisiae_D_\d

At the top of the panel you will find three checkboxes that allow you to decide which modules will be started:

Proline zero config window	×
Modules	
Start Proline Server	
Start Sequence Repository	
Start Proline Studio	
Memory Folders Server Parsing rules	

You can choose to start or not Proline Studio or Sequence Repository, Proline Server is mandatory. Depending on whether or not these options are selected, some features may not be available.

At the bottom of the panel you will find three buttons and a check box :

Do not show again		
🕗 Restore Settings	🖌 Ok	🗶 Cancel

Restore Settings allows you to reset the parameters as they were when the application was launched, reading back these parameters from configuration files.

will save the current configuration in configuration files and launch Proline Suite. A systematic check of possible errors is always done. Proline Suite won't start if there are "fatal" errors.

X Cancel will close the application without changing configuration.

Do not show again If this checkbox is selected, the configuration window will not be displayed at the next startup.

Memory Panel

This Panel allows you to configure memory allocation for all modules : Proline Studio client and server components: the datastore, the JMS server, Sequence Repository and Proline Server.

You can choose three different types of Allocation Mode :

- If **automatic mode** is selected, you only have to specify the total amount of memory that will be allocated for the Proline Suite. The system will automatically dispatch memory between components.
- In **Semi-automatic** mode, the amount of memory is defined for Proline Studio as well as for all server components.
- In **manual** mode, the amount of memory for each component should be specified by the user.

Memory	Folders	Server	Parsing rules	
	Mer Spe - In Ti - In ta - Fir	mory cify the d Automati he syster Semi-aut o dispatch ally, in m	esired memory c mode, only th n will define mer omatic mode, m ı between other anual mode, th	allocation mode and the amount of memory. e Total memory should be specified. nory for each component. emory amount for Proline Studio can be specified as well as Server memory [,] components. e amount of memory for each component should be specified.
Allocatio	n mode :	Autom	atic	×
Allocatio	on			
Total m	nemory :			6 🗢 Go
-Client Proline	e Studio :			1 Go
Server	r ———			
Serve	er memory	:		5 🌲 Go
		Sequence	e repository :	1 🗘 Go
			DataStore :	1 🗘 Go
		Pr	oline server :	2 🗘 Go
			JMS :	Go
			JMS :	1 🗘 Go

Total memory :

7,1 🔹	Go
	7,1 🛉

Total memory corresponds to the total memory allocated to Proline Suite (Client+Server). This value is either specified by the user in automatic mode or calculated in other cases. An optimal setting depends on the amount of ram on your pc.

Error management

If the memory settings you specified are not valid (for example if memory is too small or too large for a component) the GUI will display pop-ups such as this one :

Proline	Zero X
?	Proline Zero can't start with current errors : The specified total memory value is below minimum required (3072)
	Would you like to exit Proline Zero ? An example proline_launcher.config file is provided in proline_launcher.config.origin file Yes No

Mounting Points Panel

A first field allows you to specify the maximum size in MB of the temporary directory used by Proline for working files.

Maximum size for temp folder : 1000	Mo
-------------------------------------	----

Once this size is reached you will be asked if you want to clear it.

Result identification files (Mascot, OMSSA or X!Tandem) as well as mzDB files (for the XIC Quantitation process) are only browsed from the Proline Server side. These folders should be specified in order to permit the server to access these files. In the same way, fasta folders path should be configured to allow Sequence Repository to browse and parse fasta files.

The *Folder* panel allows you to visualize, modify and repair mounting points and fasta folders. Mounting Points consist in a label and a path for result and mzDB folders, Fasta folder contains only paths without label.

🎾 Proline zero co	fig window		×
Modules			
Start Proline Serv	er		
Start Sequence F	epository		
Start Proline Stud	io		
Memory Folders S	erver Parsing rules		
- Ealdar			
- identif	rolders, identified by a label, containing : cation result files (for mascot. omassa).		
- quanti	ation files (in mzdb format)		
- fasta	ies (used by sequence repository)		
You can	also specify a maximum size for temp folder, in order to be asked to dear it i	fit's size is abov	e specified one.
Maximum size for	emp folder : 1000		Mo
	1000		MO
Add Mounting Point			
Add Hodinary Point	T		
List of Folders			
-Result folder			
mascot_data:	C:\Users\CD272832\IdeaProjects\proline-zero\data\sample\data\mascot	Ø	A
wwwz:	C:\Users\CD272832\Desktop\captures d'ecran	1	Î
x:	C:\Users\CD272832\Desktop\PDF et TUTOS		â
Mzdb folder			
mzdb_files:	C:\Users\CD272832\Documents\securité	P	A
cc :	C: \Users \CD272832 \Desktop \GridBagLayoutArticle	Ø	î
-Fasta files folde	S		
Folder 1:	C:\Users\CD272832\Desktop\Typesafe Config_files	I	Î
Do not show agair			
🕘 Restore Settir	gs	🖌 Ok	💥 Cancel

To add a mounting point, simply click on the Add button

A dialog box will show up :

🛨 Add Mo	ounting Point	\times
Data type :	Result folder ~	
Label :		
Path :		D
	🖉 Clear 🕂 Add 🔀 Cano	cel

It is possible to add three different types of mounting points :

mzDB files, Mascot Files and Fastas Files.

At first select the type of the mount point you want to add in the menu bar :

Data type :	mzDB folder	~
You must t	nen enter the label and the path.	

Once you are done click on the add button : + Add

The validity and uniqueness of the path is systematically checked. The uniqueness of the label is also required. If the entry is valid the mounting point you entered is saved and displayed inside the list. If not, pop-ups will help you to solve the problem



The clear button 🖉 Clear erases the entries in the form.

All mounting points are displayed in the list :

mascot_data:	C: \Users \CD272832 \IdeaProjects \proline-zero \data \sample \data \mascot
label :	C:\Users\CD272832\Desktop\ProlineZeroArchitecture - Proline - BioProj _ SYNAPSE_files
label2 :	C:\Users\CD272832\Desktop\PDF et TUTOS
1zdb folder	
mzdb_files:	C:\Users\CD272832\IdeaProjects\proline-zero\data\sample\data\mzdb
label4 :	C:\Users\CD272832\Desktop\Typesafe Config_files
label3 :	C:\Users\CD272832\Desktop\GridBagLayoutArticle
asta files folder	······································
Folder 1:	C:\Users\CD272832\Desktop\DOCUserV3

It is possible to modify a mounting point by clicking on the Edit button to the right of the mount point.

By clicking on the edit button a dialog box opens :

Maximum s	🥖 Edit Mo	unting Point	×	Мо
List of For Result masco	Data type : Label : Path :	Mzdb folder label4 C:\Users\CD272832\Desktop\Typesafe Config files		
Mzdb fr		Clear Update	X Cancel	
	label4 : C:\U label3 : C:\U	sers\CD272832\Desktop\Typesafe Config_files sers\CD272832\Desktop\GridBagLayoutArticle		

After modifying the values, the user finalizes the edition by clicking on the "update" button

If the fields are correctly filled in, i.e. if the label and the path are not already used, the update is done and the dialog is closed.

Update can be canceled at anytime with cancel button

The mounting points can be deleted by clicking on the button 💼 .

The default mounting points (with a blue label) cannot be deleted hence there is no button

and a small icon is displayed instead <a> . Although these elements are not erasable It is possible to change their paths.

Error management

At startup Proline checks the validity of all mount point paths. If a path is not valid it will be displayed in red.

mzdb_files: C:\Users\CD272832\IdeaProjects\proline-zero\data\sample\data\mzdbx

A warning is also displayed before the window is visible:



It is possible (and **required**) to repair an invalid path with the edit button. Proline also checks for the presence of at least one mount point. In case of failure a warning will be displayed. A warning is also displayed when a default mount point is missing although it is not considered an error.

Parsing Rules Tab

This tab is dedicated to the visualization and management of parsing rules used by "Sequence Repository" module.

A parsing rule has four elements:

- 1. a label,
- 2. two regular expressions (Fasta Version Rule and Protein accession Rule)
- 3. a list of "fasta name rules" of variable size.

noquies						
Start Proline Ser	ver					
Start Sequence I	Repository	r				
Start Proline Stu	dio					
lemory Folders	Server Pa	arsing rules				
Parsi Spacif	ng kules	ulan unod by Conuonco i	Dopository to rotriovo			
Protei	n accessio	n from fasta entry.	Repository to retrieve			
A com	plete desc Forum "Pa	ription is given in Proline rsing rules for Sequence	Admin documentation Repository" topic			
0.011		ing face for bequeree	reportery topic			
Add pa	arsing rule					
ridd pr	a bing raic					
efault Protein Acce	ssion Rule	: >(\S+)				
List of parsing rules						
Label	Unipro	t	Fasta Name Rule:	VAUP_; VAS_cere	visiae_D_\d	
Easta Version Bulk	(2.0)	(2)Decevil) (*)) facto	Dratain Accession Dulay	Shuf9311[A1][*1	()(C +)	
Fasta version Rule	: [_(::)]	(?:Decoy))_(.*)\.fasta	Protein Accession Rule;	>\w{2}\\[^^\]_^\	(()>+)	
Label	Unipro	t_FirstAcc	Fasta Name Rule:	SP_Human_isoform	n	
Easta Version Rule	e isofo	rm (*)\ fasta	Protein Accession Rule:	>\w{2}\\([^\[]+)	M.	
				s weed the many	u	
Label	FirstIC)	Fasta Name Rule:	Acinetobacter_D;	; \APEntomop	hilaL48
Easta Version Bulk	. (200)	(2:Docov)) (*)) facta	Protein Accession Pula	>/[^/]*)///C+		
	(:.0]	(::::::::::::::::::::::::::::::::::::::	Protein Accession Rule.			
Label	Dollari	Rule	Fasta Name Rule:	\AMolliSibericum_	; PandoraNe	ocaledonia_D
Fasta Version Pule	. (2·DI	(2:Decov)) (*)\ fasta	Protein Accession Pule:	>([^¢]*)/c+		
	(:)	(::::::::::::::::::::::::::::::::::::::	Trotein Accession Adie.	×(1 4) 701		
Label	FullRu	e	Fasta Name Rule:	P53 D		
	(2.0)	(2.2.)) ((2)) ()		-		
Fasta Version Rule	: [_(?:D]	(?:Decoy))_(.*)\.fasta	Protein Accession Rule:	>(.*)		
						Test
Do not show again	n					

Add A Parsing Rule

To add a parsing rule click on the button **t**. Following dialog will be opened.

🛨 Add Parsing Rule		×
Label:		Add Fasta Name Rules:
Fasta Version Rule:		Fasta Name Rules:
Protein Accession Rule:		Rule
Test		
Fasta entry:	Test	
protein name extracted :		
	🖉 Clear	🕂 Add 🛛 💥 Cancel

Add Fasta Name Rules:				
Fasta4				
Fasta Name Rules:				
Rule				
Fasta 1	1			
Fasta2	1			
Fasta3	Û			

It is possible to enter as many fasta names as you want, they are displayed dynamically in a table :

After filling in the four fields, click on the button **+** Add . If the input is valid the parsing rule will be saved and displayed.

Visualization of the Parsing rules in a block with the four fields :

Label:	FirstID	Fasta Name Rule:	Acinetobacter_D; \APEntomophilaL48
Fasta Version Rule:	_(?:D (?:Decoy))_(.*)\.fasta	Protein Accession Rule:	>([^\]*)\ \S+



Edit Parsing Rules

Editing and deleting parsing rules is possible by clicking on the edit button

and the

delete button

The editing is done in a dialog box :

🥟 Edit Parsing Rule			×
Label:	FirstID		Add Fasta Name Rules:
Fasta Version Rule:	_(?:D (?:Decoy))_(.*)fasta		Fasta Name Rules:
Protein Accession Rule:	>([^\]*)\ \\$+		Rule
Test			Acinetobacter_D 🍿 🔺
Fasta entry:		Test	VAPEntomophila
protein name extracted :			ACAST_
		🖉 Clear	🛨 Update 🔀 Cancel

The fields are pre-filled with the initial values. It is possible to safely cancel the changes at any time with the cancel button

After clicking on update Proline checks if all fields are completed and if the label is not already allocated to another parsing rule. If everything is correct the parsing rule is updated and displayed in the GUI. If not, pop-ups will guide you to solve the problem.

Testing one parsing rule (local test)

It is common to make errors when using regular expressions, so Proline offers the possibility to test if they are correct.

The user can test the parsing rule in the dialog box. Just click on the test button. A new window is displayed and it is possible to submit a line of identification of a fasta file. If the regular expression is valid, the protein name is displayed. If the match is not correct, the user can modify the regular expression and run the test again.

🥖 Edit Parsing Rule	×
Eiro+TD	Add Fasta Name Rules:
Label:	±
Fasta Version Rule: _(?:D (?:Decoy))_(.*)\	fasta Fasta Name Rules:
Protein Accession Rule: >([^\]*)\ \S+	Rule
Test	Acinetobacter_D 🍿 🔨
Fasta entry: line tested	Test VAPEntomophila
protein name extracted : result of tes	t ChloroPlastid
	🖉 Clear 🕂 Update 💥 Cancel

Global test

To perform this test click on the button test at the bottom of the Parsing Rule Panel.

		P Test
Do not show again		
🕗 Restore Settings	🖌 Ok	💥 Cancel

This test is applied to all fasta files in the directories specified in the configuration file. The files are analyzed one by one.

For each file, Proline tries to find a parsing rule that contains a regular expression that matches with the name of the file.

The fasta file is parsed and three lines are selected then protein names are extracted using the regular expression.

The extracted protein names are displayed in a dialog box:

🕕 Glob	bal test	results			×
0	Globa This d match	al test alog displays the results of the test applied to all the fasta files contained in es, the default regular expression will be used.	side fasta directories. For each file	, Proline determines the appropriate parsing rule. If none	
Statistic	cs —				^
A total	of 65 f	asta files were scanned during the test			П
Default	: proteir	a accession rule: $w{2} w+ (w+)$ used for 70 % of files			
	3	parsing rule was found for 30% of fasta files			
	56	protein accession has been extracted in 56% of the files	parsed		
Fasta f	file:	Ribo-13Prot-mouse-SEQCE.fasta	Used parsing rule:	Uniprot Ribo	
Fasta en	ntry:	>sp P62082 RS7_MOUSE 40S ribosomal protein S7 OS=Mus mu	Protein accession extracted:	RS7_MOUSE	
Fasta en	ntry:	>sp P63323 RS12_MOUSE 40S ribosomal protein S12 OS=Mus	Protein accession extracted:	RS12_MOUSE	
Fasta en	ntry:	>sp Q9CQR2 RS21_MOUSE 40S ribosomal protein S21 OS=Mus	Protein accession extracted:	RS21_MOUSE	
Fasta f	file:	Ribo-13Prot-mouse-SEQCE.fasta	Used parsing rule:	Uniprot Ribo	
Fasta en	ntry:	>sp P62082 RS7_MOUSE 40S ribosomal protein S7 OS=Mus mu	Protein accession extracted:	RS7_MOUSE	
Fasta en	ntry:	>sp P63323 RS12_MOUSE 40S ribosomal protein S12 OS=Mus	Protein accession extracted:	RS12_MOUSE	
Fasta en	ntry:	>sp Q9CQR2 RS21_MOUSE 40S ribosomal protein S21 OS=Mus	Protein accession extracted:	RS21_MOUSE	~
				🖌 ОК	:

Some statistics are displayed.

Error handling:

If no parsing rule is recorded, a warning is displayed, so it is recommended to enter at least one parsing rule.

Server Tab

Server tab deals with server parameters, please consider that some settings are complex to manage and you should only change them if you are an advanced user.

F S V It	Folder ipecify datastore (Postgresql or H2) port to be used. VARNING : once datastore has been initialized this SHOULD NOT be changed. t may cause Proline Zero to be unstable. This may be done only in some special cases
DataStore po	ort: 4569 Test available 🖌
Advanced	settings

At first the current datastore port is displayed :

DataStore port :	4569	Test available	✓
Advanced settin	ngs		

Datastore represents the database used, it can be either PostgreSQL or H2. If the port is unavailable a cross × will

appear in place of the green label \checkmark .

By clicking on the advanced settings button a new dialog box opens :

🔊 Proline zero	advanced config wind	wol		\times
Fold	er			
Spec may Othe whid Note may	ify different ports to be run at the same time), r server specific setting h specify the number of : This parameter depend stop responding,	used by JMS s are also ava process runn ds on comput	Server (this is usefu ilable such as "Serve ing at the same time er configuration, if to	ll when 2 ProlineZero er Thread pool size" on server side. so high the computer
∟Ports choice —				
	JMS Server Port :	5445	Test available	⊻
<u>AC</u>	1S Batch Server Port :	5455	Test available	⊻
	<u>JNP Server Port :</u>	1099	Test available	⊻
:	JNP RMI Server Port :	1098	Test available	⊻
Server d	lefault timeout (s) :	120		
Serve	r thread pool size :	1		
	JVM path :	C: \Christop	ohe \ogiciels \jdk-17	
Force Data	Store update			
			🖌 Ok	X Cancel

In that tab you can visualize the ports to be used by JMS Server (this can be useful when two ProlineZero run at the same time).

A test is possible for each configurable port, simply click on the test available button, this will tell you if the port is already in use or available.

JMS Server Port :	5445	Test available	¥

The green label \checkmark on the right indicates that the JMS server is available (here on port 5445). If not a cross \Join will appear.

Server default timeout :

Server default timeout (s) :	120
------------------------------	-----

This is the time in seconds before considering a component is unavailable.

Server thread pool size :

Server thread pool size :	1	
---------------------------	---	--

Server thread pool size specifies the number of processes running at the same time on the server side.

It is also possible to specify the path for the JVM on your pc. The JVM is the java virtual machine used to run Proline. If you need to change that parameter , simply click on the folder

icon 🔑 and choose the path of the JVM on your machine.



Finally a checkbox allows you to force the update of the database.

Force DataStore update

Error management

Proline will check if the JVM path is well configured, then it will verify that ports are not used by two servers at the same time. Then it will test the availability of the ports for the database JMS, JMS batch, JNP and JNP RMI.

Conclusion

You have read the entire document, congratulations!!! By now you know how to configure

Proline using the GUI. You can click on the Ok button will check if there are errors. If this is not the case you have finished configuration otherwise you still have to fix the fatal errors. You can launch Proline if configuration contains only minor errors.

From now on you can refer to this documentation for further learning: Proline User Doc