MassXpert3 is part of the MsXpertSuite software package
Modelling, simulating and analyzing ionized flying species

# MassXpert3 User Manual

## Modelling and simulating mass spectrometric data of linear polymers

---

## MassXpert3 10.1.2

# MassXpert3 User Manual : Modelling and simulating mass spectrometric data of linear polymers

by Filippo Rusconi

November 15, 2025 , 10.1.2

MsXpertSuite - mass spectrometry software suite

http://www.msxpertsuite.org/

This book is part of the MsXpertSuite project.

The MsXpertSuite project is the successor of the MassXpert project. This project now includes various independent modules:

- MassXpert, program to model polymer chemistries and simulate mass spectrometric data;
- MineXpert, program to visualize and mine mass spectral data (mass spectrum, drift spectrum, XIC chromatograms) starting from the TIC chromatogram.

10.1.2

Revision History

Revision 10.1.2 November 2025 fr

Full port of MassXpert to the new XpertMass project's libraries. Add new chapter on JavaScript features now in MassXpert.

Revision 7.0.0 November 2022 fr

The manual has not any content changed. Only this revision corresponds to a software bump matching a full code port to Qt6.

Revision 6.1.0 October 2022 fr

Entirely reworked the user manual to make it more modern from the coding point of view. The users do not see any difference with the previous version.

Revision 6.0.3 20210202 fr

Removed all mml elements (MathML) that made rendering the document too difficult. Replaced the formulae with figures. Adapted the text;

Revision 5.7.0 20190115 fr

Finished porting of the historical LaTeX-based documentation to the DocBook/DAPS/FOP publishing system. Please, see Colophon for details;

Revision not_set 20150918 fr

Update the manual to reflect the changes in the way ion product masses are computed in massXpert and update the various examples. Now, the fragmentation specification must produce fragments that are uncharged. The software will then ionize the/gc fragments using the ionization rule currently in use in the XpertEdit module;

Revision not_set 20121205 fr

Update the manual to document the various new ways to import data in the m/z list in the XpertMiner module;

Revision not_set 20111022 fr

Update the manual to describe the new fragmentation of oligomer containing cross-linked monomers feature;

Revision not_set 20110829 fr

Update the manual to describe a number of new features;

Revision not_set 20100429 fr

Updated the section about the definition of monomers to document the new feature about computing the mass difference between any two monomers in the definition;

Revision not_set 20100427 fr

Updated the section about the cleavages of polymer sequences (XpertEdit chapter) to document the new feature that allows to only perform a cleavage in the currently selected region of a polymer sequence;

Revision not_set 20090619 fr

Updated the section about the chemical pad (XpertCalc chapter) to reflect improvements in the use of the chemical pad buttons (either immediate evaluation of the formula or mere insertion of the formula in the formula line edit widget);

Revision not_set 20090617 fr

Updated the section about the chemical pad (XpertCalc chapter) to reflect improvements in the graphical display of the chemical pad buttons as programmed in the chem_pad.conf configuration files;

Revision not_set 20090401 fr

Although not visible in the documentation, I wanted to publically extend my warm thanks to Lionel Élie Mamane, who helped me along these last months with the Debian packaging of massXpert. Note that this work also proved useful for other areas in the project.

Revision not_set 20090220 fr

Updated the XpertEdit chapter to show how to configure the options about the number of decimals to be used for display of numerals in the program;

Revision not_set 20090205 fr

Updated the XpertEdit chapter to show the simplified polymer sequence editing feature whereby editing of the sequence might be performed by clicking on monomer items in the list of all the monomers defined in the polymer chemistry definition. Fixed small bug in the documentation of about multi-region selection behaving as oligomers or residual chains.

Revision not_set 20081211 fr

Updated the XpertEdit chapter to show the feature by which it is now possible to force the calculation engine to take into account the left/right end modification(s) when calculating the masses of a sequence region that does not encompass the left/right end of the polymer sequence. This new feature was essential in trying to perform full simulations of the molecular heterogeneity of the telokin protein (Rusconi et al. 1997 Biochemistry). Added a paragraph about max count of chemical modifications of a given monomer at once in the XpertDef chapter and another one in the XpertEdit chapter to explain its working;

Revision not_set 20080911 fr

Finally indexed the whole document. Performed some minor modifications so that the documentation system does not produce HTML files anymore (the HTML production was not really worth it anyways);

Revision not_set 20080805 fr

Updated the user manual with a bunch of updated screen shots;

Revision not_set 20080730 fr

Updated the XpertMiner chapter to illustrate the m/z--z mass list matching feature;

Revision not_set 20080708 fr

Updated the XpertEdit chapter to illustrate the new multi-cleavage feature;

Revision not_set 20080701 fr

Updated the XpertEdit chapter to illustrate the new multi-region and multi-selection features;

Revision not_set 20080529 fr

Changed the install instructions for the Mac OS X system;

Revision not_set 20080527 fr

Added a chapter about XpertMiner to document the new features in that module. Some fixes here and there.

Revision not_set 20080526 fr

Modified the documentation to reflect switch to version 3 of the GNU General Public License;

Revision not_set 20080425 fr

The installation chapter was updated to illustrate the installation of the software in the Mac OS X system;

Revision not_set 20080424 fr

The installation chapter was updated to reflect the changes in the way the package might be installed (the package is now relocatable, provided the user indicates where the directories are located);

Revision not_set 20080402 fr

The XpertDef chapter was updated to detail the new way of defining fragmentation specifications where the side chain is decomposed in the gas-phase. The section about fragmentations is now much better documented;

Revision not_set 20080330 fr

The XpertEdit chapter was updated to include a description of the new fragmentation/mass searching data in-place filtering. A section is now devoted to data filtering;

Revision not_set 20080325 fr

The XpertEdit chapter was updated to include a description of the new sequence cleavage data in-place filtering;

Revision not_set 20080318 fr

The XpertEdit chapter was updated to include a description of the find sequence motif feature;

Revision not_set 20080313 fr

The XpertEdit chapter was updated to include a new paragraph about monomer cross-linking as this is now implemented in the software;

Revision not_set 20080221 fr

The XpertEdit chapter was updated to include a new figure of the polymer modification procedure and to describe the enhanced modification procedure;

Revision not_set 20080215 fr

The XpertEdit chapter was updated to include a new figure of the monomer modification procedure and to describe the enhanced modification procedure;

Revision not_set 20071217 fr

The chapter about installation of massXpert has been rewritten to reflect changes in the building of massXpert and in the installation of Debian GNU/Linux and Fedora core GNU/Linux packages;

Revision not_set 20071216 fr

The chapter about polymer chemistry definitions has been refactored to reflect the rewriting of the corresponding code. Added a small section about m/z ratio calculation, that was missing, although the feature was added a long time ago;

Revision not_set 20070922 fr

The new multi-charged cleavage and fragmentation oligomers have been documented;

Revision not_set 20070819 fr

Switched back to version 2 of the GPL in the Appendices chapters, as massXpert cannot be licensed otherwise: the Qt libraries are licensed using version 2 of the GPL without the ``or any later version, at your option'' wording;

Revision not_set 20070728 fr

Updated the XpertDef chapter (modifications) to show the new ``targets'' feature. Updated the XpertEdit chapter to show the new monomer modification dialog.

Revision not_set 20070719 fr

Added explanation on the arbitrary formula-based polymer sequence ends modification.

| | | |
|---|---|---|
| Revision not_set | 20070713 | fr |

Revision of the whole document for a better printed output;

| | | |
|---|---|---|
| Revision not_set | 20070710 | fr |

Added a section to the XpertEdit chapter about the data mining mass list lab feature added recently. Mentioned the installation of Debian packages;

| | | |
|---|---|---|
| Revision not_set | 20070630 | fr |

Switched the project and all accompanying documentation and other data files to GPL version 3;

| | | |
|---|---|---|
| Revision not_set | 2007 mid-june | fr |

Start of the writing by taking inspiration of the GNU polyxmass manual.

# DEDICATION

To María Cecilia

To all the admirable people acting in the *“Free Software Movement”* for a better and more ethical computing world

To all involved in the development of the K Desktop Environment (KDE)

To all the readers who helped me with this manual.

# CONTENTS

PREFACE v

1 GENERALITIES 1

1.1 ON CHEMICAL FORMULÆ AND CHEMICAL REACTIONS 1

1.2 THE MASSXPERT FRAMEWORK DATA FORMAT 2

1.3 CHEMICAL ENTITY NAMING POLICY 2

2 BASICS IN POLYMER CHEMISTRY 4

2.1 POLYMERS? WHERE? EVERYWHERE! 4

2.2 VARIOUS BIOPOLYMER STRUCTURES 5

PROTEINS 5 • NUCLEIC ACIDS 7 • SACCHARIDES 9

2.3 TO SUM UP 11

2.4 POLYMER CHAIN DISRUPTING CHEMISTRY 12

POLYMER CLEAVAGE 12 • POLYMER FRAGMENTATION 14

3 XPERTDEF: DEFINITION OF POLYMER CHEMISTRIES 23

3.1 THE ISOTOPES 24

3.2 THE POLYMER CHEMICAL ENTITIES 25

THE MONOMERS 26 • THE MODIFICATIONS 28 • THE CROSS-LINKERS 29 • THE CLEAVAGE SPECIFICATIONS 31 • THE FRAGMENTATION SPECIFICATIONS 34

3.3 SAVING THE DEFINITION 39

4 XPERTEDIT: A POWERFUL EDITOR AND SIMULATION CENTER 40

4.1 XPERTEDIT INVOCATION 40

4.2 XPERTEDIT OPERATION: IN MEDIAS RES 40

4.3 THE EDITOR WINDOW MENU 44

4.4 Editing Polymer Sequences 46

Multi-Character Monomer Codes 47 • Unambiguous Single-/Multi-Character Monomer Codes 48 • Erroneous Monomer Codes 48 • Simplified Editing 49

4.5 Finding sequence motifs 50

4.6 Importing Sequences 50

Importing From The Clipboard 50 • Importing From Raw Text Files 52

4.7 Multi-region Selections 53

4.8 Polymer Sequence Modification 54

Selected Monomer(s) Modification 54 • Whole Sequence Modification 57

4.9 Monomer Cross-linking 58

4.10 Sequence Cleavage 60

4.11 Oligomer Fragmentation 63

4.12 Mass Searching 68

4.13 Oligomer Data Filtering 70

4.14 m/z Ratio Calculations 72

4.15 Monomeric And Elemental Composition 73

4.16 pKa, pH, pI and Charges 74

Ionized Group(s) In Monomers 75 • Ionized Group(s) In Modifications 80 • pH, pI and Charge Calculations 81

4.17 General Options 82

5 XpertCalc: a Powerful Mass Calculator 83

5.1 XpertCalc Invocation 83

5.2 An Easy Operation 83

5.3 The Programmable Calculator 85

5.4 The Log Book Recorder 87

5.5 The m/z Ratio Calculator 88

5.6 THE ISOTOPIC PEAKS CALCULATOR 88

6 XPERTMINER: A DATA MINER 89

6.1 XPERTMINER INVOCATION 89

6.2 MZLAB: MINING M/Z RATIOS 89

6.3 CREATING A NEW INPUT M/Z LIST 90

FILLING M/Z LISTS WITH DATA 91 • IMPOSING THE MASS TYPE: MONO OR AVG 95

6.4 WORKING ON ONE INPUT M/Z LIST 95

AVAILABLE CALCULATIONS 96 • OUTPUT OF THE CALCULATIONS 96 • INTERNAL WORKINGS 97

6.5 WORKING ON TWO INPUT M/Z LISTS 97

OUTPUT OF THE CALCULATIONS 97 • TRACING THE DATA 98

7 XPERTSCRIPT: A SCRIPTING CONSOLE 100

7.1 XPERTSCRIPT INVOCATION 100

7.2 RUNNING SCRIPTS IN THE SCRIPTING CONSOLE 101

7.3 LOADING SCRIPT FILES IN THE SCRIPTING CONSOLE 101

7.4 ONE ELABORATE EXAMPLE OF SCRIPT 101

8 UTILITIES 106

8.1 CALCULATING ISOTOPIC CLUSTERS 106

THE ISOSPEC GRAPHICAL USER INTERFACE IN MINEXPERT 108 • SHAPING MASS PEAK CENTROIDS INTO WELL-PROFILED PEAKS 119

8.2 CONFIGURATION OF THE MASSXPERT AND MINEXPERT INTERCOMMUNICATIONS 123

DYNAMIC CONFIGURATION OF THE MASSXPERT AND MINEXPERT INTERCOMMUNICATIONS 124

9 DATA CUSTOMIZATIONS 126

A GNU GENERAL PUBLIC LICENSE VERSION 3 133

# PREFACE

## 1 SOFTWARE FEATURE OFFERINGS AND INTENDED AUDIENCE

This manual is about the MsXpertSuite mass spectrometric software suite, a software environment that contains two modules:

- *MassXpert* module (currently *MassXpert3*): Allows users to define brand new polymer chemistries and use these polymer chemistry definitions to model linear polymer sequence. Once modelled, a polymer sequence can undergo chemical reactions (enzymatic or chemical cleavages, gas-phase fragmentations...). The obtained results are a model of what a mass spectrum would look like if the modelled experiment had actually been carried over up to the mass spectrometry analysis;

- *MineXpert* module (currently *MineXpert3*): Allows users to load mass spectrometry data from mzML files, visualize and mine the data throughout all their depth. The mass data visualization starts at the TIC chromatogram level and deepens to the mass spectra, the drift spectra, the XIC chromatograms.

As such, this manual is intended for people willing to learn how to use the comprehensive MsXpertSuite software package.

Mass spectrometry has gained popularity across the past twenty years or so. Indeed, developments in polymer mass spectrometry have made this technique appropriate to accurately measure masses of polymers as heavy as many hundreds of kDa, and of any chemical type.

There are a number of utilities—sold by mass spectrometer constructors with their machines, usually as a marketing “plus”—that allow predicting/analyzing mass spectrometric data obtained on polymers. These programs are usually different from a constructor to another. Also, there are as many mass spectrometric data prediction/analysis computer programs as there are different polymer types. You will get a program for oligonucleotides, another one for proteins, maybe there is one program for saccharides, and so on. Thus, the biochemist/massist, for example, who happens to work on different biopolymer types will have to learn to use several different software packages. Also, if the software user does not own a mass spectrometer, chances are he will need to buy all these software packages.

The MsXpertSuite mass spectrometric software is designed to provide *free* solutions to all these problems by providing the following features:

- MassXpert:
    - Model *ex nihilo* polymer chemistry definitions (in the XpertDef module that is part of the MassXpert program);
    - Peform simple yet powerful mass computations to be made in a mass desktop calculator that is both polymer chemistry definition-aware and fully programmable (that's the XpertCalc module also part of the MassXpert program);
    - Edit polymer sequences on a polymer chemistry definition-specific basis, along with chemical reaction simulations, finely configured mass spectrometric computations... (all taking place in the XpertEdit module that is the main module of the MassXpert program);
    - Customize the way each monomer will show up graphically during the program operation (in the XpertEdit module);
    - Edit polymer sequences with immediate visualization of the mass changes elicited by the editing activity (in the XpertEdit module);
    - Open an unlimited number of polymer sequences at any given time and of any given polymer chemistry definition type (in the XpertEdit module);
    - Use a scripting console to automate a large number of tasks, including the generation of complex synthetic mass spectra of biopolymer intact molecules or of sets of biopolymer (bio)chemical lysate fragments (like peptides, for example) (in the XpertScript module).
- MineXpert:
    - Load mass spectometry data files in the mzML format, thanks to the excellent libpwiz library of ProteoWizard [1] fame;
    - Display the data in powerful ways in a unified graphical user interface. The interface was designed to integrate all the most useful characteristics of the various proprietary environments known by the author, thanks to the excellent libqcustomplot [2] library;
    - Configure the way mass spectrometry data integrations are performed and optionally configure and apply a Savitzky-Golay smoothing;

---

[1] http://proteowizard.sourceforge.net/ .

[2] http://qcustomplot.com/ .

- Perform data mining by performing data integrations in various ways;
- Ion mobility mass spectrometry data are supported with an automatic m/z = f(dt) color map plot calculation;
- A specific data integration mode allows easy quantitation of spectral data at any level (TIC chromatogram, mass spectrum, drift spectrum);
- Innovative data analysis recording allows to store the features mined during the data mining sessions in flexible ways that allow further data processing, like injection in databases;
- A JavaScript scripting environment allows taking the control of the software and of all the widgets from a script file;
- Convert data from mzML to the private (albeit open) database file format that allows to load data much faster. MineXpert can slice big data files into smaller chunks retaining all the data selected by the user in the most flexible ways.
- Use a scripting console to automate a large number of tasks.

## 2 FEEDBACK FROM THE USERS

We are always grateful to any constructive feedback from the users.

The MsXpertSuite software team might be contacted *via* the following addresses

```
msxpertsuite@msxpertsuite.org  = general mailing list about msXpertSuite
bug-reports@msxpertsuite.org   = report bugs found in msXpertSuite software
```

FIGURE 1: ADDRESSES TO REPORT FEEDBACK TO

## 3 PROJECT HISTORY

This is a brief history of MsXpertSuite.

- *1998–2000*

  The name MassXpert comes from a project I started while I was a post-doctoral fellow at the École Polytechnique (Institut Européen de Chimie et Biologie, Université Bordeaux 1, Pessac, France);

  The MassXpert program was published in *Bioinformatics* ( RUSCONI, F. AND BELGHAZI, M. DESKTOP PREDICTION/ANALYSIS OF MASS SPECTROMETRIC DATA IN PROTEOMIC PROJECTS BY USING MASSXPERT . BIOINFORMATICS, 2002, 644–655 (HTTPS://ACADEMIC.OUP.COM/BIOINFORMATICS/ARTICLE/18/4/644/243311) ↗ ).

At that time, MS-Windows was at the Windows NT 4.0 version and the next big release was going to be *“you'll see what you'll see ”* : MS-Windows 2000.

When I tried MassXpert on that new version (one colleague had it with a new machine), I discovered that my software would not run normally (the editor was broken). The Microsoft technical staff would advise to *“buy a new version of the compiler environment and rebuild ”* . This was a no-go: I did not want to continue paying for using something I had already produced with legitimate software.

- *2001–2006*

  During fall 1999, I decided that I would stop using Microsoft products for my development. At the beginning of 2000 I started as a CNRS research staff in a new laboratory and decided to start fresh: I switched to GNU/Linux (I never looked back). After some months of learning, I felt mature to start a new development project that would eventually become an official GNU package: GNU polyxmass .

  The GNU polyxmass software, much more powerful than what the initial MassXpert software used to be, was published in *BMC Bioinformatics* in 2006 ( Rusconi, F. GNU polyxmass: a software framework for mass spectrometric simulations of linear (bio-)polymeric analytes . BMC Bioinformatics, 225– (https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-7-226) ↗ ).

  Following that publication I got a lot of feedback (very positive, in a way) along the lines: — *“Hey, your software looks very interesting; only it's a pity we cannot use it because it runs on GNU/Linux, and we only use MS-Windows and MacOSX! ”* .

- *2007–2016*

  In december 2006, I decided to make a full rewrite of GNU polyxmass. The software of which you are reading the user manual is the result of that rewrite. I decided to “recycle” the MassXpert name because this software is written in C++, as was the first MassXpert software. Also, because the first MS-Windows-based MassXpert project is not developped anymore, taking that name was kind of a “revival” which I enjoyed. However, the toolkit I used this time is not the Microsoft Foundation Classes (first MassXpert version) but the Trolltech Qt framework (see the About Qt menu in the *Help* menu in MassXpert).

  Coding with the Qt libraries has one big advantage: it allows the developer to code once and to compile on the three main platforms available today: GNU/Linux, MacOSX, MS-Windows. Another advantage is that the Qt libraries are wonderful software, technically and philosophically (Free Software).

  The rewritten software was published in 2009 ( Rusconi, F. massXpert 2: a cross-platform software environment for polymer chemistry modelling and simulation/analysis of mass spectrometric data .Bioinformatics, 2009, 2741–2742 (https://academic.oup.com/bioinformatics/article/25/20/2741/194220) ↗ ).

- *2016–2022*

  In 2016, I started a new project about visualization of mass spectrometric data. The project developed pretty quickly, as we needed at the mass spectrometry facility a software that would allow to cope efficiently with ion mobility mass spectrometric experimental data. MineXpert was thus started.

To bundle both MassXpert and MineXpert in a single software suite, I bought the MsXpertSuite website HTTP://MSXPERTSUITE.ORG ↗ and created that new name.

- *2023–*

  The fundamental classes used in both MassXpert and MineXpert were structured in two private libraries: `libmass` and `libmassgui`. In 2023, I still used two copies of these libraries source code trees in each one of the two projects: one copy in the MassXpert source code tree and one copy in the MineXpert source code tree, although I was actually managing a single copy, making synchronizations of the private libraries in every direction... I happened to get fed up with this way of working and I finally decided to extract these copies ouf the source code trees of MassXpert and MineXpert and to create a new XpertMass project containing two shared object libraries (dll files in MS-Windows):

  - *libXpertMassCore*;
  - *libXpertMassGui*;

  I also decided that I would stop bundling both MassXpert and MineXpert into a single package, now letting the users choose what package theyt actually want to download and install. See the updated MsXpertSuite website HTTP://MSXPERTSUITE.ORG ↗

## 4 PROGRAM AND DOCUMENTATION AVAILABILITY AND LICENSE

The programs and all the documentation that are shipped along with the MsXpertSuite software suite are available at HTTP://WWW.MSXPERTSUITE.ORG ↗ . Most of the time, a new version is published as source, as binary install packages for MacOSX and MS-Windows. No GNU/Linux are created outside of the autbuilder of the various distributions. As a Debian Developer, Filippo Rusconi creates Debian [3] packages that are uploaded on the distribution servers. These packages are available using the system's software management infrastructure (like apt , for example).

The software and all the documentation are all provided under the Free Software license *GNU General Public License, Version 3, or later, at your option* . For an in-depth study of the *Free Software* philosphy I kindly urge the reader to visit HTTP://WWW.GNU.ORG/PHILOSOPHY ↗ .

---

[3] HTTP://WWW.DEBIAN.ORG/ ↗

# 1 Generalities

In this chapter, I wish to introduce some general concepts around the MassXpert program and the way data elements are named in this manual and in the program.

The MassXpert mass spectrometry software suite has been designed to be able to “work” with every linear polymer. Well, in a certain way this is true… A more faithful account of the MassXpert's capabilities would be: “*The MassXpert software suite works with whatever polymer chemistry the user cares to define; the more accurate the polymer chemistry definition, the more accurate MassXpert will be*”.

For the program to be able to cope with a variety of possibly very different polymers, it had to be written using some *abstraction layer* in between the mass calculations engine and the mere description of the polymer sequence. This abstraction layer is implemented with the help of “polymer chemistry definitions”, which are files describing precisely how a given polymer type should behave in the program and what its constitutive entities are. The way polymer chemistry definitions are detailed by the user is the subject of a chapter of this book (see menu *XpertDef* of the program). However, in order to give a quick overview, here is a simple situation: a user is working on two polymer sequences, one of chemistry type “protein” and another one of chemistry type “DNA”. The protein sequence reads “ATGC”, and the DNA sequence reads “CGTA”. Now imagine that the user wants to compute the mass of these sequences. How will MassXpert know what formula (hence mass) each monomer code corresponds to? There must be a way to inform MassXpert that one of the sequences is a protein while the other is a DNA oligonucleotide: this is done upon creation of a polymer sequence; the programs asks of what chemistry type the sequence to be created is. Once this “chemical parentage” has been defined for each sequence, MassXpert will know how to handle both the graphical display of each sequence and the calculations for each sequence.

## 1.1 On Chemical Formulæ and Chemical Reactions

Any user of MassXpert will inevitably have to perform two kinds of chemical simulations:

- Define the formula of some chemical entity;
- Define a given chemical reaction, like a protein monomer modification, for example.

While the definition of a formula poses no special difficulty, the definition of a chemical reaction is less trivial, as detailed in the following example. The lysyl residue has the following formula: $C_6H_{12}N_2O$. If that lysyl residue gets acetylated, the acetylation reaction will read this way:—“An acetic acid molecule will condense onto the ε amine of the lysyl side chain”. This can also read:—“An acetyl group enters the lysyl side chain while a hydrogen atom leaves the lysyl side chain; water is lost in the process”. The representation of that reaction is:

$$\text{R-NH}_2 + \text{CH}_3\text{COOH} \leftrightharpoons \text{R-NH-CO-CH}_3 + \text{H}_2\text{O}$$

When the user wants to define that chemical reaction, she can use that representation: "-$H_2O$+$CH_3COOH$", or even the more brief but chemically equivalent one: "-H+$CH_3CO$". In MassXpert, the chemical reaction representation is considered a valid formula.

## 1.2 The MassXpert Framework Data Format

All the data dealt with in MassXpert are stored on disk as XML-formatted files. XML is the *eXtensible Markup Language*. This "language" allows to describe the structure of a document. The structure of the data is first described in a section of the document that is called the *Document Type Definition, DTD*, and the data follow in the same file. One of the big advantages of using such XML format in MassXpert is that it is a text format, and not a binary one. This means that any data in the MassXpert package is human-readable (even if the XML syntax makes it a bit difficult to read data, it is actually possible). Try to read one of the polymer chemistry definition XML files that are shipped with this software package, and you'll see that these files are pure text files (the same applies for the `*.mxp` XML polymer sequence files). The advantages of using text file formats, with respect to binary file formats are:

- The data in the files are readable even without the program that created them. Data extraction is possible, even if it costs work;
- Whenever a text document gets corrupted, it remains possible to extract some valid data bits from its uncorrupted parts. With a binary format, data are chained from bit to bit; loosing one bit lead to automatic corruption of all the remaining bits in the file;
- Text data files are searchable with standard console tools (`sed`, `grep`...), which make it possible to search easily text patterns in any text file or thousands of these files in one single command line. This is not possible with binary format, simply because reading them require the program that knows how to decode the data and the powerful console-based tools would prove useless.

## 1.3 Chemical Entity Naming Policy

Unless otherwise specified, the user is *strongly* advised *not* to insert any non-alphanumeric-non-ASCII characters (space, %, #, $...) in the strings that identify polymer chemistry definition entities. This means that, for example, users must refrain from using non-alphanumeric-non-ASCII characters for the atom names and symbols, the names, the codes or the formulæ of the monomers or of the modifications, or of the cleavage specifications, or of the fragmentation specifications… Usually, the accepted delimiting characters are - and _. It is important not to cripple these polymer data for two main reasons:

- So that the program performs smoothly (some file-parsing processes rely on specific characters (like # or %, for example) to isolate sub-strings from larger strings);
- So that the results can be easily and clearly displayed when time comes to print all the data.

# 2 BASICS IN POLYMER CHEMISTRY

This chapter will introduce the basics of polymer chemistry. The way this topic is going to be covered is admittedly biased towards mass spectrometry and biological polymers. Moreover, the aim of this chapter is to provide the reader with the specialized words that will later be used to describe and explain the (inner) workings of the MassXpert program. This manual is not a ``crash course'' in biochemistry.

## 2.1 POLYMERS? WHERE? EVERYWHERE!

Indeed, polymers are everywhere. If you ask somebody to show you something polymeric, he/she will point you at the first plastic object in the vicinity. Right, plastic materials are made of hydrocarbon polymers. We also have many different polymers in our body. Proteins are polymers, complex sugars are polymers, DNA (the so-called "molecule of heredity" is a *huge* polymer. There are polymers in wine, in wood... Where? Everywhere!

The *Oxford Advanced Learner's Dictionary of Current English* gives for *polymer* the following definition: —*"natural or artificial compound made up of large molecules which are themselves made from combinations of small simple molecules"*.

A polymer is indeed made by covalently linking small simple molecules together. These small simple molecules are called *monomer*s, and it is immediate that a *polymer* is made of a number of monomers. A general term to describe the process that leads to the formation of a polymer is *polymerization*. It should be noted that there are many ways to polymerize monomers together. For example, a polymer might be either linear or branched. A polymer is linear if the monomers that are polymerized can be joined at most two times. The first junction links the monomer to an elongating polymer (thus making it the new end of the elongating polymer which, by the way, is longer than before by one unit) and the second junction links the new elongating polymer's end to another monomer. This process goes on until the reaction is stopped, the point at which the polymer reaches its *finished state*. A branched polymer is a polymer in which at least one monomer is able to contract more than two bonds. It is thus clear that a single monomer linked three times to other monomers will yield a "T-structure", which is nothing but a branched structure.

In the following sections we'll describe a number of different kinds of polymers. Each time, they will be described by initially detailing the structure of their constitutive monomers; next the formation of the polymer is described. At each step we shall try to set forth each polymer characteristics in such a manner as to introduce the way MassXpert "thinks polymers" and to introduce specialized terminologies. Once the basic chemistries (of the different polymers) have all been described, we will enter a more complex subject that is of enormous importance to the mass spectrometry specialist: polymer chain disrupting chemistry. We shall see that this terminology actually involves two kinds of chemistries: cleavage, on the one hand, and fragmentation, on the other hand.

While MassXpert is basically oriented to linear single-stranded polymer chemistries, it can also be used to simulate highly complex polymer chemistries. Biological polymers are the main focus of this manual, however all the concepts described here may be applied with no modification to synthetic polymer chemistries.

## 2.2 VARIOUS BIOPOLYMER STRUCTURES

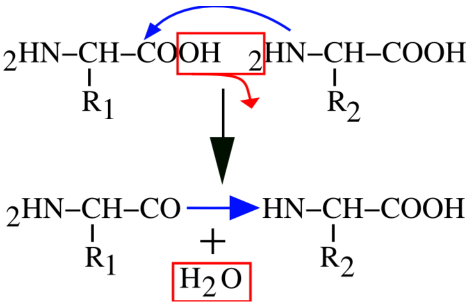
Biopolymers are amongst the most sophisticated and complex polymers on earth and it certainly is not a mistake to take them as examples of how monomers (be these complex or not) can assemble covalently into life-enabling polymers. In this section we will visit three different polymers encountered in the living world: proteins, nucleic acids and polysaccharides. We shall be concerned with 1) the monomers' structure, 2) the polymerization reaction and 3) the final end-capping reaction responsible for putting the polymer in its finished state.

### 2.2.1 PROTEINS

These biopolymers are made of amino acids. There are twenty major amino acids in nature, and each protein is made of a number of these amino acids. The combinations are infinite, providing enormous diversity of proteins to the living world.

A protein is a polar polymer: it has a left end and a right end, and polymerization actually occurs from left to right (from N-terminus to C-terminus, see below). FIGURE 2.1, “ PEPTIDIC BOND FORMATION BY CONDENSATION. ” shows that the chemical reaction at the basis of protein synthesis is a *condensation*. A protein is the result of the condensation of amino acids with each other in an orderly polar fashion. A protein has a left end, called *N-terminus; amino terminal end* and a right end, called *C-terminus; carboxyl terminal end*. The left end is an amino group ($_2$HN–) corresponding to the non-reacted amino group of the amino acid. Upon condensation of a new amino acid onto the first one, the carboxyl group of the first amino acid reacts with the amino group of the second amino acid. A water molecule is released, and the formation of an amide bond between the two amino acids yields a dipeptide. The right end of the dipeptide is a carboxyl group (–COOH) corresponding to the un-reacted carboxyl group of the last amino acid to have “polymerized in”.

The bond formed by condensation of two amino acids is an amide bond, also called—in protein chemistry—a *peptidic bond*. The elongation of the protein is a simple repetition of the condensation reaction shown in FIGURE 2.1, “ PEPTIDIC BOND FORMATION BY CONDENSATION. ” , granted that the elongation *always* proceeds in the described direction (a new monomer arrives to the right end of the elongating polymer, and elongation is done from left to right).

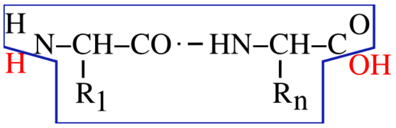


The left end monomer $R_1$ is condensed to the right end monomer $R_2$ to yield a peptidic bond. A water molecule is lost during the process.

**FIGURE 2.1:** **PEPTIDIC BOND FORMATION BY CONDENSATION.**

Now we should point at a protein chemistry-specific terminology issue: we have seen that a protein is a polymer made of a number of monomers, called amino acids. In protein chemistry, there is a subtlety: once a monomer is polymerized into a protein it is no more called a monomer, it is called a *residue*. We may say that a residue is an amino acid less a water molecule.

From what we have seen until now, we may define a protein this way: —*"A protein is a chain of residues linked together in an orderly polar fashion, with the residues being numbered starting from 1 and ending at n, from the first residue on the left end to the last one on the right end"*. This definition is still partly inexact, however. Indeed, from what is shown in FIGURE 2.2, " END CAPPING CHEMISTRY OF THE PROTEIN POLYMER. ", there is still a problem with the extremities of the residual chain: what about the amino group on the left end of a protein (the amino group sits right onto the first amino acid of the protein), and what about the carboxyl group of the right end of a protein (the carboxyl group sits right onto the last amino acid of the protein)? Because these groups lie at the extremities of the residual chain, they remained unreacted during the polymerization process. But because we are simulating a residual chain using residues and not amino-acids, we still need to put the residual chain in its finished state: by *capping* the left end with a proton *cap* (so as to complete the amino group) and the right end with a hydroxyl cap (so as to complete the carboxyl group). The capping of the residual chain extremities ensures that the polymer is in its finished state, and that it cannot be elongated anymore. The proton is the *left cap* of the protein polymer and the hydroxyl is the *right cap* of the protein polymer.



A protein is made of a chain of residues and of two caps. The left cap is the N-terminal proton and the right cap is the C-terminal hydroxyl. Altogether, the residual chain (enclosed here in the blue polygon) and both the H and OH red-colored caps do form a complete protein polymer in its finished state.

**FIGURE 2.2:** **END CAPPING CHEMISTRY OF THE PROTEIN POLYMER.**

Now comes the question of unambiguously defining the structure of a protein. It is commonly accepted that the simple ordered sequence of each residue code in the protein, from left to right, constitutes an unambiguous description of the protein's primary structure (that is its sequence). Of course, proteins have three-dimensional structures, but this is of no interest to a program like MassXpert, which is aimed at calculating masses of polymers. To enunciate unambiguously the sequence of a protein, one would use a symbology like this:

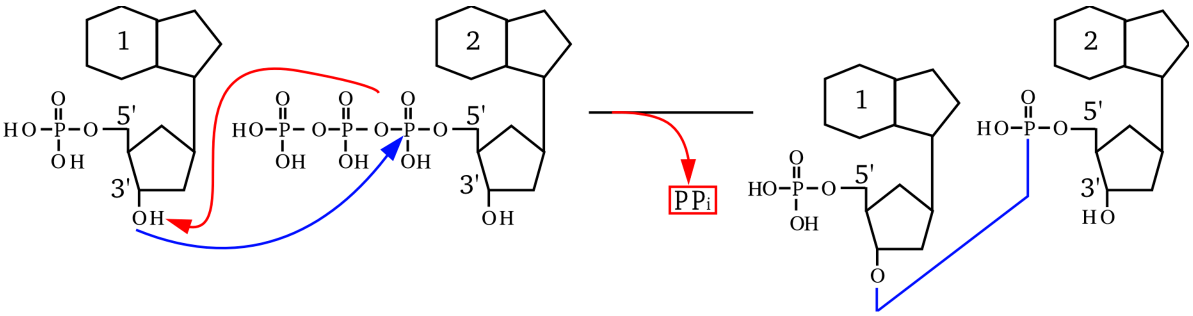
- Using the 3-letter code of the amino acids:
  Ala Gly Trp Tyr Glu Gly Lys

- Using the 1-letter code of the amino acids:
  A G W Y E G K
  Alanine is thus the residue 1 and Lysine is the last residue (n = 7)

### 2.2.2 NUCLEIC ACIDS

These biopolymers are more complex than proteins, mainly because they are composed of monomers *(nucleotides)* that have three different chemical parts, and because those parts differ in DNA and RNA. A nucleotide is the nucleic acid's brick: *a nucleotide consists of a nitrogenous base combined with a ribose/deoxyribose sugar and with a phosphate group*. There are two different kinds of nucleic acids: deoxyribonucleic acid (DNA, the sugar is a deoxyribose) and ribonucleic acid (RNA, the sugar is a ribose). DNA is most often found in its double stranded form, while RNA is most often found in single strand form. There are four nitrogenous bases for each: Adenine, Thymine, Guanine, Cytosine for DNA; in RNA only one of these bases changes: Thymine is replaced by Uracile. As for proteins, nucleic acids are polar polymers: the polymerization process is polar, from left to right (sometimes left is up and right is down in certain vertical representations found mainly in textbooks).

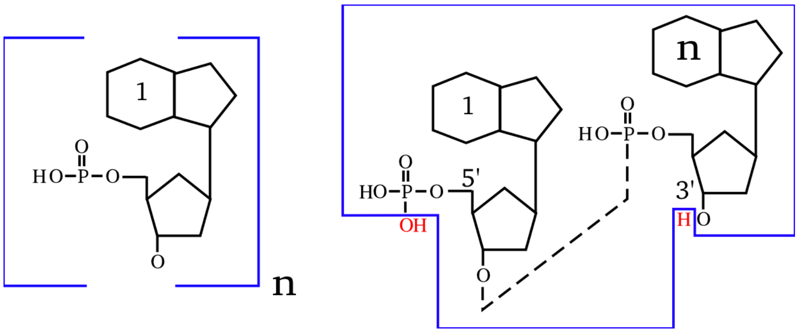
This manual is not to teach biochemistry, which is why the structure of the monomers is not described in atomic detail. However, since it is important to understand how the polymerization occurs, FIGURE 2.3, " PHOSPHODIESTER BOND FORMATION BY ESTERIFICATION. " represents the polymerization reaction mechanism between a nucleotide and another one, to yield a dinucleotide. That reaction is a *trans-esterification*. A nucleic acid has a left end—*5' end; often this end is phosphorylated*—and a right end—*3' end; hydroxyl end*. The trans-esterification reaction is the attack of the phosphorus of the new (deoxy)nucleotide triphosphate by the 3'OH of the right end of the elongating nucleotidic chain. Upon trans-esterification, an *inorganic pyrophosphate* ($PP_i$) is released, and the formation of a phosphodiester bond between the two nucleotides yields a dinucleotide. The elongation of the nucleic acid polymer is a simple repetition of this esterification reaction so that the chain growth is always in the 5'→3' direction. This is achieved in the living cells by what is called the *5'→3' polymerase enzymatic activity*.



The arriving monomer (on the right) has its triphosphate on the 5' carbon of the sugar esterified by nucleophilic attack of the first phosphorus by the alcohol function beared by the 3' carbon of the (deoxy)ribose sugar ring of the left monomer. The bond that is formed is a phosphodiester bond, with release of a pyrophosphate group (PP$_{\mathrm{i}}$). Note that the sugar and nitrogenous bases are schematically represented in this figure.

**FIGURE 2.3: PHOSPHODIESTER BOND FORMATION BY ESTERIFICATION.**

The conventional representation of a nucleic acid involves showing the 5' end on the left, and the 3' end on the right, horizontally. Sometimes, to clearly indicate that the left end is phosphorylated, while the right end is not, the ends are indicated as ``5'P'' and ``3'OH''. Figure 2.4, “ End capping chemistry of the nucleic acid polymer. ” shows a simple way to formalize what a nucleic acid polymer is. The molecule represented on the left is the “monomer” in the sense that the polymer is made of n monomers. On the right side of that figure, the polymer made of n monomers is shown as a residual chain (inside the blue polygon box) that got capped with OH on its left end and H on its right end (red-colored atoms). Thus, in the case of the nucleic acid polymers, the left cap is a hydroxyl and the right cap is a proton. This anecdotically happens to be the exact converse of what was described earlier for proteins.



A nucleic acid is made of a chain of nucleotides (left formula) and of two caps. The left cap is the hydroxyl group that belongs to the terminal phosphate of the 5' carbon of the sugar. The right cap is the proton that belongs to the hydroxyl group of the 3' carbon of the sugar ring (right formula). Altogether, a finished nucleic acid polymer is made of the nucleotidic chain (enclosed here in the blue polygon), made of the repetitive elements (one of which is shown on the left), and of the two caps (red-colored OH and H, out of the box on the right).

**FIGURE 2.4: END CAPPING CHEMISTRY OF THE NUCLEIC ACID POLYMER.**

Now comes the question of unambiguously defining the structure of a nucleic acid. It is commonly accepted that the listing of the named nitrogenous bases in the nucleic acid—from left (5′ end) to right (3′ end)—constitutes an unambiguous description of the nucleic acid sequence. To enunciate the sequence of a gene, one would use a symbology like this:

for a DNA, using the 1-letter code of the nitrogenous bases:

A T G C A G T C

for an RNA, using the 1-letter code of the nitrogenous bases:

A U G C A G U C

Adenine is thus the base 1 and Cytosine is the last base (n = 8)

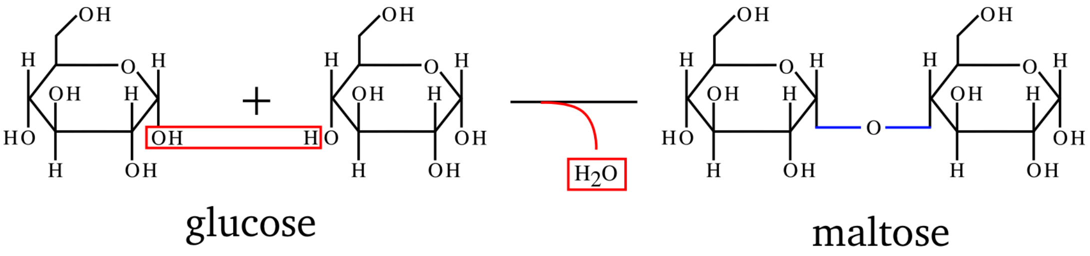
### 2.2.3 SACCHARIDES

These biopolymers are certainly amongst the most complex ones in the living world. This is mainly due to the fact that saccharides are usually heavily modified in living cells with a huge variety of chemical modifications. Furthermore, the ramifications in the polymer structure are more often the normal situation than not. Interestingly, these molecules are first thought of as the "fuel" for the cell, which is certainly far from being total nonsense, but it is also undoubtful that their structural role is extremely important (often in combination with proteinaceous material). Another interesting aspect of their ability to form complex structures is their use as "key" systems for identification processes: a number of complex sugars are located on the cell walls and provide "recognition patterns" for the other cells to deal with…

Nonetheless, the general picture is not that complex, if the way monomers are polymerized together is the only concern (which is the case in this manual). As far as we are concerned, in fact, the polymerization mechanism is a simple condensation (much like what has been described for proteins), yielding a sugar bond. Indeed, some people use the same terminology: a monomeric sugar becomes a residue once polymerized in the saccharidic chain. There are two main different kinds of sugars: *pentoses* (in $C_5$) and *hexoses* (in $C_6$); it should be noted, however, that there is a variety of other common molecules, like *sialic acids*, *heptoses*…

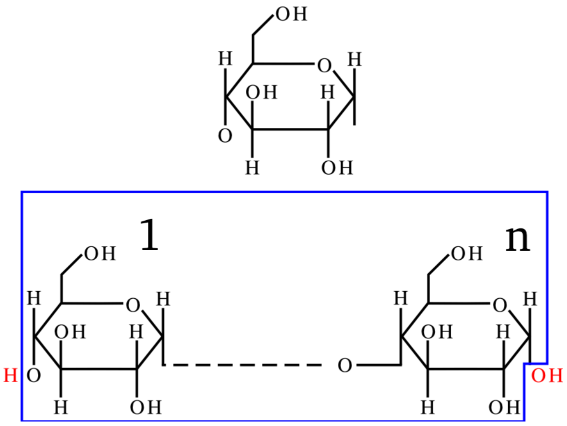
Like already seen for proteins and nucleic acids, a saccharidic polymer is polar: it has a left end and a right end. The terminology regarding the ends of a saccharidic polymer is rather unexpected at first sight: the left end is said to be the *non-reducing end* while the right end is said to be the *reducing end*. Historically this was observed with monosaccharides (also called *monoses*), which reduced cupric ($Cu^{2+}$) ions, thus getting oxydized themselves on the carbonyl (when in the open ring aldehydic form).

FIGURE 2.5, " OSIDIC BOND FORMATION BY CONDENSATION. " shows the polymerization reaction between a sugar and another one (2 glucose monomers, actually), to yield a maltose disaccharide. The polymerization mechanism is a simple condensation. The elongation of the saccharidic polymer is a simple repetition of this condensation reaction so that the chain growth is always in the same orientation, from the non-reducing end to the reducing end. The conventional representation of a polysaccharide involves showing the non-reducing end on the left, and the reducing end on the right, horizontally. Figure 2.6, " End capping chemistry of the polysaccharidic polymer. " shows a simple way to formalize what a saccharidic polymer is. The top formula is the representation of the monomer. The bottom formula represents a polysaccharide, with the repetitive elements boxed (there are n monomers polymerized). The atoms shown in red (outside the boxed repetitive elements) are the saccharidic polymer caps. Thus, we see clearly that in the case of polysaccharides, the left cap is a proton and the right cap is a hydroxyl. This anecdotically happens to be identical to proteins and the exact converse of what we described previously for nucleic acids.



The two monomers are subject to condensation with loss of one molecule of water.

**FIGURE 2.5: OSIDIC BOND FORMATION BY CONDENSATION.**



A polysaccharide is made of a chain of osidic residues (blue-boxed formula) and of two caps (red-colored atoms). The left cap is the proton group that belongs to the non-reducing end of the polymer. The right cap is the hydroxyl group that belongs to the reducing end of the polymer

**FIGURE 2.6: END CAPPING CHEMISTRY OF THE POLYSACCHARIDIC POLYMER.**

Now comes the question of unambiguously defining the structure of a saccharidic polymer. It is commonly accepted that the simple ordered sequence of the named monoses in the saccharidic polymer, from left (non-reducing end) to right (reducing end), constitutes an unambiguous description of the glycan sequence. To enunciate the sequence of a glycan, one would use a symbology like this:

Using a 3-letter code:

Ara Gal Xyl Glc Hep Man Fru

Arabinose is thus the monose 1 and Fructose is the last monose (n = 7)

Incidentally, this is where the ability of MassXpert to handle monomer codes of non-limited length comes in handy!

## 2.3 To Sum Up

We made a rapid overview of the three major polymers in the living world. A great many other polymers exist around us. Table 2.1, “ Quick comparison of three biopolymers with examples of monomers ” tries to sum up all the informations gathered so far. Note that the formulæ given for the monomers are the “residual” ones. For example, the formula of the glycyl residue corresponds to the formula of the Glycine monomer less one molecule of water. Many synthetic polymers are much simpler than the ones we have rapidly reviewed, and it should be clear that, if MassXpert can deal with the complex biopolymers described so far, it certainly will be very proficient with less complex synthetic polymers. Describing the formation of polymers is one thing, but we also have to describe how to disrupt polymers. This is what we shall do in the next section.

TABLE 2.1: QUICK COMPARISON OF THREE BIOPOLYMERS WITH EXAMPLES OF MONOMERS

| polymer | name | code | formula | left cap | right cap |
|---|---|---|---|---|---|
| protein | | | | H | OH |
| | Glycine | G | $C_2H_3O_1N_1$ | | |
| | Alanine | A | $C_3H_5O_1N_1$ | | |
| | Tyrosine | T | $C_9H_9O_2N_1$ | | |
| nucleic acid | | | | OH | H |
| | Adenine | A | $C_{10}H_{12}O_5N_5P_1$ | | |
| | Cytosine | C | $C_9H_{12}O_6N_3P_1$ | | |
| saccharide | | | | OH | H |
| | Arabinose | Ara | $C_5H_8O_4$ | | |
| | Heptose | Hep | $C_7H_{12}O_8$ | | |

## 2.4 POLYMER CHAIN DISRUPTING CHEMISTRY

The “polymer chain disrupting chemistry” was mentioned earlier as a complex subject that was of *enormous* importance to the mass spectrometrist. This is why that subject will be treated in a pretty thorough manner. First of all it should be noted that a chemical modification of a polymer does not necessarily involve the perturbation of the chain structure of the polymer. Here, however, we are concerned specifically with a number of chemical modifications that yield a polymer chain perturbation; *cleavage* and *fragmentation*:

**Cleavages.** These are chemical processes by which a cleaving agent will act directly on the polymer chain making it fall into at least two separated pieces (the *oligomers*). As a result of the cleavage reaction, groups originating in the cleaving molecule remain attached to the polymer at the precise cleavage location;

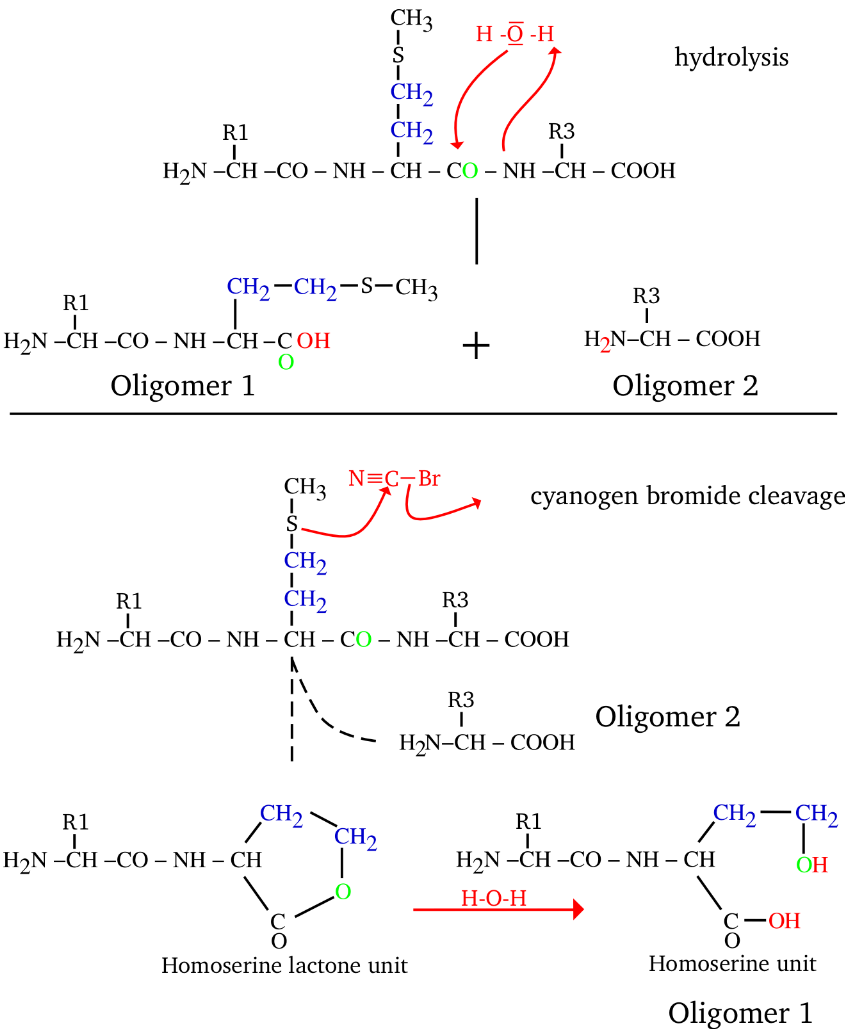
**Fragmentations.** These are chemical processes by which the polymer structure is disrupted into separated pieces (the *fragments*) mainly because of energy-dependent electron doublet rearrangements leading to bond breakage.

### 2.4.1 POLYMER CLEAVAGE

We said above that, upon cleavage of a polymer, the cleaving molecule reacts with it, and by doing so directly or indirectly *“dissolves”* an inter-monomer bond. A polymer cleavage always occurs in such a way as to generate a set of *true* polymers (smaller in size than the parent polymer, evidently, which is why they are called *oligomers*). Indeed, let us take the example shown in FIGURE 2.7, “ PROTEIN CLEAVAGE BY WATER AND CYANOGEN BROMIDE ” , where a tripeptide (a very little protein, containing a methionyl residue at position 2) is submitted either to a water-mediated cleavage (hydrolysis, upper panel) or to a cyanogen bromide-mediated cleavage (lower panel). The two cases presented in this figure are similar in some respects and different in others:

- In the first case the molecule that is responsible for the cleavage is water, while in the second case it is cyanogen bromide;
- In both cases the bond that is cleaved is the inter-monomer bond (in protein chemistry this is a peptidic bond);
- In both cases the Oligomer 2 has the same structure;
- The structures of the Oligomer 1 species differ when produced using water or cyanogen bromide as the cleaving molecule.

The difference between hydrolysis and cyanogen bromide cleavage is in the generation of the Oligomer 1 species: the cyanogen bromide cleavage has a side effect of generating a homoserine as the right end monomer of Oligomer 1, while hydrolysis generates a genuine methionine monomer. This is because water reverses in a very symmetrical manner what polymerization did (hydrolysis is the converse of condensation), while cyanogen bromide did some chemical modification onto the generated Oligomer 1 species.



A tripeptide is cleaved at position 1 either by hydrolysis (top) or by cyanogen bromide (bottom). Cyanogen bromide cleaves specifically on the right of a methionine monomer. Upon cleavage, the methionyl monomer gets converted into homoserine by the cyanogen bromide reagent.

FIGURE 2.7: **PROTEIN CLEAVAGE BY WATER AND CYANOGEN BROMIDE**

Nonetheless, the reader might have noted that—interestingly—all the four oligomers do effectively have their left cap (a proton) and their right cap (the hydroxyl). This means that in both water- and cyanogen bromide-mediated cleavages, all the generated oligomers are indeed true polymers in the sense that: 1) they are a chain of monomers (modified or not) and 2) they are correctly capped ( *i.e.* they are polymers in their finished state).

This is important because it is the basis on which we shall make the difference between a cleavage process and a fragmentation process. Thus, the MassXpert definition of an oligomer might be: *an oligomer is a polymer (of at least one monomer) in its finished state that was generated upon cleavage of a longer polymer.*

When the polymer cleavage reaction precisely reverses the reaction that was performed for the same polymer's synthesis, there is no special difficulty. But when the cleavage reaction modifies the substrate, then this should be carefully modelled. How? To answer this question we might start by comparing the two different Oligomer 1 species that were yielded upon the water-mediated and the cyanogen bromide-mediated cleavage reactions: “the hydrolysis-generated Oligomer 1 is equal to the cyanogen bromide-generated Oligomer 1 +S1 +C1 +H2 -O1”; this is a big difference! The observations we did so far might be worded this way: *Whenever a protein undergoes a cyanogen bromide-mediated cleavage, the “-C1H2S1+O1” chemical reaction should be applied to the resulting oligomers if and only if they have a methionine monomer at their right end.* In MassXpert's jargon, this logical condition is called a *cleavage rule* (described later; see Section 3.2.4, “ The cleavage specifications ” ).

Well, all this sounds reasonable. But what about the “normal” case, when the cleavage is done using water? Nothing special: the mass of the oligomer is calculated by summing the mass of each monomer in the oligomer (since the monomers are not modified, this is easily done) and the masses corresponding to the left and right caps (these are defined in the polymer chemistry definition; in our present case it would be a proton on the left end, and a hydroxyl on the right end). In this way, the oligomer complies with its definition, which states that it is a faithful polymer made of monomers and that it is in its finished state.

Yes, but then how will MassXpert manage to calculate the mass of the modified oligomer, like our Oligomer 1 in the case of the cyanogen bromide-mediated cleavage? Simple enough: in a first step it does exactly the same way as for the unmodified oligomer. Next, each oligomer is checked for presence or absence of a methionine residue on its right end. If a methionine is found, the mass corresponding to the “-C1H2S1+O1” chemical reaction is applied. And that's it.

In the previous cyanogen bromide example, the logical condition was involving the identity of the oligomers' right end monomer, but other examples can involve not the right end monomer, but the left end monomer, if some chemical modification was to occur to the monomer sitting right of the cleavage location. In this case the user would have to analyse the situation and provide MassXpert with the proper chemical reaction by stating something analog to: *if and only if they have a Xyz monomer at their left end.* This introduction to polymer cleavage abstraction should be enough to later delve into the cleavage specification definition as MassXpert conceives it and that is thoroughly detailed at Section 3.2.4, “ The cleavage specifications ” .

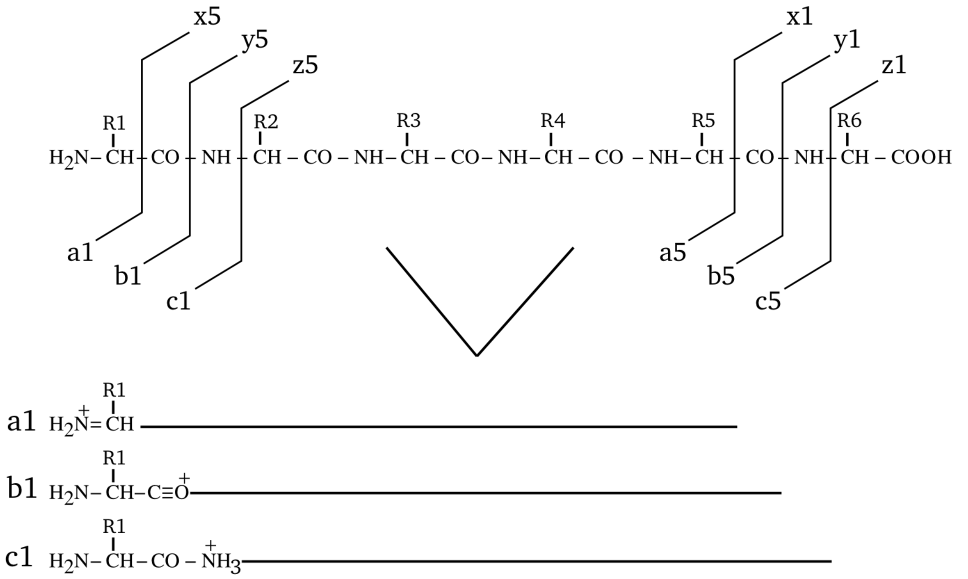
### 2.4.2 Polymer Fragmentation

In a fragmentation process, the bond that is broken is not necessarily the inter-monomer bond. Indeed, fragmentations are oft-times high energy chemical processes that can affect bonds that belong to the monomers' internal structure. This is one of the reasons why fragmentations do differ from cleavages: they are specific of the polymer type in which they occur. Hydrolyzing a protein and an oligosaccharide is just the same process, from a chemical point of view. But fragmenting a protein or an oligosaccharide are truly different processes because the way that the fragmentation happens in the polymer sequence is so much dependent on the nature of each monomer that makes it.

Another peculiarity of the fragmentations, compared with the cleavages that were described above, is the fact that there is no cleaving molecule starting the process. Instead, a fragmentation process is often initiated by an intra molecular electron doublet rearragement that propagates more or less in the polymer structure to eventually break it. Fragmentations are mainly a gas phase process, not some reaction that happens in solution as a result of putting in contact the polymer and some reagent. It is precisely because no cleaving molecule is involved in the fragmentation process that the fragments are not necessarily capped like a normal polymer should be; and this is another really important difference between cleavage and fragmentation. The following examples should illustrate these concepts: protein and nucleic acid fragmentation.

#### 2.4.2.1 PROTEIN FRAGMENTATION

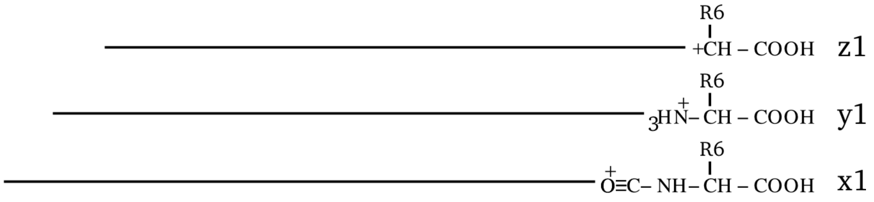
There is a pretty important number of different kinds of fragments that can be generated upon fragmentation of peptides. We are going to detail the most common ones; the user is invited to use the MassXpert' fragmentation-specification grammar to add less frequent (or newly discovered) fragmentation types. Note that the fragmentation schemes below apply to positively-charged precursor ions. To compute the product ions' masses obtained in negative mode fragmentation experiments, then, simply remove as many protons as required. For example, to switch from a fragment positively charged once (+H), then remove a first proton to go back to the uncharged state and then remove another proton to yield the deprotonated (thus singly negatively charged) ion product. The requirement to be able to computed masses for the positively- and negatively-charged ion products imposes a specific way to defined fragmentation specifications in the XpertDef module (to be detailed later in this manual).



m (a1) = m (monomer 1) - C1O1 + left cap (H1)

m (b1) = m (monomer 1) + left cap (H1)

m (c1) = m (monomer 1) + N1H3 + left cap (H1)



✲ m (z1) = m (monomer 6) -N1H1 + right cap (O1H1) (variant: +H1)

m (y1) = m (monomer 6) + H2 + right cap (O1H1)

m (x1) = m (monomer 6) + C1O1 + right cap (O1H1)

✲ Note how a z fragment is identical to a [y -NH3] fragment.
In some cases (high CID energy) the z fragment is often seen as a species of mass z+1

---

canonical monomer immonium ions

An hexapeptide is fragmented in the seven most widely encountered manners, such as to generate a, b, c, x, y, z and immonium fragment ions. The figure illustrates the position of the cleavage for each kind of fragment (exemplified using the case of the smallest fragment possible) and the mass calculation method is described for each fragment kind; consider that each fragment bears only *one positive* charge.

FIGURE 2.8: PROTEIN FRAGMENTATION PATTERNS MOST WIDELY ENCOUNTERED.

As can be seen from Figure 2.8, “ Protein fragmentation patterns most widely encountered. ” , the fragmentations do generate fragments of three categories: the ones that include the left end of the precursor polymer (a, b, c), the ones that include the right end of the precursor polymer (x, y, z), and finally the special case in which the fragment is an *internal fragment*, like the immonium ions. When looking at the fragmentations described in the figure it becomes immediately clear why a fragmentation cannot be mistaken for a cleavage: the ionization of the fragment is not necessarily due to the captation of a proton by the fragment. Furthermore, we can also see that a fragmentation is not a cleavage because the fragment that is generated is *absolutely* not necessarily what we call a polymer, in the sense that the fragment might not be capped the same way as the precursor polymer is (that is, the fragment is not in its finished polymerizaton state).

The two observations above should make clear to the reader that calculating masses for fragments is a more difficult process than what was described above for the oligomers. Indeed, while it was simple to calculate the mass of an oligomer (by simply adding the masses of its constitutive monomer units, plus the left and right caps, plus ionization), here there is no chemical formalism generally applicable to all the fragment types. This is why the specification of the fragmentation is left to the user's responsibility.

By looking at Figure 2.8, “ Protein fragmentation patterns most widely encountered. ” , the reader should have noticed that the fragment naming scheme takes into consideration the fact that the fragment bears the left or the right end of the precursor polymer (or none, also). Indeed, the numbering of fragments holding the left end of the precursor polymer sequence begins at the left end, and for fragments that hold the right end, at the right end. Thus the third fragment of series *a* (*a3*) would involve monomers [1→] and the third fragment of series *y* (*y3*) would involve monomers [6→] (in the figure, these left-to-right and right-to-left directions are symbolized using arrows). Therefore, it should appear to the reader how important—when specifying a fragmentation—it is to clearly indicate from which end of the precursor polymer the fragment is generated (in MassXpert's jargon this is “LE” for left end, “RE” for right end and “NE” for no end). MassXpert knows what action it should take when it encounters one of these three specifications; for example, if a “LE” specification is found for a given fragmentation specification, MassXpert adds to the fragment's mass the mass corresponding to the left cap of the precursor polymer.

The mass calculations for the different fragment product ion types is shown in Figure 2.9, “ Peptide product ion mass calculation for the different ion series ” .

$$a_i = LC + \sum_{1}^{i} M_i - CO$$

$$b_i = LC + \sum_{1}^{i} M_i$$

$$c_i = LC + \sum_{1}^{i} M_i + NH_3$$

$$x_i = RC + \sum_{1}^{i} M_i + CO$$

$$y_i = RC + \sum_{1}^{i} M_i + H_2$$

$$z_i = RC + \sum_{1}^{i} M_i - NH$$

$$z_i = RC + \sum_{1}^{i} M_i - NH + H \text{ (HCD)}$$

$$imm_i = M_i + H - CO$$

The way masses are computed from peptide fragments change according to the fragmention pattern that produces product ions of different series. *LC* and *RC* mean *left cap* and *right cap*, respectively.

**FIGURE 2.9: PEPTIDE PRODUCT ION MASS CALCULATION FOR THE DIFFERENT ION SERIES**

If we take the *a* fragment series, the FIGURE 2.8, “ PROTEIN FRAGMENTATION PATTERNS MOST WIDELY ENCOUNTERED. ” indicates that the fragments include the left end and that their last monomer lacks its carbonyl group (see, on top of FIGURE 2.8, “ PROTEIN FRAGMENTATION PATTERNS MOST WIDELY ENCOUNTERED. ” , that the *a1* arrow goes between the CαH and the CO of monomer 1). So we would say that each fragment of the *a* series should be challenged with the following chemical treatments: 1) addition of the mass corresponding to the left cap (proton), 2) removal of the mass corresponding to the lacking CO group. This way we have the mass of fragment *a1*. If we were interested in the fragment *a4* we would have summed the masses of monomers 1 to 4, added the mass of the left cap, and finally removed the mass of a CO.

For the *x* series of fragments we do not add the left cap anymore, but replace it with the right cap, since the fragments hold the right end of the precursor polymer. Note also that the numbering of the monomers using the variable *i* in the following mathematical expressions goes from right to left (contrary to what happened for the *a, b, c* fragment series. All the fragments that hold the precursor polymer right end are numbered this way, so this applies to fragments *x, y, z*.

In low energy CID, the *z* fragments are expressed in a distinct way, when compare to high energy CID.

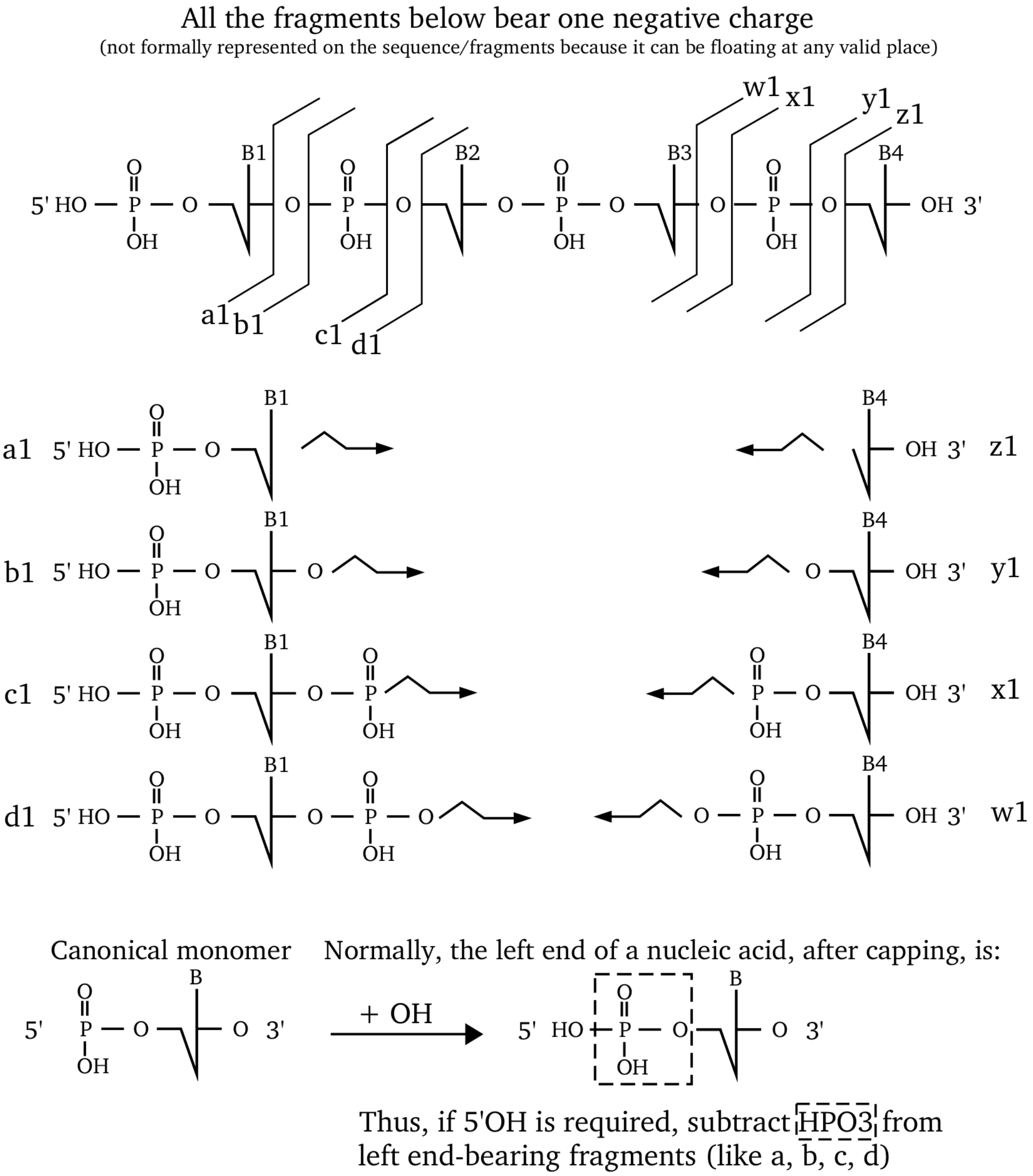
In the *immonium* fragment series, the fragments are internal fragments in the sense that they do not hold neither of the two precursor polymer's ends. MassXpert understands that the user is speaking of this kind of fragment when the “from which end” piece of data —in the fragmentation specification—states “NE” instead of “LE” or “RE” (see SECTION 2.4.2, “ POLYMER FRAGMENTATION ” ). The mass calculation for these fragments does not take into account the monomers surrounding the one for which the calculation is done. The mass for an immonium ion —at position *i* in the precursor polymer—will be the mass of the monomer at position *i*, less the mass of a CO, plus the mass of a proton.

### 2.4.2.2 NUCLEIC ACIDS FRAGMENTATION

The fragmentations that can be obtained with nucleic acids are numerous and it is more complicated than with proteins to describe them fully. The main reason for this is that there are a big number of fragmentation combinations because of the loss of nitrogenous bases from the skeleton. The mechanisms by which this loss happens are fairly complex, and I am not going to detail any of them. FIGURE 2.10, “ DNA FRAGMENTATION PATTERNS MOST WIDELY ENCOUNTERED. ” shows the most common fragmentations (without taking into consideration the potential loss of bases). An example of fragment is given for each fragment series (pretty the same way as we did before for proteins). Note that the fragment representations are aimed at helping the reader to figure out what the product ion is, not taking into account where the negative charge lies on the fragment, since this charge can float around at every de-protonatable group. All the fragments shown bear one and one only charge.

Another remark pertaining to the ionization mode of the ion products: the fragmentation schemes below apply to negatively-charged precursor ions (by loss of a proton, typically). To compute the product ions' masses obtained in positive mode fragmentation experiments, then, simply add as many protons (or any other cationic ionization agent) as required. For example, to switch from a fragment negatively charged once (-H), then add a first proton to go back to the uncharged state and then add another proton to yield the monoprotonated (thus singly positively charged) ion product. The requirement to be able to computed masses for the positively- and negatively-charged ion products imposes a specific way to defined fragmentation specifications in the XpertDef module (to be detailed later in this manual).

The reader might have noticed at the bottom of FIGURE 2.10, “ DNA FRAGMENTATION PATTERNS MOST WIDELY ENCOUNTERED. ” that a provision is made in the case the fragmented molecular species are not 5′ end-phosphorylated but 5′ end-hydroxylated. Indeed, the canonical monomer is such that, upon polymerization and left capping, the 5′ end is phosphorylated. However, oft-times the oligonucleotides are synthesized chemically without the 5′ end phosphate group, thus ending in hydroxyl. This special case should be accounted for by applying to all the fragments that bear the left end of the precursor polymer the following chemical reaction: -$HPO_3$. This chemical reaction should be applied *in addition* to the chemical reaction that yields the fragment *per se*.



A short DNA sequence is fragmented in the eight most widely encountered manners, such as to generate a, b, c, d, w, x, y, z fragment ions. The figure illustrates the position of the cleavage for each kind of fragment (exemplified using the case of the smallest fragment possible). and the mass calculation method is described for each fragment kind; considering that each fragment is protonated only once (+1).

**FIGURE 2.10:** **DNA FRAGMENTATION PATTERNS MOST WIDELY ENCOUNTERED.**

Exactly as done earlier for the protein fragments, the mathematical expressions used to calculate the mass of different series of nucleic acid fragments are provided in Figure 2.11, “ Nucleic acids product ion mass calculation for the different ion series ” . In these calculations it is assumed that the left end of the precursor polymer is phosphorylated (5'P) and the reader should bear in mind that this precise phosphate might itself be expelled by the fragmentation. The fragment naming schemed detailed earlier for proteins applies to nucleic acids in the very same manner.

$$a_i = LC + \sum_{1}^{i} M_i - O$$

$$b_i = LC + \sum_{1}^{i} M_i$$

$$c_i = LC + \sum_{1}^{i} M_i - HPO_2$$

$$d_i = LC + \sum_{1}^{i} M_i - HPO_3$$

$$w_i = RC + \sum_{1}^{i} M_i + O$$

$$x_i = RC + \sum_{1}^{i} M_i$$

$$y_i = RC + \sum_{1}^{i} M_i - HPO_2$$

$${}_{zi} = RC + \sum_{1}^{i} M_i - HPO_3$$

The way masses are computed from oligonucleotide fragments change according to the fragmention pattern that produces product ions of different series. *LC* and *RC* mean *left cap* and *right cap*, respectively.

**FIGURE 2.11:** **NUCLEIC ACIDS PRODUCT ION MASS CALCULATION FOR THE DIFFERENT ION SERIES**

In the *a* fragment series, the fragments most often appear with base loss. There are also a variety of fragments for which a base is lost.

### 2.4.2.3 More Complex Patterns Of Fragmentation

Before finishing with fragmentations, it is necessary to describe a powerful feature of the fragmentation specification grammar available in MassXpert. This feature was required for the fragmentation of oligosaccharides and also sometimes for proteins. When the fragmentation (the bond breakage reaction itself) occurs at the level of certain monomers, it might be necessary to be able to specify some particular chemistry that would arise on the monomer in question.

We have seen in the cleavage documentation that, upon cleavage of a protein sequence with cyanogen bromide, for example, a particular chemical reaction had to be applied to the oligomers that were generated with a methionine monomer as their right end monomer. Well, in a fragmentation specification it is possible to apply comparable chemical reactions but in a more thorough manner. Indeed, while in the cleavage it was possible to say something like: —*"Apply a given chemical reaction to the oligomer if the right end monomer is Xyz"*, in the fragmentation the logical condition can be bound not only to the identity of the currently fragmented monomer, but also (optionally) to the identity of the previous and/or next monomer in the precursor polymer sequence. For example: —*"Apply a given chemical reaction if fragmentation occurs at the level of 'Xyz' monomer only if it is preceded by a 'Yxz' monomer and followed by a 'Zyx' monomer"*.

These logical conditions are called *fragmentation rules*. A *fragmentation specification* can hold as many rules as necessary. All of this is described in great detail at SECTION 2.4.2, " POLYMER FRAGMENTATION " .

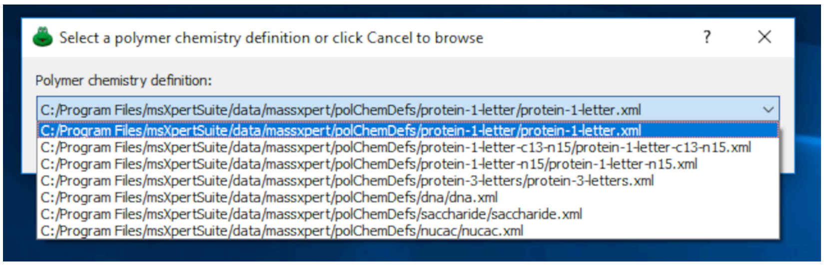
#### 2.4.2.4 To Sum Up

To sum up all what we have seen so far with polymer chain disrupting chemistries:

- Fragments are never ionized automatically; ionization (gain/loss of a charged group) is necessarily integrated in the fragmentation specification.
- A polymer sequence gets fragmented into fragments when a bond breakage occurs, without the help of any exterior molecule, at any level of the polymer structure, with no limitation to the inter-monomer bond; monomer-specific chemical reactions can be modelled into the fragmentation specification using any number of fragrules;
- Oligomers are automatically capped—*on both ends*—using the rules described in the precursor polymer's definition;
- Fragments are capped automatically only—*on the end they hold, if any*—using the rules described in the precursor polymer's definition;
- Oligomers are automatically ionized (if required by the user) using the rules described in the precursor polymer's definition;
- Fragments are never ionized automatically; ionization (gain/loss of a charged group) is necessarily integrated in the fragmentation specification.

# 3 XpertDef: Definition of Polymer Chemistries

After having completed this chapter the reader will be able to accomplish the very first steps needed to use MassXpert's features at best: the normal workflow, indeed, is to first make a polymer chemistry definition, in order to be able to edit polymer sequences of that specific definition. The XpertDef module is made available in MassXpert by pulling down the *XpertDef* menu item from the program's menu. It is possible to start a new polymer chemistry definition from scratch, but it is certainly usually easier to first duplicate a polymer chemistry definition shipped with MassXpert and then open that copy and edit it. Please, refer to chapter DATA-CUSTOMIZATION ↗ , for an explanation of how this is safely done.



```
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/protein-1-letter/protein-1-letter.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/protein-1-letter/protein-1-letter.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/protein-1-letter-c13-n15/protein-1-letter-c13-n15.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/protein-1-letter-n15/protein-1-letter-n15.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/protein-3-letters/protein-3-letters.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/dna/dna.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/saccharide/saccharide.xml
C:/Program Files/msXpertSuite/data/massxpert/polChemDefs/nucac/nucac.xml
```

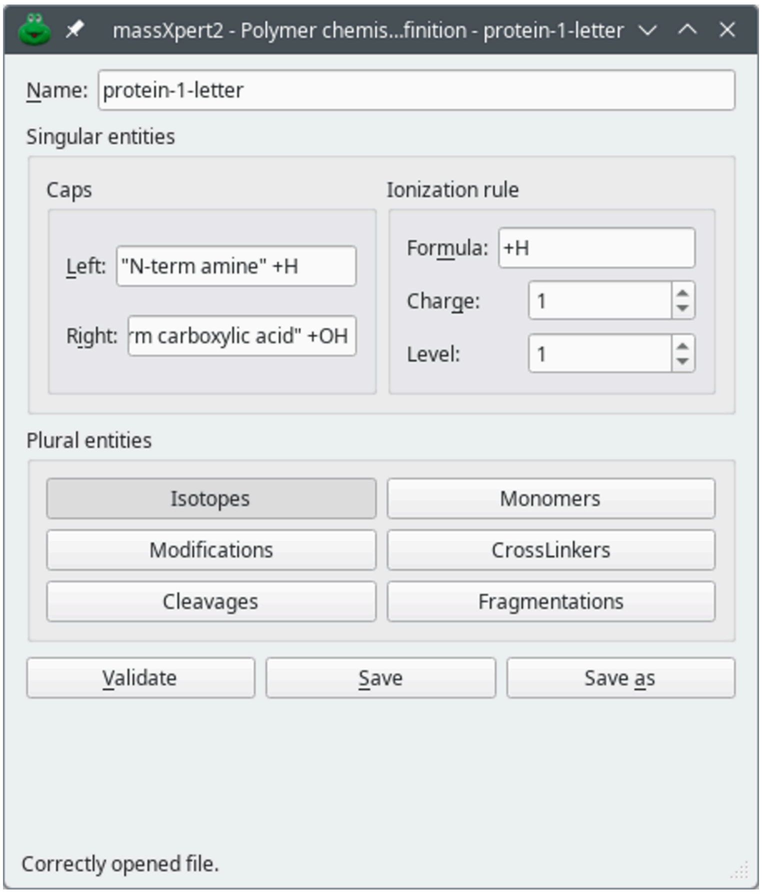
It is possible to immediately select a polymer chemistry definition already registered with the system, or open an arbitrary file by browsing the filesystem (click the *cancel* button, hidden in this figure, if so desired).

FIGURE 3.1: **SELECT ONE POLYMER CHEMISTRY DEFINITION FILE.**

To open a polymer chemistry definition, the user may either select one that is already registered with the system, and that appears listed in the drop-down list widget shown in figure FIGURE 3.1, “ SELECT ONE POLYMER CHEMISTRY DEFINITION FILE. ” or click the *cancel* button so as to open one definition file by browsing the filesystem. In the polymer chemistry definition window that shows up, the user accomplishes two different tasks:

- Define the name of the polymer chemistry definition;
- Define “singular” data like the left cap and the right cap of the polymer, the ionization rule governing the default ionization of the polymer sequence;
- Define the isotopes needed to operate the different polymer chemistry entities (these are “plural” data) ;
- Define all the polymer chemistry entities needed to work on polymer sequences (all these are also “plural data”) .

The definition of the isotopes and of all the chemical entities belonging to a given polymer chemistry are collectively called a *polymer chemistry definition*. The polymer chemistry definition window that shows up is shown in figure FIGURE 3.2, “ POLYMER CHEMISTRY DEFINITION WINDOW. ” .



Name: protein-1-letter

Left: "N-term amine" +H

Right: rm carboxylic acid" +OH

Formula: +H

Charge: 1

Level: 1

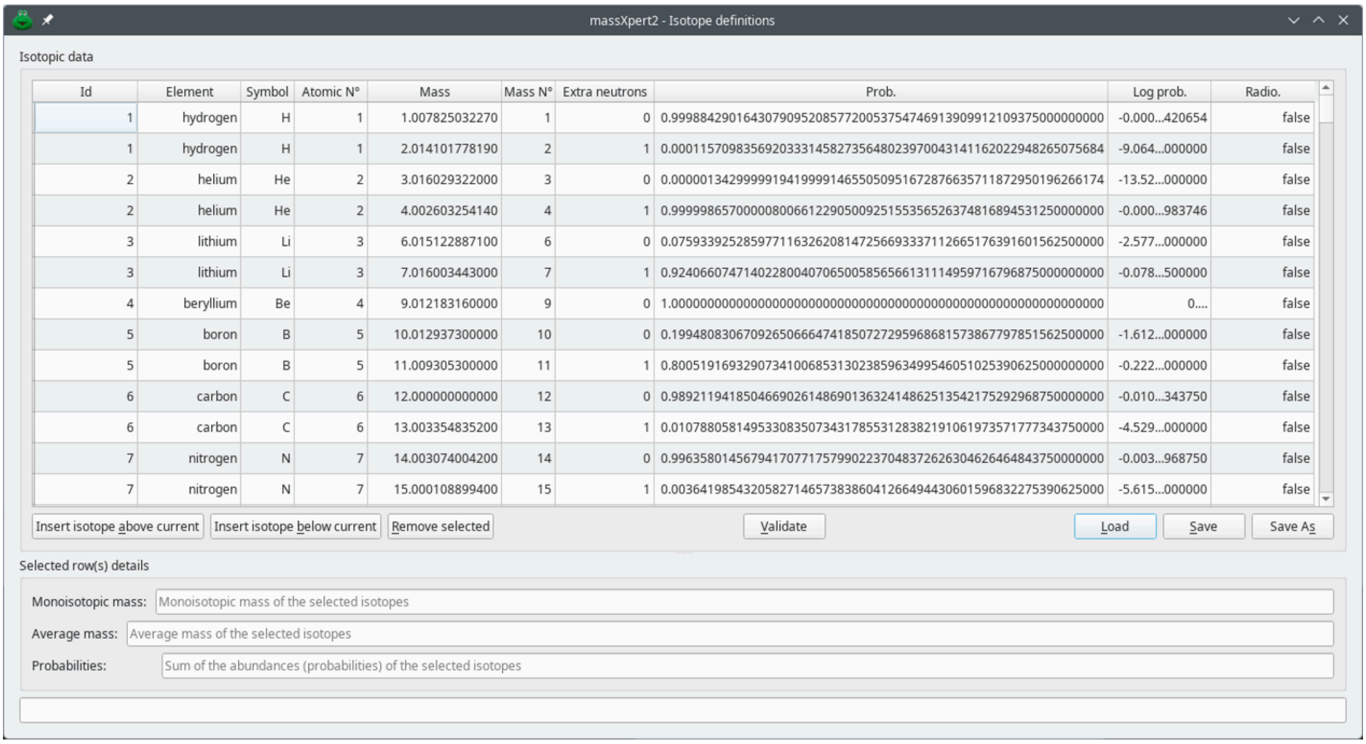
Correctly opened file.

All the polymer chemistry entities are defined in this window. the different buttons dealing with isotopes, monomers, modifications, cross-linkers, cleavage and fragmentation specifications open up specific dialogs (see below).

FIGURE 3.2: POLYMER CHEMISTRY DEFINITION WINDOW.

## 3.1 The isotopes

The definition of the isotopes is performed through the user interface shown in figure Figure 3.3, “ Isotopes definitions ” ( *Isotopes* button in the polymer chemistry definition window). In this dialog, the user defines all the isotopes that collectively form the chemical elements (at least one isotope per atom, logically). In the table view, each row corresponds to an isotope.



| Id | Element | Symbol | Atomic N° | Mass | Mass N° | Extra neutrons | Prob. | Log prob. | Radio. |
|---|---|---|---|---|---|---|---|---|---|
| 1 | hydrogen | H | 1 | 1.007825032270 | 1 | 0 | 0.999884290164307909520857720053754746913909912109375000000000 | -0.000...420654 | false |
| 1 | hydrogen | H | 1 | 2.014101778190 | 2 | 1 | 0.000115709835692033314582735648023970043141162022948265075684 | -9.064...000000 | false |
| 2 | helium | He | 2 | 3.016029322000 | 3 | 0 | 0.000001342999991941999914655050951672876635711872950196266174 | -13.52...000000 | false |
| 2 | helium | He | 2 | 4.002603254140 | 4 | 1 | 0.999998657000008006612290500925155356526374816894531250000000 | -0.000...983746 | false |
| 3 | lithium | Li | 3 | 6.015122887100 | 6 | 0 | 0.075933925285977116326208147256693337112665176391601562500000 | -2.577...000000 | false |
| 3 | lithium | Li | 3 | 7.016003443000 | 7 | 1 | 0.924066074714022800407065005856566131114959716796875000000000 | -0.078...500000 | false |
| 4 | beryllium | Be | 4 | 9.012183160000 | 9 | 0 | 1.000000000000000000000000000000000000000000000000000000000000 | 0.... | false |
| 5 | boron | B | 5 | 10.012937300000 | 10 | 0 | 0.199480830670926506664741850727295968681573867797851562500000 | -1.612...000000 | false |
| 5 | boron | B | 5 | 11.009305300000 | 11 | 1 | 0.800519169329073410068531302385963499546051025390625000000000 | -0.222...000000 | false |
| 6 | carbon | C | 6 | 12.000000000000 | 12 | 0 | 0.989211941850466902614869013632414862513542175292968750000000 | -0.010...343750 | false |
| 6 | carbon | C | 6 | 13.003354835200 | 13 | 1 | 0.010788058149533083507343178553128382191061973571777343750000 | -4.529...000000 | false |
| 7 | nitrogen | N | 7 | 14.003074004200 | 14 | 0 | 0.996358014567941707717579902237048372626304626464843750000000 | -0.003...968750 | false |
| 7 | nitrogen | N | 7 | 15.000108899400 | 15 | 1 | 0.003641985432058271465738386041266494430601596832275390625000 | -5.615...000000 | false |

Each row of the table view corresponds to an isotope. For any given element, there must be at least one isotope.

**FIGURE 3.3: ISOTOPES DEFINITIONS**

The table view is editable. Each cell, when double-clicked, switches to an edit mode in which the user might enter a value. A series of buttons below the table view trigger various actions. To add a new isotope, either above or below the currently selected one, click of the two *Insert isotope...* buttons. To remove isotopes, select them and click the *Remove selected* button. The three buttons on the right hand side of the window, allow loading, saving and saving with a new file name the isotopes currently displayed in the table view.

When the user selects one or more rows of the table, the program updates some data specific to the selected isotopes. As long as the selected isotopes all have the same chemical element, they are taken into account for the calculation of the average mass (*Average mass*). The *Probabilities* line edit widget displays the sum of the abundances of the selected isotopes (only if these are of the same chemical element).

When the user finishes editing the probability of a given isotope, the program automatically updates that probability's logarithm value and refreshes the corresponding *Log prob.* cell on the same row.

## 3.2 THE POLYMER CHEMICAL ENTITIES

Once the atoms have been properly defined (note that such atoms are already available in the distributed package), it is possible to start entering data for the other polymer chemical entities (figure FIGURE 3.2, “ POLYMER CHEMISTRY DEFINITION WINDOW. ” . These are often defined using chemical formulas, which explain why it is necessary to first define the atoms.

The following are the data that need to be entered so as to obtain a usable polymer chemistry definition:

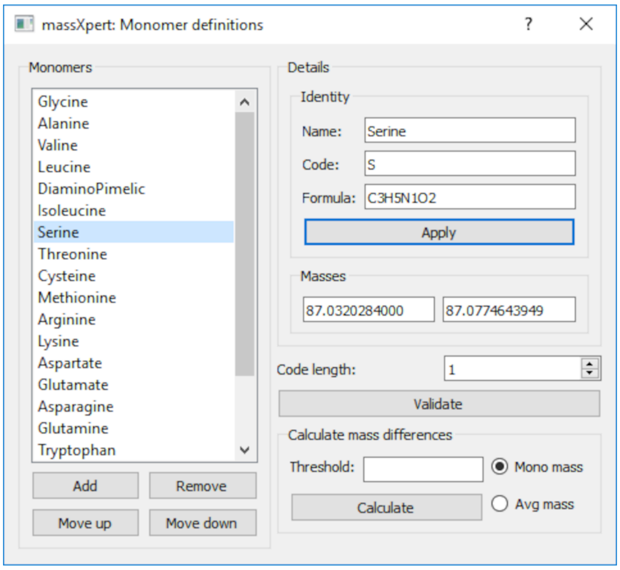
- The polymer chemistry definition's name: *protein-1-letter*;
- The chemical capping reactions that should happen on the left end and on the right end of the polymer sequence:
    - *+H*: left capping of the polymer sequence. proteins are capped at the n-terminal end with a proton;
    - *+OH*: right capping of the polymer sequence. proteins are capped at their c-terminal end with a hydroxyl group;
- The *ionization rule* describes the manner in which the polymer sequence should be ionized by default, when the mass is calculated. This rule actually holds three elements:
    - *+H*: chemical reaction that ionizes the polymer sequence. in the example, all the polymer sequences of polymer chemistry definition “protein-1-letter” are protonated by default;
    - *1*: charge that is brought by the chemical agent ionizing the polymer (the formula above). In the example, a protonation reaction brings a single positive charge.
    - *1*: ionization level, that is, the number of times that the ionization (above) must be performed by default on any polymer sequence of this chemistry definition. In this example, monoprotonation is set as the default ionization rule.

At this point, time has come to deal with “plural” data. the first chemical entities to deal with are monomers.

### 3.2.1 THE MONOMERS

The monomers are the constitutive blocks of the polymer sequence. in the MassXpert's jargon, “monomer” stands *not* for the molecule that may be used to perform a polymer synthesis; it stands for this molecule *less* the chemical group(s) that were eliminated upon polymerization. In the case of the biological polymers, the creation of chemical link between two monomers invariably leads to the loss of a water molecule (that is also called a condensation reactions in organic chemistry).

Click onto the *monomers* button, which triggers the opening of the dialog window shown in figure FIGURE 3.4, “ MONOMER DEFINITION ” .



Name: Serine

Code: S

Formula: C3H5N1O2

Masses: 87.0320284000 87.0774643949

Code length: 1

Threshold:

● Mono mass

○ Avg mass

Each monomer is defined with a name, a code and a chemical formula.

FIGURE 3.4: **MONOMER DEFINITION**

The way this dialog is operated is similar to what was described for the atom definition, unless it is simpler, because monomers are non-deep objects: there are no contained objects. One data element is critical: the number of characters that might be used to define the code of the element cannot be greater than the value entered in the *code length* spinbox widget [1] .

The fundamental rule is the following:

## WARNING

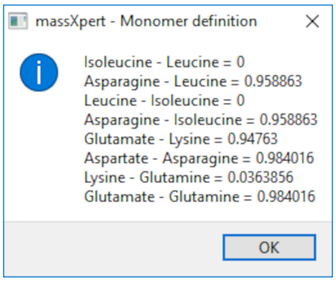
The first character of a monomer code must be uppercase, while the remaining characters (if any) must be lowercase. That means that—if *code length* is 3—“a”, “al”, “ala” would be perfectly fine, while “alan”, “al”, “a”, “ala” would be wrong.

Each time a formula is either displayed by selecting a new monomer in the list or modified by editing it in its line edit widget, the monoisotopic and average masses are recalculated.

As of version 2.3.5, it is possible to calculate the mass difference between any two monomers in the definition. This is useful, for example, to grasp the resolution and mass accuracy requirements for a given polymer definition. The user sets a threshold to filter the results (in the example, that *mono mass* threshold was set to *1*. The results of such a calculation are displayed in figure FIGURE 3.5, “ MONOMER MASS DIFFERENCES ” .

---

[1] Allowing more than one letter to craft monomer codes might seem trivial at first. But that design decision triggered the requirement for non-trivial algorithms throughout all the code of the of program. this is easily understandable at least in the polymer sequence editor: how are monomer codes keyed-in if “a” and “ala” are valid monomer codes in a polymer chemistry definition? the magic is described in the chapter about XpertEdit,



Isoleucine - Leucine = 0
Asparagine - Leucine = 0.958863
Leucine - Isoleucine = 0
Asparagine - Isoleucine = 0.958863
Glutamate - Lysine = 0.94763
Aspartate - Asparagine = 0.984016
Lysine - Glutamine = 0.0363856
Glutamate - Glutamine = 0.984016

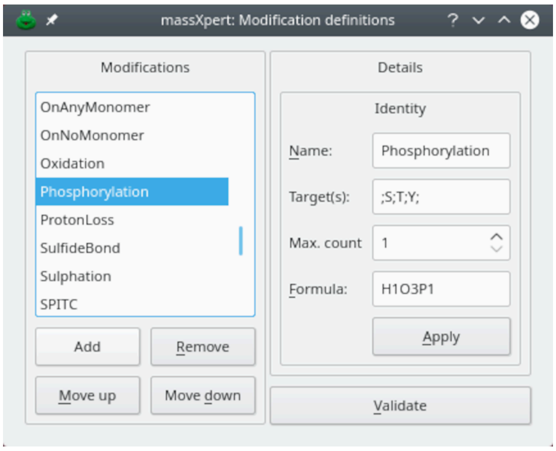
The mass difference between any two monomers in the definition is computed and displayed only if it is less or equal to a threshold (see figure FIGURE 3.4, " MONOMER DEFINITION "

FIGURE 3.5: **MONOMER MASS DIFFERENCES**

After addition of the monomers it is always a good idea to validate them by clicking *validate* .

## 3.2.2 THE MODIFICATIONS

Polymers are often either chemically or biochemically modified. in nature, biopolymers are modified more often than not. Some of the more common modifications in the protein reign are phosphorylation, acetylation and methylation, for example. Nucleic acids are modified with a sheer number of chemical modifications, saccharides also. The MassXpert software provides entire freedom to define any number of intelligent modifications, that is, modifications with any chemical formula but also that are knowledgeable of what monomers they can modify. indeed, it would make no sense to phosphorylate a glycyl residue in a protein, for example.



OnAnyMonomer
OnNoMonomer
Oxidation
Phosphorylation
ProtonLoss
SulfideBond
Sulphation
SPITC

Name: Phosphorylation
Target(s): ;S;T;Y;
Max. count 1
Formula: H1O3P1

Each modification is defined with a name, targets, a count number and a chemical formula.

FIGURE 3.6: **MODIFICATION DEFINITION**

Click *modifications* to open the dialog window shown in figure FIGURE 3.6, " MODIFICATION DEFINITION " . In the example shown, the *phosphorylation* modification is being defined. a modification is defined by:

- The name of the modification: *name*;
- A ";"-separated list of codes of the monomers that might be modified by this modification: *targets*;

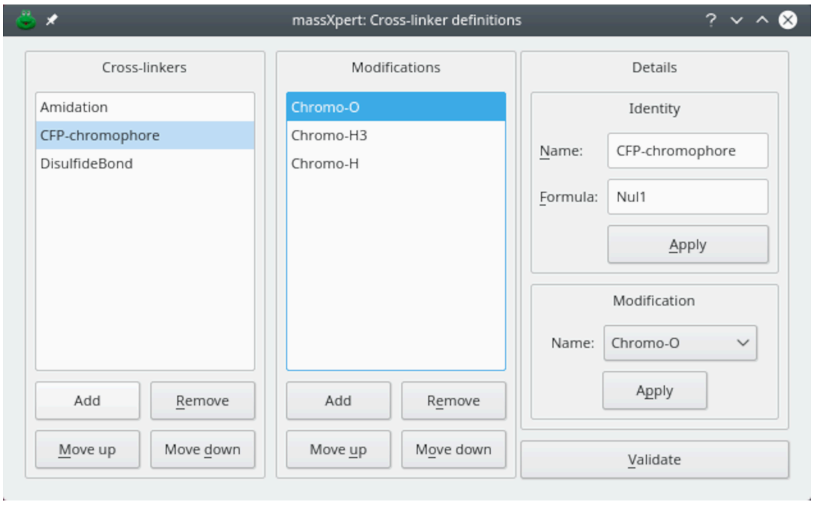
- The maximum number a given monomer might be modified with this modification (*max. count*). This feature is essential when working on methylation of proteins, for example, with arginyl and lysyl residues being multi-methylated;
- The formula that defines the modification chemical reaction, as explained in SECTION 1.1, " ON CHEMICAL FORMULÆ AND CHEMICAL REACTIONS " (*formula*). Note that, in the example of the figure, for *phosphorylation*, the formula is a net formula. That formula could be more explicit by entering *-h+h2po3*. The net formula is thus the one visible on the figure.

The *phosphorylation* reaction can thus be read like this: —"the polymer looses a proton and gains h2po3". the *phosphorylation* is being defined as having *s;t;y* targets only, that means that when the user will try to modify non-seryl or non-threonyl or non-tyrosinyl monomers, the program will complain that these monomers are not targets of *phosphorylation*. there is, however, and for maximum flexibility, the possibility to override these target-limiting data when modifying monomers. When a monomer is modified with this modification, its masses will change according to the net mass of this *phosphorylation* "reaction".

### 3.2.3 THE CROSS-LINKERS

Polymers are often either chemically or biochemically modified by interconnecting monomers from the same polymer sequence. In the protein reign, one classical example of intra-sequence cross-linking is the formation of disulfide bonds. Another wonderful example is the formation of the fluorophore in the fluorescent proteins: there is a chemical reaction involving the side chains of three consecutive residues going on, resulting in the formation of a complex intra-sequence cross-link. each side chain of the three monomers involved undergoes a chemical modification.

Cross-linkers are defined in the dialog window shown in figure FIGURE 3.7, " CROSS-LINKER DEFINITION " . This dialog window is opened by clicking *crosslinkers*.



Name: CFP-chromophore
Formula: Nul1

Name: Chromo-O

each cross-linker is defined using a name, a formula and either no modification or as many modifications as there are monomers involved in the formation of the cross-link.

FIGURE 3.7: **CROSS-LINKER DEFINITION**

The formation of cross-link between one or more monomers often involves chemical reactions to occur at the level of the engaged monomers. Cross-linkers defined in MassXpert should refer to these modifications as modification objects already available in the polymer chemistry definition. Note that, in some cases, it is not necessary to define modifications to occur at the level of the cross-linked monomers.

## Note

When a cross-link does not involve any specific modification, as defined in the polymer chemistry definition, then a chemical formula must be entered in the *formula* edit box widget, otherwise the cross-link definition will have no effect. In the figure example, the *cfp-chromophore* cross-linker is +*nul*, that is there is no chemical reaction defined for the cross-linker *per se*.

The example described in figure Figure 3.7, “ Cross-linker definition ” , corresponds to the cross-linking reaction involved in the formation of the chromophore of the cyan fluorescent protein. That reaction involves the three following monomers: $^{65}$threonyl $^{66}$tyrosinyl $^{67}$glycyl. Each monomer undergoes a distinct chemical modification: “-o”, “-h3” and “-h”, respectively. Three modifications were thus defined: *chromo-o*, *chromo-h3* and *chromo-h*, in that specific order, as these modifications are going to be sequentially applied to their corresponding monomer in the cross-linking reaction.

### Warning

When multiple modifications are used, the number of these modifications must match the number of monomers involved, and their order must match the order with which the monomers are cross-linked. if no modification is defined, then, the chemical reaction that occurs upon cross-linking might be defined in the *formula* of the cross-linker.

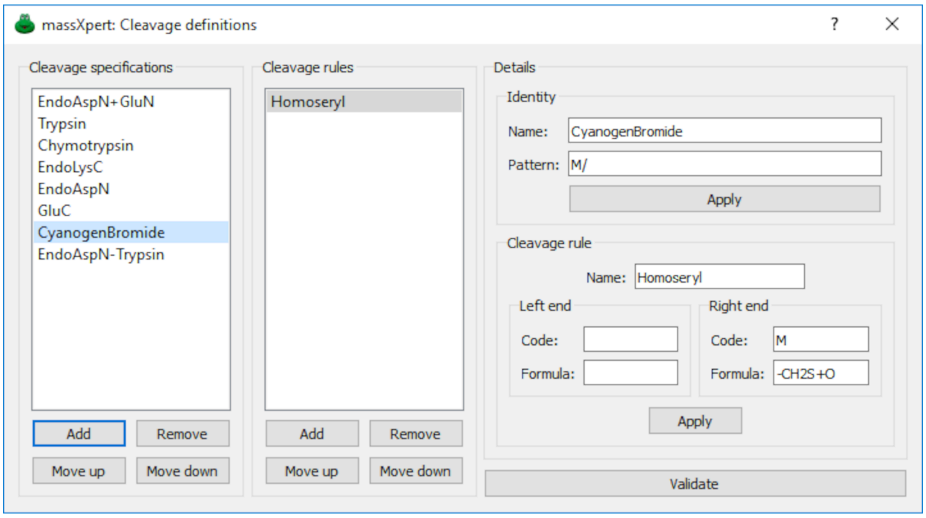
## 3.2.4 The cleavage specifications

It is common practice—in biopolymer chemistry, at least—to cut a polymer into pieces using molecular scissors like the following:

- Proteases, for proteins;
- Nucleases, for nucleic acids;
- Glycosidases, for saccharides…

### Note

Not only biological scissors can be defined, but also chemical ones, like cyanogen bromide, for example, that cleaves at a methionyl residue. MassXpert allows the user to define such kind of chemical scissor.



massXpert: Cleavage definitions

Cleavage specifications

EndoAspN+GluN
Trypsin
Chymotrypsin
EndoLysC
EndoAspN
GluC
CyanogenBromide
EndoAspN-Trypsin

Cleavage rules

Homoseryl

Details

Identity

Name: CyanogenBromide
Pattern: M/

Cleavage rule

Name: Homoseryl

Left end
Code:
Formula:

Right end
Code: M
Formula: -CH2S+O

Each cleavage specification is defined using a name, a cleavage pattern and any number of cleavage rules.

FIGURE 3.8: **CLEAVAGE SPECIFICATION DEFINITION**

For each different polymer type, the molecular scissors are specific. indeed, a protease will not cleave a polysaccharide. This is why cleavage specifications belong to polymer chemistry definitions. In the example of figure FIGURE 3.8, " CLEAVAGE SPECIFICATION DEFINITION " , the definition of the *cyanogenbromide* cleavage specification is detailed (this organic reagent cleaves right of methionyl residues). The *cyanogenbromide* cleavage specification is qualified as so:

- *cyanogenbromide*: the name of the cleavage agent;
- *m/*: the sequence pattern recognized by the cleaving agent. In this case, the cleavage agent cleaves the protein right after *m* residues;
- The *cleavage rule* groupbox allows the user to define the cleavage rules that might be added to the cleavage specification. The case of the cyanogen bromide reagent is interesting in this regard:
  - *left code* and *left formula* are two line edit widgets for the special cases of cleaving agents that not only cut a polymer sequence (usually it is a hydrolysis) but that also modify the substrate in such a way that must be taken into account by MassXpert so that it computes correct molecular masses for the resulting oligomers. These rules are optional. However, if *left code* is filled with something, then it is compulsory that *left formula* be filled with something valid also, and conversely. The cyanogen bromide/protein reaction does not involve any chemical modification (apart from the cleavage) of the monomer left of the generated peptide, so these edit wigets are left blank.
  - *right code* and *right formula* *m* and *-ch2s+o3*, respectively. Same explanation as above. this cleavage rule stipulates that upon cleavage of a protein using cyanogen bromide, the methionyl residue that gets effectively cleaved must be converted to a homoseryl residue. see below for a detailed explanation.

Here are some examples of more complex cleavage patterns:

- *trypsin = k/;r/;-k/p*: "trypsin cuts right of a 'k' and right of a 'r'. But it does not cut right of a 'k' if this k is immediately followed by a p";
- *endoaspn = /d*: "endoaspn cuts left of a d";
- *hypothetical = t/ys; pgt/hyt; /mnop; -k/mnop*: "hypothetical cuts after 't' if it is followed by ys and also cuts after 't' if preceded by pg and followed by hyt. Also, hypothetical cuts prior to 'm' if 'm' is followed by nop and if 'm' is not preceded by 'k'".

## Note

Please, *do* note that the letters in the examples above correspond to monomer codes and *not* to monomer names. If, for example, we were defining a “trypsin” cleavage specification pattern—in a protein polymer chemistry definition with the standard 3-character monomer codes—we would have defined it this way: “trypsin = lys/;arg/;-lys/pro”.

Now comes the time to explain in more detail what the *left code* and *left formula* (along with the *right* siblings) are for. For this, we shall consider that we have the following polymer sequence (1-character monomer codes): “thismwillmbecutmandthatmalso”. If that sequence had been cleaved using cyanogen bromide and if the cleavage had been total,[2] that would have generated the following oligomers: “thism willm becutm andthatm also”. but if there had been partial cleavages, one or more of the following oligomers would have been generated: “thismwillm becutmandthatm also willmbecutm andthatmalso” and so on…

Now, the biochemist knows that when a protein is cleaved with cyanogen bromide, the cleavage occurs effectively right of monomer “m” (this we knew already) *and* the “m” monomer that underwent the cleavage is changed from a methionyl residue to an homoseryl residue (this chemical change involves this formula: “-ch2s+o”). Amongst all the oligomers generated above, there are two oligomers that should not undergo the cleavage rule “-ch2s+o”: “also” and “andthatmalso”. indeed, these two oligomers were generated by the “cyanogenbromide” cleavage, but were not actually cleaved at the right side of a methionyl residue, because they correspond the the right end terminal part of the protein sequence (even if one them does contain a “m” residue; the cleavage did not *occur* at that residue).

This example should clarify why the definition clearly stipulates—in the cleavage specification for “cyanogenbromide”—that the oligomers resulting from this cleavage should “undergo the ‘-ch2s+o’ formula only if they have a ‘m’ as their right end monomer code”. These *cleavage rule*s need to be defined in a very careful way: imagine that—in some experiments involving cyanogen bromide —that reagent would cleave right of “c” (cysteine) residues, but with no chemical modification of the “c” monomer[3] . In this case, it would be suitable to put the flexibility of MassXpert at work by specifying that the generated oligomers should ``undergo the “-ch2s+o” formula'' only if they have a “m” as their right end monomer, so that “c”-terminated oligomers are not chemically modified. Thus the cleavage pattern might be safely defined: “m/;c/”…

---

[2] Cleavage occurs at every possible position, right of each monomer “m”.

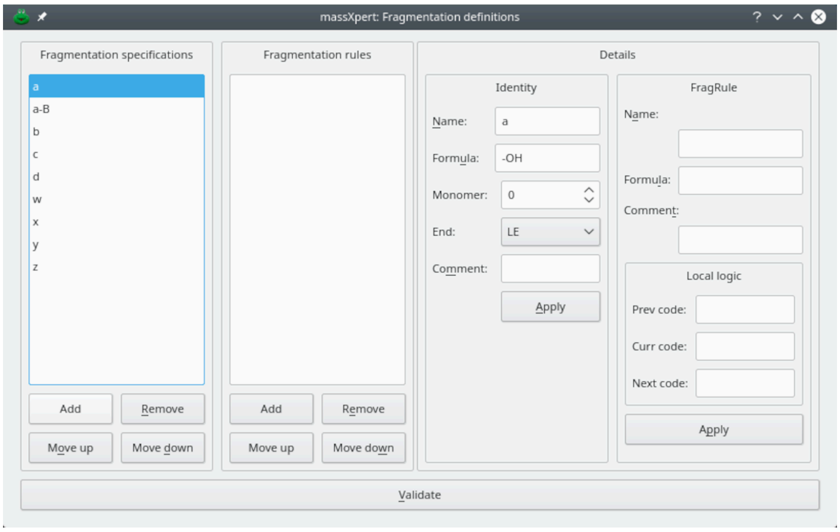
[3] This is a purely hypothetical situation that i never observed personally.

### 3.2.5 THE FRAGMENTATION SPECIFICATIONS

The specification of fragmentation events in a polymer chemistry definition is not a trivial task. In this section three different cases will be described, from simple to more complex. One major rule is that the fragmentation specification should be crafted in such a manner that the resulting fragment is neutral. The ionization of the fragments will be then automatically performed by MassXpert upon calculation of the ion products according to the current ionization rule. This is a major improvement over previous versions, that forced the user to define fragmentation specifications by assuming a product ion of a given ionization ([m+h]$^{+}$ for proteins or [m-h]$^{-}$ for nucleic acids, for example).

#### 3.2.5.1 SIMPLE FRAGMENTATION PATTERNS

One simple example of polymer chain fragmentation is the formation of *a* fragments with a nucleic acid (dna, in this example). Bond cleavage occurs right before the sugar-carbon-linked oxygen of the phosphoester bond linking one deoxyribonucleotide to the next. Thus, the molecular weight of the fragment corresponds to the sum of the monomer masses from the left end of the polymer up to and including the monomer being decomposed *less* one oxygen. note that this specification yields a nucleic acid ion product that is protonated. we thus need to remove a proton to change its charge to 0, thus the formula of the *a* fragmentation pattern is "-oh" this is illustrated in figure FIGURE 3.9, " FRAGMENTATION RULE DEFINITION " : the name of the specification for fragmentation pattern "a" is *a*; the formula associated to this fragmentation pattern is *-oh*; the fragment encompasses the *le* (for "left end") of the polymer chain; the *monomer* value is set to *0*, which will be explained later.



a
a-B
b
c
d
w
x
y
z

Name: a
Formula: -OH
Monomer: 0
End: LE

Each fragmentation rule is defined using a name, a formula and a local logic, that is a set of logical conditions which must be verified for the fragmentation specification to be applied to the fragment.

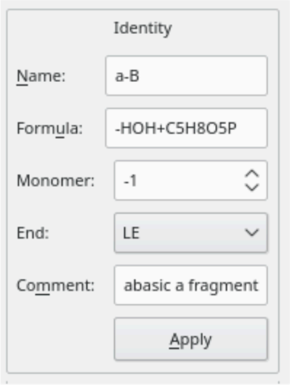
FIGURE 3.9: **FRAGMENTATION RULE DEFINITION**

### 3.2.5.2 MORE COMPLEX FRAGMENTATION PATTERNS

In nucleic acids gas-phase chemistry, it often happens that not only fragmentation occurs at the level of the phospho-ribose skeleton, but also at the level of the nucleic base. These fragmentation patterns are called abasic patterns. The decomposition of the base occurs at the monomer position where the fragmentation occurs. For example, if a “atgc” oligonucleotide is fragmented according to pattern *a* but with nucleic base decomposition, and that fragmentation occurs at position 1, then the computation of the mass should occur like represented in figure FIGURE 3.10, “ FRAGMENTATION DEFINITION WITH GENERIC SPECIFICATION ” . This figure illustrates a number of things, amongst which some known basics. the top left panel show what the configuration would be in the fragmentation definition window for this kind of fragmentation. The top right panel shows the basic constituents of the dna polymer chemistry definition: the caps are *oh* on the left end and *h* on the right end; the circled formula is the skeleton (also called backbone) and the base attached to the deoxyribose ring identifies the nucleotide. that base might be adenine, guanine, cytosine, thymine. in the “dna” polymer chemistry definition, the monomers are made of the skeleton (formula c5h8o5p) plus the formula of the base, which is understandable. The following paragraphs detail two ways to configure a base-loss fragmentation pattern.

**Using a monomer-generic specification.** Now, if we want to compute the mass of the a-b#1 fragment, that is fragmentation occurs according to pattern *a* right after the “a” monomer *plus* decomposition of the base (in our case this is an adenine, see figure FIGURE 3.10, “ FRAGMENTATION DEFINITION WITH GENERIC SPECIFICATION ” ). note that the decomposition of the base is accompanied by the formation of an insaturation on the sugar moiety of which the net formula is -h. We thus have to:

- Apply an adapted specification for *a* fragments: removal of the h due to the insaturation on the sugar and also removal of the oh related to the formation of an *a* fragment: the *-hoh* component of the formula;
- Remove one *full monomer* with *monomer* set to *-1* (this equals to the removal of both the skeleton and the side chain—the adenine, here);
- Add back the skeleton: the *+c5h8o5p* component of the formula;
- Add the left end cap, since *a* fragments start at the left *end* of the fragmented polymer sequence: *le* (“left end”). That *le* bit of information will be transformed into a +oh formula, since this is the formula of the left end cap for nucleic acids.



Identity

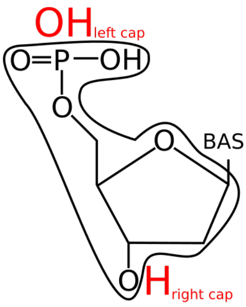
Name: a-B

Formula: -HOH+C5H8O5P

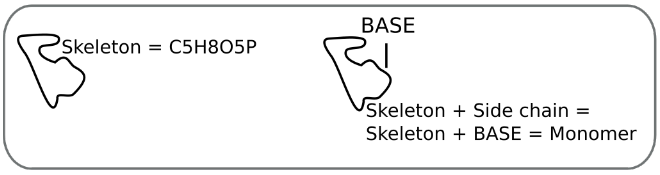
Monomer: -1

End: LE

Comment: abasic a fragment



```
Adenine:  C5H4N5   (monomer: C10H12N5O5P) mono: 313.058
Guanine:  C5H4N5O  (monomer: C10H12N5O6P) mono: 329.053
Cytosine: C4H4N3O  (monomer: C9H12O6N3P)  mono: 289.040
Thymine:  C5H5N2O2 (monomer: C10H13N2O7P) mono: 304.050
```



```
a#1= monomer - OH (gives neutral fragment)
             + left cap =
     monomer - OH + OH = 313 - 17 + 17 = 313

a-B#1= monomer - OH2 (gives neutral fragment)
               + left cap
               - 1x(side chain) =
       monomer - OH2 + OH - monomer + skeleton =
       313 -18 + 17 - 313 + C5H8O5P =
       -1 + 179 = 178 (This is the mass of the neutral fragment)
```

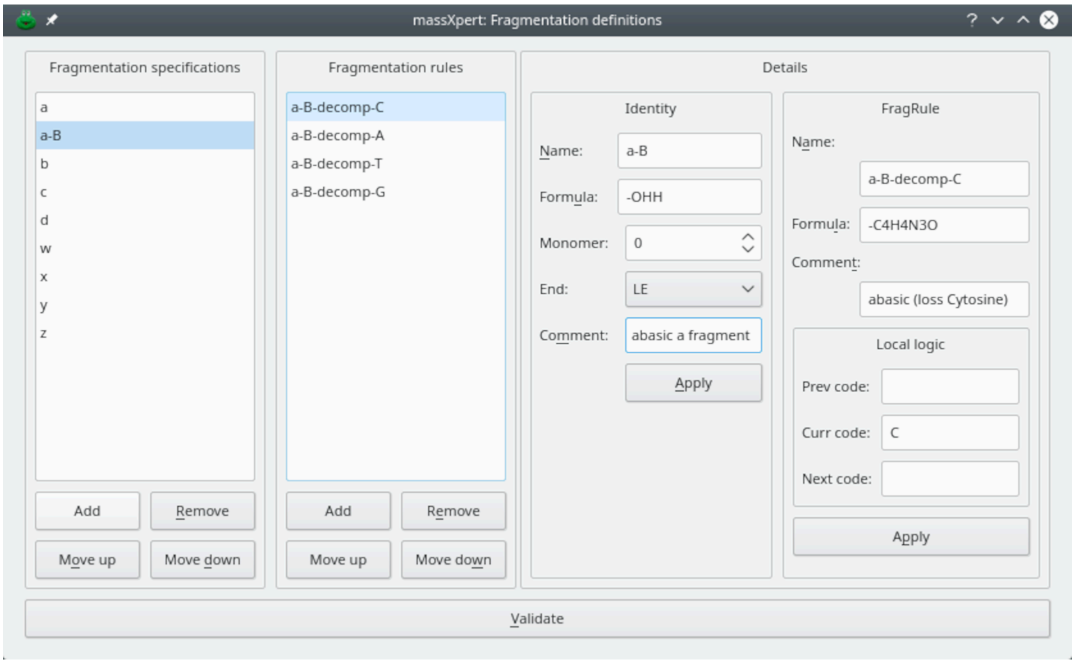
Fragmentation patterns that involve the decomposition of the nucleic base need specific configuration adjustments. Here, the removal of the nucleic base is done by first removing the whole monomer and then readding the skeleteon.

FIGURE 3.10: **FRAGMENTATION DEFINITION WITH GENERIC SPECIFICATION**

The advantage of working this way is that we need not specify a fragmentation rule for each different monomer in the sequence (see below, for how this might be done). Indeed, by specifying *monomer* to be *-1*, we indicate —without knowing the monomer identity—to the mass calculation engine that once the fragmentation has occurred in the polymer chain, the mass of the monomer that got fragmented should be subtracted from the fragment mass. That subtraction removes, however too much material, as we do not want to loose the skeleton, we only want to loose the base (adenine, in our example). This is why we ask in the fragmentation specification formula that the skeleton be added (the *+c5h8o5p* component of the formula). Because the skeleton does not change along the polymer chain, even if the base itself changes, this computation method is generalizable, and because of this the polymer chemistry definition works.

This whole process of defining a fragmentation pattern that needs to “know” what monomer is being fragmented so as to compute the fragment masses correctly, can be performed by using fragmentation rules. This is described below.

**Using a monomer-specific specification.** Another way of achieving what was described above is by using fragmentation rules, whereby the fragment's mass computation is made conditional to one or more conditions that should be verified. figure FIGURE 3.11, “ FRAGMENTATION DEFINITION WITH SPECIFIC RULES ” shows how the *a-b* fragmentation pattern might be defined using fragmentation rules.



Fragmentation specifications: a, a-B, b, c, d, w, x, y, z

Fragmentation rules: a-B-decomp-C, a-B-decomp-A, a-B-decomp-T, a-B-decomp-G

Identity

Name: a-B

Formula: -OHH

Monomer: 0

End: LE

Comment: abasic a fragment

FragRule

Name: a-B-decomp-C

Formula: -C4H4N3O

Comment: abasic (loss Cytosine)

Local logic

Prev code:

Curr code: C

Next code:

The fragmentation for pattern *a* with decomposition of the nucleic base at the location of the nuclotide undergoing decomposition is defined using a name, a formula and a *“local logic”*, that is, a set of logical conditions which must be verified for the fragmentation rule(s) to be applied to the fragment.

FIGURE 3.11: **FRAGMENTATION DEFINITION WITH SPECIFIC RULES**

The *a-b* fragmentation specification comprises 4 rules, one rule for each available monomer in the "dna" polymer chemistry definition: "a", "t", "g" and "c". The figures illustrates the definition of the fragmentation specification *a-b* which stipulates that the mass of the fragment should be computed this way:

- For the fragmentation specification part, everything is like for fragments of type *a*, that is, the formula is merely *-ohh* and the end is *le* (see above, for explanations);
- But there is one rule (*a-b-decomp-c*) which adds some *local logic* for the fragmentation specification: the formula "*-c4h4n3o*" should be applied upon calculation of the fragment's masses if the monomer at which the fragmentation actually occurs (*curr code*) is *c*, *i.e.* , if it is a cytosine. The "*-c4h4n3o*" formula is the formula of cytosine (the nucleic base, *not* the monomer).
- The other rules (for *curr code* values *a*, *t* and *g* are identical to the *a-b-decomp-c* one unless the *curr code* is "a", "t" or "g" and the formula to be removed is the formula of the corresponding dna base.

The fragmentation rule-based definition of fragmentation pattern *a-b* yields identical results as for the more generalizable method described earlier ( USING A MONOMER-GENERIC SPECIFICATION ).

### 3.2.5.3 EVEN MORE COMPLEX FRAGMENTATION PATTERNS

Note that in saccharide chemistry, the fragmentation patterns are extremely complex, and often totally depend on the nature of the monomers local to the fragmentation site. For example, the fragmentation behaviour at position "e" in a sequence "dear" might be different than in a sequence "dera". MassXpert had to be able to model these complex situations, and this is done using fragmentation rules where the local logic involves defining the *prev code* and/or the *next code* for a given *curr code* at which the fragmentation occurs. For example, one specific fragmentation pattern for fragmentation at "e" in sequence "dear" might be defined this way:

- *prev code*: *d*;
- *curr code*: *e*;
- *next code*: *a*.

Instead of that fragmentation rule, one would have for fragmentation at "e" in sequence "dera" the following rule:

- *prev code*: *d*;
- *curr code*: *e*;
- *next code*: *r*.

Note the change for *next code*, from *a* to *r*. Also, be aware that the "prev", "curr" and "next" notions are polar, that is, they depend on the value of *end* (that is *le* or *re*). For example, if we wanted to model the fragmentation pattern at "e" for a fragment of *end re*, similar to what was done above with sequences "dear" and "dera", we would have set the local logical like this:

For sequence "dear":

- *prev code*: *a*;
- *curr code*: *e*;
- *next code*: *d*.

For sequence "dera":

- *prev code*: *r*;
- *curr code*: *e*;
- *next code*: *d*.

This highly flexible fragmentation specification allows for definition of highly complex fragmentation behaviours of biopolymers.

## 3.3 SAVING THE DEFINITION

Once the polymer chemistry definition is completed, the user can save it to an xml file. Prior to actually writing to the file, the program checks the validity of all the chemical entities in the definition. this check can be triggered manually by clicking the *validate*. If an error is found, it is reported so that the user may identify the problem and fix it.

The location where the file should be saved, and the manner that it may be made available to MassXpert is to be described in a later chapter. It is, in fact, very important that MassXpert knows where to find newly defined polymer chemistries so as to be able to use them when sequences of that polymer chemistry are created or used.

# 4 XPERTEDIT: A POWERFUL EDITOR AND SIMULATION CENTER

After having completed this chapter you will be able to perform sophisticated polymer chemistry simulations on polymer sequences—that can be edited in place—along with automatic mass recalculations.

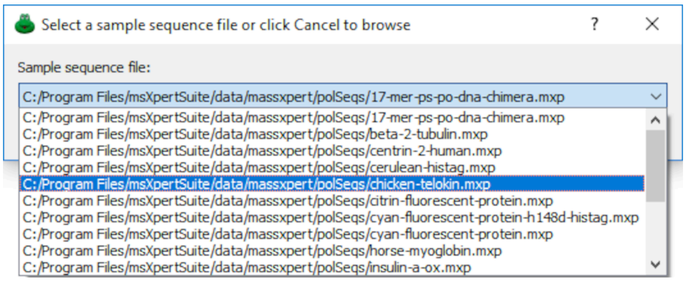
## 4.1 XPERTEDIT INVOCATION

The XpertEdit module is easily called by pulling down the *XpertEdit* menu item from the MassXpert program's menu. The user may start the XpertEdit module by:

- Opening a sample polymer sequence;
- Creating a new polymer sequence;
- Loading a polymer sequence from disk.

## 4.2 XPERTEDIT OPERATION: *In Medias Res*

The first manner to start an XpertEdit session is by opening a sample sequence out of the list of sequences that were shipped along with MassXpert. The Open Sample Sequence menu item from the *XpertEdit* menu opens the dialog box shown in FIGURE 4.1, " SELECTION OF A SAMPLE POLYMER SEQUENCE " . The drop-down widget in this dialog window lists all the polymer sequence files that were shipped along with MassXpert. Simply select one item and click *OK*. To select another polymer sequence file, click *Cancel*, which will trigger the system's file selection dialog for you to browse to the location where the polymer sequence file is stored. The process is identical to the normal polymer sequence file opening (see below).



Sample sequence file: C:/Program Files/msXpertSuite/data/massxpert/polSeqs/17-mer-ps-po-dna-chimera.mxp

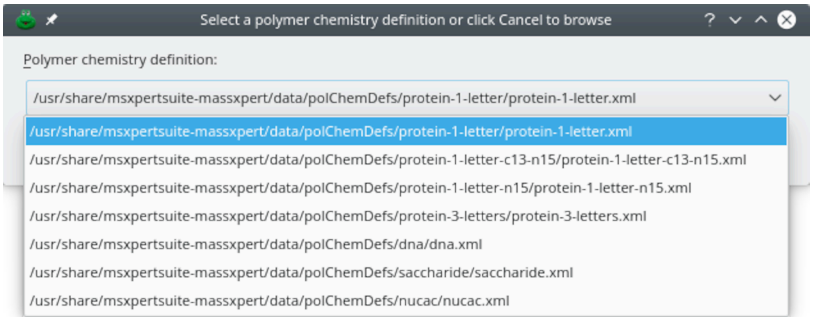
```
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/17-mer-ps-po-dna-chimera.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/beta-2-tubulin.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/centrin-2-human.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/cerulean-histag.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/chicken-telokin.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/citrin-fluorescent-protein.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/cyan-fluorescent-protein-h148d-histag.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/cyan-fluorescent-protein.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/horse-myoglobin.mxp
C:/Program Files/msXpertSuite/data/massxpert/polSeqs/insulin-a-ox.mxp
```

MassXpert ships with a number of sample polymer sequences which are designed to allow an easy demonstration of the XpertEdit features. This selection dialog lists all the polymer sequence files that were shipped along with MassXpert.

FIGURE 4.1: **SELECTION OF A SAMPLE POLYMER SEQUENCE**

The second way to start an XpertEdit session is by creating a new polymer sequence ( New Sequence menu item from the *XpertEdit* menu). The program immediately asks to select a polymer chemistry definition, as shown in Figure 4.2, “ Selection of the polymer chemistry definition ” . The drop-down widget lists all the polymer chemistry definitions currently registered on the system. If the polymer chemistry definition is not listed, clicking onto *Cancel* will let the user browse the disk in search for a polymer chemistry definition file [1] . Once the polymer chemistry definition has been selected and successfully parsed by the program, the user is presented with an empty sequence editor.

The third way to start an XpertEdit session is by opening an existing polymer sequence file. Once the sequence file has been opened, the user is presented with a sequence editor as represented in Figure 4.3, “ The XpertEdit main window ” . At this point, when the user starts editing a sequence, the characters entered at the keyboard, or pasted from the clipboard, will be interpreted using the polymer chemistry definition that was selected in the initialization window described above.



Select a polymer chemistry definition or click Cancel to browse

Polymer chemistry definition:

/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter/protein-1-letter.xml

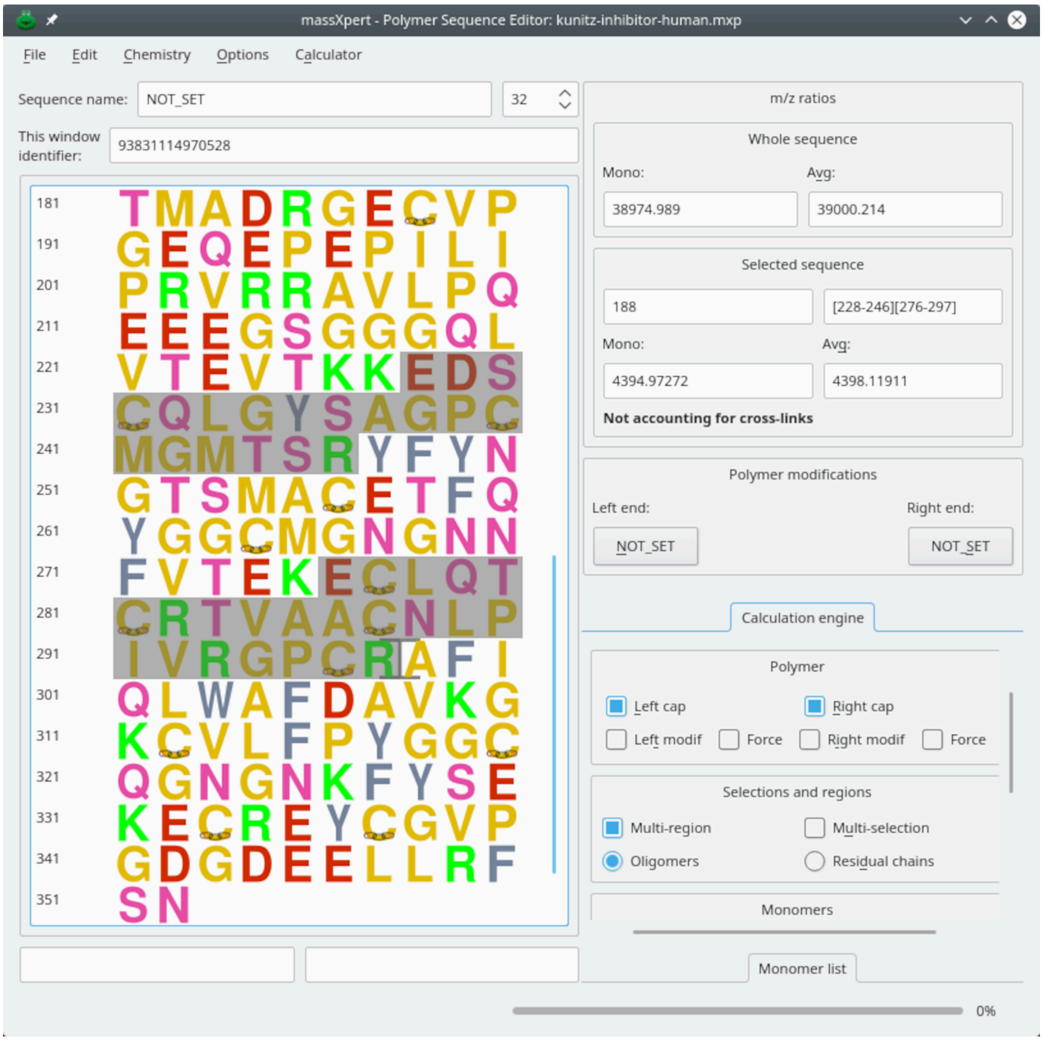
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter/protein-1-letter.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter-c13-n15/protein-1-letter-c13-n15.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter-n15/protein-1-letter-n15.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-3-letters/protein-3-letters.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/dna/dna.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/saccharide/saccharide.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/nucac/nucac.xml

When creating a new polymer sequence, it is necessary to first indicate of what polymer chemistry definition the polymer sequence will be. This window lists all the polymer chemistry definitions currently available on the system.

**FIGURE 4.2: SELECTION OF THE POLYMER CHEMISTRY DEFINITION**

---

[1] Once the sequence is saved, the polymer chemistry definition file *must* be registered or the sequence file will not be loadable. This is described in a later chapter.



Sequence name: NOT_SET 32

This window identifier: 93831114970528

181 TMADRGECVP
191 GEQEPEPILI
201 PRVRRAVLPQ
211 EEEGSGGGQL
221 VTEVTKKEDS
231 CQLGYSAGPC
241 MGMTSRYFYN
251 GTSMACETFQ
261 YGGCMGNGNN
271 FVTEKECLQT
281 CRTVAACNLP
291 IVRGPCRAFI
301 QLWAFDAVKG
311 KCVLFPYGGC
321 QGNGNKFYSE
331 KECREYCGVP
341 GDGDEELLRF
351 SN

Whole sequence: Mono: 38974.989 Avg: 39000.214

Selected sequence: 188 [228-246][276-297] Mono: 4394.97272 Avg: 4398.11911

**Not accounting for cross-links**

Polymer modifications: Left end: NOT_SET Right end: NOT_SET

Polymer: ☑ Left cap ☑ Right cap ☐ Left modif ☐ Force ☐ Right modif ☐ Force

Selections and regions: ☑ Multi-region ☐ Multi-selection ● Oligomers ○ Residual chains

This figure shows a polymer sequence displayed in an XpertEdit editor window.

FIGURE 4.3: THE XPERTEDIT MAIN WINDOW

Now, of course, editing a polymer sequence is not enough for a mass spectrometric-oriented software suite; what we want is *compute masses!* The mass calculation process is immediately visible on the right hand side of the sequence editor shown in FIGURE 4.3, " THE XPERTEDIT MAIN WINDOW " . The *m/z ratios* frame box widget contains two items:

- *Whole Sequence* A frame box widget displaying the *Mono* and *Avg* masses of the whole polymer sequence, irrespective of the current selection;
- *Selected Sequence* A frame box widget displaying the *Mono* and *Avg* masses of the currently selected region of the polymer sequence.

The user may change the mass calculation engine configuration at any point in time using the widgets in the *Calculation Engine* tool box that contains the following configurable parameters:

- *Polymer*

  - If *Left Cap* is checked, the left cap of the polymer sequence will be taken into account;
  - If *Right Cap* is checked, the right cap of the polymer sequence will be taken into account;
  - If *Left Modif* is checked, the modification of the polymer sequence's left end will be taken into account. Note that if *Force* is checked also, then the modification is taken into account even when selecting a region of the sequence that does not encompass the monomer located at that end;
  - The behaviour of *Right Modif* is the same as above, but for the right end modification;
- Selections and regions
  - If *Multi-region* is checked, the sequence editor allows more than one region to be selected at any given time (there is no limitation whatsoever on the number of selected regions);
  - If *Multi-selection* is checked, the sequence editor allows not only the selection of multiple regions at any given time, but also the selection of totally or partially overlapping regions.
  - When multiple regions are selected, each selected region behaves like an oligomer if *Oligomers* is selected; that is, it gets its left and right end caps added (if the corresponding calculation engine configuration item is activated);
  - When multiple regions are selected, the different regions behave like residual chains if *Residual chains* is selected; that is, the left and end caps are added only once (if the corresponding calculation engine configuration item is activated).
- Monomers
  - If *Modifications* is checked, the monomer modifications will be taken into account;
  - If *Cross-links* is checked, the cross-links in the polymer sequence will be taken into account.

### WARNING

Only cross-links fully encompassed by the selected sequence region(s) will be taken into account for the *Selected sequence* mass calculations. If any number of cross-links are not fully encompassed by the currently selected sequence region, then that number is displayed along with the following label visible in the *Selected sequence* group box : *Incomplete cross-links:*.

- Ionization

- The $+H$ formula represents the ionization agent formula (that is, a protonation);
- The *Unitary charge* value is set to *1* because, in the example, a protonation brings a single positive charge;
- The *Ionization level* set to *1*, that is, in the example, the polymer must undergo a single protonation.

When any parameter listed above is changed, the recalculation of the masses—for both the *Whole sequence* and the *Selected sequence*—is triggered and the new masses are updated in their respective line edit widgets, described earlier. The fact that the user can specify ionization rules should make it clear that the values that are displayed are actually m/z ratios (as long as one ionization is required).

## 4.3 THE EDITOR WINDOW MENU

The menu bar in the polymer sequence editor displays a number of menu items, reviewed below:

- *File*
    - *File*→ Close : close the sequence;
    - *File*→ Save : save the sequence under a new file name. If the sequence has no filename yet, the user is invited to select a filename;
    - *File*→ Save As : save the sequence in a new file;
    - *File*→ Import Raw : try to import the sequence. If invalid monomer code characters are found, the user is given a chance to revise the imported sequence;
    - *File*→ Export to Clipboard : copy the sequence and all the data (masses and calculation options) to the clipboard, in the form of simple text;
    - *File*→ Export to File : write the sequence to file and all the data (masses and calculation options), in the form of simple text (if a file name was already selected, otherwise the user is invited to select a file into which the data are to be written);
    - *File*→ Select export file : select a file into which the data are to be written.
- Edit
    - *Edit*→ Copy Copies the current selected region(s) (if any) to the clipboard. If there are more than one region currently selection, then the user is informed that the copied sequence will correspond to these two sequences joined together.

**WARNING**

Be aware, that the order in which the region sequences are joined is the order in which the regions were selected, and not the order in which the sequences appears in the whole polymer sequence.

  - *Edit*→ Cut copy the current selection (if any) to the clipboard and removes it from the sequence. Note that it is not yet possible to cut more than one selected region in one single operation;
  - *Edit*→ Paste : paste the sequence from the clipboard into the sequence at the current cursor location. If the pasted sequence is found to contain characters not valid for the current polymer chemistry definition, the user is given a chance to revise the pasted sequence. If one sequence region was selected, it is replaced with the pasted sequence. If more than one sequence region was selected, the operation cannot be performed and the user is informed;
  - *Edit*→ Find Sequence : find a sequence motif in the polymer sequence.

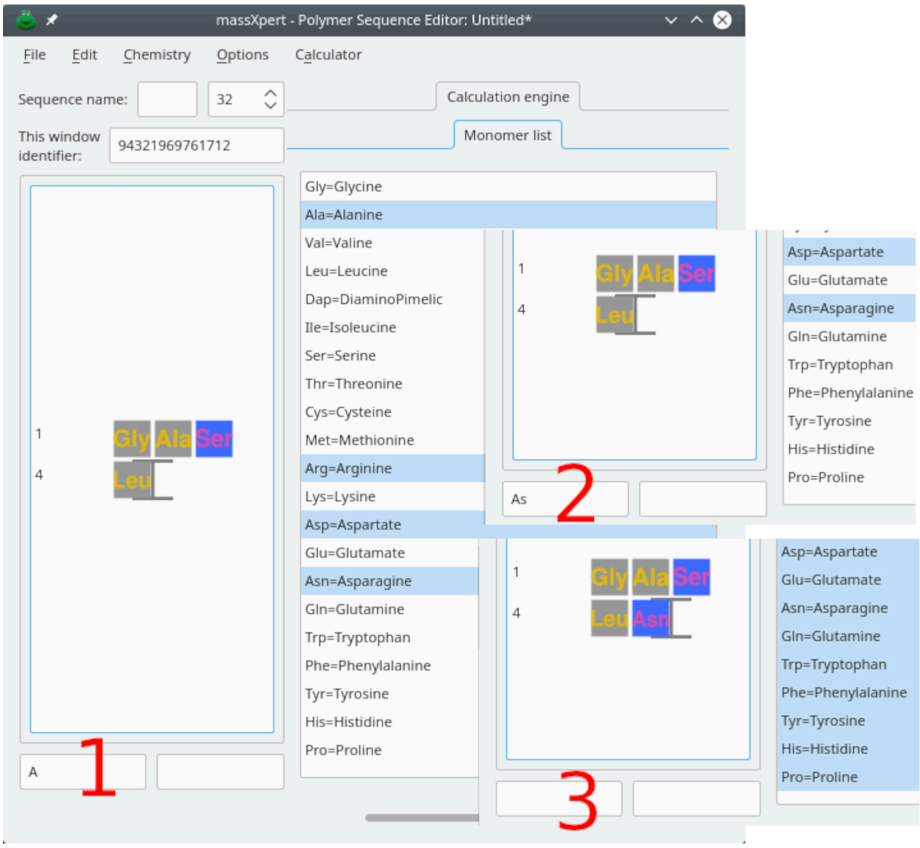
- *Chemistry*
  - *Chemistry*→ Modify Monomer(s) : modify (or unmodify) one or more monomers in the polymer sequence;
  - *Chemistry*→ Modify Polymer : set (or unset) the left (or right, or both) modification(s) of the polymer sequence;
  - *Chemistry*→ Cross-link Monomers : cross-links monomers;
  - *Chemistry*→ Cleave : perform a chemical/enzymatical cleavage of the polymer sequence;
  - *Chemistry*→ Fragment : perform the gas phase fragmentation of the currently selected oligomer;
  - *Chemistry*→ Mass Search : search in an arbitrary manner for any sequence having a mass matching the searched mass;
  - *Chemistry*→ Compute m/z Ratios : calculate a range of m/z ratios with a given ionization agent starting from a given m/z ratio and a given ionization status;

    - *Chemistry*→ Determine Compositions : calculate the monomeric/elemental composition of the whole polymer sequence or of the current selection;
    - *Chemistry*→ pKa pH pI : perform acidity, pH and isoelectric point calculations on the whole sequence or on the current selection.
- *Options*
    - *Options*→ Decimal places : set the number of decimal places to be used to display the numerical values.

## 4.4 EDITING POLYMER SEQUENCES

As described earlier, in the chapter about the XpertDef module, a polymer chemistry definition may allow more than one character to qualify the codes of the monomers (see CHAPTER 3, XPERTDEF: DEFINITION OF POLYMER CHEMISTRIES ). It was noted also that it is not because the number of allowed characters is *3*, for example, that all the monomer codes of the polymer chemistry definition must be defined using three characters: *3* is the *maximum* number of characters that may be used.

### 4.4.1 MULTI-CHARACTER MONOMER CODES



Sequence name: 32

This window identifier: 94321969761712

Gly=Glycine
Ala=Alanine
Val=Valine
Leu=Leucine
Dap=DiaminoPimelic
Ile=Isoleucine
Ser=Serine
Thr=Threonine
Cys=Cysteine
Met=Methionine
Arg=Arginine
Lys=Lysine
Asp=Aspartate
Glu=Glutamate
Asn=Asparagine
Gln=Glutamine
Trp=Tryptophan
Phe=Phenylalanine
Tyr=Tyrosine
His=Histidine
Pro=Proline

A

As

This figure shows the process by which a multi-letter monomer code is entered in the polymer sequence editor.

FIGURE 4.4: MULTI-CHARACTER CODE SEQUENCE EDITING IN XPERTEDIT

This section deals with the editing of a polymer sequence for which monomer codes can be made of more than one character. Figure FIGURE 4.4, " MULTI-CHARACTER CODE SEQUENCE EDITING IN XPERTEDIT " shows the case of a polymer sequence for which the polymer chemistry definition allows three characters to define monomer codes. The example is based on the following real-world situation: the user wants to edit the sequence by insertion —at the end of the sequence (right of the Leu residue). The new monomer to be appended to the sequence is "Asn".

After keying-in `A` (panel 1), no sequence modification is visible in the sequence editor. Instead, an "A" character is now displayed in the left line edit widget under the sequence. The reason of this apparently odd behaviour is that the polymer chemistry definition allows up to 3 characters to describe a monomer code. If no monomer vignette is displayed in the polymer sequence, that means that more than one monomer code start with an "A" character: XpertEdit cannot figure out which monomer code was actually meant by the user when keying-in `A` .

There is a way, called *code completion*, to know which monomer code(s)—in the current polymer chemistry definition—do start with the keyed-in character(s) (currently, "A"). The user can at any moment always enter the *"code completion mode"* by hitting the `Enter` key. This is what is shown in the panel 1st, right hand side

*Monomer List* listview widget (click on that *Monomer List* label to show that list if it is not already visible). We see that, in the current polymer chemistry definition, four monomer codes start with an “A” character, and these are “Ala”, “Arg”, “Asp” and “Asn” (as highlighted in the code completion monomer list).

Because we now know that the code we are to key-in is “Asp”, we key-in a `s` . The result is shown in the small panel 2. What we see here is that, this time also, nothing changed in the polymer sequence. What changed is that the character string in the left line edit widget below the sequence is now “As”. Let's key-in once more the `Enter` key. This time, only two items are highlighted: “Asp” and “Asn” in the code completion monomer list (panel 2nd). This is easily understood: there are only two monomer codes that start with the two letters “A” and “s” (“As”) that we have keyed-in so far. At this time, we key-in a last character: `n` . At this point, the monomer is effectively inserted in the polymer sequence, as the “Asn” monomer left of the cursor, as shown in panel 3. Note how the bottom edit widget is now cleared: the re are no more letters in the buffer awaiting to be completed to form a full monomer code. Also, by entering `Enter` , all the monomer codes are selected in the list of monomer codes available for editing the sequence: since a new monomer code might be entered all possibilities are open.

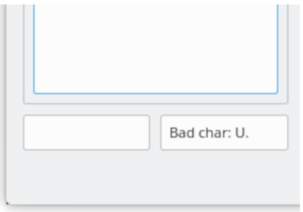
### 4.4.2 UNAMBIGUOUS SINGLE-/MULTI-CHARACTER MONOMER CODES

Let's imagine that we have a polymer chemistry definition that allows up to 3 characters for the definition of monomer codes, but that we have one of these monomer codes (let's say the one for the “Glutamate” monomer) that is one-letter-long: “E”. This monomer code “E” is the only one in the polymer chemistry definition to start with an “E” character. In this case, when we key-in `E` , we'll observe that the monomer code is immediately validated and that its corresponding monomer vignette is also immediately inserted in the polymer sequence. This is because, *if there is no ambiguity*, XpertEdit will immediately validate the code being edited.

The mechanism described above means that the user is absolutely free to define *only single-character monomer codes* in a polymer chemistry definition that allows multi-character monomer codes; the behaviour of the program is thus to behave exactly as if the multi-character code feature were inexistent in the program: each time a new uppercase letter is keyed-in, it is automatically validated and the corresponding monomer is created in the sequence.

### 4.4.3 ERRONEOUS MONOMER CODES

The typing error detection system triggers immediate alerts whenever the code beign keyed-in is incorrect. This is described in FIGURE 4.5, “ BAD CODE CHARACTER IN XPERTEDIT SEQUENCE EDITOR ” . If the user enters an uppercase character not matching any monomer code currently defined in the polymer chemistry definition, or a lowercase character as the first character of a monomer code, the program immediately complains in the right line edit widget below the sequence. In this case, the monomer code is *not* put into the left text widget, which means it is simply ignored.



Bad char: U.

This figure shows the feedback that the user is provided by the code editing engine, when a bad character code is keyed-in.

FIGURE 4.5: **BAD CODE CHARACTER IN XPERTEDIT SEQUENCE EDITOR**

If you start keying-in valid monomer character codes, like for example we did earlier with “As”, and you want to erase these characters because you changed your mind, hit the `Esc` key to remove any character entered previously. These characters will disappear sequentially, at each `Esc` key press, from the line edit widget below the sequence. For example, let's say you have already keyed-in `A` and `s` . In this case the left line edit widget displays these two characters: “As”. Now, if you change your mind, not willing to enter “Asp” monomer code anymore, but “Gly” instead, all you have to do is to key-in `Esc` once for the “s” character (which disappears) and once more to remove the remaining “A” character. At this point it is possible to start fresh with the “Gly” monomer code by keying-in sequentially `G` , `l` and finally `y` .

### WARNING

Do not use the `<–` key to erase erroneously entered characters, you would delete monomers from the sequence!

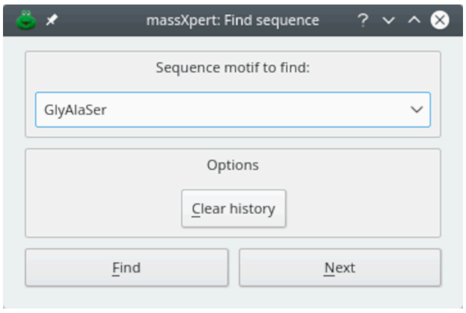
## 4.4.4 SIMPLIFIED EDITING

When the monomer codes of a given polymer chemistry definition are too numerous or too long to remember, one simplified editing strategy is by using the list of available monomers located on the right side of the sequence editor (widget labelled *Monomer list*). The items in the list are active: if double-clicked, an item will see its corresponding monomer code inserted in the sequence at the current cursor location. This list thus makes it easy to “visually” edit the polymer sequence without having to remember all the codes in the polymer chemistry definition.

## 4.5 FINDING SEQUENCE MOTIFS

Finding sequence motifs in the polymer sequence is performed by selecting the *Edit*→ Find Sequence menu item. The dialog window is shown in FIGURE 4.6, " FINDING A SEQUENCE MOTIF IN THE POLYMER SEQUENCE " . When performing the first search in a polymer sequence, the *Find* button should be used. This will trigger a search starting at the beginning of the polymer sequence. For each successive search, the *Next* button should be used.

Each searched sequence motif will be stored in a history list that is made available by dropping down the combo box widget where the sequence motif is entered. The *Clear history* button will erase all the searched sequence motifs from the history, thus resetting it.



GlyAlaSer

The first iteration should be performed by clicking onto the *Find* button, and each following iteration should be performed using the *Next* button.

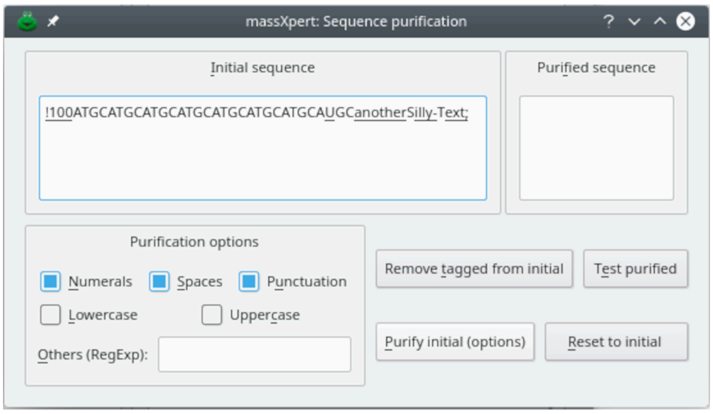
**FIGURE 4.6: FINDING A SEQUENCE MOTIF IN THE POLYMER SEQUENCE**

## 4.6 IMPORTING SEQUENCES

Very often, the user will make a sequence search on the web and be provided with a polymer sequence that is crippled with non-code characters. That web output might either be saved in a text file for future reference or copied to the clipboard for immediate use in MassXpert. The two cases are reviewed below.

### 4.6.1 IMPORTING FROM THE CLIPBOARD

XpertEdit provides a convenient way to spot non-valid characters in a text and to let the user "purify" the imported sequence. A clipboard-imported sequence is systematically parsed. When invalid characters are found, the window depicted in FIGURE 4.7, " CLIPBOARD-IMPORTED SEQUENCE ERROR-CHECKING " is presented to the user for her to make appropriate adjustments (in this example we tried to copy from clipboard the following sequence: "!100 ATGCATGC ATGCATGC ATGCATGC ATGCAUGC anotherSilly-Text;").



!100ATGCATGCATGCATGCATGCATGCATGCAUGCanotherSilly-Text;

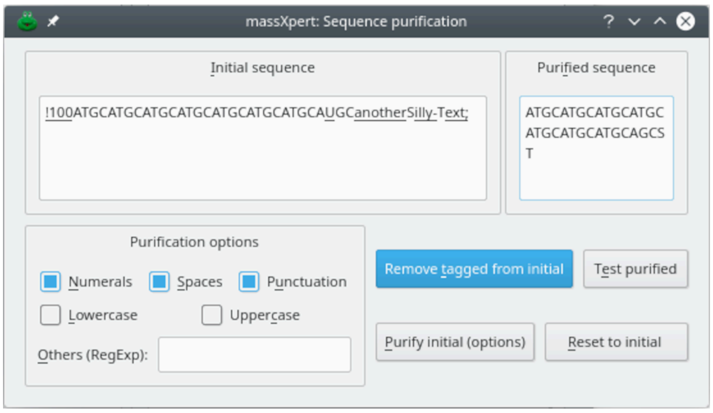
☑ Numerals ☑ Spaces ☑ Punctuation
☐ Lowercase ☐ Uppercase
Others (RegExp):

If a sequence that is imported through the clipboard to the XpertEdit sequence editor contains invalid characters, the user is provided with a facility to “purify” the sequence. This facility is provided to the user through the window depicted in this figure.

FIGURE 4.7: **CLIPBOARD-IMPORTED SEQUENCE ERROR-CHECKING**

As soon as a characer does not correspond to any valid monomer code, it is tagged, and the sequence is presented to the user in a text edit widget (*Initial Sequence*) with the all the invalid characters tagged by underlining. At that point, if the user clicks the *Remove Tagged From Initial* button, all the tagged characters will be automatically removed and the purified sequence will show up in the *Purified Sequence* text edit widget.

Also, the user is provided with automatic “purification” procedures whereby it is possible to remove one or more classes of characters from the imported sequence (*Purification Options* frame widget). Checking one or more of the *Numerals* or *Spaces* or *Punctuation* or *LowerCase* or *Uppercase* checkbuttons, or even entering other user-specified regular expressions in the *Other (RegExp)* line edit widget, will elicit their removal from the imported sequence after the user clicks the *Purify Initial (Options)* button.



!100ATGCATGCATGCATGCATGCATGCATGCAUGCanotherSilly-Text;

ATGCATGCATGCATGC
ATGCATGCATGCAGCS
T

☑ Numerals ☑ Spaces ☑ Punctuation
☐ Lowercase ☐ Uppercase
Others (RegExp):

There are a number of ways to purify a sequence. Here the *Remove Tagged From Initial* button was clicked. The purified sequence shows up in the *Purified Sequence* text edit widget.

FIGURE 4.8: **CLIPBOARD-IMPORTED SEQUENCE PURIFICATION**

When you are confident that almost all the erroneous characters have been removed ( FIGURE 4.8, " CLIPBOARD-IMPORTED SEQUENCE PURIFICATION " ), you can click the *Test Purified* button, which will trigger a "re-reading" of the sequence in the *Purified Sequence* text edit widget. If erroneous characters are still found, they are tagged.

### NOTE

For maximum flexibility, you are allowed an immediate and direct editing of the purified sequence in the *Purified Sequence* text edit widget (that is, that text edit widget is *not* read-only).

Once the sequence if finally depured from all the invalid characters, you can select it in the text edit widget and paste it in the XpertEdit sequence editor. This time, the paste operation will be error-free.

### 4.6.2 IMPORTING FROM RAW TEXT FILES

It might be of interest to be able to import a sequence from a raw file. To this end, the user is provided the menu *File*→ Import Raw that opens up a file selection window from which to choose the file to import. The program then iterates in the lines of that file and checks their contents for validity. If errors are found, then the same process as described earlier for clipboard-imported sequences is started ( SECTION 4.6.1, " IMPORTING FROM THE CLIPBOARD " ). The user can then purify the sequence imported from the file and finally integrate that sequence in the polymer sequence currently edited. Note that if any sequence portion is currently selected, it will be replaced by the one that is being imported.

## 4.7 MULTI-REGION SELECTIONS

MassXpert implements a sophisticated multi-region selection model. Two selection modes are available:

- *Multi-region selection mode:* In this mode, it is possible to select more than one region in the polymer sequence. In all cases below, make sure that the *Multi-region* checkbutton is checked in *Selections and regions* group box. This is how these selections are performed:
    - *With the mouse:* Left-click and drag to make the first selection. Go with the mouse cursor at the beginning of new selection, hold the `Ctrl` key down while left-clicking and dragging to perform the second region selection. Continue as may times as necessary;
    - *With the keyboard:* Position the cursor at the beginning of the first region to be selected, hold the `Ctrl`–`Shift` keys down while moving the cursor with the direction keys (the arrow keys of the keyboard). Hold the `Ctrl` key down and use the direction keys to go to the beginning of the new region selection, press the `Shift` key and hold it down while moving the cursor with the direction keys to actually perform the region selection.
- *Multi-selection region mode:* In this mode (which requires the multi-region selection mode to be enabled), it is possible to perform selections that overlap. For example, one could select the sequence "MAMISGM" and then select the sequence "SGMSGRKAS". The overlapping sequence is thus "SGM".

Being able to select multiple regions and/or to select multiple times the same region involves some configurations, as far as calculating relevant masses is concerned. Indeed, whatever the selection mode that is enabled, each time one selection (overlapping with another or not) is added or removed, masses are recalculated for the current selection [2] . The way the multi-region selections and the multi-selection regions are handled, from the mass calculation standpoint, is configured as follows:

- *Regions are oligomers:* In this configuration, each selection behaves as an oligomer, and thus should normally be capped on both its left and right ends. This is typically the situation when the user wants to simulate the formation of a cross-linked species arising from the cross-linking of two oligomers: each oligomer is capped on both its ends;
- *Regions are residual chains:* In this configuration, each selection behaves as a residual chain, and thus the oligomer resulting from the multi-region selections is capped on its left and right ends only once. This situation is typically encountered when simulating partial cleavages by first selecting an oligomer, checking

---

[2] "Selection", here, is thus used to collectively represent all multi-region selections and multi-selection regions at any given time in the polymer sequence editor.

its mass and then continuing selection to simulate a longer oligomer resulting from a partial cleavage. Also, the situation might be encountered when there are multiple repeated sequence motifs in a polymer sequence and mass data are difficult to analyze.

## 4.8 Polymer Sequence Modification

It very much often happens that (bio) chemists use chemical reactions to modify the polymer sequence they are working on. Mass spectrometry is then often used to check if the reaction proceeded properly or not. Further, in nature, chemical modifications of biopolymer sequences are very often encountered. For example, protein sequences get often modified as a means to regulate their function (phophorylations, for example, or acetylations, methylations…). Nucleic acid sequences are very often and extensively modified with modifications such as methylations…

It is thus crucial that MassXpert be able to model with high precision and flexibility the various chemical reactions that can be either made in the chemistry lab or found in nature. The MassXpert program provides two different chemical modification processes:

- A process by which monomers belonging to the polymer sequence can be individually modified;
- A process by which the whole polymer sequence can be modified, either on its left end or on its right end or even on both ends.

### 4.8.1 Selected Monomer(s) Modification

There are a number of manners in which monomers can be modified in a polymer sequence. Figure Figure 4.9, “ Modification of a monomer in a polymer sequence ” shows the simplest manner: the user first selects the monomer vignette to be modified and calls the *Chemistry*→ Modify Monomer(s) menu. A window shows up where all the modifications currently available in the polymer chemistry definition are listed. Because a monomer vignette was initially selected in the editor window, the *Selected Monomer* target radiobutton is on by default.

#### Note

Note that if a sequence was selected when the monomer modification task was started, then, selecting *Current selection* would be required to modify all the monomers in the selection. Alternatively, if this is not what is required, re-selecting the right monomer in the sequence and selecting *Current selection* will ensure the modification applies only on the currently selected monomer.

It is then simply a matter of choosing the right modification from the *Available modifications* list and clicking onto the *Modify* button. The target(s) of a given modification (as selected in the *Target* frame widget) can be identified according to:

- The *Selected Monomer* frame will display data in its two line edit widgets if a single monomer vignette was selected at the time the monomer modification action was invoked (exactly as in Figure 4.9, “ Modification of a monomer in a polymer sequence ” ).

### Warning

Only the monomer of which the code and the position are displayed will be modified (even if it is no more selected or if the sequence has changed and the monomer at the displayed position is not the same anymore).

- If the *Current Selection* radiobutton widget is selected, the modification should be performed on all the monomers that are *currently* selected, that is, if the selection changed after the modification window was displayed, the new selection is modified, not the old one;
- If the *Monomers Of Same Code* radiobutton widget is selected, all the monomers in the sequence that have their code identical to the one shown in the *Current selection* line edit widget are modified;
- If the *Monomers From The List* radiobutton widget is selected, all the monomers in the polymer sequence having a code corresponding to any code selected in the *Available Monomers* list are modified;
- If the *All Monomers* radiobutton widget is selected, all the monomers of the polymer sequence are modified;

This figure shows how the chemical modification of monomer(s) can be performed.

**FIGURE 4.9: MODIFICATION OF A MONOMER IN A POLYMER SEQUENCE**

Note that there is one checkbox widget (*Override target limitations*) that requires explanation. In the chapter about the definition of polymer chemistries ( Chapter 3, XpertDef: Definition of Polymer Chemistries ) the definition of modifications was detailed, and the “target” notion was explicited. If, during a monomer modification, MassXpert detects that the user is trying to modify a monomer that is not a target of the modification at hand, it will complain, as shown in the *Messages* text edit widget of Figure 4.9, “ Modification

OF A MONOMER IN A POLYMER SEQUENCE” ). In this example, indeed, the user tried to modify monomer *Leucine* with *Phosphorylation*, which is not possible because modification *Phosphorylation* has been defined a not having monomer *Leucine* as any of its targets. Another situation where target limitations might show up, is when trying to modify a monomer more than authorized by the *Max. count* number of times that monomer might be modified at once with that modification. For example, when working of methylation of proteins, it might happen that lysyl residues get methylated more than one at a time (tri-methylation occurs often in histones). If the chemical modification was defined in XpertDef with a max count of 2 and a third chemical modification is asked on a given target monomer, then the program refuses to perform the modification. To override this limitation, check the *Override target limitations* checkbox widget.

The general concept about this is : the *Override target limitations* checkbox widget is unchecked by default so that the user does not do mistakes without knowing. However, flexibility is desirable, and that checkbox widget can be checked if required.

As a result of the monomer modification, the monomer vignette gets modified. FIGURE 4.9, “ MODIFICATION OF A MONOMER IN A POLYMER SEQUENCE ” shows one phosphorylated Seryl residue at position 13: a transparent graphics object (a red “P”) was overlaid onto the corresponding seryl monomer vignette. If the user modifies a monomer with a modification that has no corresponding SVG -formatted file defined for its graphical rendering in file `modification_dictionary`, then a default modification rendering is used.

The user is responsible for correctly reading the messages that might be published in the *Messages* text edit widget. It is important to understand that, when a monomer is modified, its previous modification (if any) is overwritten with the new one. The user is invited to experiment a bit with the monomer modification process, so as to be confident of the results that she is going to obtain when real polymer chemistry work is to be modelled in MassXpert.

### TIP

If the modification to be applied is not readily available in the list of modifications defined in the polymer chemistry definition, then it is possible, by checking the *Define modification* check button widget to manually define a modification. This procedure leads to the modification of the target monomer(s) exactly as if the modification had been selected from the list of available modifications. But, because the modification has a name not known to the polymer chemistry definition, the editor cannot modify the monomer vignette with a predefined transparent raster image. Thus, as seen on FIGURE 4.10, “ RENDERING OF A MONOMER MODIFICATION IN A POLYMER SEQUENCE ” , the modified residue gets visually modified using the default transparent raster image (4 interrogation marks, one at each corner of the monomer vignette square).

This figure shows how the chemical modification of monomer(s) is graphically rendered. The “S” residue is modified using an “Phosphorylation” modification. The “I” residue is modified with a modification that has no associated graphical vignette. The default vignette is thus used.

**FIGURE 4.10:** RENDERING OF A MONOMER MODIFICATION IN A POLYMER SEQUENCE

It is perfectly feasible to modify a single monomer more than once (with the same modification or not ; for example a tri-methylation with a methylation modification). This is why when the window depicted in FIGURE 4.9, “ MODIFICATION OF A MONOMER IN A POLYMER SEQUENCE ” shows up, the two lists at the right hand side show the monomers currently modified and the modification(s) that are currently set to these modified monomers. Selecting one item from the *Modified monomers* list will show only the modifications set to that monomer in the *Modifications* list. If all the modifications in the polymer sequence are to be displayed then, checking the *All modifications* check box widget will trigger the display of all the modifications set to any monomer in the whole polymer sequence.

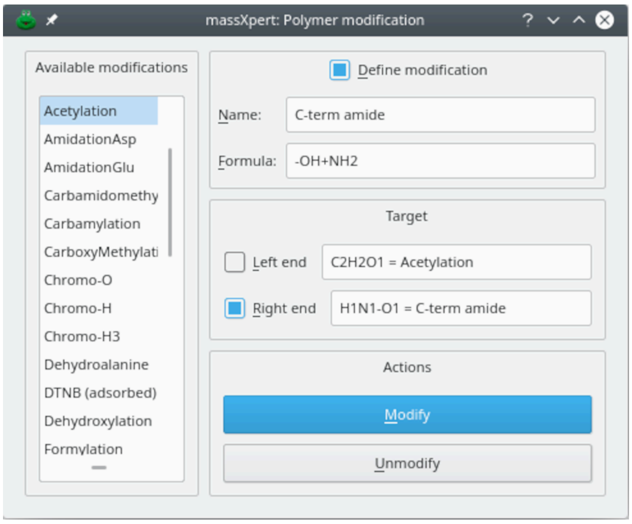
Unmodification of monomers is easily performed by selecting any number of items from the *Modifications* list and clicking the *Unmodify* button.

### WARNING

It should be noted that once a monomer modification dialog window has been opened, the polymer sequence should not be edited. This is because the modification/unmodification process takes for granted that the polymer sequence is still identical to what it was when the monomer modification dialog was opened. Mecanisms are there to ensure that the irreparable does not happen, but this warning is in order.

### 4.8.2 WHOLE SEQUENCE MODIFICATION

As described above, it is possible to modify any monomer in the polymer sequence; when any modified monomer is removed, the modification associated to it goes away with it. The modifications that we describe here are not of this kind. They can be applied either to the left end of the polymer sequence or to its right end (or both). But these modifications do belong to the polymer sequence *per se* and are not removed from it—even if the polymer sequence is edited by removing the left end monomer or the right end monomer. This is why these modifications are *polymer modifications* and not monomer modifications.



☑ Define modification

Name: C-term amide

Formula: -OH+NH2

☐ Left end C2H2O1 = Acetylation

☑ Right end H1N1-O1 = C-term amide

This figure shows how simple it is to permanently modify a polymer sequence on either or both its left/right ends.

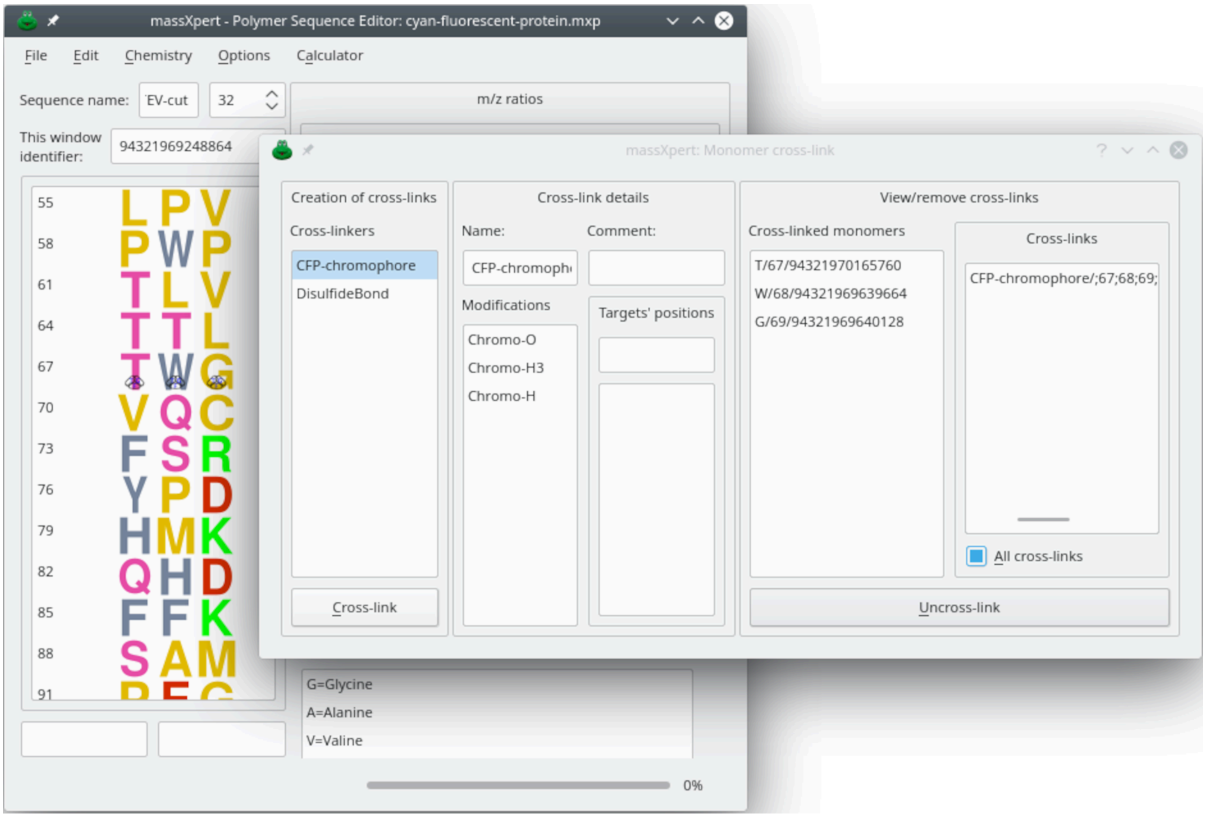
FIGURE 4.11: MODIFICATION OF THE LEFT END OF A POLYMER SEQUENCE

The way in which a polymer sequence is modified using *polymer modifications* is much easier than the previous *monomer modifications* case. The modification window is opened by choosing the *Chemistry*→ Modify Polymer menu. The FIGURE 4.11, “ MODIFICATION OF THE LEFT END OF A POLYMER SEQUENCE ” shows that window. The modification is absolutely easy to perform, with a clear feedback provided to the user (by listing the permanent modifications in two line edit widgets located in front of the *Target* checkbuttons *Left End* and *Right End*.

If a formally-defined modification is not available, you can define manually the modification that is needed (check the *Define modification* checkbox to that effect) and then apply to the polymer end of interst. This is illustrated in Figure FIGURE 4.11, “ MODIFICATION OF THE LEFT END OF A POLYMER SEQUENCE ” for the C-terminal end (the right end) of the protein. The modification object used is created on-the-fly by the program and gets saved in the file as if the user had selected a modification out of the list of available modifications. In the example, the polymer sequence was modified on its left end using the “Acetylation” modification available in the polymer chemistry definition and was amidated (formula *-OH+NH2*) with a manually-defined modification called *C-term amide*. The polymer sequence editor window displays the left end and right end modifications as labels of buttons located in the *Polymer modifications* groupbox widget.

## 4.9 MONOMER CROSS-LINKING

A cross-link is a covalent bond that links a monomer with one or more other monomer. A monomer might be cross-linked more than once. The dialog window in which the user might define cross-links is shown in FIGURE 4.12, “ CROSS-LINKING OF MONOMERS ” .



Sequence name: EV-cut 32

This window identifier: 94321969248864

Cross-linkers: CFP-chromophore, DisulfideBond

Name: CFP-chromophore

Comment:

Modifications: Chromo-O, Chromo-H3, Chromo-H

Targets' positions:

Cross-linked monomers:
T/67/94321970165760
W/68/94321969639664
G/69/94321969640128

Cross-links: CFP-chromophore/;67;68;69;

☑ All cross-links

G=Glycine
A=Alanine
V=Valine

This figure shows the window in which monomers can be cross-linked together. A cross-link (as defined in the current polymer chemistry definition) is selected and the targets are specified in the *Targets' positions* text line edit widget in the form of monomer positions separated by ";" semicolumns.

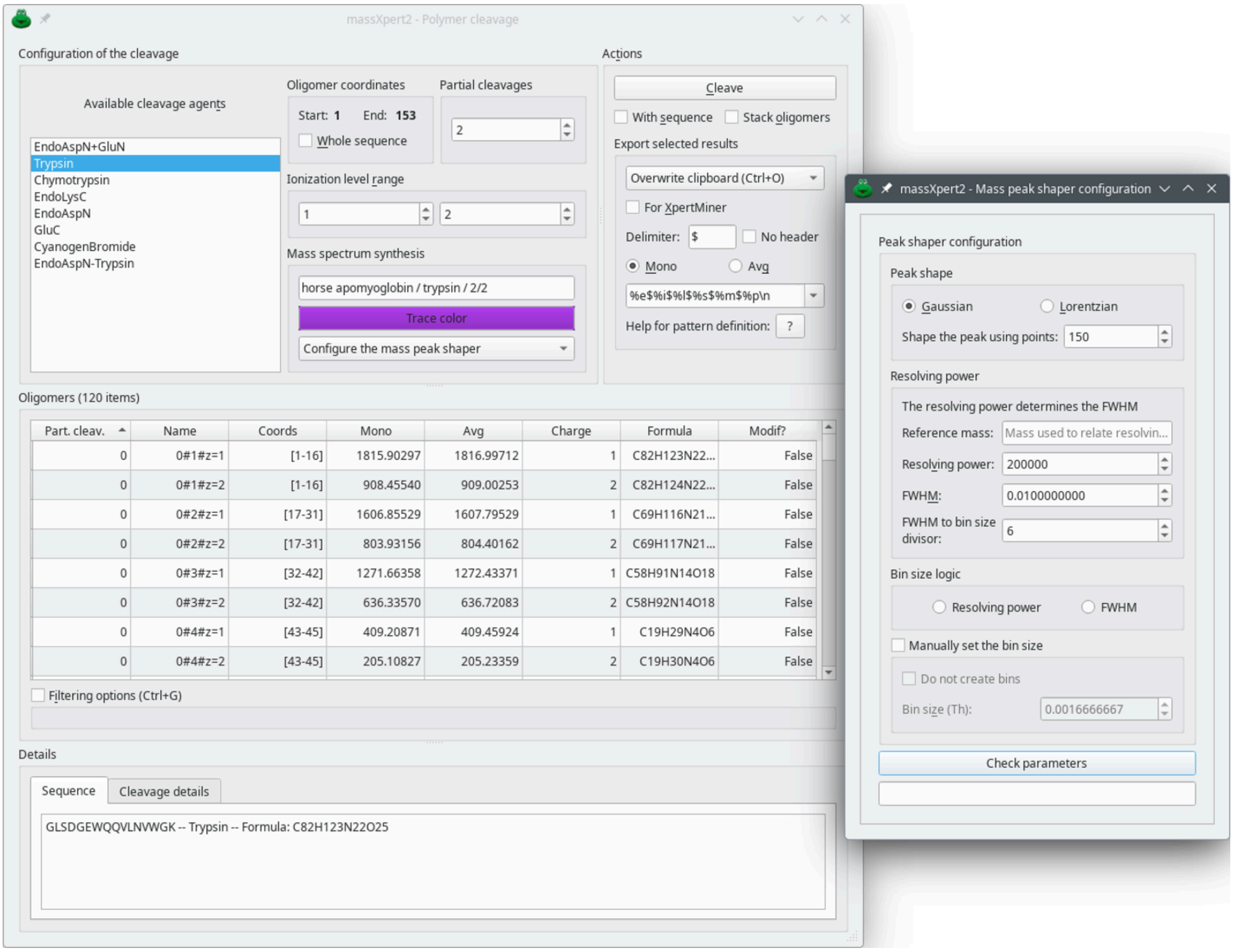
FIGURE 4.12: CROSS-LINKING OF MONOMERS

Cross-linkers were defined in the section about XpertDef (see page SECTION 3.2.3, " THE CROSS-LINKERS " ). A cross-linker might either define no modification to be applied to the cross-linked monomers or the same number of modifications as there are monomers cross-linked. For example, fluorescent proteins have a chromophore that is made by reaction of three residues (Threonyl [or Seryl]–Tryptophanyl [or Tyrosinyl or Phenylalanyl]–Glycyl), as shown in FIGURE 4.12, " CROSS-LINKING OF MONOMERS " . When cross-linking with the fluorescent protein cross-linker, there must be three monomers involved as these are three modifications defined in the cross-linker.

When any monomer involved in a cross-link is edited off a polymer sequence, the cross-link(s) it was involved in are automatically dissolved and destroyed. Destruction of a cross-link might be performed by selecting the cross-link in the *Cross-links* list widget at the right hand side of the dialog window depicted in FIGURE 4.12, " CROSS-LINKING OF MONOMERS " and by clicking the *Uncross-link* button.

## 4.10 SEQUENCE CLEAVAGE

It happens very often that polymer sequences get cleaved in a sequence-specific manner. These specific cleavages do occur very often in nature, and are made by enzymes that do cleave biopolymer sequences, like the glycosidases (cleaving saccharides), the proteases (cleaving proteins) or the nucleases (cleaving nucleic acids). But the scientist also uses purified enzymes or chemicals to perform such cleavages in the test tube. MassXpert must be able to perform those cleavages *in silico*.



Start: 1 End: 153

☐ Whole sequence

Partial cleavages: 2

Ionization level range: 1 – 2

horse apomyoglobin / trypsin / 2/2

☐ With sequence ☐ Stack oligomers

Overwrite clipboard (Ctrl+O)

☐ For XpertMiner

Delimiter: $ ☐ No header

● Mono ○ Avg

%e$%i$%l$%s$%m$%p\n

Oligomers (120 items)

| Part. cleav. | Name | Coords | Mono | Avg | Charge | Formula | Modif? |
|---|---|---|---|---|---|---|---|
| 0 | 0#1#z=1 | [1-16] | 1815.90297 | 1816.99712 | 1 | C82H123N22... | False |
| 0 | 0#1#z=2 | [1-16] | 908.45540 | 909.00253 | 2 | C82H124N22... | False |
| 0 | 0#2#z=1 | [17-31] | 1606.85529 | 1607.79529 | 1 | C69H116N21... | False |
| 0 | 0#2#z=2 | [17-31] | 803.93156 | 804.40162 | 2 | C69H117N21... | False |
| 0 | 0#3#z=1 | [32-42] | 1271.66358 | 1272.43371 | 1 | C58H91N14O18 | False |
| 0 | 0#3#z=2 | [32-42] | 636.33570 | 636.72083 | 2 | C58H92N14O18 | False |
| 0 | 0#4#z=1 | [43-45] | 409.20871 | 409.45924 | 1 | C19H29N4O6 | False |
| 0 | 0#4#z=2 | [43-45] | 205.10827 | 205.23359 | 2 | C19H30N4O6 | False |

GLSDGEWQQVLNVWGK -- Trypsin -- Formula: C82H123N22O25

● Gaussian ○ Lorentzian

Shape the peak using points: 150

Reference mass: Mass used to relate resolvin...

Resolving power: 200000

FWHM: 0.0100000000

FWHM to bin size divisor: 6

○ Resolving power ○ FWHM

☐ Manually set the bin size

☐ Do not create bins

Bin size (Th): 0.0016666667

This figure shows the window in which polymer sequence cleavages are performed. One cleavage specification is selected, the number of allowed partial cleavages and the ionization level range is set. The results are displayed in the same window. The cleavage might be performed on the currently selected polymer sequence region or the whole sequence.

FIGURE 4.13: **POLYMER SEQUENCE CLEAVAGE WINDOW**

Starting a polymer sequence cleavage is a matter of having a polymer sequence opened in an editor window and selecting the *Chemistry*→ Cleave menu. The user is provided with a window where a number of cleavage specifications are listed ( Figure 4.13, “ Polymer sequence cleavage window ” ), along with options that allow customizing the production of oligomers. The parameters that can be set are the following:

- *Available cleavage agents*: list of the cleavage agent specifications available in the polymer chemistry definition (see Section 3.2.4, “ The cleavage specifications ” );
- *Oligomer coordinates*: when the window is opened, this groupbox widget lists the coordinates of the currently selected region of the polymer sequence. Either leave the values as they are shown or check *Whole sequence*. If *Whole sequence* is checked, the cleavage will be performed over all the sequence length. Otherwise it will be performed over the sequence in the *Start–End* range. This feature, which was introduced in version 2.3.0, is useful so as to simulate a first cleavage of a polymer sequence and then a second cleavage of a selected oligomer using a different cleavage agent. In protein chemistry, that would be useful to explore possibilities of double sequential cleavages of a protein, first with EndoAspN, for example, and then with Trypsin.
- Setting the *Partial cleavages* number defines if the cleavage must be total (value of *0*) or if missed cleavages are allowed.
- Setting the *Ionization level range* defines what charge state the generated oligomer will have.

A number of other features might be configured, either on the way oligomers should be exposed or on the way they should be exported to a number of destinations:

- Checking *With sequence* tells the cleavage engine to store their sequence in the generated oligomers; see Section 4.10, “ Sequence Cleavage ” (page 63) .
- In normal operation, when multiple cleavages are performed by clicking *Cleave*, the new oligomer set replaces the one obtained previously. However, it might be useful to generate in the same unique list all the oligomers generated for different cleavages. To stack oligomers from different cleavages, check *Stack oligomers*.
- It is possible to export the oligomers selected in the tableview widget in a number of ways. Select the destination in the drop down menu widget. A specific export format might be defined and the mass type to be exported might be selected. When exporting for the XpertMiner module, check *For XpertMiner*.

### NOTE

If the list of monoisotopic or average masses is desired in the form of a text list, right-clicking onto the tableview widget will allow copying to the clipboard either the monoisotopic or the average masses. Also, it is possible to either export the data to the clipboard or to a file or even to drag the displayed oligomer items in a text editor. Only the selected items in the treeview widget will be exported.

- *Mass spectrum synthesis* This group box is about the configuration of the way a mass spectrum might be calculated and displayed in another program. The *Trace color* button allows one to select a color with which to plot the mass spectrum's trace once exported to a mass data visualizing program (MineXpert). The line edit widget above the button allows one to define a title for the mass spectral trace plotted. Finally, the drop down menu holds three menus, as described below:

  - *Configure the mass peak shaper* Click this menu item to display the *Mass peak shaper configuration* dialog window as shown on FIGURE 4.13, " POLYMER SEQUENCE CLEAVAGE WINDOW " . The configuration of the mass peak shaper works in this dialog window exactly as described in FIGURE 8.10, " SETTING-UP OF THE CENTROID MASS PEAK SHAPING PROCESS " .

  - *Synthesize the mass spectra* This button actually triggers the computation of the mass spectra (the isotopic clusters) of all the selected fragmentation oligomers currently selected in the table view. The calculation is only performed if the previous configuration step has yielded a valid isotopic cluster shaper configuration.

  - *Mass spectrum to clipboard* This menu item copies the mass spectrum to the clipboard. For this menu to be available, a mass spectrum synthesis must have been performed successfully, otherwise the menu reads *Mass spectrum not yet available*.

### NOTE

Upon successful completion of a mass spectrum synthesis, the obtained mass spectral trace is automatically made available to any requesting program *if and only if* as server is started on the MassXpert side and *if* a client is started on the MineXpert side. See for details. When the mass spectral trace is successfully displayed in MineXpert via the network, the trace color and title are honoured. When the trace is copied to the clipboard and then loaded in MineXpert, the color and title are not transferred along with the mass spectral data.

The *Details* frame widget at the bottom of the window displays a number of informative data. In particular, the *Sequence* tab widget displays the sequence of the oligomer currently selected in the *Oligomers* table view along with the name of the cleavage agent which it arose from. The *Cleavage Details* tab widget displays the mass calculation engine configuration at the time the *last* cleavage was performed (checked items mean that the corresponding feature was on, unchecked items mean that the related feature was off). In the example (Figure FIGURE 4.13, " POLYMER SEQUENCE CLEAVAGE WINDOW " ), the mass calculation for the oligomers did not account for the monomer modifications nor for the left/right end modifications of the polymer, nor for the cross-links.

When the user triggers a cleavage, the mass calculation engine configuration currently set in the sequence editor is used for the calculation of the mass of the oligomers obtained *per* the cleavage. This process allows an easy change in the mass calculation engine configuration between one cleavage and another so as to allow comparison of masses obtained for the same cleavage but with different mass calculation engine configurations.

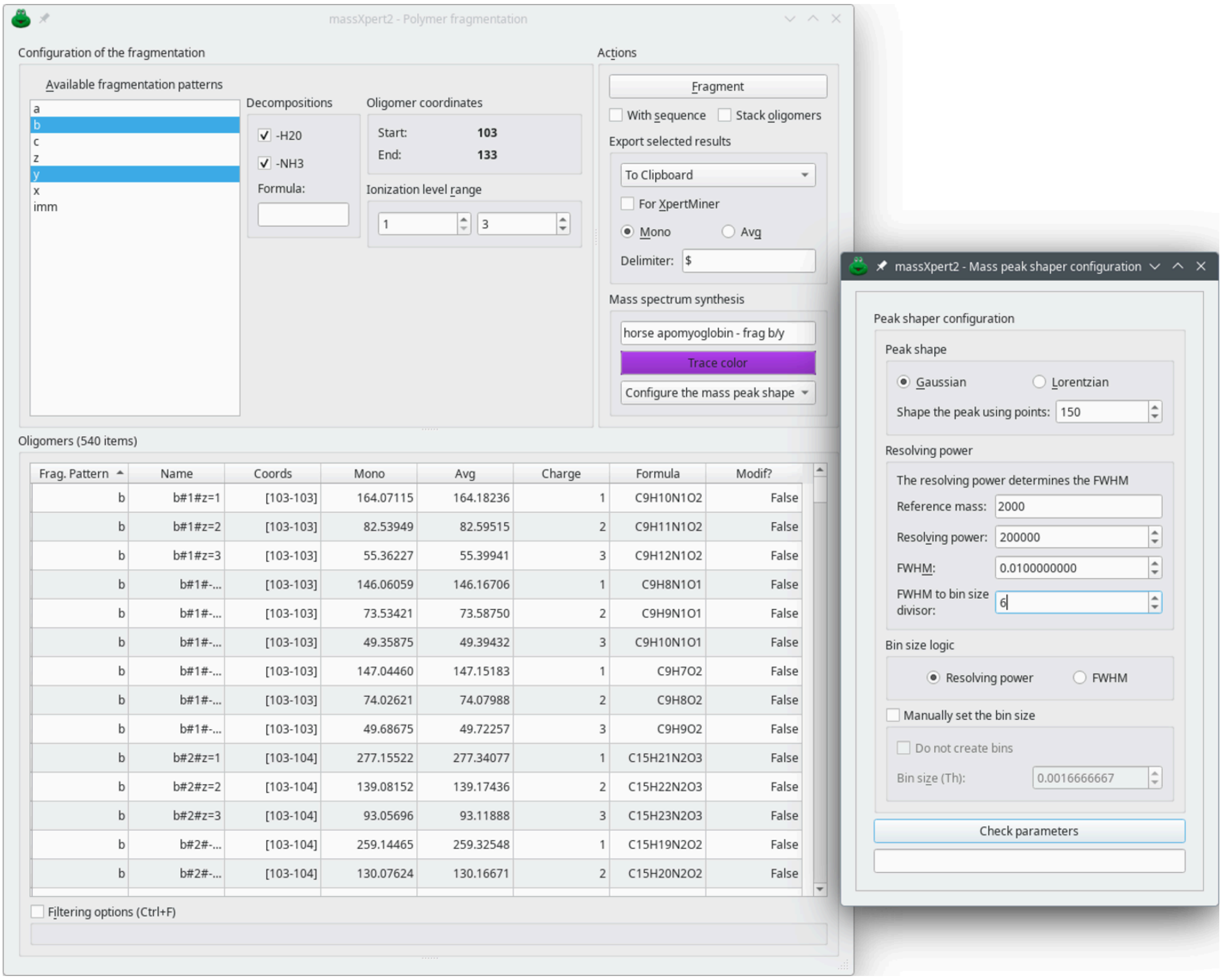
For oligomer data filtering, please refer to SECTION 4.13, " OLIGOMER DATA FILTERING " .

### TIP

It is possible to display any given fragment in the polymer sequence editor simply by double-clicking its corresponding row in the table view widget.

## 4.11 OLIGOMER FRAGMENTATION

It happens very often that polymer sequences need to be fragmented in the gas phase (in the mass spectrometer) so that structure characterizations may be performed. For protein chemistry, this happens very often in order to get sequence information for a given peptide ion selected in the gas phase. MassXpert must be able to perform those fragmentations *in silico*. Let's see how an oligomer can be fragmented using MassXpert.



Configuration of the fragmentation

Available fragmentation patterns: a, b, c, z, y, x, imm (b and y selected)

Decompositions: ☑ -H20 ☑ -NH3 Formula:

Oligomer coordinates: Start: 103 End: 133

Ionization level range: 1 3

☐ With sequence ☐ Stack oligomers

Export selected results: To Clipboard

☐ For XpertMiner

● Mono ○ Avg

Delimiter: $

Mass spectrum synthesis: horse apomyoglobin - frag b/y

Oligomers (540 items)

| Frag. Pattern | Name | Coords | Mono | Avg | Charge | Formula | Modif? |
|---|---|---|---|---|---|---|---|
| b | b#1#z=1 | [103-103] | 164.07115 | 164.18236 | 1 | C9H10N1O2 | False |
| b | b#1#z=2 | [103-103] | 82.53949 | 82.59515 | 2 | C9H11N1O2 | False |
| b | b#1#z=3 | [103-103] | 55.36227 | 55.39941 | 3 | C9H12N1O2 | False |
| b | b#1#-... | [103-103] | 146.06059 | 146.16706 | 1 | C9H8N1O1 | False |
| b | b#1#-... | [103-103] | 73.53421 | 73.58750 | 2 | C9H9N1O1 | False |
| b | b#1#-... | [103-103] | 49.35875 | 49.39432 | 3 | C9H10N1O1 | False |
| b | b#1#-... | [103-103] | 147.04460 | 147.15183 | 1 | C9H7O2 | False |
| b | b#1#-... | [103-103] | 74.02621 | 74.07988 | 2 | C9H8O2 | False |
| b | b#1#-... | [103-103] | 49.68675 | 49.72257 | 3 | C9H9O2 | False |
| b | b#2#z=1 | [103-104] | 277.15522 | 277.34077 | 1 | C15H21N2O3 | False |
| b | b#2#z=2 | [103-104] | 139.08152 | 139.17436 | 2 | C15H22N2O3 | False |
| b | b#2#z=3 | [103-104] | 93.05696 | 93.11888 | 3 | C15H23N2O3 | False |
| b | b#2#-... | [103-104] | 259.14465 | 259.32548 | 1 | C15H19N2O2 | False |
| b | b#2#-... | [103-104] | 130.07624 | 130.16671 | 2 | C15H20N2O2 | False |

☐ Filtering options (Ctrl+F)

Peak shaper configuration

Peak shape: ● Gaussian ○ Lorentzian

Shape the peak using points: 150

Resolving power

The resolving power determines the FWHM

Reference mass: 2000

Resolving power: 200000

FWHM: 0.0100000000

FWHM to bin size divisor: 6

Bin size logic: ● Resolving power ○ FWHM

☐ Manually set the bin size

☐ Do not create bins

Bin size (Th): 0.0016666667

This figure shows the window in which oligomer fragmentations are performed. One or more fragmentation patterns might be selected in one fragmentation step.

FIGURE 4.14: **OLIGOMER FRAGMENTATION WINDOW**

Starting an oligomer fragmentation is a matter of having a polymer sequence opened in an editor window, selecting the sequence region to be fragmented and finally selecting the *Chemistry*→ Fragment menu. The user is provided with a window where a number of fragmentation specifications are listed ( FIGURE 4.14, “ OLIGOMER FRAGMENTATION WINDOW ” ). As detailed for the cleavage of polymers, these fragmentation specifications are listed by looking into the polymer chemistry definition corresponding to the polymer sequence of which an oligomer is to be fragmented.

Select the fragmentation specification(s) of interest, set the ionization range required for the generated fragment oligomers (the same as for polymer sequence cleavage) and click *Fragment* . Upon successful termination of the fragmentation reaction, the generated fragments are displayed in the *Oligomers* tableview widget.

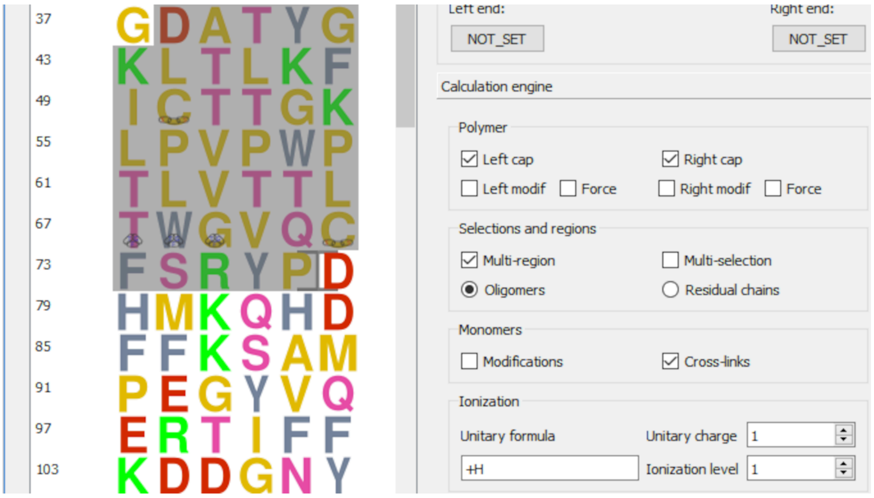
As detailed for the cleavage of polymer sequences, the *Details* frame widget displays data about the fragments generated and the way masses were calculated for them.

It is possible to take into account cross-links that are beared by monomers contained in the oligomer.

### Warning

Only cross-links that are fully contained in the oligomer are taken into account. Partial cross-links, that is, cross-links involving at least one monomer outside of the oligomer, are ignored.

The cross-links are accounted for during the fragmentation of an oligomer if *Cross-links* is checked in the calculation engine configuration panel of the sequence editor window (Figure FIGURE 4.15, " STARTING A FRAGMENTATION THAT ACCOUNTS THE CROSS-LINKS " ). This is a partial view of the cyan fluorescent protein, with the "TWG" chromophore tripeptide. We added a disulfide bond cross-link between two cysteinyl residues, only for the example (this is not biologial!).



37 G D A T Y G
43 K L T L K F
49 I C T T G K
55 L P V P W P
61 T L V T T L
67 T W G V Q C
73 F S R Y P D
79 H M K Q H D
85 F F K S A M
91 P E G Y V Q
97 E R T I F F
103 K D D G N Y

Left end: NOT_SET
Right end: NOT_SET

Calculation engine

Polymer
☑ Left cap
☑ Right cap
☐ Left modif ☐ Force
☐ Right modif ☐ Force

Selections and regions
☑ Multi-region
☐ Multi-selection
● Oligomers
○ Residual chains

Monomers
☐ Modifications
☑ Cross-links

Ionization
Unitary formula: +H
Unitary charge: 1
Ionization level: 1

The polymer sequence in the editor is selected making sure that all the cross-links are included in it. The calculation engine is configured to account for the cross-links.

FIGURE 4.15: STARTING A FRAGMENTATION THAT ACCOUNTS THE CROSS-LINKS

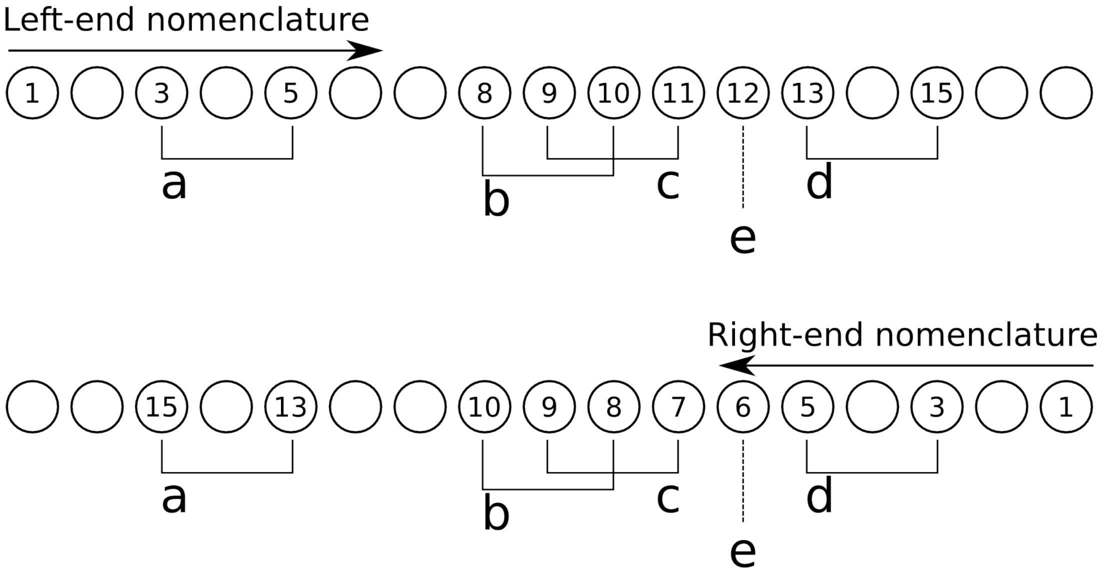
If we select the oligomer region [38–77] and that we ask for a fragmentation, the fragmentation results will take into account both cross-links only in the case the generated fragments encompass fully one or more cross-links.

The following calculation rationale applies:

- Fragments b (left end) from $b_1$ (D) to $b_{12}$ (up to I) do not take into account the cross-links as both are outside of its scope;
- Fragments $b_{13}$ (up to C) to $b_{34}$ (up to Q) do not take into account the cross-links because the outer cross-link (disulfide bond between cysteine residues) is not complete (the second cysteine is left out of the fragment);

- Fragments b$_{35}$ (up to C) to b$_{40}$ (up to P) do take into account both cross-links because both are contained in the fragments;
- Likewise, the only y fragments (right end) that do take into account the cross-links are the fragments y$_{28}$ (up to C) and all the remaining, as for these fragments, the cross-links are both fully contained.

A more complex cross-linked structure example is shown in Figure Figure 4.16, “ Complicated cross-linking situation ” , where the 17-mer oligomer has 4 fully-encompassed cross-links and one partial. This oligomer is used as an example of how the fragmentation computation is handled my MassXpert.



This figure shows a complicated cross-linking situation with an oligomer that has five cross-links, four of which are fully encompassed by the oligomer and one that involves a monomer outside of the oligomer.

**FIGURE 4.16: COMPLICATED CROSS-LINKING SITUATION**

The calculation of the fragments for this oligomer involves the following steps:

- Calculate regions of the oligomer that involve cross-links either overlapping or not. The regions are thus the following: [3–5], [8–11] and [13–15]. Note that the cross-link involving monomer 12 is never taken into account as it involves also a monomer outside of the oligomer;
- For fragments that have the left end of the oligomer (``Left-end nomenclature''), the following rationale is used:
    - Fragments →1 and →2 do not have any cross-link;
    - Fragments →3 to →4 do not account for cross-link a because that cross-linke is not fully encompassed by the fragments;
    - Fragments →5 to →10 account only for the cross-link a as this is the only cross-linked region to be fully encompassed by these fragments;

  - Fragments →11 to →14 account for cross-links a, b and c as they are all fully encompassed in the fragments;
  - Fragments →15 to →16 account for all cross-links, a, b, c, d as they are all fully encompassed in the fragments;
- For fragments that have the right end of the oligomer (Right-end nomenclature), the following rationale is used:
  - Fragments 1← and 2← do not have any cross-link;
  - Fragments 3← and 4← do not account for cross-link d because that cross-link is not fully encompassed by the fragments;
  - Fragments 5← and 6← account for cross-link d because it is fully encompassed in these fragments;
  - Fragments 7← to 9← only account for cross-link d because cross-links b and c (which make one cross-linked region) are not fully encompassed by these fragments;
  - Fragments 10← to 14← account for cross-links d, c and b, but not for cross-link a as this last cross-link is not fully encompassed in these fragments;
  - Fragments 15← and 16← account for all the cross-links of the oligomer.

### WARNING

It is necessary to repeat one more time that cross-links that involve monomer(s) outside of the oligomer are ignored. The user is alerted whenever this situation is encountered.

The various widgets in the *Actions* groupbox widget are very similar to the ones found in the polymer sequence cleavage window FIGURE 4.13, “ POLYMER SEQUENCE CLEAVAGE WINDOW ” ).

For oligomer data filtering, please refer to SECTION 4.13, “ OLIGOMER DATA FILTERING ” .

Exactly as described for the cleavage oligomer mass spectrum synthesis, it is possible to synthesize a mass spectrum corresponding to all the fragmentation oligomers currently selected in the table view widget. This process has been described at ??? (PAGE 62) .

### TIP

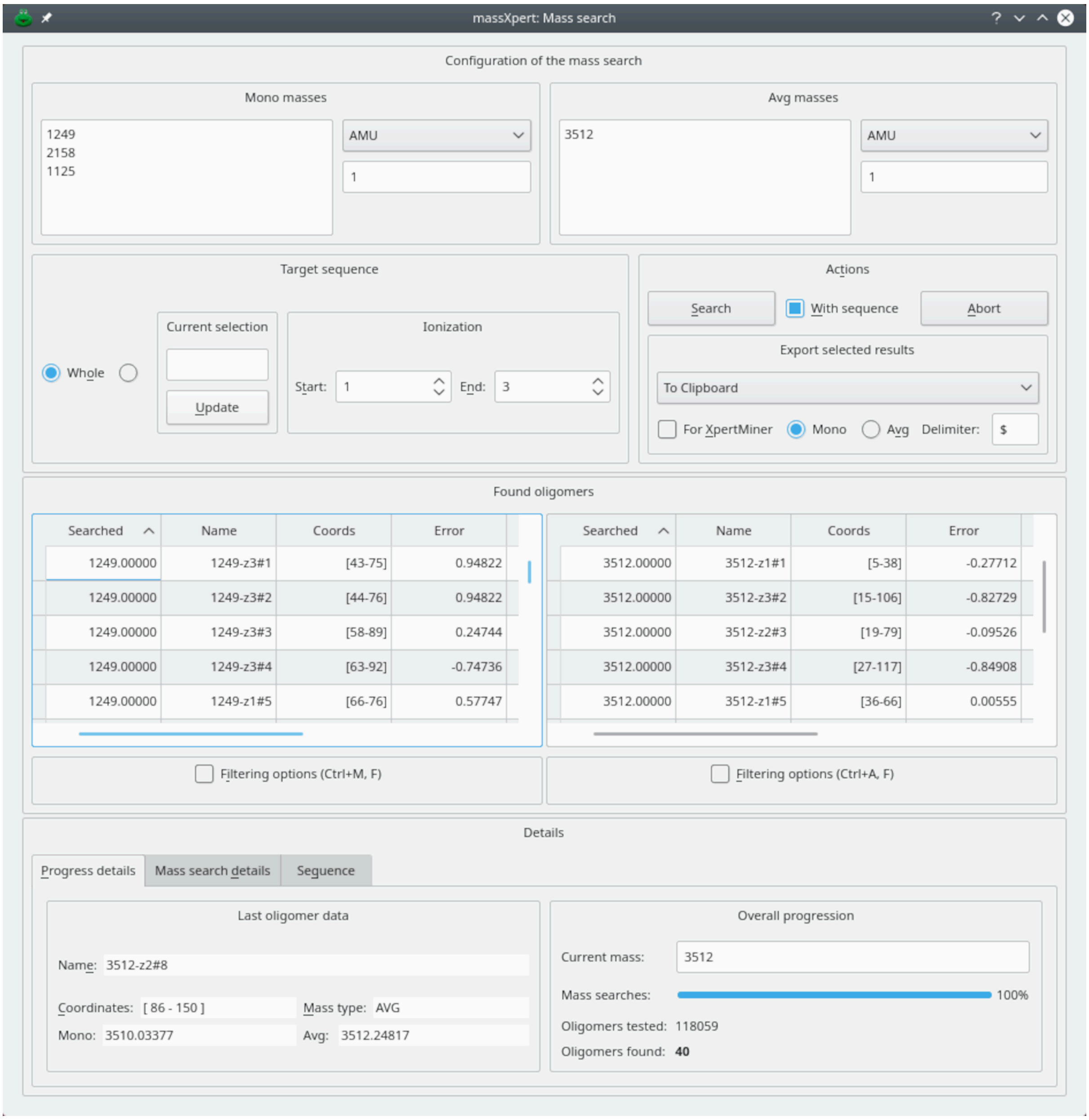
In order to visualize a given oligomer, listed in the table view widget, inside of the sequence editor window, double-click its corresponding row; the corresponding sequence will be highlighted in the sequence editor.

## 4.12 MASS SEARCHING

It may happen that the scientist needs to know if some arbitrary sequence region would have a given mass. MassXpert allows for mass searching operations in the polymer sequence. This is done by using the menu *Chemistry*→ Mass Search . The window illustrated in FIGURE 4.17, “ SEARCHING MASSES IN A POLYMER SEQUENCE ” shows up and the user enters masses to search for. A number of parameters are to be detailed:

- *Targets*: should the masses be searched for in the whole sequence or in the currently selected region only?
- *Ionization*: should different levels of ionization be calculated when calculating masses for the potential oligomers matching the searched mass? For example, one finds in an electrospray ionization experiment mass spectrum a peak at m/z 1245. It is not possible to know the ionization level for that ion. On could imagine that this value is for a monopronotonated or for a multiprotonated species. If we wanted to assess this, we might ask that the mass be searched for by computing a range of possible ionization levels between *Start* level *1* and *End* level *4* (admitting that for that experiment this is what one would expect).

Once the masses have been searched for, if results are found they are displayed in the same window in the *Oligomers* table view widgets (the left one for the mono masses and the right one for the avg masses).



Mono masses: 1249, 2158, 1125 — AMU — 1

Avg masses: 3512 — AMU — 1

● Whole ○

Start: 1 End: 3

☑ With sequence

To Clipboard

☐ For XpertMiner ● Mono ○ Avg Delimiter: $

| Searched | Name | Coords | Error |
|---|---|---|---|
| 1249.00000 | 1249-z3#1 | [43-75] | 0.94822 |
| 1249.00000 | 1249-z3#2 | [44-76] | 0.94822 |
| 1249.00000 | 1249-z3#3 | [58-89] | 0.24744 |
| 1249.00000 | 1249-z3#4 | [63-92] | -0.74736 |
| 1249.00000 | 1249-z1#5 | [66-76] | 0.57747 |

| Searched | Name | Coords | Error |
|---|---|---|---|
| 3512.00000 | 3512-z1#1 | [5-38] | -0.27712 |
| 3512.00000 | 3512-z3#2 | [15-106] | -0.82729 |
| 3512.00000 | 3512-z2#3 | [19-79] | -0.09526 |
| 3512.00000 | 3512-z3#4 | [27-117] | -0.84908 |
| 3512.00000 | 3512-z1#5 | [36-66] | 0.00555 |

☐ Filtering options (Ctrl+M, F)

☐ Filtering options (Ctrl+A, F)

Name: 3512-z2#8

Coordinates: [ 86 - 150 ]

Mass type: AVG

Mono: 3510.03377

Avg: 3512.24817

Current mass: 3512

Mass searches: 100%

Oligomers tested: 118059

Oligomers found: 40

This figure shows the window in which to search for masses in a polymer sequence.

FIGURE 4.17: **SEARCHING MASSES IN A POLYMER SEQUENCE**

The various widgets in the *Actions* groupbox widget are very similar to the ones found in the polymer sequence cleavage window FIGURE 4.13, “ POLYMER SEQUENCE CLEAVAGE WINDOW ” ).

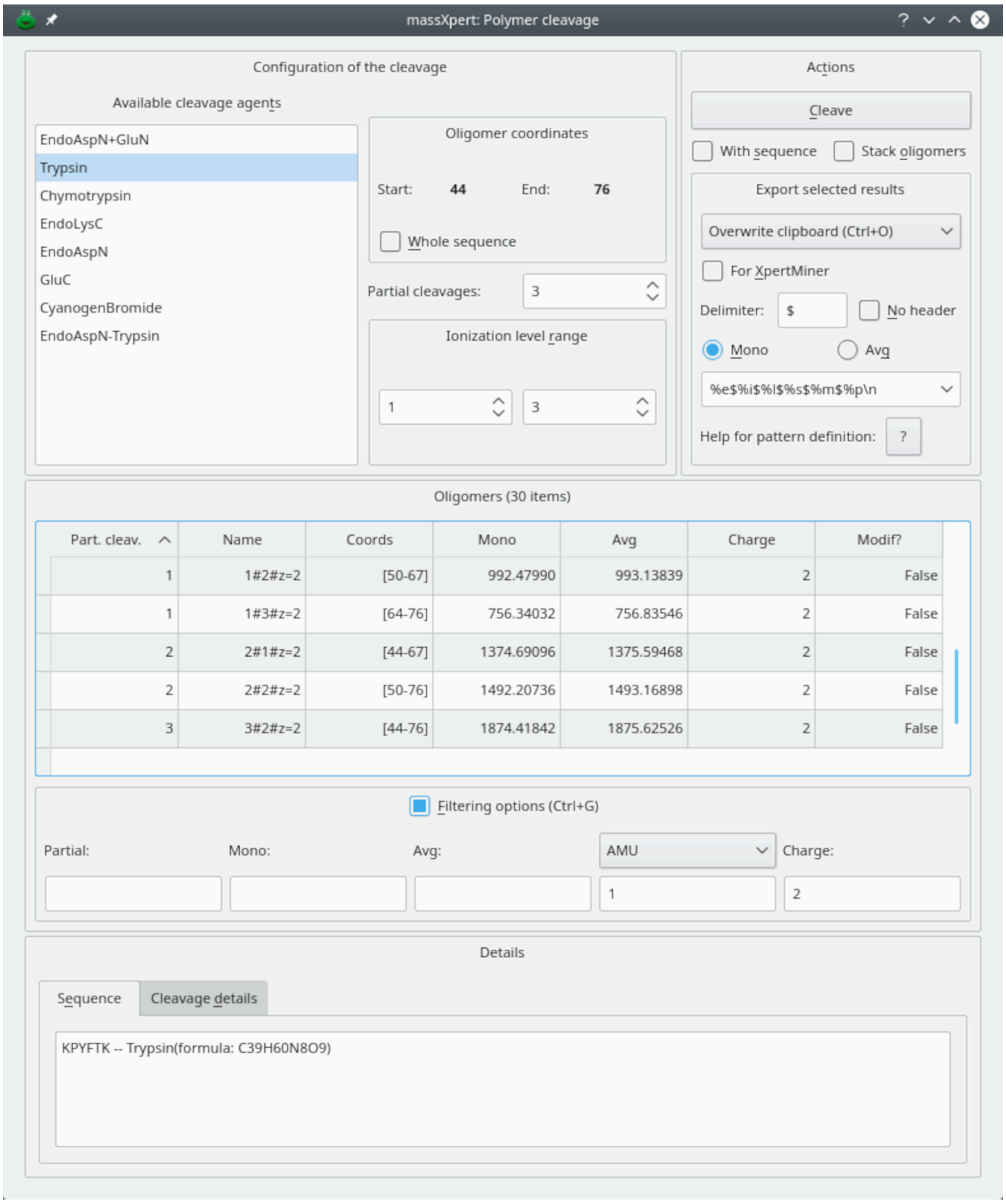
For oligomer data filtering, please refer to SECTION 4.13, “ OLIGOMER DATA FILTERING ” .

## TIP

If you want to visualize into the sequence editor window a given oligomer, as listed in the tableview widget, double-click its item; the corresponding sequence will be highlighted in the sequence editor.

## 4.13 OLIGOMER DATA FILTERING

Oligomer-generating simulations, like polymer sequence cleavages or fragmentations or mass searches, produce a very large amount of data. It is often desirable to be able to filter quickly some specific data out of these bunch of data… In all these three simulation contexts, the results that are displayed in the corresponding dialog windows are easily filtered using the mechanism illustrated in FIGURE 4.18, “ OLIGOMER DATA FILTERING ” . To enter filtering mode, check *Filtering options*; the line edit widgets will appear for you to start entering filters.



Start: 44 End: 76

Partial cleavages: 3

Ionization level range: 1 3

Delimiter: $

%e$%i$%l$%s$%m$%p\n

Oligomers (30 items)

| Part. cleav. | Name | Coords | Mono | Avg | Charge | Modif? |
|---|---|---|---|---|---|---|
| 1 | 1#2#z=2 | [50-67] | 992.47990 | 993.13839 | 2 | False |
| 1 | 1#3#z=2 | [64-76] | 756.34032 | 756.83546 | 2 | False |
| 2 | 2#1#z=2 | [44-67] | 1374.69096 | 1375.59468 | 2 | False |
| 2 | 2#2#z=2 | [50-76] | 1492.20736 | 1493.16898 | 2 | False |
| 3 | 3#2#z=2 | [44-76] | 1874.41842 | 1875.62526 | 2 | False |

Partial: Mono: Avg: AMU 1 Charge: 2

KPYFTK -- Trypsin(formula: C39H60N8O9)

This figure shows how oligomer data can be filtered. The *Filtering options* groupbox contains four line edit widgets where filtering might be triggered: *Partial*, *Mono*, *Avg*, *Charge*. The filtered data are displayed in the same window (this example is for polymer sequence cleavage simulation data). Here, the filter is applied to the charge level of the oligomers, only showing those having a charge of *2*.

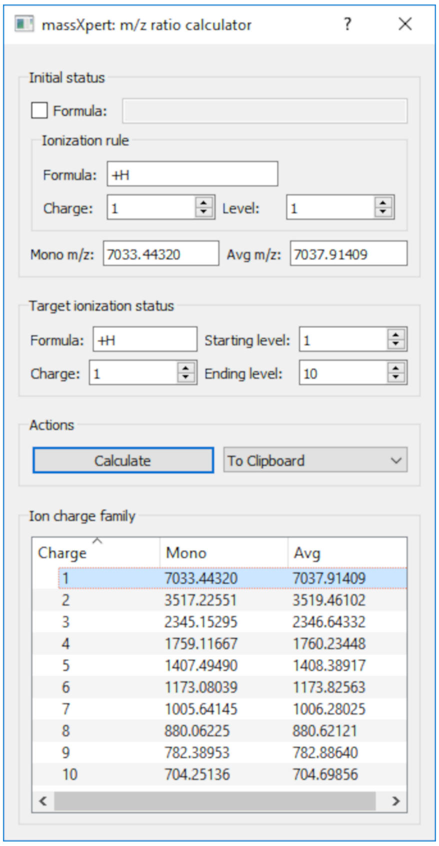
FIGURE 4.18: **OLIGOMER DATA FILTERING**

Filtering on the data is easily performed by entering the options in the *Filtering options* group box ( FIGURE 4.18, " OLIGOMER DATA FILTERING " ). For any filtering operation, only one criterium can be used, that is, for example, filtering can occur only on the basis of the monoisotopic mass or of the average mass, but not on both masses. For example, if one wanted to filter a huge set of data against a specific monoisotopic mass of 850 plus or minus 3 atomic mass units, it would simply be a matter of setting the monoisotopic mass to be *850* with a tolerance of *3 AMU* in the corresponding line edit widgets contained in the *Filtering options* groupbox. To perform that filtering action, first set the tolerance value (*3*) in its line edit widget and next set the monoisotopic mass value to be *850* in the corresponding line edit widget. While the cursor *is still* in the *Mono* line edit where *850* was entered, press the keyboard key combination `Ctrl`–`Enter` . The filtering will be immediate and the table view will show the data that passed the filter. Note that the combo box widget holding the unit of the tolerance (in the example, that unit is *AMU*, that is, “atomic mass unit”) and the line edit widget where the tolerance value proper is set (*3* in the example) do not trigger any filtering by themselves; these widgets are only useful in conjunction with other oligomer data : *Mono*, *Avg*, *Error* line edit widgets (depending on the dialog window the filtering occurs: cleavage, fragmentation or mass search). In our example, thus, the filtering would be spoken like this: —“Only show the oligomers for which the monoisotopic mass is 850 plus or minus 3 atomic mass units”.

To exit the data filtering mode, uncheck *Filtering options* and all the initial data will be displayed, irrespective of any data in the line edit widgets described above.

## 4.14 m/z RATIO CALCULATIONS

In electrospray ionization, a given polymer sequence might be charged a large number of times. The tool shown in FIGURE 4.19, “ CALCULATION OF RANGES OF M/Z RATIOS ” shows how to compute a range of m/z ratios starting from one m/z value for a given charge and a given ionization agent. It is also possible to switch ionization agent on-the-fly.



Initial status

☐ Formula:

Ionization rule

Formula: +H

Charge: 1

Level: 1

Mono m/z: 7033.44320

Avg m/z: 7037.91409

Target ionization status

Formula: +H

Starting level: 1

Charge: 1

Ending level: 10

Ion charge family

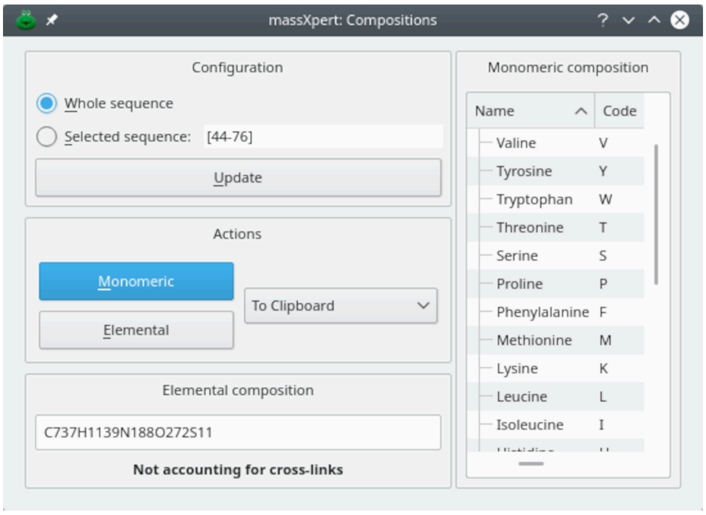
| Charge | Mono | Avg |
|---|---|---|
| 1 | 7033.44320 | 7037.91409 |
| 2 | 3517.22551 | 3519.46102 |
| 3 | 2345.15295 | 2346.64332 |
| 4 | 1759.11667 | 1760.23448 |
| 5 | 1407.49490 | 1408.38917 |
| 6 | 1173.08039 | 1173.82563 |
| 7 | 1005.64145 | 1006.28025 |
| 8 | 880.06225 | 880.62121 |
| 9 | 782.38953 | 782.88640 |
| 10 | 704.25136 | 704.69856 |

This figure shows the window in which to perform the calculation of different m/z ratios starting from one m/z value with a given ionization agent.

FIGURE 4.19: **CALCULATION OF RANGES OF M/Z RATIOS**

## 4.15 Monomeric And Elemental Composition

The *Chemistry*→ Determine Compositions menu triggers the window shown in Figure 4.20, “ Determination of the compositions ” . The elemental composition is determined using the calculations engine configuration currently set in the polymer sequence editor window.



● Whole sequence

○ Selected sequence: [44-76]

C737H1139N188O272S11

**Not accounting for cross-links**

| Name | Code |
|---|---|
| Valine | V |
| Tyrosine | Y |
| Tryptophan | W |
| Threonine | T |
| Serine | S |
| Proline | P |
| Phenylalanine | F |
| Methionine | M |
| Lysine | K |
| Leucine | L |
| Isoleucine | I |

This figure shows how to determine the monomeric and elemental compositions for the whole sequence or the current selection.

FIGURE 4.20: DETERMINATION OF THE COMPOSITIONS

## 4.16 pKa, pH, pI and Charges

When preparing biochemical experiments, very often users need to know how many charges a given polymer sequence will bear at any given pH. Equally important is the ability to know at which pH value the polymer sequence will have a net charge near to zero. The pH value for which a given polymer sequence has a net charge near to zero (typically this means that the number of positive charges equals the number of negative charges) is called the isoelectric point—the pI.

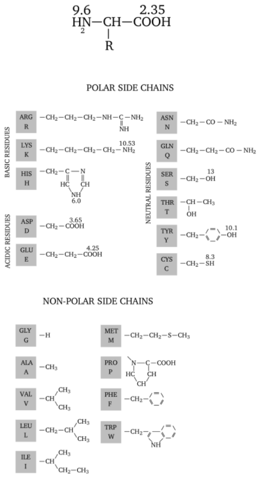
Such computations are pretty computer-intensive and require a very precise knowledge of the chemical structure of the different monomers that take part in the definition of the polymer chemistry. A file, called `pka_ph_pi.xml` is located in the polymer chemistry definition directory. This file lists all the chemical groups that are possibly charged; each monomer of the polymer definition is represented by a <*monomer*> element in which data are defined for any chemical group of that monomer that might bear a charge at any given pH. You can find the listing of the `pka_ph_pi.xml` file in . We'll discuss any aspect of this file's contents in the next sections with enough detail that the user will be able to write one such file for her specific polymer chemistry.

At the moment, two entities in the polymer chemistry definition might have chemical groups bearing charges: monomers and modifications. We will first review monomers, and modifications next.

## 4.16.1 Ionized Group(s) In Monomers

Monomers are the building blocks of polymer sequences. These blocks must have at least two reactive groups so that they can be polymerized into a polymer sequence thread. Reactive groups are often chargeable groups; for example, the amino group of amino-acids is such that it gets protonated (positively charged) at a pH inferior to its pKa. Similarly, the carboxylic acid group of amino-acids is deprotonated (negatively charged) at physiological pH.

### 4.16.1.1 Some Theory First



All of the twenty amino-acids are represented here, which each amino-acid's lateral chain fully represented. Above each chemical group—for which the value makes sense from a biological perspective—the pKa value is indicated.

**FIGURE 4.21: DIFFERENT PKA VALUES FOR A NUMBER OF AMINO-ACIDS' CHEMICAL GROUPS**

For the non-biochemist reader, amino-acids involved in the formation of proteins have always at least two chemical groups that are of inverted electrical charge, at physiological pH values (see Figure Figure 4.21, “ Different pKa values for a number of amino-acids' chemical groups ” ):

- The amino group (called $\alpha NH_2$) has a typical pKa value of 9.6. This means that, at physiological pH values (between 6.5 and 7.5), the amino group will find the environment rather acidic, and will thus be protonated, leading to a positively-charged species ($\alpha NH_3^+$);
- The carboxylic group (called αCOOH) has a typical pKa value of 2.35. This means that, at physiological pH values, the carboxylic group will be in a rather basic environment, and will thus be deprotonated, leading to a negatively-charged species ($\alpha COO^-$).

It should be clear that, at physiological pH values the two α-chemical groups have a net charge of 0. But proteins are charged, and this is because some of the twenty common amino-acids have other chemical groups beyond the two others already described. Indeed, some amino-acids have lateral chains that bear groups that might be charged depending on the pH: seryl residues have an alcohol group that has a pKa of 13, for example; that means that it is almost always uncharged (form ROH at physiological pH values). The lateral chain of lysine has a pKa of 10.53, which means that at pH values below this pKa value, the $\epsilon NH_2$ gets protonated, introducing a positive charge in the protein. Similarly, amino-acids glutamate and aspartate do have a lateral chain ended with a $\gamma COOH$ and a $\beta COOH$, respectively. Their pKa values are below 4.5, and thus the groups are negatively charged a physiological pH values.

When the net charge of a polymer sequence has to be computed for a given pH condition, the program iterates in the sequence, and for each monomer will check which one of its chemical group(s) is possibly charged. For this to happen, it is required that a number of data be known for each monomer's chemical group that might play a role in the determination of the polymer sequence's electrical charge. Thus, for each chemical group a number of data should be listed in the `pka_ph_pi.xml` file (please, see that file in the file):

- the chemical group's <*name*> element is required. For example, "$\alpha NH_2$" or "$\epsilon NH_2$" or "$\alpha COOH$";

- the chemical group's <*pka*> element is optional, but is the basis for the charge calculation. For example, *9.6* for the "$\alpha NH_2$" or *2.35* for "$\alpha COOH$";

- the <*acidcharged*> element is required if the <*pka*> element is given. This element is responsible for telling if the chemical group is charged (positively) when the pH is lower than pKa (that is when the medium is acidic with respect to the pKa). For example, an amine is positively charged when it is in its acidic form (protonated); a carboxylic acid is *not* charged when it is in its acidic form;

- there can be none, one or more <*polrule*> element(s) for each chemgroup. The <*polrule*> element gives informations about the way the chemical group at hand might be "trapped" (or not) in the formation of inter-monomer bonds (while the monomer is polymerized into the polymer sequence). The value "left_trapped" means that the chemical group ceases to be involved in charge calculations as soon as it has a monomer at its left end. The value "right_trapped" means the same as above, but when a monomer is polymerized at its right end. For a chemical group that is "left_trapped", we understand that it is only effectively evaluated if it is at the left end of the polymer sequence, since in this case it does not have a monomer at its left side. Conversely, a chemical group that has a <*polrule*> element with value “right_trapped”, will be evaluated only if the monomer is actually the right end monomer in the polymer sequence. Finally, the typical lateral chains of amino-acids have a <*polrule*> element with a value “never_trapped”, as these chemical groups do not take part in the formation of the inter-monomer bond;

- there can be none, one or more <*chemgrouprule*> element(s) for each chemgroup. A chemgrouprule element should contain the following:
  - there must be an <*entity*> element that indicates what is the chemical entity being dealt with in the current chemgroup element. Valid values for this element are “LE_PLM_MODIF”, “RE_PLM_MODIF” or “MNM_MODIF”;
  - there must be a <*name*> element naming the chemical entity properly;
  - there must be an <*outcome*> element telling what action should be taken when encountering the <*entity*> on the chemgroup. Valid values are either “LOST” or “PRESERVED”.

#### 4.16.1.2 Understanding By Example

Let us take some examples in order to make sure we actually understand the process of describing how an electrical net charge is calculated for a given polymer sequence and at any given pH value.

Let us see the example of the aspartate amino-acid, of which the lateral chain is nothing but $CH_2COOH$:

```
<monomer> <code>D</code> <mnmchemgroup>
<name>N-term NH2</name> <pka>9.6</pka>
<acidcharged>TRUE</acidcharged>
<polrule>left_trapped</polrule> <chemgrouprule>
<entity>LE_PLM_MODIF</entity>
<name>Acetylation</name>
<outcome>LOST</outcome> </chemgrouprule>
</mnmchemgroup> <mnmchemgroup> <name>C-term
COOH</name> <pka>2.36</pka>
<acidcharged>FALSE</acidcharged>
<polrule>right_trapped</polrule> </mnmchemgroup>
<mnmchemgroup> <name>Lateral COOH</name>
<pka>3.65</pka>
<acidcharged>FALSE</acidcharged>
<polrule>never_trapped</polrule> <chemgrouprule>
<entity>MONOMER_MODIF</entity>
```

```
<name>AmidationAsp</name>
<outcome>LOST</outcome> </chemgrouprule>
</mnmchemgroup> </monomer>
```

We see that the code of the monomer for which acid-basic data are being defined is “D” and that this monomer has three chemical groups that might bring electrical charges. These chemical groups are described by three <*mnmchemgroup*> elements that we will review in detail below (see Figure 4.21, “ Different pKa values for a number of amino-acids' chemical groups ” ).

The first <*mnmchemgroup*> element is related to the $\alpha NH_2$ amino group of the amino-acid:

- <*name*>*N-term NH2*</*name*> The name of the chemical group is not immediately useful, but will be used when reports are to be prepared for the calculation;
- <*pka*>*9.6*</*pka*> This element is optional. However, of course, if the chemical group might be electrically charged, the pKa value will be essential in order to compute the charge that is brought by this chemical group at any given pH;
- <*acidcharged*>*TRUE*</*acidcharged*> This element is also optional, however, if the previous element is given, then this one is compulsory. Telling if the conjugated acid form is charged (that is protonated) is essential in order to know what sign the charge has to be when the chemical group is ionized. The value “TRUE” indicates that when the pH is lower than the pKa, the chemical group is charged, thus protonated (in the form $NH_3^+$). Consequently, if the pH is higher than the pKa, the chemical group is neutral (in the form $NH_2$);

- *<polrule>left_trapped</polrule>* This element indicates that the chemical group should only be taken into account in the eventuality that the monomer bearing it (code "D") is the left end monomer of the polymer sequence. This can easily be understood, as this chemical group is responsible for the establishment of the inter-monomer bond towards the left end of the polymer sequence;
- *<chemgrouprule>* This element provides further details on the chemistry that this chemical group might be involved in:
    - *<entity>LE_PLM_MODIF</entity>* This element indicates that the supplementary data in the current *<chemgrouprule>* element are pertaining to the $\alpha NH_2$ chemical group *only* in case the polymer sequence is left end-modified (that is with a permanent left end modification) and the monomer (code "D") is located at the left end of the polymer sequence (that is: it is the first monomer of the sequence for which the electrical charge—or pI—calculation is to be performed).
    - *<name>Acetylation</name>* This element goes further in the detail of the potential chemistry of the $\alpha NH_2$ chemical group: if the left end permanent modification is "Acetylation", then the current chemgrouprule element can be further processed, otherwise it should be abandoned;
    - *<outcome>LOST</outcome>* This element actually indicates what should be done with the chemical group for which the chemgrouprule is being defined. What we see here is: —*"If the $\alpha NH_2$ chemical group, belonging to a 'D' monomer located at the left end of a polymer sequence, is modified permanently with an 'Acetylation' left end modification, it should not be taken into account when computing the charge that it could bring to the polymer sequence".*

The second *<mnmchemgroup>* element is related to the αCOOH carboxylic group of the amino-acid:

- *<name>C-term COOH</name>* Same remark as above;
- *<pka>2.36</pka>* Same remark as above;
- *<acidcharged>FALSE</acidcharged>* Same remark as above. However, as we can see, the value indicates that the acid conjugate (form COOH) does not bring any charge. This means that when the basic conjugate is predominant (that is when pH > pKa), it brings a negative charge: the form is $COO^-$;
- *<polrule>right_trapped</polrule>* The chemical group should not be evaluated if a monomer is linked to it at its right side. That means that the current chemical group is only evaluated if the monomer bearing it is located at the right end of the polymer sequence. This is easily understood, as the αCOOH chemical group is involved in the formation of the inter-monomer bond towards the right end of the polymer sequence.

The third *<mnmchemgroup>* element is related to the βCOOH carboxylic group of the amino-acid:

- *<name>Lateral COOH</name>*;
- *<pka>3.65</pka>*;

- *<acidcharged>FALSE</acidcharged>*;
- *<polrule>never_trapped</polrule>* This element indicates that, whatever the position of the monomer bearing the chemical group in the polymer sequence (left end, right end or middle), the chemical group is to be evaluated;
- *<chemgrouprule>* This element provides further details on the chemistry that the chemical group at hand (βCOOH) might be involved in:
    - *<entity>MONOMER_MODIF</entity>* This element indicates that the supplementary data in the current *<chemgrouprule>* element are pertaining to the βCOOH chemical group *only* in case the monomer bearing the chemical group is chemically modified;
    - *<name>AmidationAsp</name>* This is the modification by which the monomer should be modified in order to have the *<chemgrouprule>* element effectively evaluated;
    - *<outcome>LOST</outcome>* This element actually indicates that if the monomer bearing the chemical group is modified with an “AmidationAsp” chemical modification, then the chemical group should not be evaluated any more for the electrical charge —or pI— calculations, since reacting a carboxylate group with an amino group produces an amide group which is not easily chargeable at physiological pH values.

At this point we should have made it clear how the charge calculations can be configured for the different monomers in the polymer chemistry definition. As usual, the more the polymer chemistry definition is sophisticated, the more sophisticated the computations are allowed.

### 4.16.2 Ionized Group(s) In Modifications

In the excerpt from the `pka_ph_pi.xml` file below, we see that chemical modifications can also bring charges. The example of the chemical modification “Phosphorylation” shows that when a monomer is phosphorylated, two chemical groups are brought in: the first has a pKa value of 1.2 (that is it will always be deprotonated at physiological pH values), the second has a pKa value of 7 (that is it will be divided by half in a protonated (not charged) form and in an un-protonated (negatively charged) form, leading to a net electrical charge of -0.5.

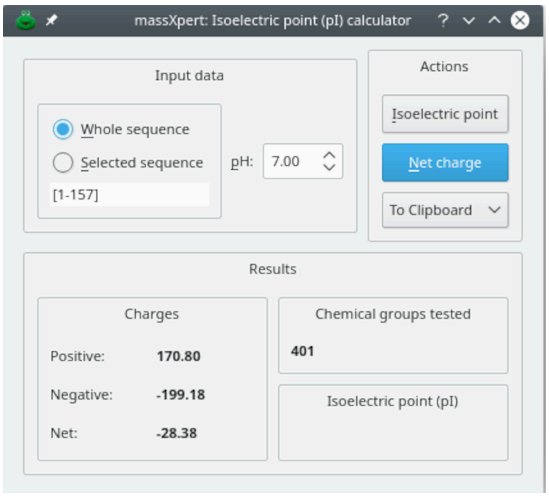
```
<modif> <name>Phosphorylation</name>
<mdfchemgroup> <name>none_set</name>
<pka>1.2</pka> <acidcharged>FALSE</acidcharged>
</mdfchemgroup> <mdfchemgroup>
<name>none_set</name> <pka>6.5</pka>
```

```
    <acidcharged>FALSE</acidcharged> </mdfchemgroup>
    </modif>
```

At this point we should be able to study the way computations are actually performed in the XpertEdit module.

### 4.16.3 pH, pI and Charge Calculations

The user willing to compute charges (positive, negative, net) or the isoelectric point for the current polymer sequence uses the menu *Chemistry*→ pKa pH pI which triggers the appearance of the window shown in FIGURE 4.22, “ ACIDO-BASIC COMPUTATIONS: NET CHARGES ” .



● Whole sequence
○ Selected sequence
[1-157]
pH: 7.00

Positive: 170.80
Negative: -199.18
Net: -28.38

Chemical groups tested: 401

This figure shows the options that can be set for the calculation of the charges beared by the polymer sequence.

FIGURE 4.22: ACIDO-BASIC COMPUTATIONS: NET CHARGES

This figure shows that the user can calculate the charges (positive, negative and net) beared by the polymer sequence (either the whole sequence or the current selection) by setting the *pH* value at which the computation should take place. It is also possible to calculate the isoelectric point by clicking onto the *Isoelectric Point* button.

Note that the computations might involve the permanent left/right modifications of the polymer sequence, as well as the monomer chemical modifications. To configure the way net charge—or pI—calculations are performed, use the calculations engine configuration of the sequence editor window.

## 4.17 GENERAL OPTIONS

One of the options that are valued most by users is to be able to set the number of decimal places used to diplay numbers. The settings should apply in a distinct manner depending on the different entities for which numerical values are to be displayed. The following are the default values (and recommended ones):

- Atoms (and all related entities (isotopic masses, isotopic abundances): 10;
- pKa, pH, pI: 2;
- Oligomers (obtained *via* mass searches, polymer cleavages, oligomer fragmentations): 5;
- Polymers : 3;

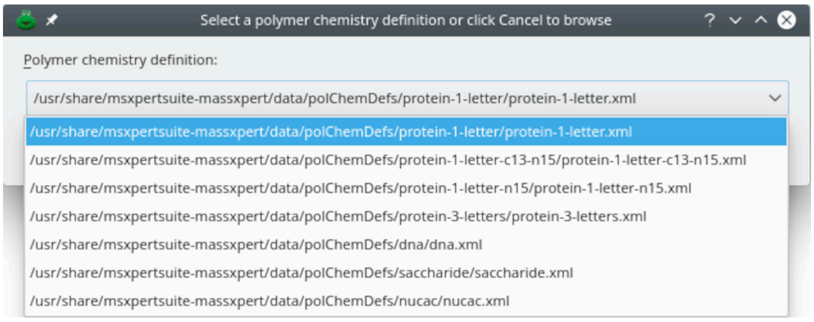
Note that modifying these values will allow immediate change of the way numerals are displayed, without needing to restart the program. Only triggering a new cleavage or a new fragmentation will update the data display according to the new options set. These options are stored on the disk and are permanent.

# 5 XpertCalc: a Powerful Mass Calculator

After having completed this chapter you will be able to perform sophisticated polymer chemistry-aware mass calculations.

## 5.1 XpertCalc Invocation

The XpertCalc module is easily called by pulling down the XpertCalc menu item from the MassXpert program's menu. The user is presented with a window to select the polymer chemistry definition that should be used for the calculations ( Figure 5.1, “ Selecting a polymer chemistry definition for use with XpertCalc ” ).



Select a polymer chemistry definition or click Cancel to browse

Polymer chemistry definition:

/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter/protein-1-letter.xml

/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter/protein-1-letter.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter-c13-n15/protein-1-letter-c13-n15.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-1-letter-n15/protein-1-letter-n15.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/protein-3-letters/protein-3-letters.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/dna/dna.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/saccharide/saccharide.xml
/usr/share/msxpertsuite-massxpert/data/polChemDefs/nucac/nucac.xml

This figure shows that the user can either select one already registered polymer chemistry definition (listed in the drop-down widget) or browse the filesystem to select one polymer chemistry definition file. Choosing a polymer chemistry definition allows to take advantage, during the mass calculations, of all the chemical entities defined therein.

FIGURE 5.1: **SELECTING A POLYMER CHEMISTRY DEFINITION FOR USE WITH XPERTCALC**

## 5.2 An Easy Operation

Once the polymer chemistry definition has been correctly selected, it is parsed by the XpertCalc module and its entities are automatically made available in the calculator window, as shown in Figure 5.2, “ Interface of the XpertCalc module ” . The way XpertCalc is operated is very easy. This is partly due to the very self-explanatory graphical user interface of the module, which is illustrated in Figure 5.2, “ Interface of the XpertCalc module ” . XpertCalc can handle a number of items that are reviewed below:

- The user may (is not obliged to) seed the calculation by setting masses manually in the *Seed masses* line edit widgets (the left line edit is for *mono* and the right one for *avg*);

### WARNING

Both monoisotopic and average m/z values need to be entered.

For example, imagine that a mass spectrum analysis session ends up like this: — *“There is a peak with m/z 1000.55, z=1 and another one roughly 80 Da more. Is it possible that the analyte showing up at m/z 1000.55 is phopshorylated?”*. The mass spectrometrist would seed the calculator with mass *1000.55* and ask that one *Phosphorylation* modification be added to it by setting *1* in front of the corresponding drop-down widget. Clicking *Apply* triggers the calculation, with the resulting masses being displayed in the *Result masses* line edit widgets. We can see that the phosphorylation of our analyte shifts its m/z value from *1000.55* to *1080.5163*.

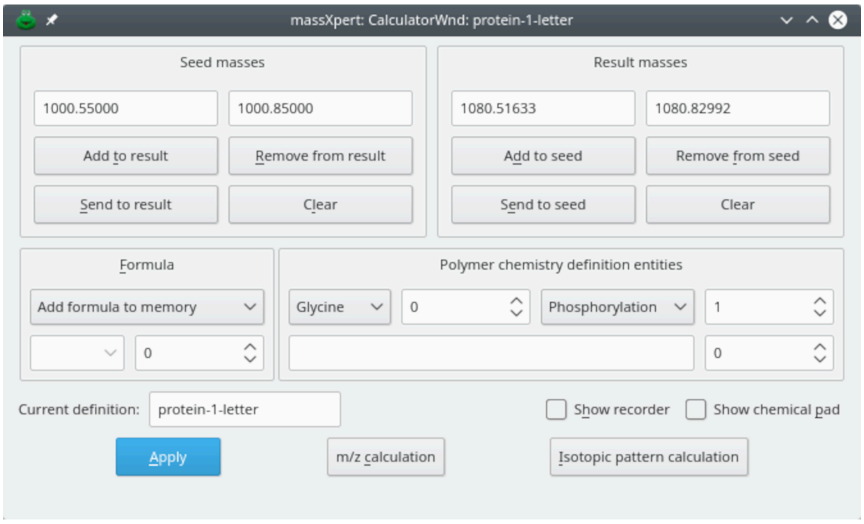
### TIP

Each time a calculation is triggered by clicking *Apply* (or the chemical pad's buttons; see below), the values already present in the *Result masses* line edit widgets are transferred to the *Seed masses* line edit widgets. This provides a 1-level undo;

- The *Formula* group box widget contains two widgets: a line edit widget where the formula is typed and a count spin box widget where the user sets the number of times that the formula should be applied. Setting the formula to $H2O$ and the count to *2* would hydrate the analyte twice.

- The *Polymer Chemistry Definition Entities* group box widget contains two drop-down widgets and a line edit widget. The drop-down widget on the left lists all the monomers defined in the *protein-1-letter* polymer chemistry definition; the drop-down widget on the right lists all the modifications defined in the *protein-1-letter* polymer chemistry definition. Each drop-down widget has its corresponding count spin box widget. In the example, the user asked that one (*1*) *Phosphorylation* modification be applied during the calculation. The line edit widget below the first row of widgets is the polymer sequence widget where the user might enter a sequence of monomers. It is possible to apply many times the sequence by setting the count spin box widget value to something greater than 1 (either positive or negative);

It is possible to perform a set of calculations in one go, that is, the user may ask for a formula, a monomer, a modification, a sequence to be accounted in one single calculation operation. Once all the chemical entities to be taken into account have been set, the user clicks *Apply* : all the entities are parsed in sequence and their mass equivalent are added to the result masses. Other prominent features of XpertCalc are described in the following sections.



Seed masses: 1000.55000 1000.85000

Result masses: 1080.51633 1080.82992

Glycine 0 Phosphorylation 1

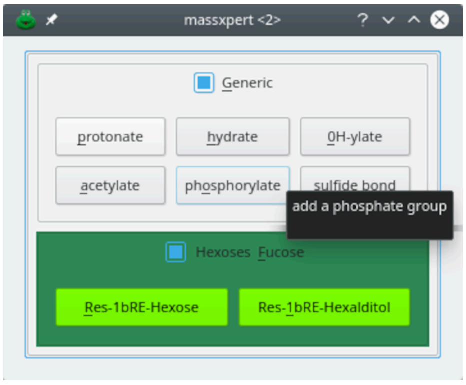
Current definition: protein-1-letter

This figure shows that the XpertCalc polymer chemistry definition-aware module can handle atoms, formulæ, monomers, modifications and even polymer sequences for computing masses.

FIGURE 5.2: **INTERFACE OF THE XPERTCALC MODULE**

## 5.3 THE PROGRAMMABLE CALCULATOR

For the scientists who work on molecules that are often modified in the same usual ways, XpertCalc features a built-in mechanism by which they can easily program their calculator. This programming involves the definition of how a *chemical pad* (or *chempad*) may be arranged, exactly the same way as a desktop calculator would display its numerical keypad.



This figure shows that the chemical pad is very similar to what a numerical calculator would display. Here, the user has programmed a number of chemical reactions.

FIGURE 5.3: **INTERFACE OF THE CHEMICAL PAD**

The chemical pad can be shown/hidden by using the *Show Chemical Pad* check box widget. An example of such a chemical pad is shown in FIGURE 5.3, “ INTERFACE OF THE CHEMICAL PAD ” , where a “protein-1-letter” polymer chemistry definition-associated chempad is featured. As shown, the user has programmed a number of chemical reactions that may be applied to the masses in the XpertCalc calculator window by simply clicking on their respective button (see Figure 5.3, “ Interface of the chemical pad ” ). The configuration of the chemical pad is very easy.

### Tip

It is recommended to copy one of the `chemPad.conf` configuration files in any of the polymer chemistry definitions distributed within MassXpert and to modify it according to the instructions at the top of the file.

One example of such a configuration file is shown below along with explanations:

```
color%aliceblue%240,248,255
❶
color%antiquewhite%250,235,215 color%aqua%0,255,255 chempad_columns%3
❷
chempadgroup%Generic
❸
chempadkey=protonate%+H1%adds a proton
❹
chempadkey=hydrate%+H2O1%adds a water molecule
chempadkey=0H-ylate%+O1H1%adds an hydroxyl group
chempadkey=acetylate%-H1+C2H3O1%adds an acetyl group
chempadkey=phosphorylate%-H+H2PO3%add a phosphate group
chempadkey=sulfide bond%-H2%oxydizes with loss of hydrogen
chempadgroup%Hexoses & Fucose%[seagreen]
❺
chempadkey%Res-1bRE-Hexose%C6H11O6%residue Hexose
(1bRE)%[lawngreen,black]
❻
chempadkey%Res-1bRE-Hexalditol%C6H12O6%residue Hexalditol
(1bRE-ol)%[lawngreen,black]
```

❶ It is possible to define as many colors as necessary (red,blue, green format, on a scale of 0–255).

❷ The calculator chemical pad shall have its buttons organized in three columns.

❸ This separator will create a group box widget labelled "Generic" that will be populated with all the items found below (until another separator is encountered).

❹ A button definition is introduced by the “chempadkey=” string. Separated by “%” characters, follow the name of the chemical reaction that will label the button (“protonate”), the chemical formula of the reaction (“+H2O1”) and finally the tooltip text that displays when the cursor stays on the button.

❺ Separator that starts a new button group box labelled “Hexoses & Fucose”. This syntax allows for the coloring of the group box widget.

❻ A button definition that also specifies the coloring of the button. “lawngreen” is the background color and “black” is the color of the text.

These buttons might be used in two distinct ways:

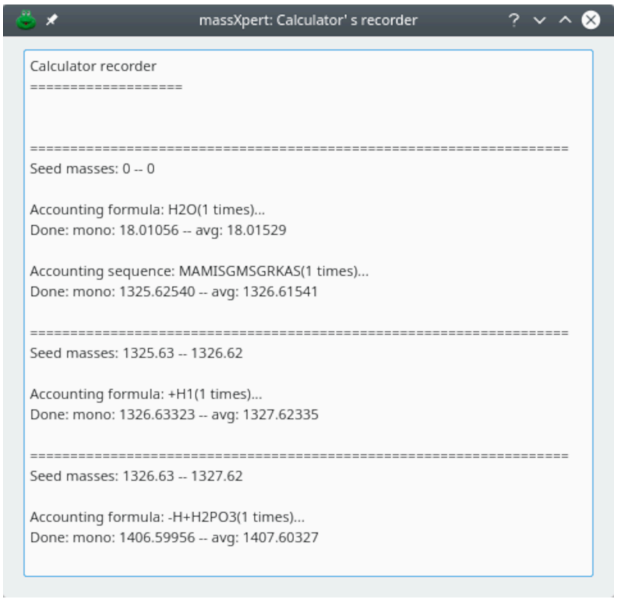
- Upon clicking the button, its formula is evaluated and the corresponding masses are added to (or subtracted from) the *Result masses*;
- Upon simultaneous clicking the button and keeping the `Ctrl` key pressed, its formula is inserted into the *Formula* line edit widget. In this case, the formula is not evaluated and the *Result masses* are not modified.

### TIP

Clicking sequentially on various chemical pad buttons, append the formulæ in the *Formula* line edit widget, which can be useful for storing the whole formula string in the memory using the *Add formula to memory* menu item from the drop-down menu before clicking *Apply*.

## 5.4 THE LOG BOOK RECORDER

Each time an action that is chemically relevant—from a molecular mass point of view—is performed, the program dumps the calculations to the XpertCalc recorder window ( FIGURE 5.4, “ THE XPERTCALC RECORDER WINDOW ” ). The recorder can be shown/hidden by using the *Show Recorder* check box widget. The text in the recorder window is editable for the user to edit the XpertCalc output, and selectable also, so that pasting to text editors or word processors is easy *via* the clipboard.



```
Calculator recorder
====================



=======================================================================
Seed masses: 0 -- 0

Accounting formula: H2O(1 times)...
Done: mono: 18.01056 -- avg: 18.01529

Accounting sequence: MAMISGMSGRKAS(1 times)...
Done: mono: 1325.62540 -- avg: 1326.61541

=======================================================================
Seed masses: 1325.63 -- 1326.62

Accounting formula: +H1(1 times)...
Done: mono: 1326.63323 -- avg: 1327.62335

=======================================================================
Seed masses: 1326.63 -- 1327.62

Accounting formula: -H+H2PO3(1 times)...
Done: mono: 1406.59956 -- avg: 1407.60327
```

This figure shows that the recorder window is a simple text edit widget that records all the mass-significant operations in the XpertCalc calculator. The text in the recorder may be selected and later used in an electronic logbook or printed.

FIGURE 5.4: **THE XPERTCALC RECORDER WINDOW**

## 5.5 THE M/Z RATIO CALCULATOR

It very often happens that the mass spectrometrist doing electrospray analyzes is faced with a challenging task: to compute by mind all the m/z ratios for a given family of charge peaks. To ease that daunting task, XpertCalc contains a m/z ratio calculator that is called by clicking onto the *m/z calculation* button.

The m/z ratio calculator has been described at SECTION 4.14, " M/Z RATIO CALCULATIONS " (see Figure FIGURE 4.19, " CALCULATION OF RANGES OF M/Z RATIOS " .

## 5.6 THE ISOTOPIC PEAKS CALCULATOR

This section is no more appropriate for version 6.0.0 and upper as it is now implemented in MineXpert2. Please refer to that software user manual.

# 6 XPERTMINER: A DATA MINER

XpertMiner is a module that has been conceived as a repository of functionalities aimed at analyzing mass data. The data to be subjected to mining can originate from:

- MassXpert-based simulations, like polymer cleavage, oligomer fragmentation or arbitrary mass searches;
- An export of a mass list from the mass spectrometer software;
- Any mass data that might have been processed outside of MassXpert and that need to be reimported in XpertMiner.

## 6.1 XPERTMINER INVOCATION

The XpertMiner module is easily called by pulling down the *XpertMiner* menu item from the MassXpert program's menu. Clicking on *XpertMiner*→ mzLab will open the mzLab window, as represented in FIGURE 6.1, “ MZLAB WINDOW ” .

## 6.2 MZLAB: MINING M/Z RATIOS

The features available in this laboratory operate on lists of m/z values in the form of (m/z,z) pairs. The mass of the ion is represented by “m”, while “z” is the charge of the ion. With the two data in the pair, the m/z ratio and the z charge, and knowing the ionization rule that ionized the analyte in the first place, it is possible to perform any mass calculation on the (m/z,z) pair.

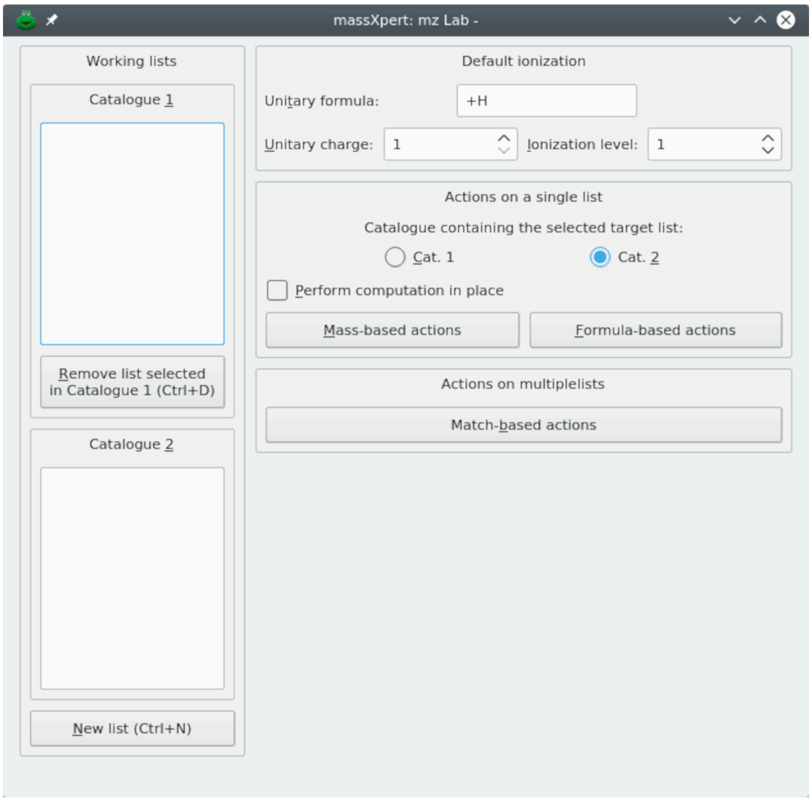
The mzLab window is represented in FIGURE 6.1, “ MZLAB WINDOW ” . This window is divided into a number of distinct parts:

- The left part (*Working lists*) contains two list widgets which will hold the names of the different working m/z lists. We call these lists “catalogues” in the following text;
- The *Default ionization* groupbox widget contains the ionization rule that is to be assumed when working on (m/z,z) pairs. If you are unsure about this concept, please read section SECTION 3.2, “ THE POLYMER CHEMICAL ENTITIES ” .
- The *Actions on a single list* groupbox widget holds a number of mining actions to be performed on a single list that is identified by being selected either in the *Cat. 1* or in the *Cat. 2* catalogue of available m/z lists. When performing computations that modify the m/z values in the list, if *Perform computation in place* is checked, then the new m/z values will replace the former ones. Otherwise, the program will ask for a new m/z list name as a new list is created to hold the new m/z values resulting from the computation.

There are two main kinds of computations that might be performed against a single m/z list:

  - *Mass-based actions* rely on masses or m/z values to perform computations;
  - *Formula-based actions* rely on formulæ to perform computations;

- The *Actions on multiple lists* groupbox widget allows one to perform actions that use two lists, for example matching masses (or m/z ratios) in two lists with a given tolerance.



Unitary formula: +H

Unitary charge: 1

Ionization level: 1

○ Cat. 1 ● Cat. 2

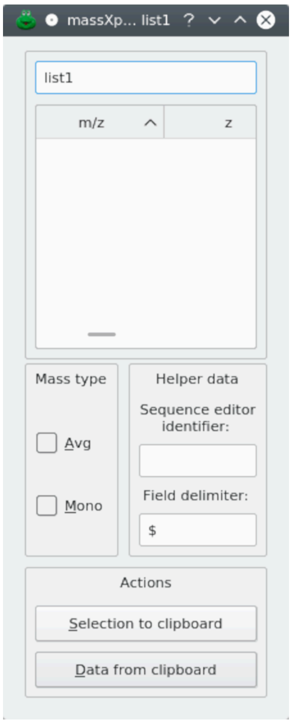
☐ Perform computation in place

XpertMiner's laboratory window. From there it is possible to create any number of m/z list dialog windows, to fill-in m/z data and start making computations, like changing the ionization; applying arbitrary masses, formulæ or m/z ratios; matching masses or m/z ratios.... See text for details.

**FIGURE 6.1: MZLAB WINDOW**

## 6.3 Creating a New Input m/z List

In order to be able to use the mzLab, it is necessary to create at least one list of (m/z,z) pairs, which is referred to by “input m/z list”, for short. To create a new input m/z list, you click *New list*. An input dialog window let's you enter the name of the new list. The new input m/z list dialog window shows up empty like in Figure 6.2, “ m/z list's empty input m/z list dialog window. ” . That kind of list is actually a table view widget that is embedded in a dialog window. The first column of the table view widget holds the m/z value, and the second column, the z value. Optionally, the name of the corresponding oligomer and its coordinates in the polymer sequence can be shown, depending on how data have been imported into the m/z list (see below).



An empty input m/z list dialog window is empty upon its creation. Filling that list is performed by either drag-and-drop or clipboard operations.

**FIGURE 6.2: M/Z LIST'S EMPTY INPUT M/Z LIST DIALOG WINDOW.**

### Tip

The list name entered by the user at creation time will be used to refer to that list in the two catalogues.

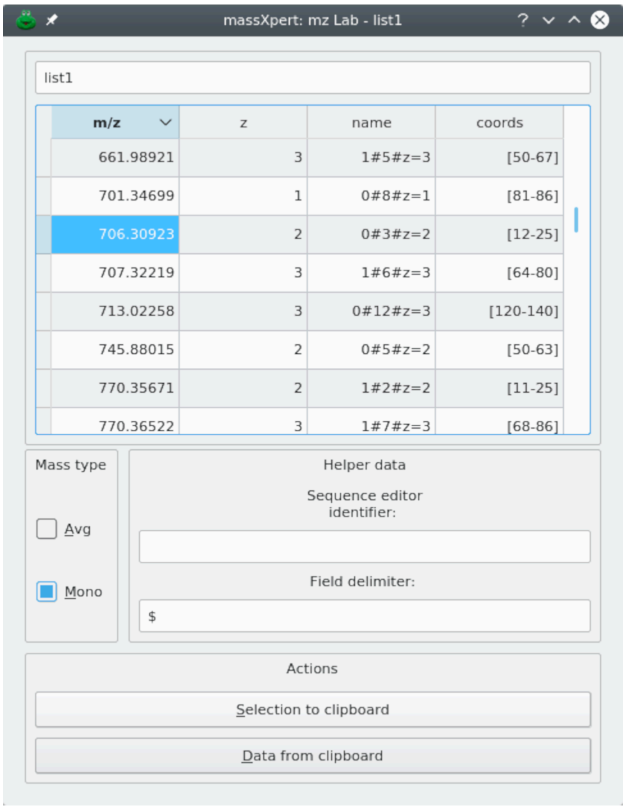
### 6.3.1 Filling m/z Lists with Data

Once a new input m/z list has been named and created, it is necessary to fill it with (m/z,z) pairs. This is performed *via* drag-and-drop or clipboard operations. There might be a number of different data sources to be used for filling the input m/z list, all reviewed in the following sections.

### 6.3.1.1 Importing Data from MassXpert Results Windows

Data from the various simulations available in MassXpert include cleavage results, fragmentation results and mass search results, which all produce oligomers that are displayed in treeview widgets, as shown in Figure 4.13, “ Polymer sequence cleavage window ” or Figure 4.14, “ Oligomer fragmentation window ” or Figure 4.17, “ Searching masses in a polymer sequence ” .

From these results windows, either select the oligomers of interest and export these data to the clipboard or perform a drag-and-drop operation to the m/z list area. Both ways produce identical results, as described in Figure 6.3, “ m/z list's data-filled input m/z list dialog window ” . One can see that the mass of the oligomers is set in the list, along with the charge, the oligomer name and, finally, the coordinates of the oligomer in the corresponding polymer.



list1

| m/z | z | name | coords |
|---|---|---|---|
| 661.98921 | 3 | 1#5#z=3 | [50-67] |
| 701.34699 | 1 | 0#8#z=1 | [81-86] |
| 706.30923 | 2 | 0#3#z=2 | [12-25] |
| 707.32219 | 3 | 1#6#z=3 | [64-80] |
| 713.02258 | 3 | 0#12#z=3 | [120-140] |
| 745.88015 | 2 | 0#5#z=2 | [50-63] |
| 770.35671 | 2 | 1#2#z=2 | [11-25] |
| 770.36522 | 3 | 1#7#z=3 | [68-86] |

☐ Avg

☑ Mono

Sequence editor identifier:

Field delimiter: $

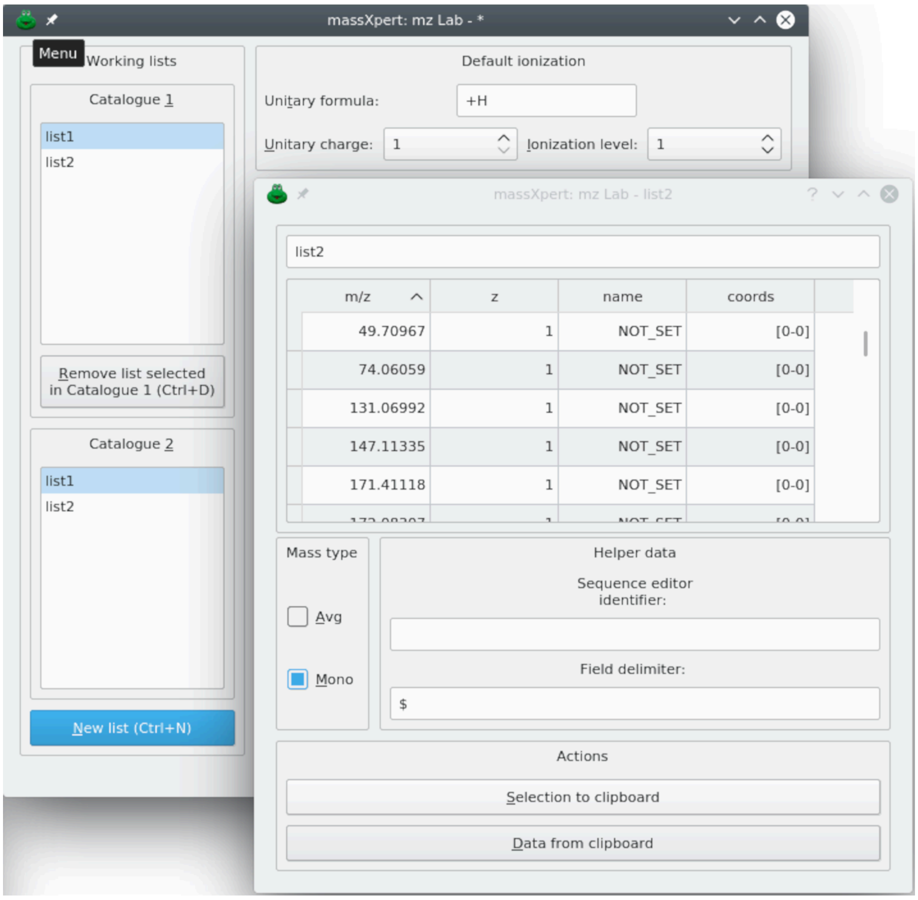
Data pasted from any MassXpert result window hold all the necessary data to fill-in fully the m/z list. The user is asked to specified if the imported data are for monoisotopic or average masses.

**FIGURE 6.3: M/Z LIST'S DATA-FILLED INPUT M/Z LIST DIALOG WINDOW**

### 6.3.1.2 Textual Data from Non-MassXpert Results Windows, Without Charge

The mass data might be first copied to the clipboard from other software and then imported in the m/z list by clicking *Data from clipboard* (a drag-and-drop operation would also work).

When the charge z is not present in the imported m/z list, then it is deduced from the ionization rule currently defined the the mzLab window (see background of FIGURE 6.4, “ M/Z LIST'S (M/Z) TEXTUAL DATA-FILLED INPUT M/Z LIST DIALOG WINDOW. ” ).



Unitary formula: +H

Unitary charge: 1

Ionization level: 1

list2

| m/z | z | name | coords |
|---|---|---|---|
| 49.70967 | 1 | NOT_SET | [0-0] |
| 74.06059 | 1 | NOT_SET | [0-0] |
| 131.06992 | 1 | NOT_SET | [0-0] |
| 147.11335 | 1 | NOT_SET | [0-0] |
| 171.41118 | 1 | NOT_SET | [0-0] |

☐ Avg

☑ Mono

Sequence editor identifier:

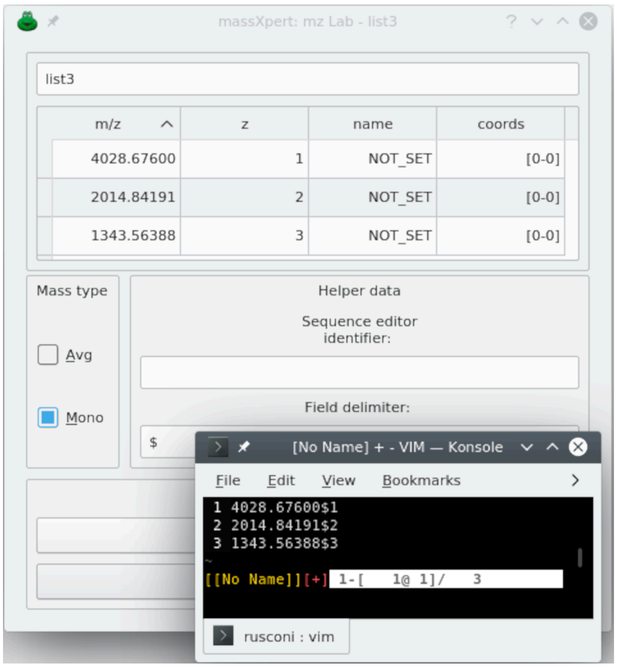
Field delimiter: $

When mass data missing the z charge value are pasted from the clipboard, the z value is assumed to be the result of the ionization rule defined in the mzLab window (the window on the background). See text for details.

FIGURE 6.4: **M/Z LIST'S (M/Z) TEXTUAL DATA-FILLED INPUT M/Z LIST DIALOG WINDOW.**

### 6.3.1.3 TEXTUAL DATA FROM NON-MASSXPERT RESULTS WINDOWS, WITH Z CHARGE SPECIFIED

When the mass data copied to the clipboard do include the m/z ratio along with the z charge, the delimiter character needs to be known. This character must be set as the *Field delimiter* in the m/z list prior to clicking *Data from clipboard* to actually import the data. This way, the program knows how to parse the (m/z,z) pairs. A drag-and-drop operation from a graphical text editor would have produced the same results. The obtained list is shown in FIGURE 6.5, “ M/Z LIST'S (M/Z,Z) TEXTUAL DATA-FILLED INPUT M/Z LIST DIALOG WINDOW. ” .



list3

| m/z | z | name | coords |
|---|---|---|---|
| 4028.67600 | 1 | NOT_SET | [0-0] |
| 2014.84191 | 2 | NOT_SET | [0-0] |
| 1343.56388 | 3 | NOT_SET | [0-0] |

Mass type
☐ Avg
☑ Mono

Sequence editor identifier:

Field delimiter: $

```
1 4028.67600$1
2 2014.84191$2
3 1343.56388$3
~
[[No Name]][+] 1-[    1@ 1]/    3
```

The textual data contain (m/z,z) pairs (delimited with “$”). The list contains both m/z and z data. See text for details.

FIGURE 6.5: **M/Z LIST'S (M/Z,Z) TEXTUAL DATA-FILLED INPUT M/Z LIST DIALOG WINDOW.**

### 6.3.1.4 GENERAL RULES ON TEXTUAL MASS DATA FORMAT

The most detailed format that is supported is the following:

m/z <delim> charge <delim> name <delim> coordinates <delim>

In this syntax, the <delim> (field delimiter) is the “$” character. Any character might be used (including spaces). The delimiter character (or string) that is set in the m/z list window must be the same as the one defined in the window from where the data originate (when using the option to export the selected oligomers data to the clipboard).

For example, data can be formatted like this:

```
    3818.05262$1$0#2#z=1$[3-39] 3834.05262$1$0#2#z=1$[3-39]
```

The compulsory datum (that is, the imported datum, either dragged and dropped or pasted from the clipboard), is the *m/z ratio*. The charge, name, coordinates fields are optional. If the charge is present, it will be taken into account while preparing the data for further use by the m/z list. If the charge is absent, it is deduced from the ionization rule currently defined in the mzLab window ( FIGURE 6.1, “ MZLAB WINDOW ” ).

### WARNING

If there is no charge value, then the other name and coordinates fields cannot be filled (or an error will result). The presence of the name and coordinates fields is optional. Note, however that the coordinates field is *fundamental* to be able to highlight the corresponding region in the XpertEdit sequence editor upon double-clicking of any given item in the m/z list. For this to be possible, the data must have been originated by drag and drop from a MassXpert simulation results window *or* the m/z list window must have been connected to a polymer sequence editor window (see below).

### 6.3.2 IMPOSING THE MASS TYPE: MONO OR AVG

When dropping data—either from MassXpert-driven simulations (cleavage, fragmentation or mass search) or from textual data originating from outside MassXpert—it is necessary to inform the input m/z list of what kind of mass it is dealt with. That is, when dropping a line like “1234.56 1”, the question is: —“The m/z 1234.56 value is a monoisotopic m/z or an average m/z?” The type of the masses dropped in an input m/z list is governed by the two radio buttons labelled *Mono* and *Avg*. The one of the two radiobuttons that is checked at the moment the drop or the clipboard-paste occurs determines the type of the masses that are dealt with. It will be possible to check the other radio button widget once a first data drop occurred, but then the user will be alerted about doing so, as this has huge implications for the calculations to be performed later.

## 6.4 WORKING ON ONE INPUT m/z LIST

Once an input m/z list has been filled with data, it becomes possible to perform calculations on these data. Because there might be any number of input m/z lists open at any given time, it is necessary to identify the input m/z list onto which to perform these calculations. The selection of the input m/z list(s) is performed in two steps: first, by indicating in which catalogue the list of interest is currently selected (select either *Cat.1* or *Cat.3*). Make sure a list name is currently selected in the proper catalogue.

### 6.4.1 AVAILABLE CALCULATIONS

There are a number of operations that might be performed, all of which are selectable in the *Actions on a single list* groupbox widget. The simulations are organized into two groups:

- *Formula-based* actions which involve processing the input m/z lists with formulæ (that is, chemical entities represented using formulæ):
    - *Apply formula* will modify the m/z list by applying to all of its members the mass corresponding to the formula entered by the user. This is where it is crucial that the mass type (mono or avg) be set correctly, because the type of the mass calculated for the formula must be of the same type as the type of the data;
    - *Increment charge by* will iterate in all the items present in the list and apply the charge increment to them. For example, one item in the list that is charged 1 will be deionized and reionized to 2 (this calculation involves the ionization rule of the oligomer, and thus its ionization formula);
    - *Reionization* will iterate in all the items present in the list and apply the new ionization rule, defined in this groupbox widget.
- *Mass-based* actions which involve processing the input m/z lists with numerical data representing masses:
    - *Apply mass* will iterate in all the items present in the list and apply the entered mass to them. The value entered by the user is a *mass*, not a m/z ratio. Thus, this computation involves, for each (m/z,z) pair in the list the sequential deionization, mass addition, reionization.
    - *Apply threshold* will remove all data items in the list for which m/z or M is less than the value set, depending on the radio button that is selected (*On m/z value* or *On M value*).

### 6.4.2 OUTPUT OF THE CALCULATIONS

Simulations performed on a single input m/z list produce a m/z list that is identical to the input list, unless for the m and/or z values, which might have changed. This means that it is perfectly possible to:

- Overwrite the initial data with the newly obtained ones (this is performed by checking the *Perform computation in place* check button widget);
- Create a new list with the newly obtained data. As a convenience for the user, the new list will be an input m/z list in which it will be possible to perform ulterior simulations. This is useful when the simulations that need to be performed are sequential in kind. To have a new list created uncheck *Perform computation in place*.

### 6.4.3 INTERNAL WORKINGS

When an operation is performed on the items of an input m/z list, say we want to make sodium adducts (that would be a formula “-H+Na”) of all the items in the list, the process involves the following steps, as detailed below for one single item of the list (which has data pair (334.341,3) and *protonation* as ionization agent).

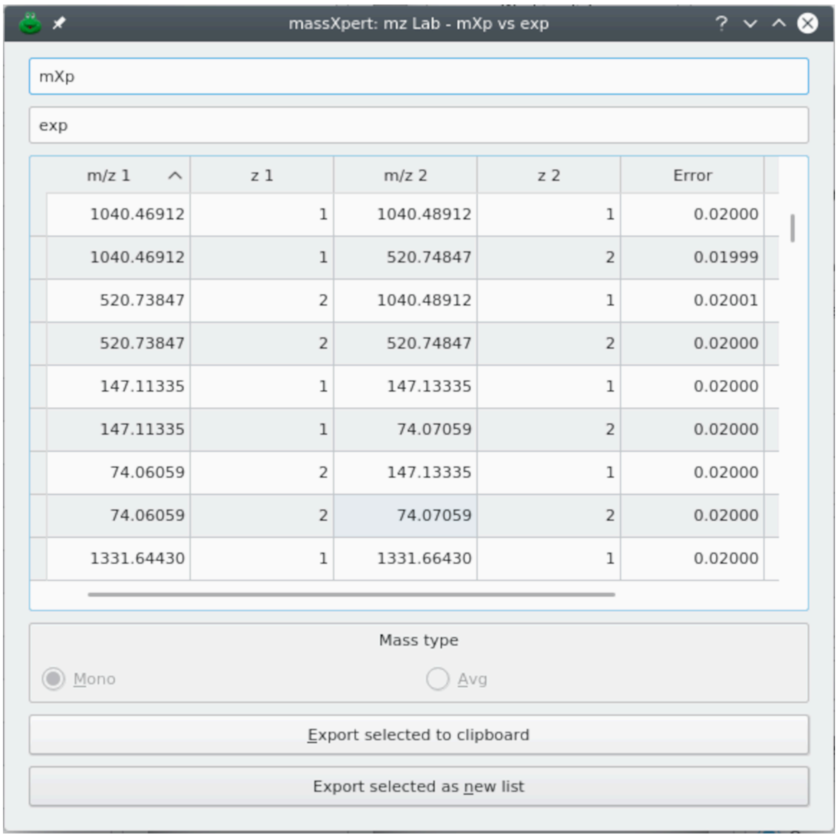
- Convert the tri-protonated analyte into a non-ionized analyte, thus getting M=1000;
- Compute the mass of the “-H+Na” formula: 21.98 Da;
- Add 1000+21.98;
- Reionize to the initial charge state: (341.67,3).

## 6.5 WORKING ON TWO INPUT M/Z LISTS

It is possible to perform calculations on two input m/z lists. These calculations are called matches. The (m/z,z) pairs of two different input m/z lists might be matched. Typically, a match operation would involve data from the mass spectrometer and data from a MassXpert-based simulation (cleavage or fragmentation, for example). In order to perform a match operation, the first input m/z list (the data from the mass spectrometer) should be selected by its name in the *Catalogue* list and the second input m/z list (the data from the simulation) should be selected by its name in the *Catalogue 2* list. Note that if the two input m/z lists are not of the same type (one is mono and the other is avg), the user will be alerted about this point.

### 6.5.1 OUTPUT OF THE CALCULATIONS

Calculations involving matches between two input lists produce an output that is displayed in an output m/z list, which is different from an input m/z list. FIGURE 6.6, “ MATCH OPERATION BETWEEN TWO M/Z LISTS, OUTPUT LIST DIALOG WINDOW. ” shows the results after having performed a match operation between an input m/z list obtained from the mass spectrometer (*Catalogue 1*) and an input m/z list obtained by simulating a cleavage with trypsin (*Catalogue 2*). The output m/z list dialog window holds all the matches along with the original data and the error.



mXp

exp

| m/z 1 | z 1 | m/z 2 | z 2 | Error |
|---|---|---|---|---|
| 1040.46912 | 1 | 1040.48912 | 1 | 0.02000 |
| 1040.46912 | 1 | 520.74847 | 2 | 0.01999 |
| 520.73847 | 2 | 1040.48912 | 1 | 0.02001 |
| 520.73847 | 2 | 520.74847 | 2 | 0.02000 |
| 147.11335 | 1 | 147.13335 | 1 | 0.02000 |
| 147.11335 | 1 | 74.07059 | 2 | 0.02000 |
| 74.06059 | 2 | 147.13335 | 1 | 0.02000 |
| 74.06059 | 2 | 74.07059 | 2 | 0.02000 |
| 1331.64430 | 1 | 1331.66430 | 1 | 0.02000 |

Mass type

● Mono ○ Avg

See text for details

**FIGURE 6.6:** MATCH OPERATION BETWEEN TWO M/Z LISTS, OUTPUT LIST DIALOG WINDOW.

### 6.5.2 TRACING THE DATA

When the data used for filling an input m/z list come from a MassXpert-based simulation it is possible to trace back the (m/z,z) pair items to the corresponding sequence in the polymer sequence editor that gave rise to these oligomers in the first place. This is only possible if:

- The way the data were fed into the input m/z list was by dragging oligomers from the treeview widgets, as described earlier;
- The polymer sequence window is still opened when the tracing back is tried.

If the data do not originate immediately from the MassXpert-based simulations (that is, the data do not originate directly from results treeview of cleavages or fragmentations or mass searches), it is still possible to perform the highlighting of corresponding oligomers in the sequence editor window, provided that the following requirements are met:

- The data imported in the m/z list of m/z list are rich, that is, they comprise the coordinates data in the right format (see the data format, SECTION 6.3.1.4, " GENERAL RULES ON TEXTUAL MASS DATA FORMAT " (PAGE 94) ;
- The proper polymer sequence is opened in a sequence editor window;
- The identifier in the *This window identifier* line edit widget of the sequence editor window above is copied into the *Sequence editor identifier* line edit widget.

In the case the above conditions are met, double-clicking onto a item of the m/z list will highlight the corresponding sequence region in the sequence editor window.

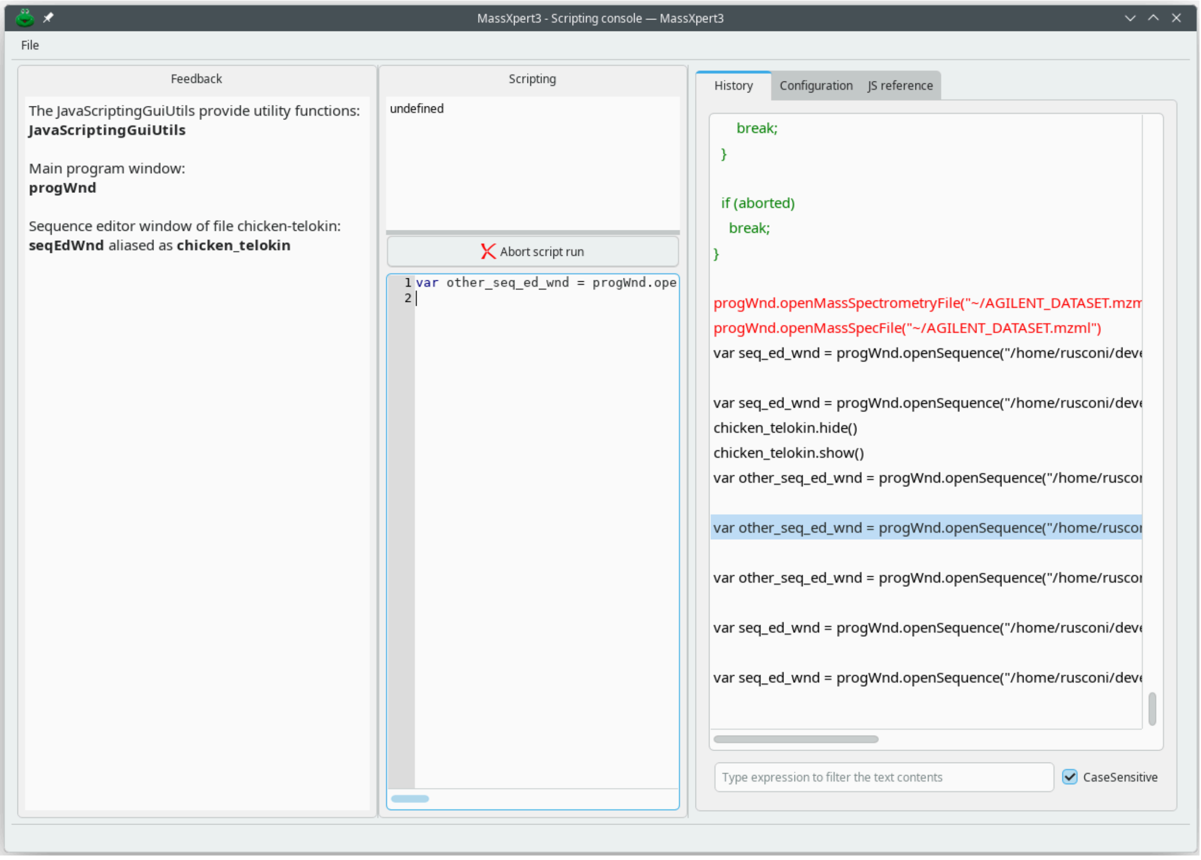
In order to trace back any given item in an input or in an output m/z list to its corresponding polyemr sequence, just activate the item while having a look at the polymer sequence whence the oligomers initially originated. Each time an item is activated by double-clicking it, its corresponding sequence region will be highlighted (selected, actually) in the polymer sequence.

# 7 XPERTSCRIPT: A SCRIPTING CONSOLE

XpertScript is a module that has been conceived as a scripting console allowing the user to automate a number of tasks.

## 7.1 XPERTSCRIPT INVOCATION

The XpertScript module is easily called by pulling down the *XpertScript* menu item from the MassXpert program's menu. Clicking on *XpertScript*→ Scripting console will open the scripting console window, as represented in FIGURE 7.1, "THE SCRIPTING CONSOLE WINDOW" .



The JavaScriptingGuiUtils provide utility functions:
**JavaScriptingGuiUtils**

Main program window:
**progWnd**

Sequence editor window of file chicken-telokin:
**seqEdWnd** aliased as **chicken_telokin**

```
undefined
```

```
var other_seq_ed_wnd = progWnd.ope
```

```
    break;
  }

  if (aborted)
    break;
}

progWnd.openMassSpectrometryFile("~/AGILENT_DATASET.mzm
progWnd.openMassSpecFile("~/AGILENT_DATASET.mzml")
var seq_ed_wnd = progWnd.openSequence("/home/rusconi/deve

var seq_ed_wnd = progWnd.openSequence("/home/rusconi/deve
chicken_telokin.hide()
chicken_telokin.show()
var other_seq_ed_wnd = progWnd.openSequence("/home/rusco

var other_seq_ed_wnd = progWnd.openSequence("/home/rusco

var other_seq_ed_wnd = progWnd.openSequence("/home/rusco

var seq_ed_wnd = progWnd.openSequence("/home/rusconi/deve

var seq_ed_wnd = progWnd.openSequence("/home/rusconi/deve
```

Scripting console window.

The window is divided into three vertical panes. The left hand side pane collects all the program's feedback to the user, so they are informed of any new variable that is created in the JavaScript environment and made available to them. The middle pane has two regions. The upper region will display all the return values after any JavaScript command has been run. The lower region is the actual script input region. The right hand side region contains mainly the history of the commands issued during the previous and current scripting sessions.

FIGURE 7.1: **THE SCRIPTING CONSOLE WINDOW**

## 7.2 RUNNING SCRIPTS IN THE SCRIPTING CONSOLE

The lower part of the middle panel of the scripting window is a text editor that has syntax colouring for JavaScript code. After having entered the script, hitting `Enter` while maintaining `Ctrl` pressed will run the script.

Upon running of the script, the output of the script is printed in the upper part of the same middle panel. Most often the output will be *undefined* (in grey colour), which is just fine. If the output is some actual value, it will be printed in green. If there was an error, the ouptut will be printed in red.

Note that each time a script is run, it is appended in the right hand side panel of the window, in the the *History* tab. The coloring of the items added to the *History*> tab follows the same logic as that described above.

## 7.3 LOADING SCRIPT FILES IN THE SCRIPTING CONSOLE

It is possible to craft ECMAscript-compliant scripts outside of the JavaScript console window, store them in files and open them for running in the JavaScript console window. To do so, select *File*→ Load a JavaScript file in the editor .

Once the script has been inserted into the script editor widget, the user can run it by hitting `Enter` while maintaining `Ctrl` pressed.

## 7.4 ONE ELABORATE EXAMPLE OF SCRIPT

The following script was used to generate a mass spectrum that represents the full molecular diversity of a protein from chicken: *Telokin* , the sequence of which is shipped along MassXpert in the sample sequences.

Telokin has been the subject of the author's Ph.D. research work and the deciphering of its incredible molecular diversity was published in *Biochemistry* in 1997. The author gives classes of mass spectrometry for biology, and that protein is the subject of the hands-on session of these classes.

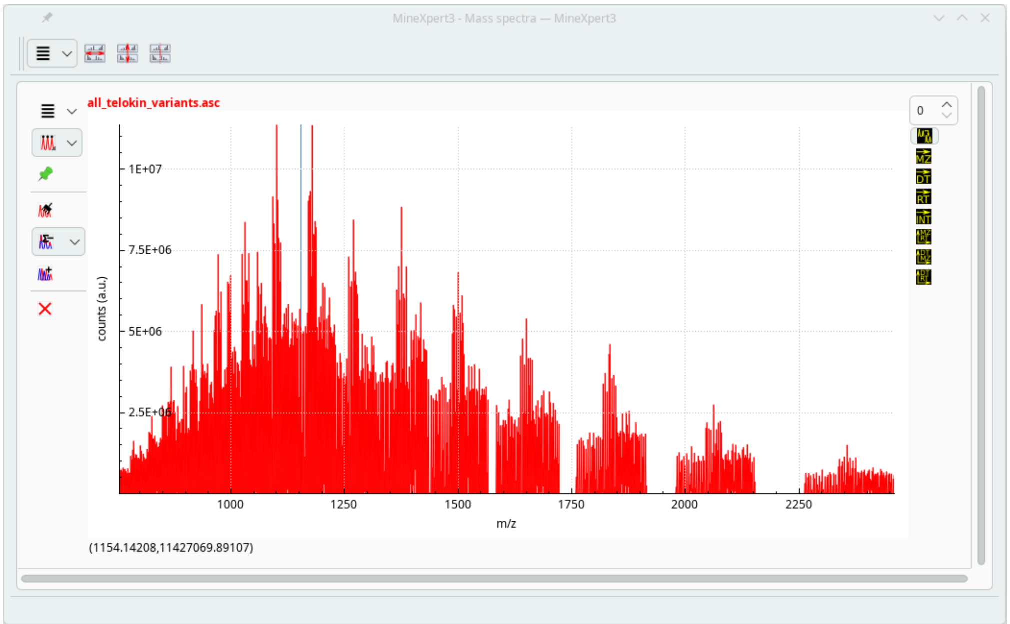
```
function getRandomWithinPercent(value, percent)
{
// Calculate the % range
const range = value * percent;

// Calculate min and max bounds
const min = value - range;
const max = value + range;

// Generate random number within the range
return Math.random() * (max - min) + min;
}
```

```
var seq_ed_wnd = progWnd.openSequence("chicken-telokin.mxp");

polymer = seq_ed_wnd.getPolymer();
polymer.setLeftEndModifByName("Acetylation");

let phosphoserine_index = 12;
polymer.modifyMonomer(phosphoserine_index, "Phosphorylation",
/*override*/ false);

calc_options = polymer.getCalcOptions();
ionizer = polymer.getIonizer();

seq_ed_wnd.setTraceColor("red");

mass_peak_shaper_dlg = seq_ed_wnd.configureMassPeakShaper();
mpscw = mass_peak_shaper_dlg.getConfigWidget();

let Mr = 17336;
let z_dict = {
7: 0.125e6,
8: 0.25e6,
9: 0.375e6,
10: 0.5e6,
11: 0.675e6,
12: 0.876e6,
13: 0.9e6,
14: 1e6,
15: 1e6,
16: 0.875e6,
17: 0.75e6,
18: 0.5e6,
19: 0.375e6,
20: 0.25e6,
21: 0.125e6
};
// let z_dict = { 1: 1e7 }

let random_intensity = 0

let nterm_indices_list = [1, 2, 7];
// let nterm_indices_list = [0]

aborted = false;

for (const nterm_index of nterm_indices_list)
{
// let removed_glu_start = 0
// let removed_glu_stop = 1
```

```
let removed_glu_start = 1;
let removed_glu_stop = 7;

for (let removed_glu = removed_glu_start; removed_glu <
removed_glu_stop; removed_glu++)
{

let start_index = nterm_index;
let end_index = polymer.size - 1 - removed_glu;

range_as_string = `[${start_index}-${end_index}]`;

seq_ed_wnd.setIndexRanges(range_as_string,
libXpertMassCore.Enums.LocationType.INDEX);

let mz = 1;

for (let z in z_dict)
{
if(checkAbort())
{
aborted = true;
break;
}

intensity = z_dict[z];

if (z == 0)
{
mz = Mr;
}
else
{
mz = Mr / z;
}

ionizer.level = z;
polymer.setIonizer(ionizer);

mass_peak_shaper_dlg.setParameters(mz, 150,
libXpertMassCore.Enums.MassPeakShapeType.GAUSSIAN, 40000, 0, 6);
mpscw.checkParameters();

// Now perform two calculations: without or with the polymer acetylation

calc_options.setPolymerEntities(libXpertMassCore.Enums.ChemicalEntity.NONE);
seq_ed_wnd.setCalcOptions(calc_options)
random_intensity = getRandomWithinPercent(intensity, 0.10);
seq_ed_wnd.setNormalizationIntensity(random_intensity);
```

```
       seq_ed_wnd.synthesizeMassSpectra();

       // Now perform the calculation with the Phosphorylation.

       calc_options.setMonomerEntities(libXpertMassCore.Enums.ChemicalEntity.MODIF);
       seq_ed_wnd.setCalcOptions(calc_options);
       random_intensity = getRandomWithinPercent(intensity, 0.10);
       seq_ed_wnd.setNormalizationIntensity(random_intensity);
       seq_ed_wnd.synthesizeMassSpectra();


       calc_options.setPolymerEntities(libXpertMassCore.Enums.ChemicalEntity.LEFT_END_MODIF
  |

       libXpertMassCore.Enums.ChemicalEntity.FORCE_LEFT_END_MODIF);
       seq_ed_wnd.setCalcOptions(calc_options);
       random_intensity = getRandomWithinPercent(intensity, 0.10);
       seq_ed_wnd.setNormalizationIntensity(random_intensity);
       seq_ed_wnd.synthesizeMassSpectra();

       // Now perform the calculation without the Phosphorylation.

       calc_options.setMonomerEntities(libXpertMassCore.Enums.ChemicalEntity.NONE);
       seq_ed_wnd.setCalcOptions(calc_options);
       random_intensity = getRandomWithinPercent(intensity, 0.10);
       seq_ed_wnd.setNormalizationIntensity(random_intensity);
       seq_ed_wnd.synthesizeMassSpectra();
       }

       if (aborted)
       break;
       }

       if (aborted)
       break;
       }
```

The output of this script is a mass spectrum that is automatically rendered in a MineXpert program's mass spectra window if the program is installed on the computer. See Figure 7.2, “Mass spectrum as generated using the script displayed in MineXpert” .

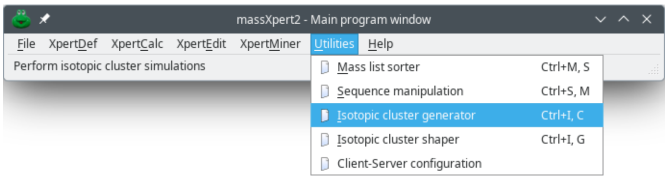


If MineXpert is installed on the computer, upon finishing running the script, MassXpert will trigger the display of the mass spectrum in MineXpert.

FIGURE 7.2: **MASS SPECTRUM AS GENERATED USING THE SCRIPT DISPLAYED IN MINEXPERT**

# 8 UTILITIES

The utilities made available in the *Utilities* menu of the main program window are shown in FIGURE 8.1, “ UTILITIES ” . These are described in detail in the following sections.



The *Utilities* menu provides useful features, which include isotopic cluster calculations, peak shaping around centroid values, and network-based communications with the companion software MineXpert.

FIGURE 8.1: **UTILITIES**

## 8.1 CALCULATING ISOTOPIC CLUSTERS

When the resolving power of the mass spectrometer is good, zooming-in on a mass peak may reveal that a given ion has given rise not to one peak but to a set of peaks that together form an “*isotopic cluster*”. The ions below the different cluster peaks are called “*isotopologues*” because they are homologues of the analyte that have different numbers (and therefore positions) of the different isotopes.

While the richness of mass spectra acquired with high resolving power machines has been neglected for many years, it is now well established that using the resolved isotopic cluster peaks for the detailed analysis of mass spectra is of great use.

When performing mass spectrometric data analyses, it is very often necessary to calculate the isotopic cluster of a given analyte. Indeed, the isotopic cluster can be a criterion helping to establish if a given detected signal (the acquired isotopic cluster) does indeed correspond to a given chemical entity, and matching the experimental isotopic cluster with one simulated for the evaluated chemical entity might prove useful. This way of ascertaining the assignment of mass peaks to specific analytes is becoming more and more desirable as mass spectrometers have ever-greater resolving power capabilities, thus providing the user with neatly resolved isotopic clusters even for oligomers of intermediate masses. In this section, the process of calculating the isotopic cluster of a given analyte defined by its chemical composition (also called elemental composition or formula) is described in detail.

It is possible to predict how a given ion (of known elemental composition *and* charge) is supposed to be revealed as an isotopic cluster in a high resolution mass spectrum. One such cluster is shown in FIGURE 8.2, “ SIMULATION OF THE ISOTOPIC CLUSTER FOR A PROTEIN ” , for a protein in its deca-protonated form $[M+10H^{+}]^{10+}$, of ionic elemental composition $\mathrm{C769H1222N210O218S2}$.

Calculated isotopic cluster for a deca-protonated protein.

**FIGURE 8.2: SIMULATION OF THE ISOTOPIC CLUSTER FOR A PROTEIN**

MineXpert provides sophisticated interface to the `libIsoSpec++` library.

> IsoSpec: Hyperfast Fine Structure Calculator
>
> Mateusz K. Łącki, Michał Startek, Dirk Valkenborg, and Anna Gambin
>
> *Analytical Chemistry*, 2017, 89, 3272–3277
>
> DOI: 10.1021/acs.analchem.6b01459

This library performs high-resolution isotopic cluster calculations. In order to run the calculations, it is necessary to have the following items ready:

- The elemental composition of the analyte (for example, $H_2O_1$). This formula needs to account for the ionization agent that is involved in the ionization of the analyte prior to its detection in the mass spectrometer. If an analyte is deca-protonated, 10 protons need to be added to its neutral-state formula.

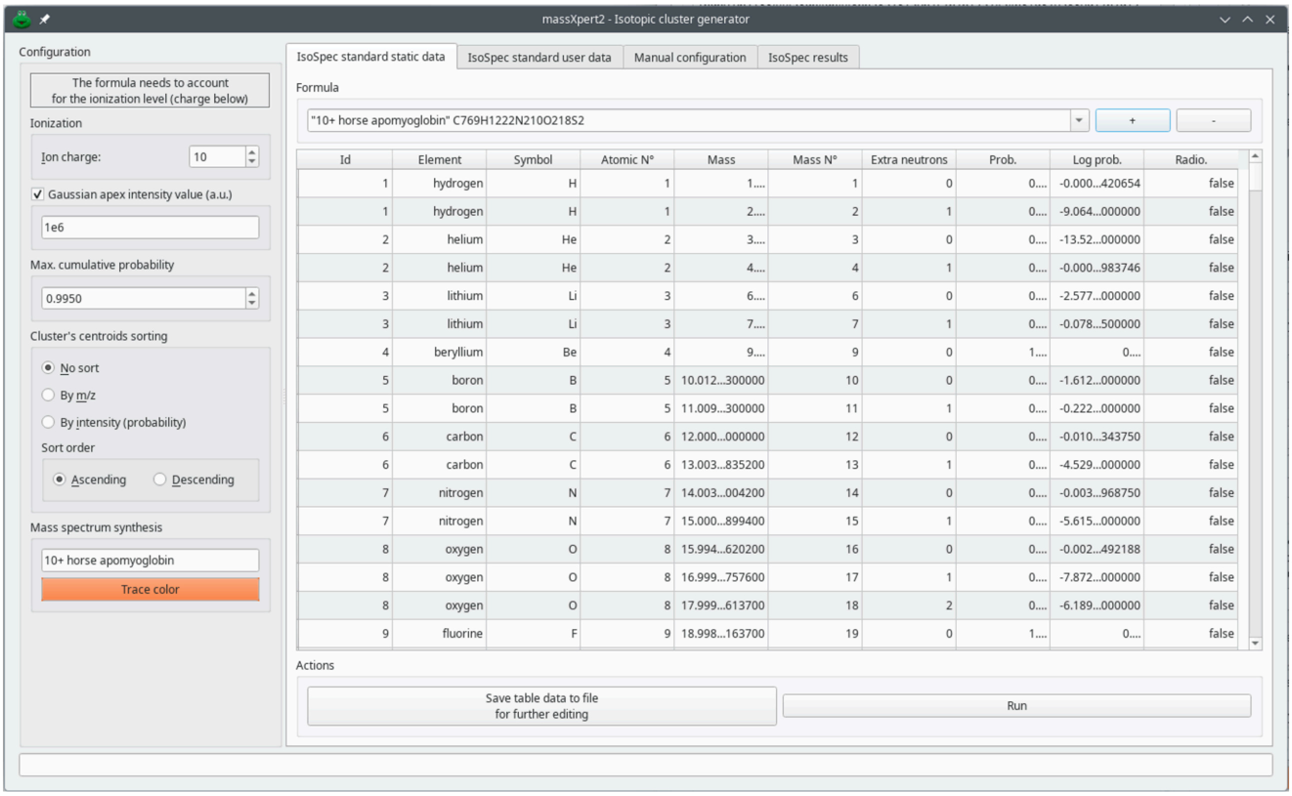
**Important**

The IsoSpec software requires that all the chemical elements of a chemical formula be indexed. This means that, for water, for example, the formula should be H2O1 (notice the index 1 after the O element symbol).

- A detailed isotopic configuration of all the chemical elements that are found in the elemental composition. MassXpert provides three interfaces to define the isotopic characteristics of the chemical elements. These will be described in the following sections.

### 8.1.1 The IsoSpec Graphical User Interface in mineXpert

Generating iosotopic clusters using the IsoSpec software package is not easily carried over, in particular because this remarkable library is designed to be highly performant. The authors rightfully put their energy into optimizations for accuracy and speed instead of investing time to design a graphical user interface. MassXpert provides that graphical user interface ( Figure 8.3, “ Isotopic cluster calculation dialog window ” ) , that shows up upon selection of the program's main window's *Utilities* › Isotopic cluster calculations menu.



Ion charge: 10

☑ Gaussian apex intensity value (a.u.): 1e6

Max. cumulative probability: 0.9950

Cluster's centroids sorting: ● No sort ○ By m/z ○ By intensity (probability)

Sort order: ● Ascending ○ Descending

Mass spectrum synthesis: 10+ horse apomyoglobin

Formula: "10+ horse apomyoglobin" C769H1222N210O218S2

| Id | Element | Symbol | Atomic N° | Mass | Mass N° | Extra neutrons | Prob. | Log prob. | Radio. |
|---|---|---|---|---|---|---|---|---|---|
| 1 | hydrogen | H | 1 | 1.... | 1 | 0 | 0.... | -0.000...420654 | false |
| 1 | hydrogen | H | 1 | 2.... | 2 | 1 | 0.... | -9.064...000000 | false |
| 2 | helium | He | 2 | 3.... | 3 | 0 | 0.... | -13.52...000000 | false |
| 2 | helium | He | 2 | 4.... | 4 | 1 | 0.... | -0.000...983746 | false |
| 3 | lithium | Li | 3 | 6.... | 6 | 0 | 0.... | -2.577...000000 | false |
| 3 | lithium | Li | 3 | 7.... | 7 | 1 | 0.... | -0.078...500000 | false |
| 4 | beryllium | Be | 4 | 9.... | 9 | 0 | 1.... | 0.... | false |
| 5 | boron | B | 5 | 10.012...300000 | 10 | 0 | 0.... | -1.612...000000 | false |
| 5 | boron | B | 5 | 11.009...300000 | 11 | 1 | 0.... | -0.222...000000 | false |
| 6 | carbon | C | 6 | 12.000...000000 | 12 | 0 | 0.... | -0.010...343750 | false |
| 6 | carbon | C | 6 | 13.003...835200 | 13 | 1 | 0.... | -4.529...000000 | false |
| 7 | nitrogen | N | 7 | 14.003...004200 | 14 | 0 | 0.... | -0.003...968750 | false |
| 7 | nitrogen | N | 7 | 15.000...899400 | 15 | 1 | 0.... | -5.615...000000 | false |
| 8 | oxygen | O | 8 | 15.994...620200 | 16 | 0 | 0.... | -0.002...492188 | false |
| 8 | oxygen | O | 8 | 16.999...757600 | 17 | 1 | 0.... | -7.872...000000 | false |
| 8 | oxygen | O | 8 | 17.999...613700 | 18 | 2 | 0.... | -6.189...000000 | false |
| 9 | fluorine | F | 9 | 18.998...163700 | 19 | 0 | 1.... | 0.... | false |

The dialog window contains two panels. The left hand side panel configures the charge for which the calculation is to be carried over, the intensity of the highest peak in the isotopic cluster, and the maximum cumulative isotopic presence probability that IsoSpec must reach during the calculation. The right hand side panel contains a tab widget to hold the configuration tabs and the results tab.

FIGURE 8.3: **ISOTOPIC CLUSTER CALCULATION DIALOG WINDOW**

An isotopic cluster calculation is most probably performed with the aim of simulating an expected isotopic cluster for an analyte that is being analyzed by mass spectrometry. It is thus logical that the analyte be in an ionized form. The way that the analyte has been ionized needs to be taken into account in the chemical formula that describes the ion for which the isotopic cluster is being calculated. For example, when determining the elemental composition of a protein in the positive ion mode, the number of protons used to ionize the protein needs to be taken into account.

### WARNING

The IsoSpec software is "charge-agnostic" in the sense that it does not know what element in the chemical formula is responsible for the ionization of the analyte. Therefore, IsoSpec does not know of (and does not care about) the charge of the analyte. The ionization level of the analyte can be handled by MassXpert if that information is set to the *Ion charge* spin box widget. By default, the charge state of the analyte is 1.

The *Gaussian apex intensity value (a.u.)* check box widget allows the user to set the intensity (y axis) that the isotopic cluster will have for its highest peak. This is useful when the isotopic cluster is to be simulated for direct comparison with an isotopic cluster in an experimental mass spectrum.

The *Max. cumulative probability* spin box widget serves to configure the extent to which IsoSpec simulates the theoretically expected isotopic cluster. A value of *0.99* tells the software to simulate enough combinations of the analyte isotopes to represent 99 % of the theoretically expected combinations.

### TIP

For large biopolymers, it might be prudent to start with a relatively low value for *Max. cumulative probability*, because setting this value too high, that is, near 1, would increase notably the calculation duration.

The *Cluster's centroid sorting* group box widget contains a few widgets that the user can use to configure the way the generated cluster's (centroid, intensity) pairs are sorted in the output.

The *Mass spectrum synthesis* group box widget contains a few widgets that the user can use to configure the title and the color of the generated spectrum (the actual generation of the spectrum on the basis of the IsoSpec output is described in detail below). The title and color of the mass spectrum will be of use when the spectrum is displayed in the companion MineXpert software program. The process by which the mass spectrum is transferred from MassXpert to MineXpert is described at SECTION 8.2, " CONFIGURATION OF THE MASSXPERT AND MINEXPERT INTERCOMMUNICATIONS ".

To perform isotopic cluster calculations, the simulation software needs to be aware of all the isotopes of all the chemical elements that enter in the composition of the ionized analyte. An isotope is defined by its mass and by the probability that it is found in nature. Carbon has two major isotopes that can be found in nature: the [$^{12}$C] most abundant isotope and the [$^{13}$C] least abundant isotope (the [$^{14}$C] isotope is irrelevant for conventional mass spectrometry unless molecules have been artificially enriched in that isotope).

There are three ways that the user might define the isotopic data to be used for the simulations. Two of them involve a configuration preparation on the part of the user. The third one, that we'll describe first, does not involve any chemical element configuration. The other ways are reviewed in the following sections.

#### 8.1.1.1 STATIC STANDARD ELEMENT TABLES SHIPPED WITHIN THE ISOSPEC LIBRARY

In order to document all the chemical elements' isotopes' characteristics, the IsoSpec library has, in its own software code headers, a number of arrays that MassXpert automatically loads up when opening the FIGURE 8.3, " ISOTOPIC CLUSTER CALCULATION DIALOG WINDOW " dialog window. [1] These data are displayed in the *IsoSpec*

---

[1] That process is performed at build time and is thus not configurable at run time.

*standard static data* tab. The table view widget data are loaded from source code each time the window is opened, hence the *static* qualifier in the tab name, meaning that the user won't be able to change these data and keep the changes between sessions.

In this tab, apart the configuration on the left hand side pane of the dialog window, all that is left to do is enter the elemental composition for which the isotopic cluster calculation is to be performed in the *Formula* combo box widget (which looks like a line edit widget with an arrow at its right side).

### TIP

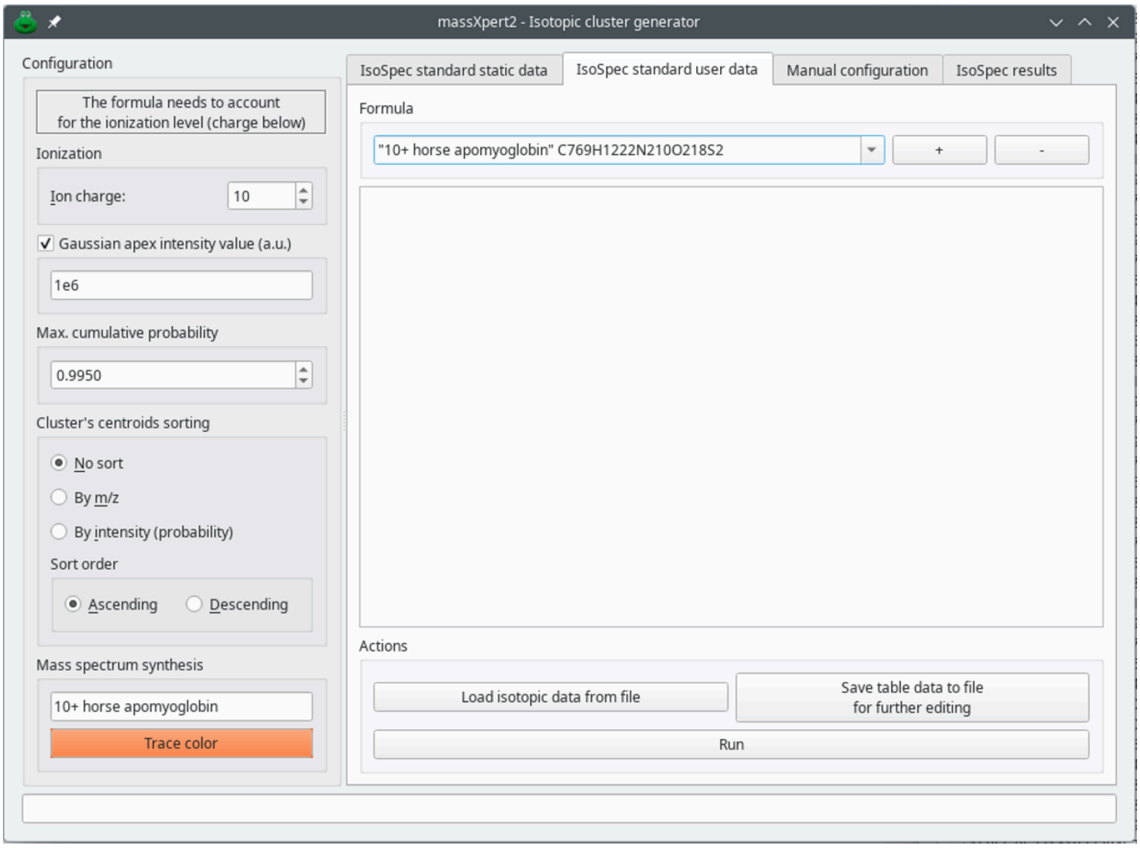
If a formula is used often, the user might want to store it with a label that explains what the elemental composition describes. To add a label to the formula, enter the label between quotes before the formula itself. To store the “titled” formula so that it will be available across sessions, click the + push button. To remove a stored formula, select in the combo box widget and click the – push button.

Once the formula has been set to its combo box widget, the charge of the ion needs to be set (*Ion charge* spin box widget) along with the *Max. cumulative probability*. To run the calculation, click *Run*.

#### 8.1.1.2 USER-MODIFIED STANDARD ELEMENT TABLE SHIPPED WITHIN THE ISOSPEC LIBRARY

While the previous section showed how to use the static element table from the IsoSpec library, this section shows a way to customize these elemental data for a specific experiment.

The user-modified version of the standard static isotopic data is available in the *IsoSpec standard user data* tab of the dialog window, which upon opening has an empty table view widget, as shown in FIGURE 8.4, “ USER-MODIFIED STANDARD ELEMENT TABLE ”.



Ionization

Ion charge: 10

☑ Gaussian apex intensity value (a.u.)

1e6

Max. cumulative probability

0.9950

Cluster's centroids sorting

● No sort

○ By m/z

○ By intensity (probability)

Sort order

● Ascending ○ Descending

Mass spectrum synthesis

10+ horse apomyoglobin

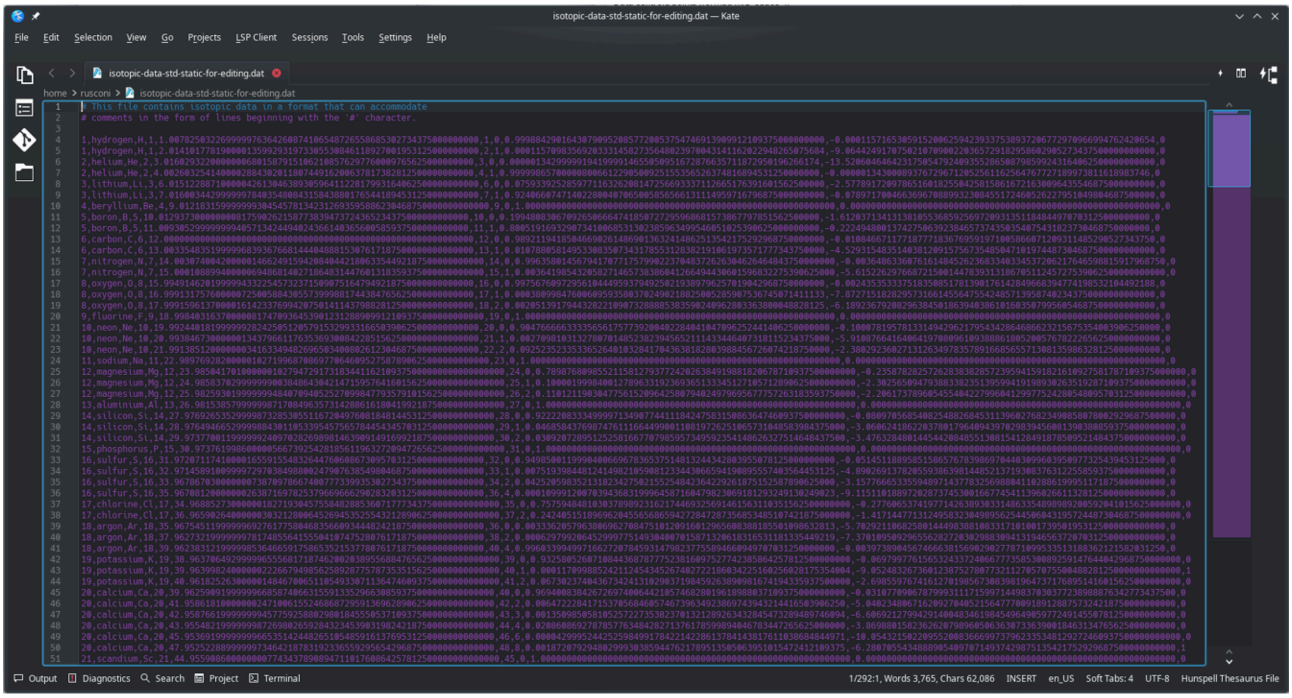
Formula

"10+ horse apomyoglobin" C769H1222N210O218S2

When started, the *IsoSpec standard user data* tab of the dialog window shows an empty table view widget. The way to populate the table view is described in the text.

FIGURE 8.4: **USER-MODIFIED STANDARD ELEMENT TABLE**

The simplest way to fill in the user-modified standard isotopic table is to export the isotopic data from the *IsoSpec standard static data* tab to a text file (*Save table data to file for further editing* button). The data are output to the file in the CSV (comma-separated-values) format, in a layout that closely mimicks the data visible in the table view. An example of the exported text is shown in FIGURE 8.5, “ ISOTOPIC EXPORTED FROM THE ISOTOPIC DATA TABLE VIEW' ”. Once the file has been saved, the data might be modified in the LibreOffice spreadsheet or in a text editor. For example, in case of an experiment involving the labeling of analytes with [$^{13}$C] at an incorporation efficiency of roughly 98 %, the [$^{12}$C] abundance (probability) would be set to to 0.02 and the [$^{13}$C] abundance to 0.98 (in reality, the abundances calculation is a bit more subtle than that).



```
# This file contains isotopic data in a format that can accommodate
# comments in the form of lines beginning with the '#' character.

1,hydrogen,H,1,1.007825032269999976364260874106548726558685302734375000000000,1,0,0.999884290164307909520857720053754746913909912109375000000000,-0.000115716530591520062594239337538937206772970966994762420654,0
1,hydrogen,H,1,2.014101778190000135992931973305530846118927001953125000000000,2,1,0.000115709835692033314582735648023970043141162022948265075684,-9.064424917075021070900220365729182958602905273437500000000000,0
2,helium,He,2,3.016029322000000006801587915106210857629776000976562500000000,3,0,0.000001342999991941999914655050951672876635711872950196266174,-13.520604646423175054792409355286508798599243164062500000000000,0
2,helium,He,2,4.002603254140002884302011807449162006378173828125000000000000,4,1,0.999998657000000800661229050092515535652637481689453125000000,-0.000001343000893767296712052561162564767727189973811618983746,0
3,lithium,Li,3,6.015122887100000426130463893059641122817993164062500000000000,6,0,0.075933925285977116326208147256693337112665176391601562500000,-2.577891720978651601825504258158616721630096435546875000000000,0
3,lithium,Li,3,7.016003442999999784035480843158438801765441894531250000000000,7,1,0.924066074714022800407065005856566131114959716796875000000000,-0.078971700466369670889932308455172460526227951049804687500000,0
4,beryllium,Be,4,9.012183159999999304545781342312693595886230468750000000000000,9,0,1.000000000000000000000000000000000000000000000000000000000000,0.000000000000000000000000000000000000000000000000000000000000,0
5,boron,B,5,10.012937300000000081759026215877383947372436523437500000000000,10,0,0.199480830670926506664741850727295968681573867797851562500000,-1.612037134131381055368592569720931351184844970703125000000000,0
5,boron,B,5,11.009305299999999405713424494024366140365600585937500000000000,11,1,0.800519169329073410068531302385963499546051025390625000000000,-0.222494800137427506392384657374350354075431823730468750000000,0
6,carbon,C,6,12.000000000000000000000000000000000000000000000000000000000000,12,0,0.989211941850466902614869013632414862513542175292968750000000,-0.010846671177187771836769591971005866071209311485290527343750,0
6,carbon,C,6,13.003354835199999968393676681444048881530761718750000000000000,13,1,0.010788058149533083507343178553128382191061973571777343750000,-4.529315483514038120915756735485047101974487304687500000000000,0
7,nitrogen,N,7,14.003074004200001466249159420840442180633544921875000000000000,14,0,0.996358014567941707717579902237048372626304626464843750000000,-0.003648633607616148452623683340334537206217646598815917968750,0
7,nitrogen,N,7,15.000108899400000694868140271864831447601318359375000000000000,15,1,0.003641985432058271465738386041266494430601596832275390625000,-5.615226297668721500144783931318670511245727539062500000000000,0
8,oxygen,O,8,15.994914620199994332254573237150907516479492187500000000000000,16,0,0.997567609729561044495937949250219389796257019042968750000000,-0.002435353337518350851781390176142849668394774198532104492188,0
8,oxygen,O,8,16.999131757600000072500588430557399988174438476562500000000000,17,1,0.000380998476006095935803702490218825005285907536745071411133,-7.872715182829573166145564755424857139587402343750000000000000,0
8,oxygen,O,8,17.999159613700001614233769942075014114379882812500000000000000,18,2,0.002051391794432822109073288885383590240962803363800048828125,-6.189236792082963845018639403861016035079956054687500000000000,0
9,fluorine,F,9,18.998403163700000081747093645390123128890991210937500000000000,19,0,1.000000000000000000000000000000000000000000000000000000000000,0.000000000000000000000000000000000000000000000000000000000000,0
10,neon,Ne,10,19.992440181999999282425051205791532993316650390625000000000000,20,0,0.904766666333356561757739200402284041047096252441406250000000,-0.100078195781331494296217954342864686623215675354003906250000,0
10,neon,Ne,10,20.993846730000001343796611763536930084228515625000000000000000,21,1,0.002709810313278070148523823945652111433446407318115234375000,-5.910876641640641970809610938886180520057678222656250000000000,0
10,neon,Ne,10,21.991385120000000341633494826965034008026123046875000000000000,22,2,0.092523523353365264010328417043638182803988456726074218750000,-2.380292360271312634978357891668565571308135986328125000000000,0
11,sodium,Na,11,22.989769282000001027199687086977064609527587890625000000000000,23,0,1.000000000000000000000000000000000000000000000000000000000000,0.000000000000000000000000000000000000000000000000000000000000,0
12,magnesium,Mg,12,23.985041701000001027947291731834411621093750000000000000000000,24,0,0.789876809855211581279377242026384919881820678710937500000000,-0.235878282572628382857239594159182161092758178710937500000000,0
12,magnesium,Mg,12,24.985837029999999003848643042147159576416015625000000000000000,25,1,0.100001999840012789633192369365133345127105712890625000000000,-2.302565094793883382351395994191989302635192871093750000000000,0
12,magnesium,Mg,12,25.982593019999999484070940525270998477935791015625000000000000,26,2,0.110121190304775615209642580794024979695677757263183593750000,-2.206173789605455404227996041299775242805480957031250000000000,0
13,aluminium,Al,13,26.981538579999998717084963573142886161804199218750000000000000,27,0,1.000000000000000000000000000000000000000000000000000000000000,0.000000000000000000000000000000000000000000000000000000000000,0
14,silicon,Si,14,27.976926535299998732853055116720497608184814453125000000000000,28,0,0.922220833349999713490774411184247583150863647460937500000000,-0.080970568540825488268453113960276823490858078002929687500000,0
14,silicon,Si,14,28.976494665299998843011053395457565784454345703125000000000000,29,1,0.046858437698747611166449900110819726251065731048583984375000,-3.060624186220378017964094397029839456081390380859375000000000,0
14,silicon,Si,14,29.973770011999999240970282698981463909149169921875000000000000,30,2,0.030920728951252581667707985957349592354148626327514648437500,-3.476328480144544208485513081541284918785095214843750000000000,0
15,phosphorus,P,15,30.973761998600000566739254281856119632720947265625000000000000,31,0,1.000000000000000000000000000000000000000000000000000000000000,0.000000000000000000000000000000000000000000000000000000000000,0
16,sulfur,S,16,31.972071174100001655915548326447606086730957031250000000000000,32,0,0.949850011999040066967836537514813244342803955078125000000000,-0.051451188958515865767839869704403099603950977325439453125000,0
16,sulfur,S,16,32.971458910099997297038498800247907638549804687500000000000000,33,1,0.007519398448124149821059081233443066594190895557403564453125,-4.890269137820559386398144852137193083763122558593750000000000,0
16,sulfur,S,16,33.967867030000000738709786674007773399353027343750000000000000,34,2,0.042520598352131823427502155254842364229261875152587890625000,-3.157766653355948971437783256988041102886199951171875000000000,0
16,sulfur,S,16,35.967081200000002638716978253796696662902832031250000000000000,36,4,0.000109991200703943683199964587160479823069181293249130249023,-9.115110188972028737453001667745411396026611328125000000000000,0
17,chlorine,Cl,17,34.968852730000001827193045755848288536071777343750000000000000,35,0,0.757594848103037898923162174469325691461563110351562500000000,-0.277606537419771426389303314863354898989200592041015625000000,0
17,chlorine,Cl,17,36.965902640000003032128006452694535255432128906250000000000000,37,2,0.242405151896962045565686594272847287356853485107421875000000,-1.417144771312495832304989562544506043195724487304687500000000,0
18,argon,Ar,18,35.967545119999996927617758046835660934448242187500000000000000,36,0,0.003336205796380696270847510120916012965608388185501098632813,-5.702921106825801444983881083317101001739501953125000000000000,0
18,argon,Ar,18,37.962732199999997817485564155504107475280761718750000000000000,38,2,0.000629799206452999775149304007015871320618316531181335449219,-7.370109509296556282720302988309413194656372070312500000000000,0
18,argon,Ar,18,39.962383121999998536466591758653521537780761718750000000000000,40,4,0.996033994997166272078459314798237755894660949707031250000000,-0.003973890456746663815690290277871099533513188362121582031250,0
19,potassium,K,19,38.963706492999996555681718746200203895568847656250000000000000,39,0,0.932580526071084436878777523816097527742385864257812500000000,-0.069799776156532433724066777358530089259147644042968750000000,0
19,potassium,K,19,39.963998240000002226679498562589287757873535156250000000000000,40,1,0.000117099885242112454345267402722186034225160256028175354004,-9.052483267360123875278077321127057075500488281250000000000000,1
19,potassium,K,19,40.961825263000001484670065110549330711364746093750000000000000,41,2,0.067302374043673424131029037198459263890981674194335937500000,-2.698559767416127019856730839819647371768951416015625000000000,0
20,calcium,Ca,20,39.962590919999996685874066315591335296630859375000000000000000,40,0,0.969400838426726974006442105746828019618988037109375000000000,-0.031077090678799931117159971449837030377238988876342773437500,0
20,calcium,Ca,20,41.958618100000002471006155246868729591369628906250000000000000,42,2,0.006472228417153705684605746739634923869743943214416503906250,-5.040234806716209270405215647770091891288757324218750000000000,0
20,calcium,Ca,20,42.958766199999999457759258802980184555053710937500000000000000,43,3,0.001350985058105257227353823701321289263432845473289489746094,-6.606921279942914004834619845496490597724914550781250000000000,0
20,calcium,Ca,20,43.955482199999998726980265928432345390319824218750000000000000,44,4,0.020860869278785776348428271376178599894046783447265625000000,-3.869880158236262079896050636307336390018463134765625000000000,0
20,calcium,Ca,20,45.953691999999996653514244826510548591613769531250000000000000,46,6,0.000042999524425259849917842214228613784143817611038684844971,-10.054321502209552008366699737962335348129272460937500000000000,0
20,calcium,Ca,20,47.952522889999997346421878319233655929565429687500000000000000,48,8,0.001872079294802999303859447621789513505063951015472412109375,-6.280705543488890540970714937429875135421752929687500000000000,1
21,scandium,Sc,21,44.955908600000000774343789089471101760864257812500000000000000,45,0,1.000000000000000000000000000000000000000000000000000000000000,0.000000000000000000000000000000000000000000000000000000000000,0
```

When saving isotopic data from the table view to a file, the generated text is in the comma-separated-values (CSV) format. In this format, the editing of the isotope masses and abundances is very easy in a text editor or in spreadsheet program.

FIGURE 8.5: **ISOTOPIC EXPORTED FROM THE ISOTOPIC DATA TABLE VIEW'**

Now that once the modifications have been performed, the file needs to be saved under the same format. It is then available for loading in the *IsoSpec standard user data* tab of the dialog window by clicking the *Load table from file* push button. The table view now shows the new isotope mass-abundance pairs. From now on, this tab may be used exactly like described above for the standard isospec *static* data-based work.

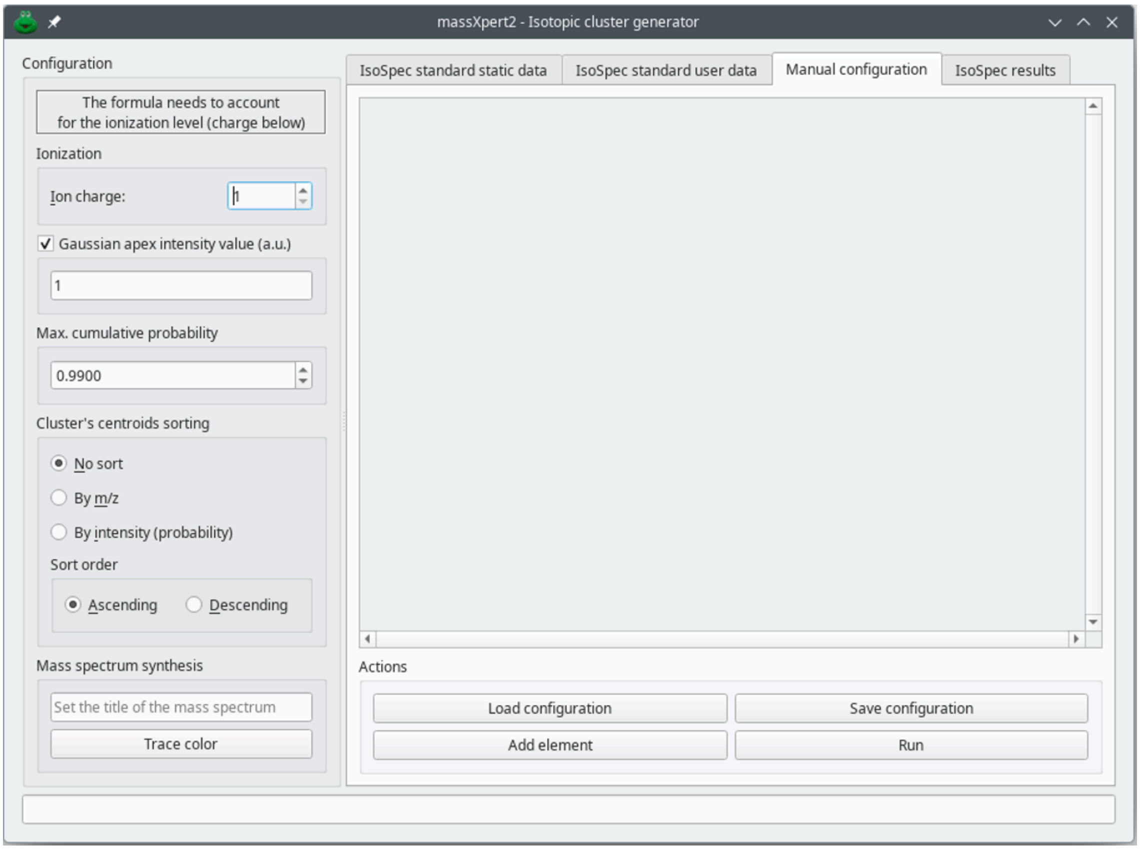
### TIP

The cells of the table view widget are editable. The cells' contents may be edited by double-clicking the cell and entering the new value. The table view widget contents might then be exported to a file exactly as described above.

Once the formula has been set to its combo box widget, the charge of the ion needs to be set (*Ion charge* spin box widget) along with the *Max. cumulative probability*. To run the calculation, click *Run*.

### 8.1.1.3 User hand-made isotopic configuration of an elemental composition

The last way the user might configure the chemical elements to be used for an isotopic cluster calculation is based on the fully manual description of the isotopes present in the elemental composition of interest. That configuration is performed in the *Manual configuration* tab of the dialog window. This method is slightly more involved than the previous one but provides also for a much greater flexibility: it allows one to create “new chemical elements” that might be required in specific labelling experiments. The manual configuration is carried over in the *Manual configuration* tab of the dialog. When start, this tab is empty, as shown in Figure 8.6, “ Hand-made user isotopic configuration of an elemental composition ” (the working sheet is blank).



Ion charge: 1

☑ Gaussian apex intensity value (a.u.): 1

Max. cumulative probability: 0.9900

● No sort ○ By m/z ○ By intensity (probability)

Sort order: ● Ascending ○ Descending

When the dialog is created, the tab is empty. To start creating element definitions, click *Add element*.

**FIGURE 8.6: HAND-MADE USER ISOTOPIC CONFIGURATION OF AN ELEMENTAL COMPOSITION**

Upon creation of the dialog window, the *Manual configuration* tab is empty, with only two rows of buttons at the bottom of the tab. To start configuring chemical elements, click *Add element* to create an “element group box” that will contains a number of widgets organized in two rows:

- Top row, a line edit widget to receive the chemical element symbol, like *C*, for example;
- On the same row, a spin box widget in which to set the number of such atoms in the formula for which the isotopic cluster is being calculated;

- A button with a "–" label that removes all the "element group box" in one go;
- The bottom row contains an "isotope frame widget" with two spin boxes for the mass of the isotope being configured (left) and its corresponding abundance (right);
- In addition to the spin boxes, two buttons, with a "plus" or a "–" label, allow one to respectively add or remove isotope frames.

> **Note**
>
> It is not possible to remove all the isotope frames from an element group box, otherwise that element group box would become useless.

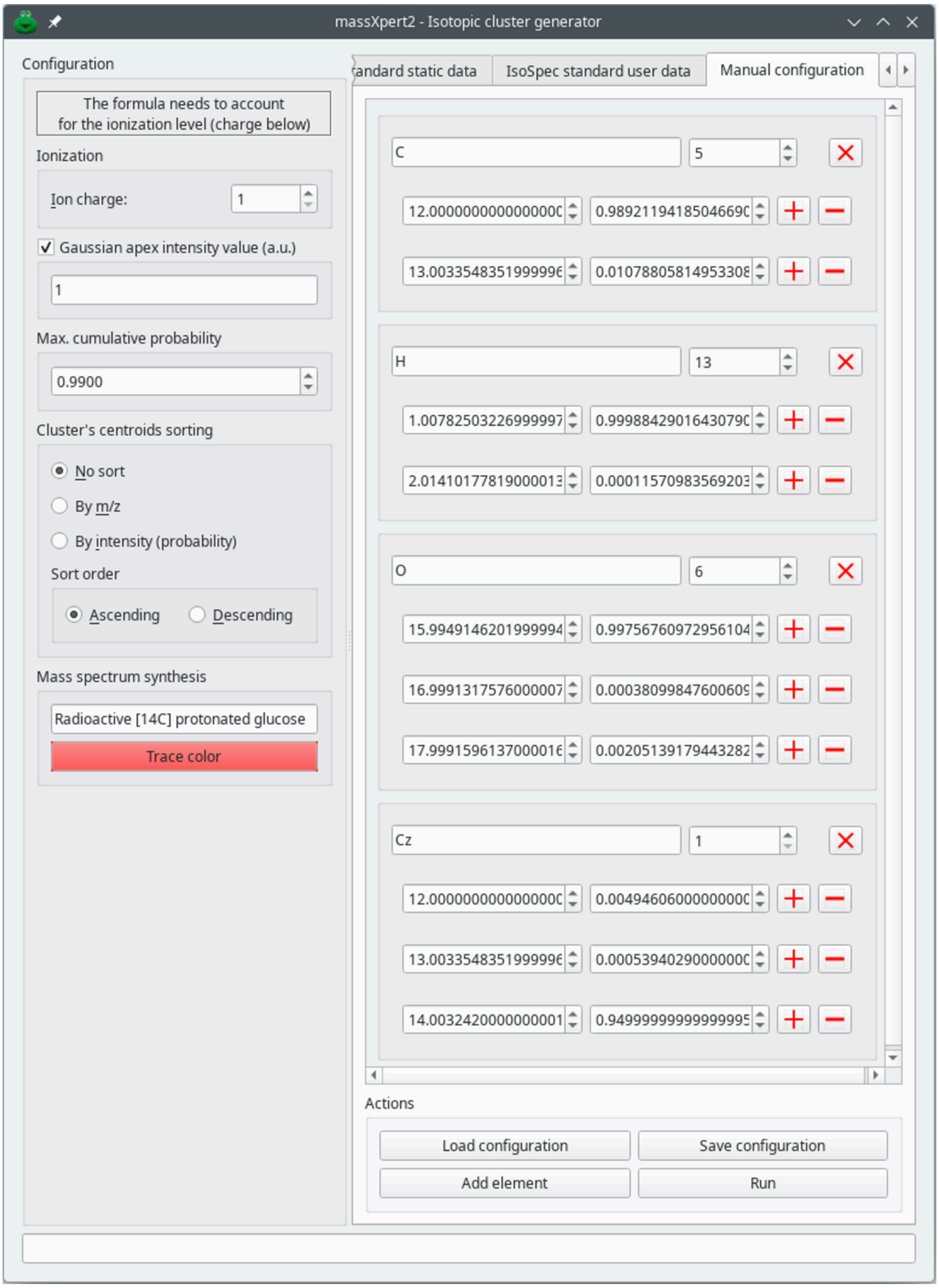
Once an isotope frame has been filled-up, a new line might be required to document another isotope for the same element. To create a new isotope frame widget, click any "plus"-labelled button in any of the isotope frames. Once a new frame is created, the spin box widgets that it contains are set to *0.00000*. Fill-in these spin boxes with mass and abundance values and go on along this path to create as many isotopes as required.

Once all the isotopes for a given chemical element have been defined, a new element might be needed. For this, click *Add element* and start the configuration of the new element as described above.

The manual isotopic configuration of the chemical elements required to perform an isotopic cluster calculation for a given formula is tedious. The user may want to save a given configuration to a file (click *Save configuration*) so that it is easier to recreate automatically all the widgets upon loading of that saved configuration (click *Load configuration*).

The final configuration is shown in FIGURE 8.7, " TYPICAL MANUAL CONFIGURATION OF THE ISOTOPIC CHARACTERISTICS OF THE CHEMICAL ELEMENTS " . The experiment that was configured above is the labelling of a glucose molecule with *Cz*, an imaginary chemical element that is like carbon but that has a $[^{14}C]$. The glucose molecule (normal formula: C6H12O6) is labelled on one single carbon atom with an efficieny of 95 %. This means that, when the labelling fails (in 5 % of the cases) the carbon atom has its isotopes with usual probabilities (compounded by the fact that the unlabeled atom is found at that position only in 5 % of the cases). The isotopic abundances for the Cz element are thus:

- For $[^{12}C]$: 0.05 * natural $[^{12}C]$ abundance;
- For $[^{13}C]$: 0.05 * natural $[^{13}C]$ abundance;
- For $[^{14}C]$: 0.95;



The formula needs to account
for the ionization level (charge below)

Ionization

Ion charge: 1

☑ Gaussian apex intensity value (a.u.)

1

Max. cumulative probability

0.9900

Cluster's centroids sorting

● No sort

○ By m/z

○ By intensity (probability)

Sort order

● Ascending ○ Descending

Mass spectrum synthesis

Radioactive [14C] protonated glucose

Trace color

C 5

12.0000000000000000 0.98921194185046690

13.0033548351999996 0.01078805814953308

H 13

1.00782503226999997 0.99988429016430790

2.01410177819000013 0.00011570983569203

O 6

15.9949146201999994 0.99756760972956104

16.9991317576000007 0.00038099847600609

17.9991596137000016 0.00205139179443282

Cz 1

12.0000000000000000 0.00494606000000000

13.0033548351999996 0.00053940290000000

14.0032420000000001 0.94999999999999995

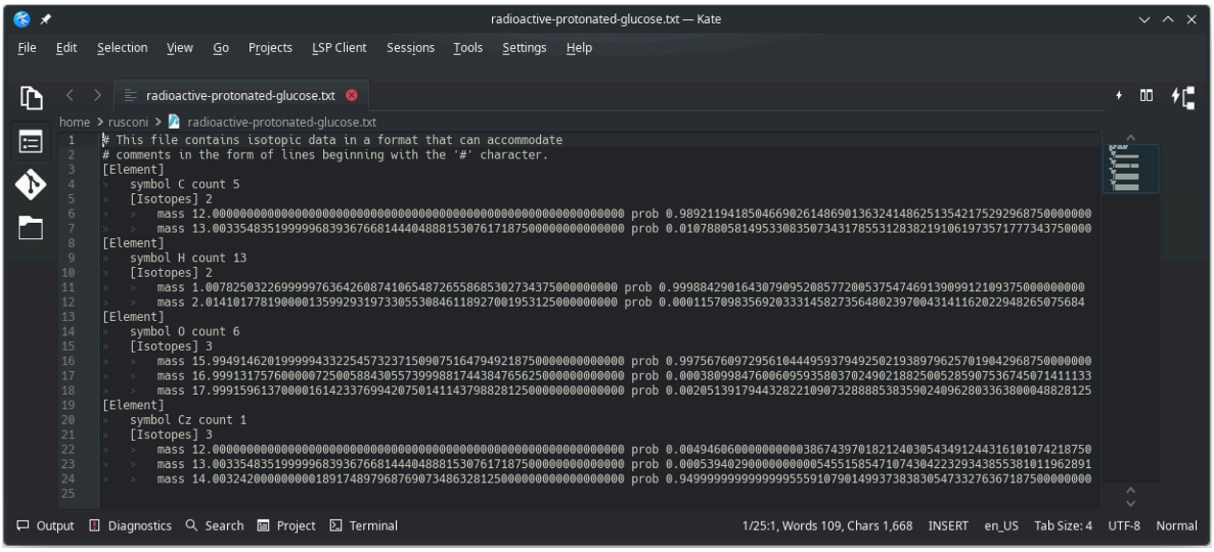
The user has configured a labelling experiment where a protonated glucose molecule is labelled at a single carbon position with a [$^{14}$C] atom (the efficiency of the labeling is 95 %).

FIGURE 8.7: TYPICAL MANUAL CONFIGURATION OF THE ISOTOPIC CHARACTERISTICS OF THE CHEMICAL ELEMENTS

## NOTE

Note that the count of unlabeled carbon atoms is 5 (and not 6), that the hydrogen count is 13 (and not 12, because the glucose is protonated), and that the [$^{14}$C]-labelled carbon atom is present only once.

When the configuration has been performed (a process that might reveal tedious), it can be exported to a text file for later reuse. The example of has been exported and is available as a text file depicted in FIGURE 8.8, “ THE USER MANUAL CONFIGURATION MIGHT BE EXPORTED TO A TEXT FILE AND LOADED BACK ”.



```
# This file contains isotopic data in a format that can accommodate
# comments in the form of lines beginning with the '#' character.
[Element]
    symbol C count 5
    [Isotopes] 2
        mass 12.000000000000000000000000000000000000000000000000000000000000 prob 0.989211941850466902614869013632414862513542175292968750000000
        mass 13.003354835199996839367668144404888153076171875000000000000000 prob 0.010788058149533083507343178553128382191061973571777343750000
[Element]
    symbol H count 13
    [Isotopes] 2
        mass 1.007825032269999976364260874106548726558685302734375000000000 prob 0.999884290164307909520857720053754746913909912109375000000000
        mass 2.014101778190000135992931973305530846118927001953125000000000 prob 0.000115709835692033314582735648023970043141162022948265075684
[Element]
    symbol O count 6
    [Isotopes] 3
        mass 15.994914620199994332254573237150907516479492187500000000000000 prob 0.997567609729561044495937949250219389796257019042968750000000
        mass 16.999131757600000725005884305573999881744384765625000000000000 prob 0.000380998476006095935803702490218825005285907536745071411133
        mass 17.999159613700001614233769942075014114379882812500000000000000 prob 0.002051391794432822109073288885383590240962803363800048828125
[Element]
    symbol Cz count 1
    [Isotopes] 3
        mass 12.000000000000000000000000000000000000000000000000000000000000 prob 0.004946060000000000386743970182124030543491244316101074218750
        mass 13.003354835199996839367668144404888153076171875000000000000000 prob 0.000539402900000000054551585471074304223293438553810119628910
        mass 14.003242000000000189174897968769073486328125000000000000000000 prob 0.949999999999999955591079014993738383054733276367187500000000
```

Use the *Save configuration* push button to export the configuration to a text file and the *Load configuration* push button to load a configuration from file.

**FIGURE 8.8: THE USER MANUAL CONFIGURATION MIGHT BE EXPORTED TO A TEXT FILE AND LOADED BACK**

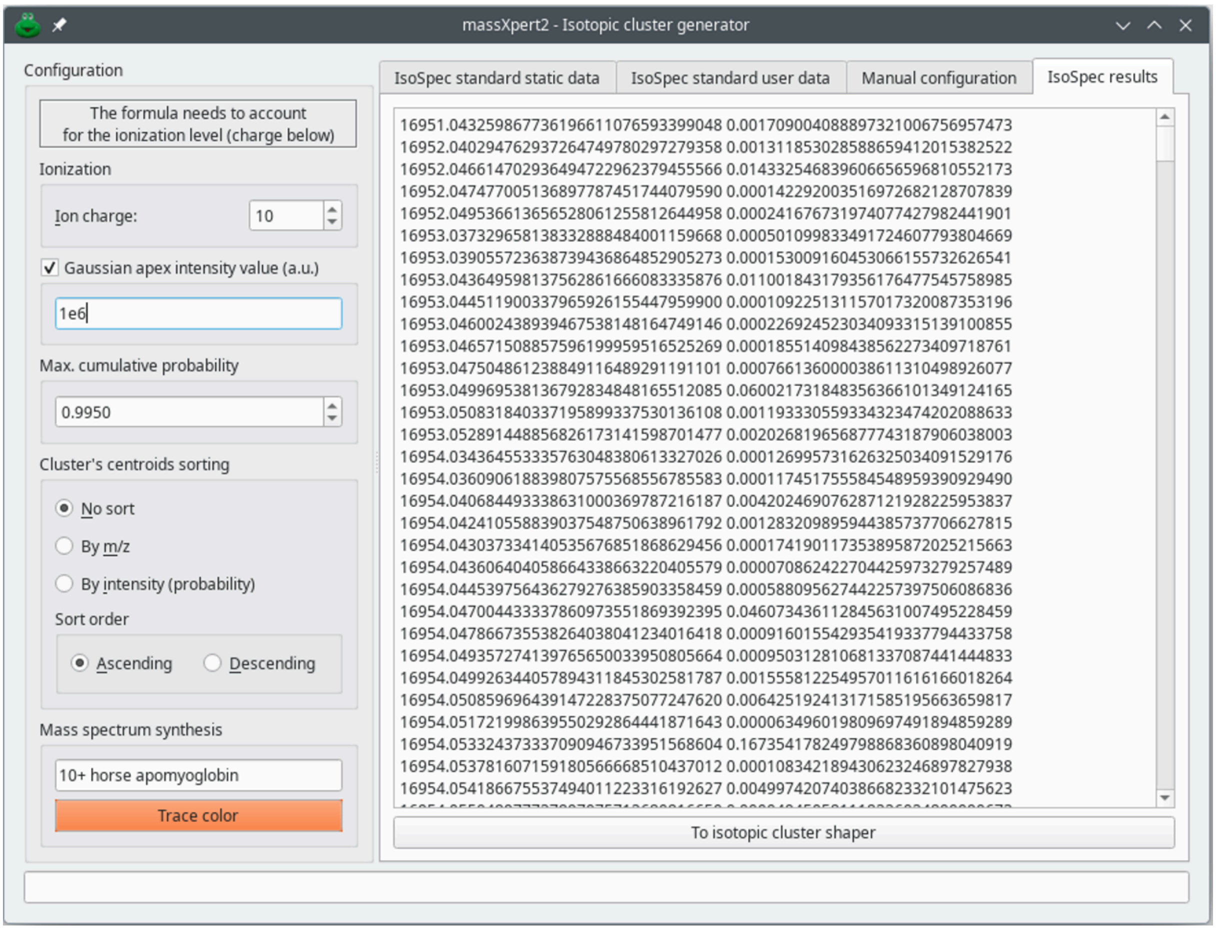
As above, once the analyte's elemental composition has been defined, as just described, click *Run*.

### 8.1.1.4 THE ISOSPEC RESULTS ARE CENTROIDS

Once the configurations have been terminated, the isotopic cluster calculations can finally be performed. In the *manual configuration* setting, the formula is automatically handled, since each chemical element that is defined goes along with the count of the corresponding atoms. In the case of the standard IsoSpec configuration (either modified or not), the user has to enter the chemical formula of the analyte in the *Formula* line edit widget.

Click *Run*. If the configuration was correct and the calculation could run properly, then the dialog window switches to the *IsoSpec results* tab ( FIGURE 8.9, “ RESULTS FROM THE ISOTOPIC CLUSTER CALCULATION ” ). That tab contains a text edit widget in which the results are displayed.

Note that the m/z values calculated by IsoSpec are “corrected” by MassXpert for the charge level that was specified in the left panel of the dialog window prior to their display in the results tab ( ).



The formula needs to account for the ionization level (charge below)

Ionization

Ion charge: 10

☑ Gaussian apex intensity value (a.u.)

1e6

Max. cumulative probability

0.9950

Cluster's centroids sorting

● No sort
○ By m/z
○ By intensity (probability)

Sort order

● Ascending ○ Descending

Mass spectrum synthesis

10+ horse apomyoglobin

Trace color

```
16951.0432598677361966110765933990048 0.0017090040888973210067569574 73
```

The IsoSpec library computes the probability of the various combinations of all the isotopes that make the elemental composition submitted to it. The results are in the form of peak centroid values along with corresponding probabilities. The sum of the probabilities corresponds to the *Max. cumulative probability* value that was set by the user.

**FIGURE 8.9:** **RESULTS FROM THE ISOTOPIC CLUSTER CALCULATION**

The results that are produced by IsoSpec represent the peak centroids of the isotopic cluster. The results are thus a set of (m/z,i) pairs that have not the characteristic shape (the profile) that is found in mass spectra. MassXpert features the ability to give a shape to the centroid peaks. For that, click the *To peak shaper* push button to open the *Peak Shaper* dialog window preloaded with the list of peak centroids.

## NOTE

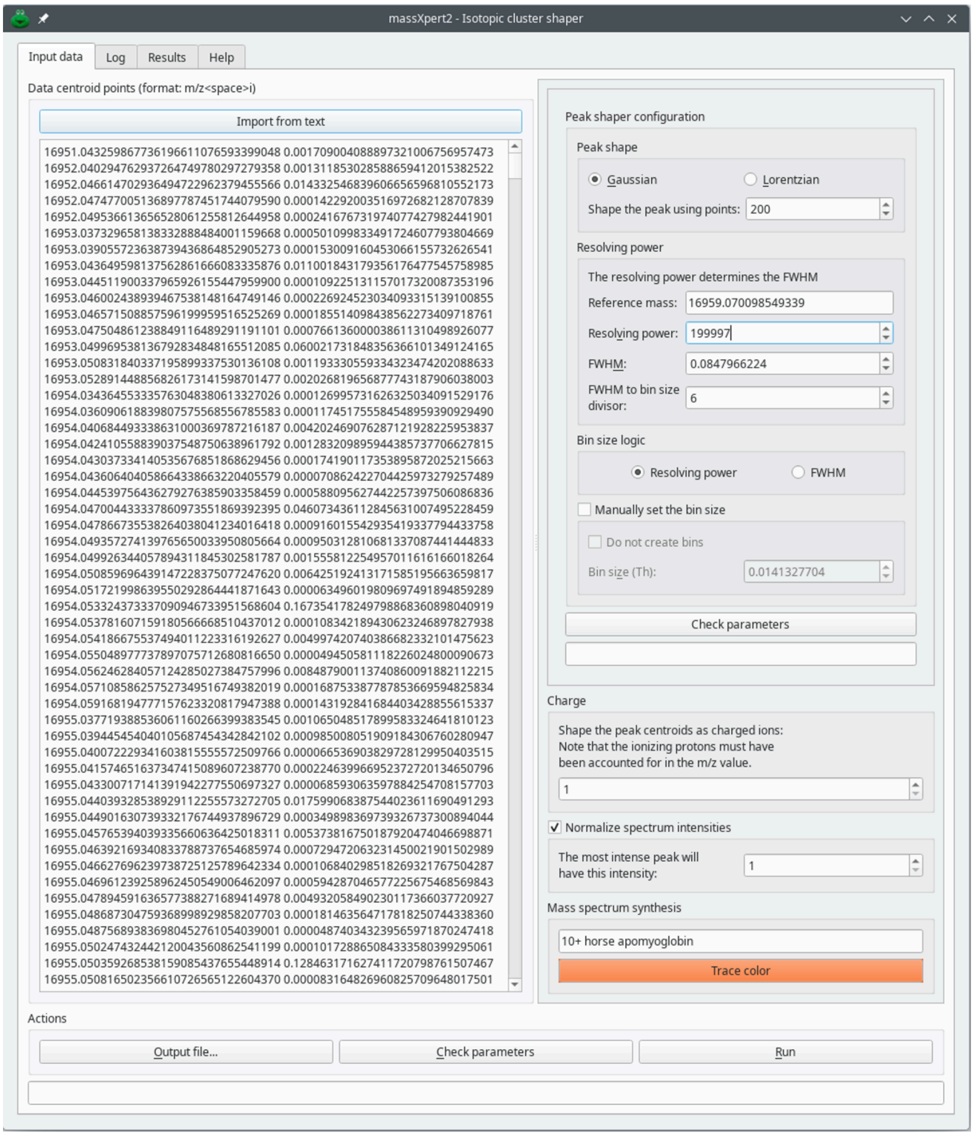
The *Peak shaper* dialog window's features might be used by themselves, not necessarily as the last step of an isotopic cluster simulation. To shape any peak starting from a centroid value and a corresponding intensity value, select the *Utilities* › Mass peak shape calculations menu item. The left hand side text edit widget appears empty. Paste into it the centroid data, one centroid per line, with the m/z value separated from the intensity by a space. Make sure to register the new centroid list by clicking the *Isotopic cluster shaper* push button.

The workings of this peak shaping feature is described in SECTION 8.1.2, " SHAPING MASS PEAK CENTROIDS INTO WELL-PROFILED PEAKS ".

### 8.1.2 SHAPING MASS PEAK CENTROIDS INTO WELL-PROFILED PEAKS

As mentioned earlier, the isotopic cluster calculation produces the isotopic cluster in the form of a set of (m/z,i) pairs that represent the centroids of the isotopic cluster's peaks and their relative intensity (probability). There are other data simulations or analysis processes that lead to having mass peaks defined by a single centroid m/z value and a corresponding intensity. Plotted to a graph, a centroid mass peak yields a bar, which is inadequate if the user is willing to simulate isotopic clusters to later compare these with experimental profiled mass peaks. The features in this dialog window aim at shaping each mass peak by creating a shape around the centroid.

The shape of mass peaks is typically Gaussian or Lorentzian (or a mix thereof). In order to convert mass peak centroids into something that resembles a real "profile" mass peak, a mathematical formula can be applied, with some parameters to configure the shapes generated. MassXpert makes the shaping of mass peaks accessible *via* the *Utilities* › Isotopic cluster shaper menu item. The window that opens up is shown in



Data centroid points (format: m/z<space>i)

```
16951.0432598677361966110765933990048 0.0017090040888973210067569574733
16952.0402947629372647497802972793358 0.0013118530285886594120153825222
16952.0466147029364947229623794555566 0.0143325468396066565968105521733
16952.0474770051368977874517440795590 0.0001422920035169726821287078399
16952.0495366136565280612558126449588 0.0002416767319740774279824419011
16953.0373296581383328884840011596668 0.0005010998334917246077938046699
16953.0390557236387394368648529052733 0.0001530091604530661557326265411
16953.0436495981375628616660833358766 0.0110018431793561764775457589855
16953.0445119003379659261554479599000 0.0001092251311570173200873531966
16953.0460024389394675381481647491466 0.0002269245230340933151391008555
16953.0465715088575961999595165252699 0.0001855140984385622734097187611
16953.0475048612388491164892911911011 0.0007661360000386113104989260777
16953.0499695381367928348481655120855 0.0600217318483563661013491241655
16953.0508318403371958993375301361088 0.0011933305593343234742020886333
16953.0528914488568261731415987014777 0.0020268196568777431879060380033
16954.0343645533357630483806133270266 0.0001269957316263250340915291766
16954.0360906188398075755685567855833 0.0001174517555845489593909294900
16954.0406844933386310003697872161877 0.0042024690762871219282259538377
16954.0424105588390375487506389617922 0.0012832098959443857377066278155
16954.0430373341405356768518686294566 0.0001741901173538958720252156633
16954.0436064040586643386632204055799 0.0000708624227044259732792574899
16954.0445397564362792763859033584599 0.0005880956274422573975060868366
16954.0470044333378609735518693923955 0.0460734361128456310074952284599
16954.0478667355382640380412340164188 0.0009160155429354193377944337588
16954.0493572741397656500339508056644 0.0009503128106813370874414448333
16954.0499263440578943118453025817877 0.0015558122549570116161660182644
16954.0508596964391472283750772476200 0.0064251924131715851956636598177
16954.0517219986395502928644418716433 0.0000634960198096974918948592899
16954.0533243733370909467339515686044 0.1673541782497988683608980409199
16954.0537816071591805666685104370122 0.0001083421894306232468978279388
16954.0541866755374940112233161926277 0.0049974207403866823321014756233
16954.0550489777378970757126808166500 0.0000494505811182260248000906733
16954.0562462840571242850273847579966 0.0084879001137408600918821122155
16954.0571085862575273495167493820199 0.0001687533877878536695948258344
16954.0591681947771576233208179473888 0.0001431928416844034288556153377
16955.0377193885360611602663993835455 0.0010650485178995833246418101233
16955.0394454540401056874543428421022 0.0009850080519091843067602809477
16955.0400722293416038155555725097666 0.0000665369038297281299504035155
16955.0415746516373474150896072387700 0.0002246399669523727201346507966
16955.0433007171413919422775506973277 0.0000685930635978842547081577033
16955.0440393285389291122555732727055 0.0175990683875440236116904912933
16955.0449016307393321767449378967299 0.0003498983697393267373008940444
16955.0457653940393356606364250183111 0.0053738167501879204740466988711
16955.0463921693408337887376546859744 0.0007294720632314500219015029899
16955.0466276962397387251257896423344 0.0001068402985182693217675042877
16955.0469612392589624505490064620977 0.0005942870465772256754685698433
16955.0478945916365773882716894149788 0.0049320584902301173660377209277
16955.0486873047593689989298582077033 0.0001814635647178182507443383600
16955.0487568938369804527610540390011 0.0000487403432395659718702474188
16955.0502474324421200435608625411999 0.0001017288650843335803992950611
16955.0503592685381590854376554489144 0.1284631716274117207987615074677
16955.0508165023566107265651226043700 0.0000831648269608257096480175011
```

Peak shaper configuration

Peak shape

● Gaussian ○ Lorentzian

Shape the peak using points: 200

Resolving power

The resolving power determines the FWHM

Reference mass: 16959.070098549339

Resolving power: 199997

FWHM: 0.0847966224

FWHM to bin size divisor: 6

Bin size logic

● Resolving power ○ FWHM

☐ Manually set the bin size

☐ Do not create bins

Bin size (Th): 0.0141327704

Charge

Shape the peak centroids as charged ions:
Note that the ionizing protons must have
been accounted for in the m/z value.

1

☑ Normalize spectrum intensities

The most intense peak will have this intensity: 1

Mass spectrum synthesis

10+ horse apomyoglobin

This dialog window allows one to configure the shaping of mass centroid peaks. Setting a spectrum name in the *Mass spectrum name* line edit widget will help recognize the result mass spectrum once displayed in the *Mass spectra* window in the MineXpert companion software (see below).

**FIGURE 8.10: SETTING-UP OF THE CENTROID MASS PEAK SHAPING PROCESS**

The mass centroid peaks are listed in the *Data centroid points (m/z,i)* text edit widget. These values may come from different origins. They can be set automatically from the isotopic cluster calculation dialog window (see SECTION 8.1.1.4, " THE ISOSPEC RESULTS ARE CENTROIDS " ); they can come from the user editing them in place; finally they can come from the user pasting text from the clipboard.

The width of the "profile" mass peak is determined either by setting the resolving power of the instrument or by setting the width of the peak at half maximum of its height (FWHM).

The profile mass peaks generated can be either of a Gaussian or of a Lorentzian shape. That parameter is configured by selecting the corresponding radio button widget. The number of points used to actually craft the shape of the peak is configurable in the *Shape the peak using points* spin box widget.

There are two ways to configure the width of the Gaussian/Lorentzian shape around the centroid peak. Either one knows the resolving power of the instrument, or one can measure the full width at half maximum (FWHM) of the experimental peak. The *Resolving power* group box widget contains all the widgets required to define that FWHM. Because the FWHM actually depends on the resolving power and the resolving power is computed using the m/z value, there needs to be a *Reference mass* provided for the calculation. If the data in the text edit widget on the left hand side pane of the window is filled with data from an isotopic cluster calculation, the reference mass is the m/z value found at the row that divides the rows into two equal parts (kind of a median row). If the data come from the user either pasting or editing text in the text edit widget, then it is their responsibility to fill in the reference mass.

To define the FWHM value, one can just enter it in the corresponding spin box widget. When the editing is finished and the focus leaves the spin box widget, the *Bin size logic* group box widget will have the *FWHM* radio button widget selected. The *Resolving power* spin box widget now contains the proper resolving power value, as computed using the FWHM value and the Reference mass.

To define the resolving power value, one can just enter it in the *Resolving power* spin box widget. When the editing is finished and the focus leaves the spin box widget, the *Bin size logic* group box widget will have the *Resolving power* radio button widget selected. The *FWHM* spin box widget now contains the proper FWHM value, as computed using the resolving power value and the Reference mass.

The end point of the configuration procedures described above is to have a proper m/z bin size for the mass spectrum that is generated by the combination of the various peaks shapes. The greater the resolving power of the instrument, the smaller the FWHM value and thus the smaller the bin size. The bin size that is computed is displayed in the *Bin size (Th)* spin box widget inside of the *Manually set the bin size* group box widget. If checked, this group box widget becomes active to let the user set the bin size manually. Its computation does not occur anymore upon modification of the values of FWHM or of the resolving power or of the reference mass.

The successful configuration of the bin size, *via* the definition of the FWHM or of the resolving power takes a bit of patience. The *Check parameters* push button triggers a verification of the configuration and provides feedback about it in the line edit widget below the push button.

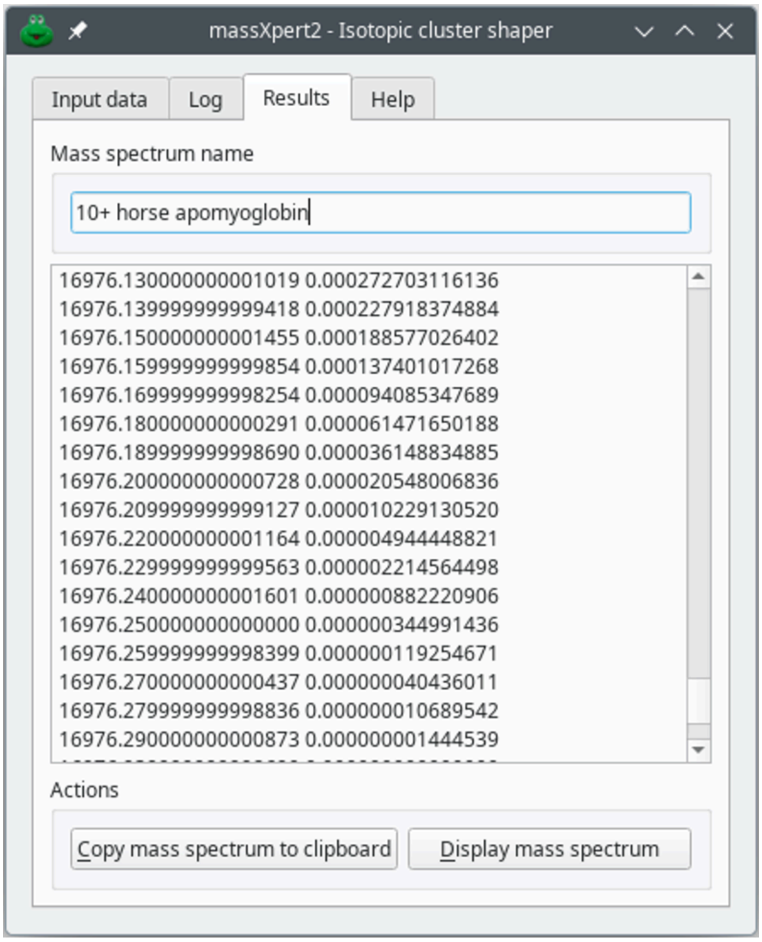
The *Charge* spin box widget allows setting the charge of the ions of which the centroid (m/z,i) pairs are listed in the text edit widget on the left hand side of the dialog window.

### Warning: Use caution with the value entered in the Charge spin box widget

When the peak centroid data have been filled in by the isotopic cluster calculation dialog window, do not change this value (let it be 1, as the default). This is because the isotopic cluster calculation has already taken into account the charge that was set at that step.

If the peak centroid data have been filled by editing in place of pasting text, then the charge needs to be set according to the actual charge of the ions for which the centroid peaks are being shaped.

Once the peak shaping parameters have been checked, click *Run* . The dialog window shifts tab to *Results*, as shown in Figure 8.11, “ Shaped peaks mass spectral data ” .



Mass spectrum name: 10+ horse apomyoglobin

```
16976.130000000001019 0.000272703116136
16976.139999999999418 0.000227918374884
16976.150000000001455 0.000188577026402
16976.159999999999854 0.000137401017268
16976.169999999998254 0.000094085347689
16976.180000000000291 0.000061471650188
16976.189999999998690 0.000036148834885
16976.200000000000728 0.000020548006836
16976.209999999999127 0.000010229130520
16976.220000000001164 0.000004944448821
16976.229999999999563 0.000002214564498
16976.240000000001601 0.000000882220906
16976.250000000000000 0.000000344991436
16976.259999999998399 0.000000119254671
16976.270000000000437 0.000000040436011
16976.279999999998836 0.000000010689542
16976.290000000000873 0.000000001444539
```

The mass spectral data corresponding to the combination of each individual “shaped” centroid peak are displayed in this tab. Plotting these data will produce nicely shaped profile peaks matching what would be seen by recording a mass spectrum on an instrument.

FIGURE 8.11: **SHAPED PEAKS MASS SPECTRAL DATA**

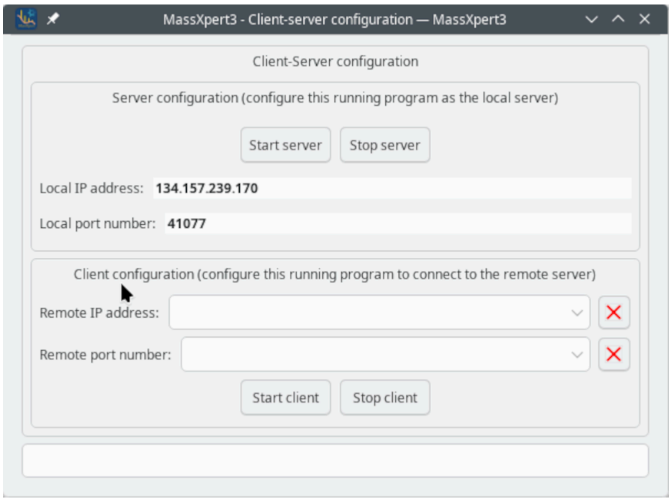
### Tip

To ensure that various isotopic cluster calculation-based mass spectra can later be recognized, the user is advised, for the different simulations, to insert distinct names in the *Mass spectrum name* line edit widget. This name will be used later when creating the mass spectrum if the user asks that the mass spectrum be displayed by clicking *Display mass spectrum*. In the eventuality that the mass spectrum name is not changed from a simulation to the other, a safeguard process ensures that names are absolutely unambiguous by appending to the mass spectrum name the time at which the mass spectrum is displayed.

Once the mass spectrum has been generated by the peak shaper, it might be copied to the clipboard, for visualization in the companion MineXpert software or any other software capable of reading space-separated-values files. It is also possible to actually trigger the automatic display of the spectrum in MineXpert by clicking *Display mass spectrum*. In this case, it is necessary to have first configured the client-server connection, as described at SECTION 8.2, “ CONFIGURATION OF THE MASSXPERT AND MINEXPERT INTERCOMMUNICATIONS ”.

## 8.2 CONFIGURATION OF THE MASSXPERT AND MINEXPERT INTERCOMMUNICATIONS

The two programs of the MsXpertSuite software suite, MassXpert and MineXpert can exchange data via the network, provided that both are set up for TCP-based communications. In MassXpert, the configuration of these communications is performed by using the *Utilities* > Client-Server configuration menu item. The configuration is performed in the dialog window shown in FIGURE 8.12, “ SETTING-UP OF THE CENTROID MASS PEAK SHAPING PROCESS ” .



Local IP address: 134.157.239.170

Local port number: 41077

This dialog window allows one to configure the FTP-based communications between the two software pieces of the MsXpertSuite software suite: MassXpert and MineXpert.

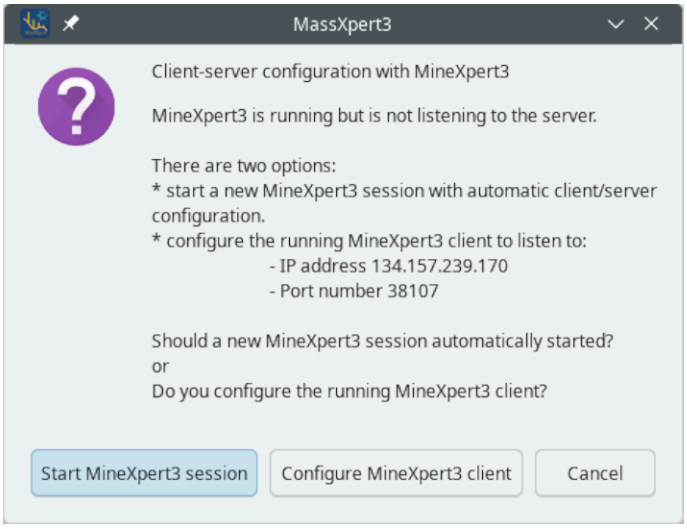
FIGURE 8.12: SETTING-UP OF THE CENTROID MASS PEAK SHAPING PROCESS

To start the server, click the *Start server* button and the IP address and port number will be displayed. To start the client, insert the IP address and port number in the *Client configuration* group box and click *Start client* .

### 8.2.1 DYNAMIC CONFIGURATION OF THE MASSXPERT AND MINEXPERT INTERCOMMUNICATIONS

Each time a task occurring in MassXpert requires to display data in MineXpert the former program checks if the latter program is already running and if an active connection between the two is up.

If an active connection between MassXpert and MineXpert is alive, the data are immediately dislayed. Instead, if the MineXpert program is effectively running, but there is no active connection between MassXpert and MineXpert, the user is prompted with a dialog box to suggest configuring a connection (see FIGURE 8.13, “DYNAMIC COMMUNICATIONS CONFIGURATION BETWEEN MASSXPERT AND MINEXPERT” ).



Client-server configuration with MineXpert3

MineXpert3 is running but is not listening to the server.

There are two options:
* start a new MineXpert3 session with automatic client/server configuration.
* configure the running MineXpert3 client to listen to:
  - IP address 134.157.239.170
  - Port number 38107

Should a new MineXpert3 session automatically started?
or
Do you configure the running MineXpert3 client?

Dynamic communications connection between MassXpert and MineXpert

If MassXpert finds that MineXpert is running but that there is no active connection between the two program, this dialog box allows the user to configure MineXpert so that a connection made. If this is done, MassXpert will send the data to MineXpert for it to display them.

FIGURE 8.13: **DYNAMIC COMMUNICATIONS CONFIGURATION BETWEEN MASSXPERT AND MINEXPERT**

Whenever MassXpert cannot find a MineXpert process, it will automatically try to start if it finds the MineXpert executable installed on the computer. In that case, if the MineXpert program can be started, the data are transferred as soon as the process and a communications connection can be set up.

# 9 DATA CUSTOMIZATIONS

In this chapter, the user will be walked trough an example of how new polymer chemistry definition data can be generated and included in the automatic “data detection system” of MassXpert (that is how new polymer chemistry definitions should be registered with the system).

Customization is typically performed by the normal user (not the Administrator nor the Root of the machine) and as such new data are typically stored in the user's “home” directory. On UNIX machines, the “home” directory is usually the `/home/username` directory, where username is the login username. On MS-Windows, that directory is typically the `C:/Documents and Settings/username` once again with username being the login username.

## TIP

Although MS-Windows pathnames use a back slash (“\”, in this book these are composed using forward slashes for a number of valid reasons. The reader only needs to replace back slashes with the forward variety.

In the next sections we will refer to that “home directory” (be it on UNIX or MS-Windows machines) as the $HOME directory, as this the standard environment variable describing that directory in GNU/Linux.

When MassXpert is executed, it automatically tries to read data configuration files from the home directory (in the `.MassXpert` directory). Once this is done, it reads all the data configuration files in the installation directory (typically, on GNU/Linux that would be the configuration data in the `/usr/share/MassXpert/data` directory or, on MS-Windows, the `c:/Program Files/MassXpert/data` directory).

We said above that MassXpert tries to read the data configuration files from the home directory. But upon its very first execution, right after installation, that directory does not exist, and in fact MassXpert creates that directory for us to populate it some day with interesting new data.

The `$HOME/MassXpert` directory should have a structure mimicking the one that was created upon installation of the software, that is, it should contain the following two directories:

- `polChemDefs`
- `polSeqs`

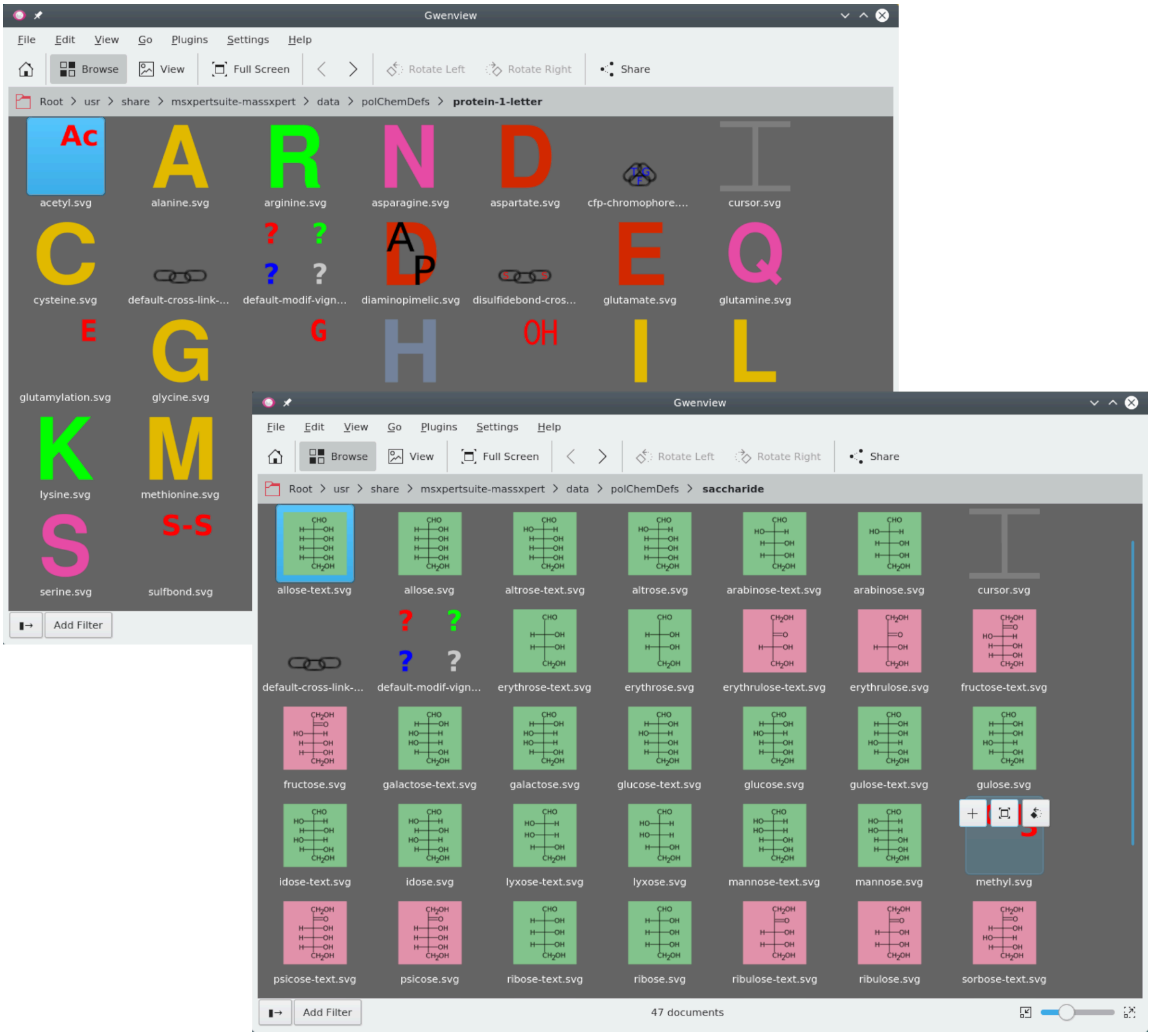
Those are the directories where the user is invited to store their personal data. In order to start a new definition, one might simply copy in the `polChemDefs` one of the polymer chemistry definitions that are shipped with MassXpert. What should be copied? An entire polymer chemistry definition directory, like for example the following:

`/usr/local/share/MassXpert/polChemDefs/protein-1-letter` or

`C:/Program Files/MassXpert3/polChemDefs/protein-1-letter`

Once that polymer chemistry definition is copied, one may start studying how it actually works. This directory contains the following kinds of files:

- `protein-1-letter.xml`: the polymer chemistry definition file. This is the file that is read upon selection of the corresponding polymer chemistry definition name in XpertDef. If the polymer chemistry definition is not yet registered with the system (described later), then open that file by browsing to it by clicking *Cancel* (see CHAPTER 3, XPERTDEF: DEFINITION OF POLYMER CHEMISTRIES );
- SVG files: *scalar vector graphics* files used to render graphically the sequence in the sequence editor. For example, `arginine.svg` contains the graphical representation of the arginine monomer. There are such graphics files also for the modifications (like, for example, the `sulphation.svg` contains the graphical representation of the sulphation modification. FIGURE 9.1, " THE POLYMER CHEMISTRY DEFINITION DIRECTORY " shows two examples of SVG files belonging to two distinct polymer chemistry definitions;
- `chemPad.conf`: configuration file for the chemical pad in the XpertCalc module;
- `monomer_dictionary`: file establishing the relationship between any monomer code of the polymer chemistry definition and the graphical SVG file to be used to render graphically that monomer in the sequence editor;
- `modification_dictionary`: file establishing the relationship between any monomer modification (see SECTION 4.8.1, " SELECTED MONOMER(S) MODIFICATION " ) and the graphical SVG file to be used to render graphically that modification onto the modified monomer in the sequence editor;
- `cross_linker_dictionary`: file establishing the relationship between any cross-link (see SECTION 4.9, " MONOMER CROSS-LINKING " ) and the graphical SVG file to be used to render graphically that cross-link onto the cross-linked monomers in the sequence editor;
- `pka_ph_pi.xml`: file describing the acido-basic data (see SECTION 4.16, " PKA, PH, PI AND CHARGES " ) pertaining to ionizable chemical groups in the different entities of the polymer chemistry definition;



Each monomer of the polymer chemistry definition ought to have a corresponding SVG file with which it has to be rendered graphically should that monomer be inserted in the polymer sequence. This example shows two SVG files corresponding to two monomers each belonging to a different polymer chemistry definition.

FIGURE 9.1: **THE POLYMER CHEMISTRY DEFINITION DIRECTORY**

The polymer sequence editor is not a classical editor. There is no font in this editor: when the user starts keying-in a polymer sequence in the editor, the small SVG graphics files are rendered into raster *vignettes* at both the proper resolution and screen size and displayed in the sequence editor. The user is totally in charge of designing the SVG graphics files for each of the monomers defined in the polymer sequence editor. Of course, reusing material is perfectly possible.

### TIP

One powerful software to edit SVG files is INKSCAPE (HTTPS://INKSCAPE.ORG/) ↗ , on any platform.

There is one constraint: that the `monomer_dictionary` file lists with precision “what monomer code goes with what SVG graphics file ” . That file has the following contents, for example, for the “protein-1-letter” polymer chemistry definition, as shipped in the MassXpert package:

```
# This file is part of the MassXpert3 project.
# The "MassXpert3" project is released ---in its entirety--- under the
# GNU General Public License and was started (in the form of the
# GNU polyxmass project) at the Centre National de la Recherche
# Scientifique (FRANCE), that granted me the formal authorization to
# publish it under this Free Software License.
# Copyright (C) 2006,2023 Filippo Rusconi
# This is the monomer_dictionary file where the correspondences
# between the codes of each monomer and their graphic file (pixmap
# file called "image") used to graphically render them in the
# sequence editor are made.
# The format of the file is like this :
#
-------------------------------------
# A%alanine.svg
# where A is the monomer code and alanine.svg is a
# resolution-independent svg file.
# Each line starting with a '#' character is a comment and is ignored
# during parsing of this file.
# This file is case-sensitive.
A%alanine.svg
C%cysteine.svg
D%aspartate.svg
E%glutamate.svg
F%phenylalanine.svg
G%glycine.svg
H%histidine.svg
I%isoleucine.svg
K%lysine.svg
L%leucine.svg
M%methionine.svg
N%asparagine.svg
```

```
P%proline.svg
Q%glutamine.svg
R%arginine.svg
S%serine.svg
T%threonine.svg
V%valine.svg
W%tryptophan.svg
Y%tyrosine.svg
```

What one sees from the contents of the file is that each monomer code has an associated SVG file. For example, when the user has to key-in a valine monomer, they key-in the code `V` and XpertEdit knows that the monomer vignette to show has to be rendered using the `valine.svg` file.

For the monomer modification graphical rendering, the situation is somewhat different, as seen in the `modification_dictionary` file:

```
# This file is part of the MassXpert project.
# The "MassXpert3" project is released ---in its entirety--- under the
# GNU General Public License and was started (in the form of the GNU
# polyxmass project) at the Centre National de la Recherche
# Scientifique (FRANCE), that granted me the formal authorization to
# publish it under this Free Software License.
# Copyright(C) 2006,2023 Filippo Rusconi
# This is the modification_dictionary file where the correspondences
# between the name of each modification and their graphic file (pixmap
# file called "image") used to graphicallly render them in the
# sequence editor are made. Also, the graphical operation that is to
# be performed upon chemical modification of a monomer is listed ('T'
# for transparent and 'O' for opaque). See the manual for details.
# The format of the file is like this :
# ------------------------------------ #
Phosphorylation%T%phospho.svg
# where Phosphorylation is the name of the modification. T indicates
# that the visual rendering of the modification is a transparent
# process (O indicates that the visual rendering of the modification
# is a full image replacement 'O' like opaque).
```

```
# phospho.svg is a resolution-independent svg file.
# Each line starting with a '#' character is a comment and is ignored
# during parsing of this file.
# This file is case-sensitive.
Phosphorylation%T%phospho.svg
Sulphation%T%sulpho.svg
AmidationAsp%O%asparagine.svg
Acetylation%T%acetyl.svg
AmidationGlu%O%glutamine.svg
Oxidation%T%oxidation.svg
```

There are two ways to render a chemical modification of a monomer:

- *Opaque* rendering: the initial monomer vignette is replaced using the one listed in the file for the modification. One example is the “AmidationGlu” modification:
  AmidationGlu%O%glutamine.svg
  Upon amidation of a glutamyl residue (“AmidationGlu” is the name of a modification in the current polymer chemistry definition), the graphical representation of the modification involves the *replacement* of the glutamyl residue vignette in the sequence editor with the new one, that happens to be in the `glutamine.svg` file. In other words, the process involves an “ *Opaque* ” overlay of the vignette for unmodified Glu with a vignette rendered by using the `glutamine.svg` file.

- *Transparent* rendering: the initial monomer vignette is overlaid with a new vignette that is read from an SVG file that has a transparent background. One example is the “Phosphorylation” modification:
  Phosphorylation%T%phospho.svg
  The monomer undergoing a phosphorylation has its vignette overlaid with a “ *Transparent* ” one showing a small red “P” that is read from the `phospho.svg` file.

When designing vignettes, the best thing to do is to convert the text to path, so that the rendering is absolutely perfect.

### Warning

It is absolutely essential, for the proper working of the sequence editor, that the SVG files be square (that is, width = height).

Once the new polymer chemistry has been correctly defined, it is time to register that new definition to the system. To recap: all the files for that definition should reside in a same directory, exactly the same way as the files pertaining to a given polymer chemistry definition are shipped in MassXpert altogether in one directory. The name of the new polymer chemistry definition should be unambiguous, with respect to other registered polymer chemistry definitions.

The way personal polymer chemistry definitions are registered is by creating a personal polymer chemistry definition catalogue file, which must comply with both following requirements:

- Be named `xxxxx-polChemDefsCat`, with "xxxxx" being a discretionary string (this might well be your login username). The requirement is that `-polChemDefsCat` be the last part of the filename.

  ### Tip

  Please *DO NOT USE* spaces or diacritical signs in your filenames. *RESTRICT* yourself to ASCII characters between [a-z], [0-9], "_" and "-".

  This is actually something very general as a recommendation in order to not suffer from severe headaches when you least expect them...

- Be located in the `$HOME/.MassXpert/data/polChemDefs` directory and have the following format:
  dna=/path/to/definition/directory/dna/dna.xml
  In this example, the "dna" polymer chemistry definition is being registered as a file `dna.xml` located in the `dna` directory, itself located in the `/path/to/definition/directory` directory;

Note that if a new polymer chemistry definition should be made available system-wide, then it is logical that its directory be placed along the ones shipped with MassXpert and a new local catalogue file might be created to register the new polymer chemistry definition.

At this point the new polymer chemistry definition might be tested. Typically, that involves restarting the MassXpert program and creating a brand new polymer sequence of the new definition type. The first step is to check if the new definition is successfully registered with the system, that is, it should show up a an available definition upon creation of the new polymer sequence. If not, then that means that the catalogue file could not be found or parsed correctly.

When problems like this one occurs, the first thing to do is to ensure that the console window (on MS-Windows it is systematically started along with the program; on GNU/Linux the way to have it is to start the program from the shell) so as to look with attention at the different messages that might help understanding what is failing.

Please, do not hesitate to submit bug reports (see SECTION 2, " FEEDBACK FROM THE USERS " for the method to contect the author for feature requests or bug reports).

# A GNU General Public License version 3

Version 3, 29 June 2007

## Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL , as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

## TERMS AND CONDITIONS

### 0. DEFINITIONS.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

## 1. SOURCE CODE.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

## 2. BASIC PERMISSIONS.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

## 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

## 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

# 5. CONVEYING MODIFIED SOURCE VERSIONS.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- **a.** The work must carry prominent notices stating that you modified it, and giving a relevant date.
- **b.** The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- **c.** You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- **d.** If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

# 6. CONVEYING NON-SOURCE FORMS.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- **a.** Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- **b.** Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable

physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

**c.** Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

**d.** Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

**e.** Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM ).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

## 7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d. Limiting the use for publicity purposes of names of licensors or authors of the material; or

e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

## 8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

## 9. ACCEPTANCE NOT REQUIRED FOR HAVING COPIES.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

## 10. AUTOMATIC LICENSING OF DOWNSTREAM RECIPIENTS.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

## 11. PATENTS.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

## 12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

## 13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

## 14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

## 15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

## 16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## 17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

## END OF TERMS AND CONDITIONS

# How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
  one line to give the program's name and a brief idea of what it does.

Copyright (C)

  year

```

```
  name of author

This program is free software: you can redistribute it and/or modify it
under the terms of the

  GNU

General Public License as published by the Free Software Foundation, either
version 3 of the License, or (at your option) any later version. This
program is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE. See the

  GNU

General Public License for more details. You should have received a copy of
the

  GNU

General Public License along with this program. If not, see
https://www.gnu.org/licenses/
.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
  program

Copyright (C)

  year



  name of author

This program comes with ABSOLUTELY NO WARRANTY; for details type ‘

  show w

’. This is free software, and you are welcome to redistribute it under
certain conditions; type ‘

  show c
```

```
    ' for details.
```

The hypothetical commands '`show w`' and '`show c`' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL , see HTTPS://WWW.GNU.ORG/LICENSES/ .

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read HTTPS://WWW.GNU.ORG/LICENSES/WHY-NOT-LGPL.HTML .

# COLOPHON

**About the author.** Filippo Rusconi is a senior researcher at the French national research council ( *Centre national de la Recherche scientifique* , CNRS ). Filippo has a background in biochemistry and organic chemistry and was trained during his Ph.D. as a bioanalytical chemist. He has extensive knowledge of analytical techniques involved in the study of biopolymers.

Filippo Rusconi is the author of a handbook about mass spectrometry for biochemists (French). The book was published by the French sci/tech publisher LAVOISIER (HTTPS://WWW.LAVOISIER.FR) ↗ .

**Colophon.** The look of this book ( PDF file) is the result of me having read many books from the O'Reilly publisher.

The typesetting of the book has been done on a Debian GNU/Linux computer using only Free Software. Use of the DocBook Authoring and Publishing Suite ( DAPS (HTTPS://GITHUB.COM/OPENSUSE/DAPS) ↗ ) from SUSE was key in the process.

The layout adopted for this book is an adaptation of the SUSE stylesheets. I would like to thank Frank Sundermeyer <fsundermeyer@opensuse.org>, Stefan Knorr <sknorr@suse.de> and Tom Schraitle ( SEE HERE (HTTPS://GITHUB.COM/TOMSCHR) ↗ ) for being helpful with all my questions.

The main font used was EBGARAMOND (HTTPS://GITHUB.COM/GEORGD/EB-GARAMOND) ↗ and the symbol/mathematical font was from the STIX PROJECT (HTTPS://WWW.STIXFONTS.ORG/) ↗ (font: STIX2Math).

The screen shots were taken with Spectacle, the screen capture program shipped along with my KDE (HTTPS://WWW.KDE.ORG/) ↗ desktop environment and resampled using The GNU image manipulation program THE GIMP (HTTPS://WWW.GIMP.ORG/) ↗ . Illustrations were done in INKSCAPE (HTTPS://INKSCAPE.ORG/) ↗ , a vectorial drawing software.