

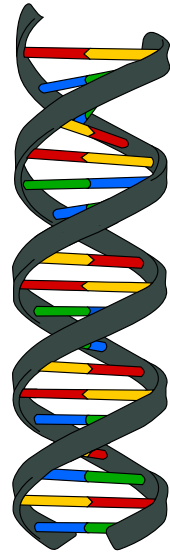


# LymPHOS<sup>2</sup>

**Organizing and Sharing Biological Data of the  
Human Lymphocyte Proteome using Python**

 **PyData** Barcelona, 19-21th April 2017

# INTRODUCTION - PROTEOME



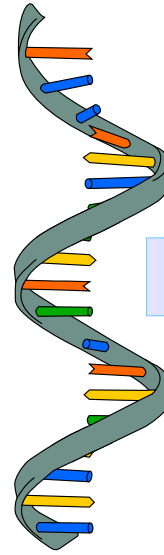
DNA

## Genome

- Set of genes of an organism.
- Static: almost the same genes for all the cell of an individual, during all its live.
- > 30.000 genes (human).



*Transcription*



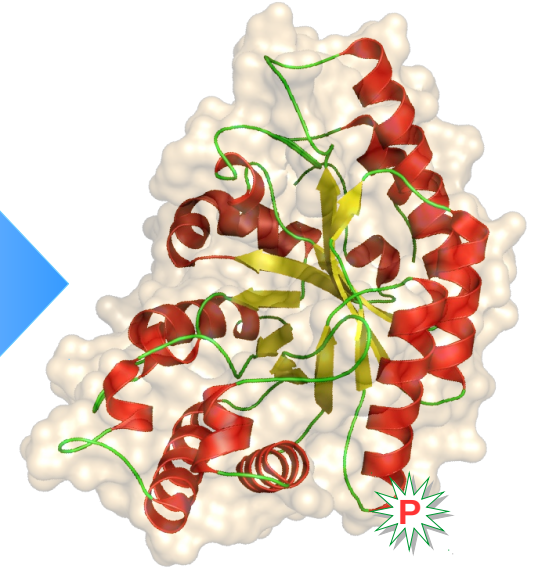
RNAs

## Transcriptome

- Set of RNAs in a cell, tissue, or organism at a defined time and conditions.
- Variable.
- Post-Transcriptional Modifications.



*Translation*



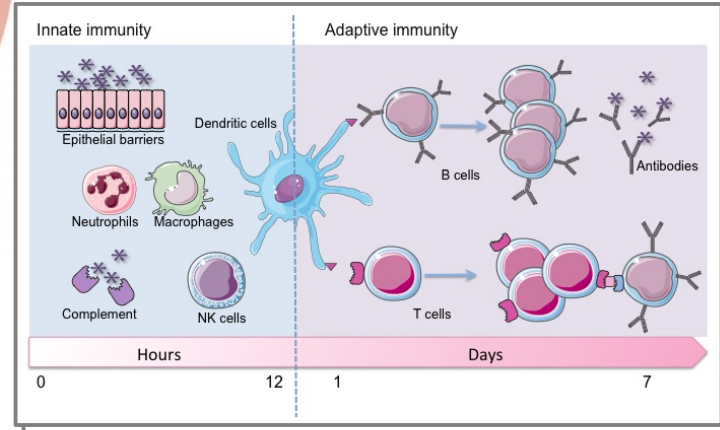
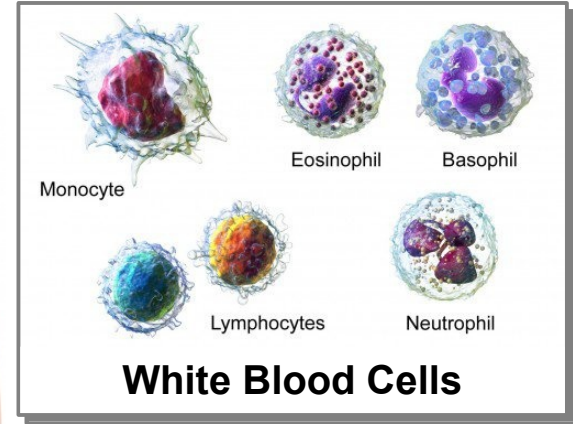
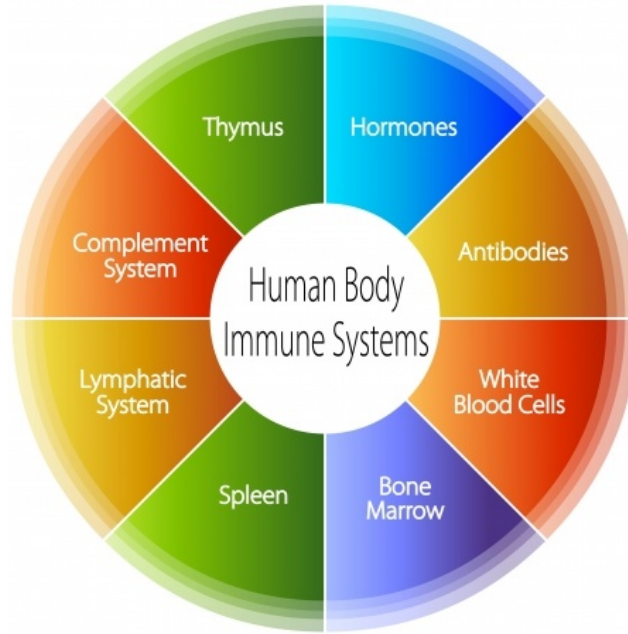
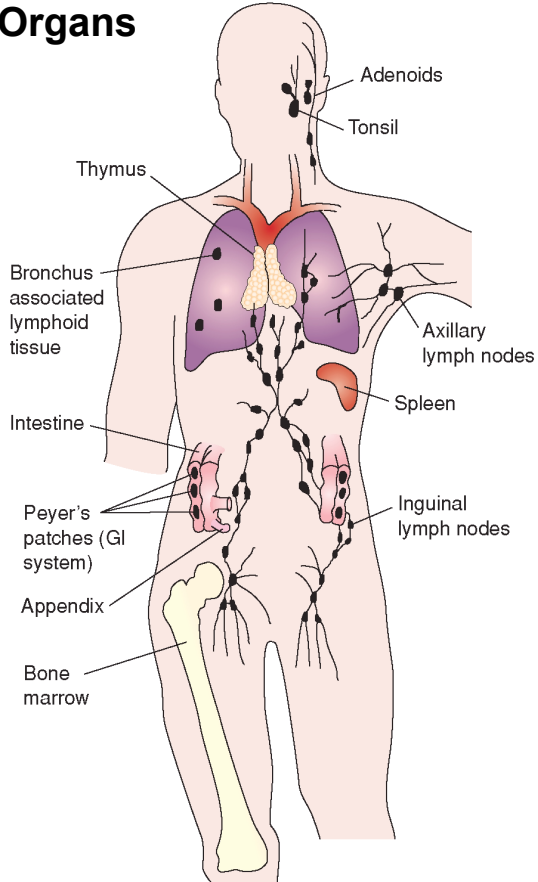
Proteins

## Proteome

- Set of proteins expressed by a cell, tissue, or organism at a defined time and conditions.
- Highly Variable.
- *Post-Translational Modifications*.
- > 300.000 possible proteins (human).

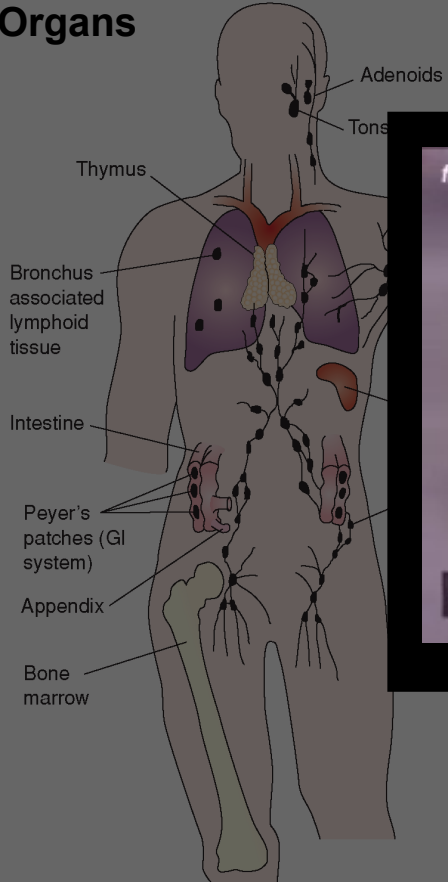
# INTRODUCTION – IMMUNE SYSTEM

## Organs

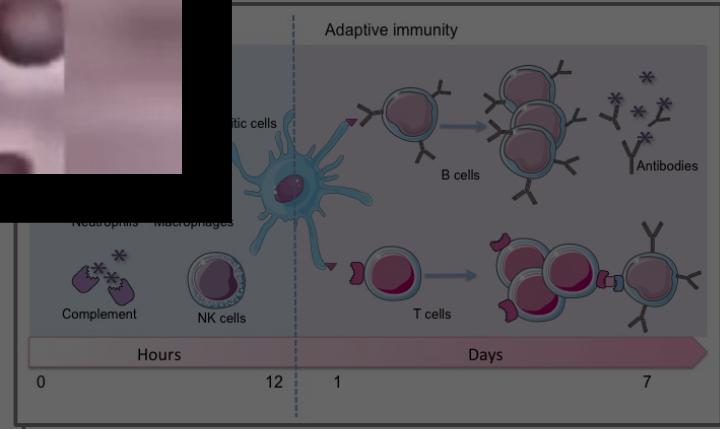
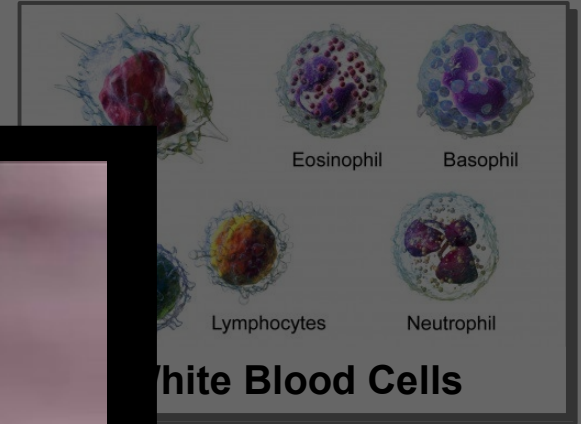


# INTRODUCTION – IMMUNE SYSTEM

## Organs

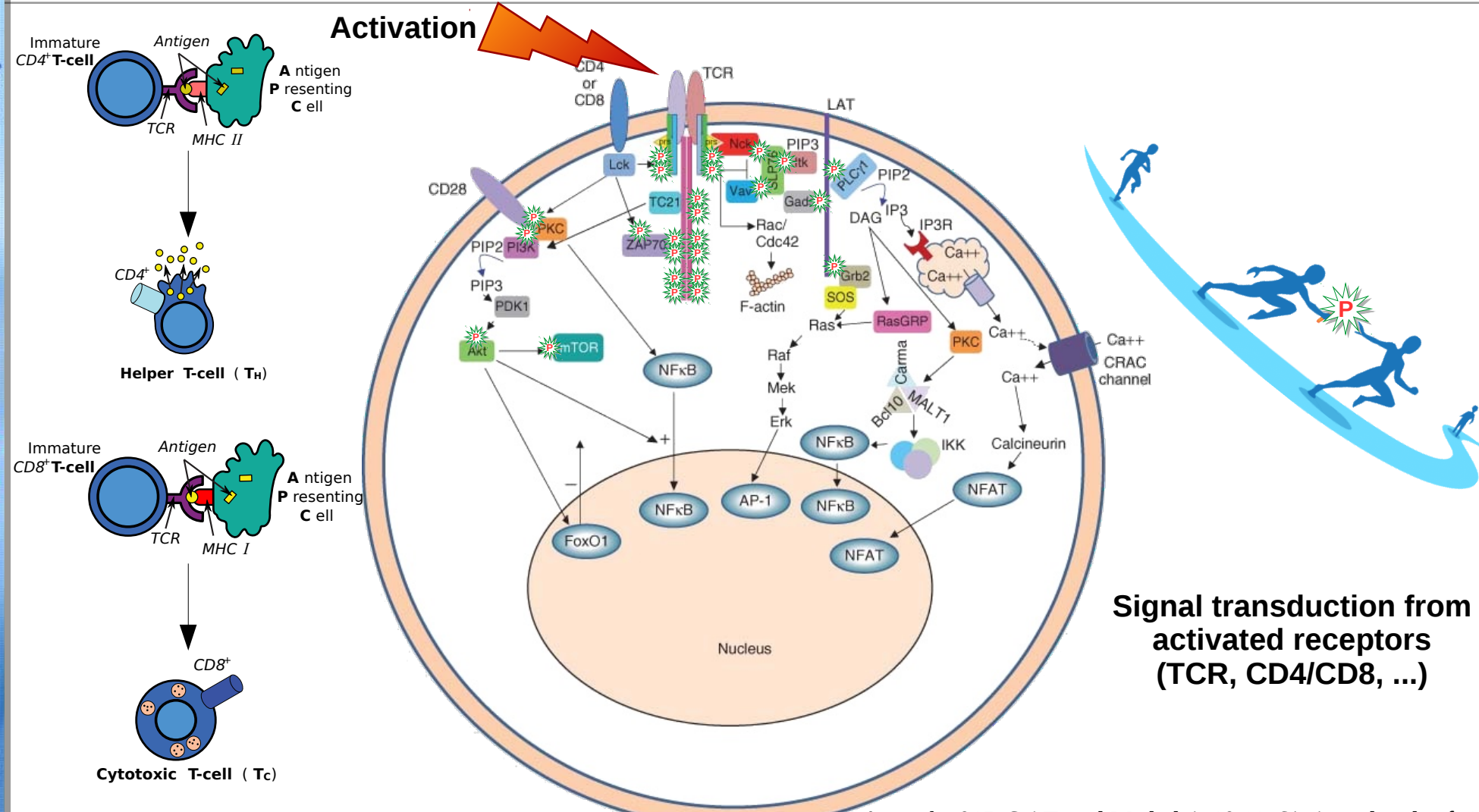


[fb.com/ScienceNaturePage](https://www.facebook.com/ScienceNaturePage)





# INTRODUCTION – T LYMPHOCYTES

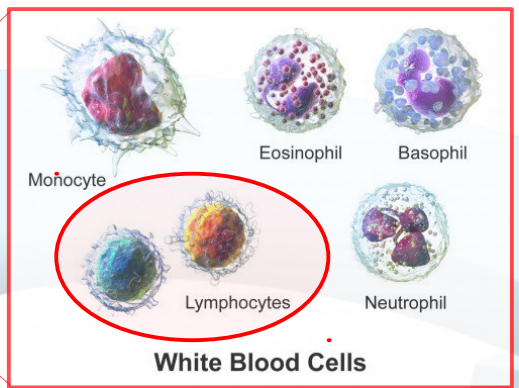
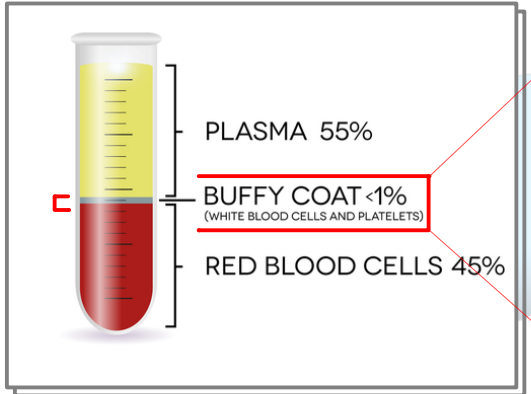


Signal transduction from activated receptors (TCR, CD4/CD8, ...)



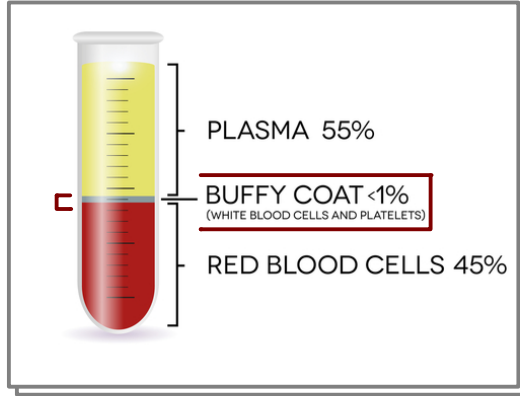
# LYMPHOS WORKFLOW – WET LAB

## Human Blood Buffy Coat Pools

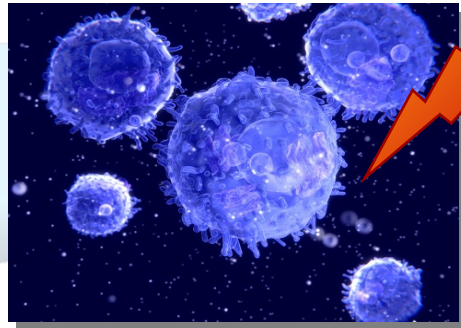


# LYMPHOS WORKFLOW – WET LAB

## Human Blood Buffy Coat Pools



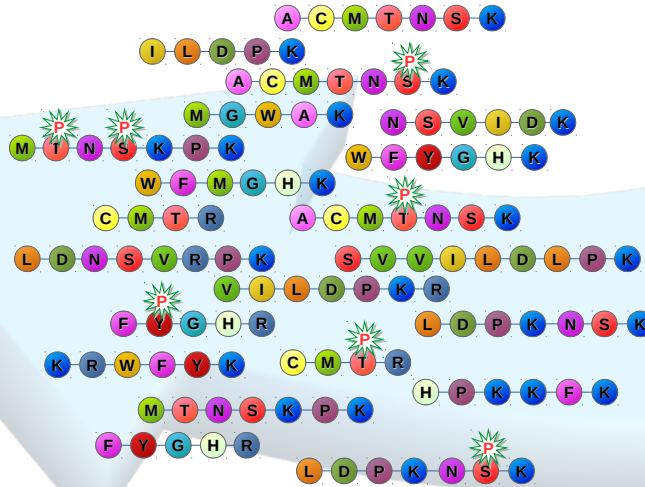
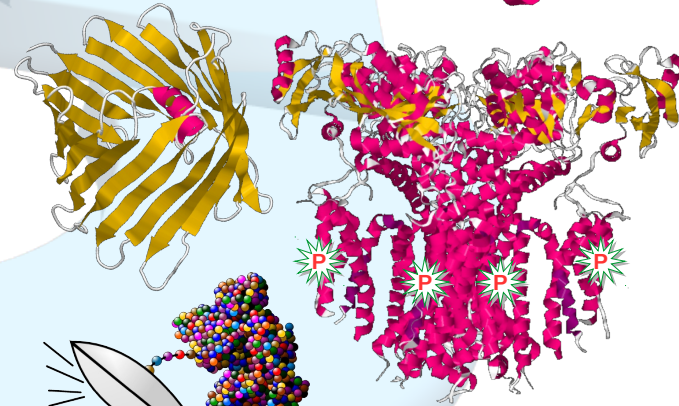
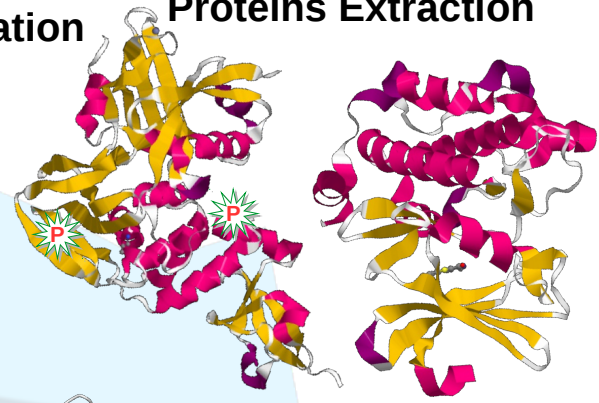
## Human T Lymphocytes



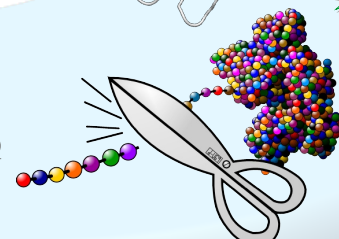
## Activation



## Proteins Extraction



## Cut/Digest Proteins with Trypsin

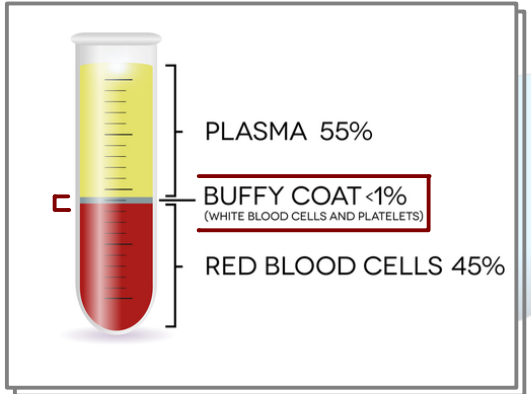


## Digestion Peptides

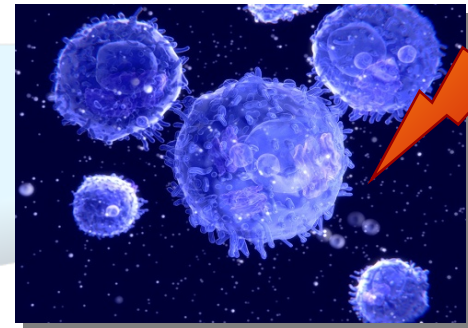


# LYMPHOS WORKFLOW – WET LAB

## Human Blood Buffy Coat Pools

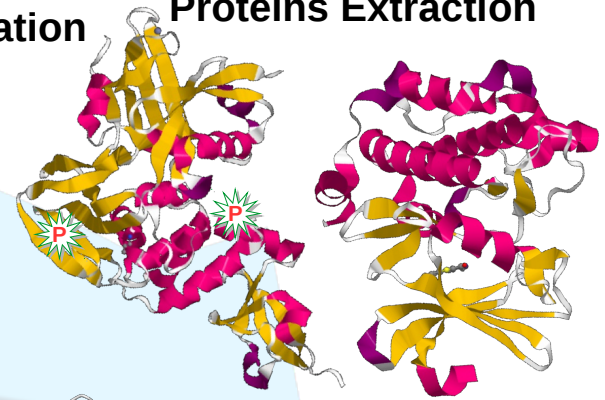


## Human T Lymphocytes

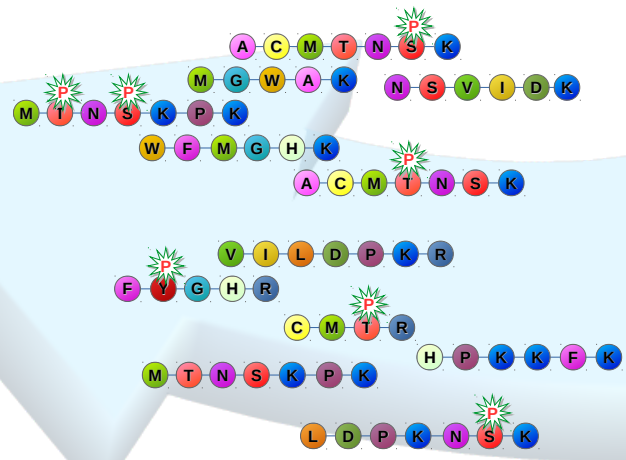


## Activation

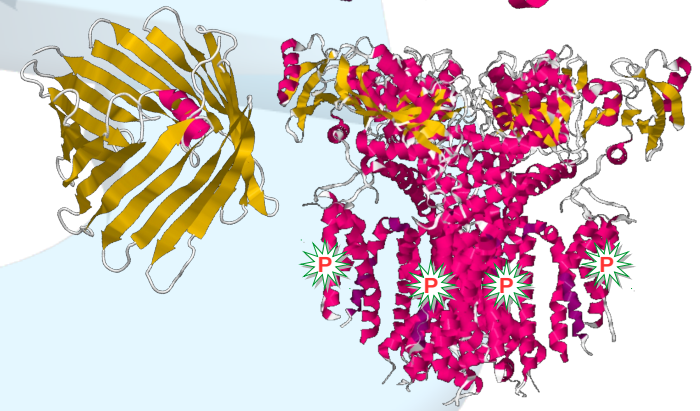
## Proteins Extraction



## Phospho-Peptides Enrichment



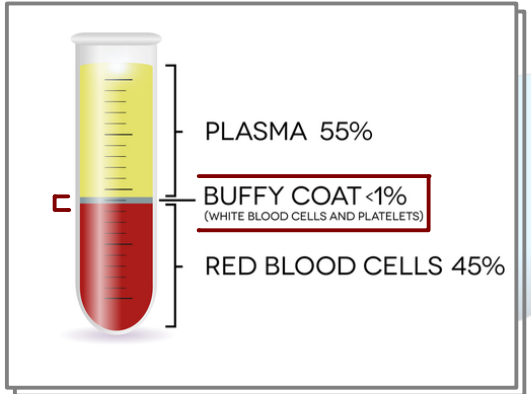
## Digestion Peptides



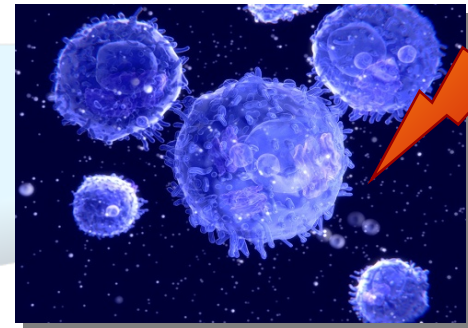


# LYMPHOS WORKFLOW – WET LAB

## Human Blood Buffy Coat Pools



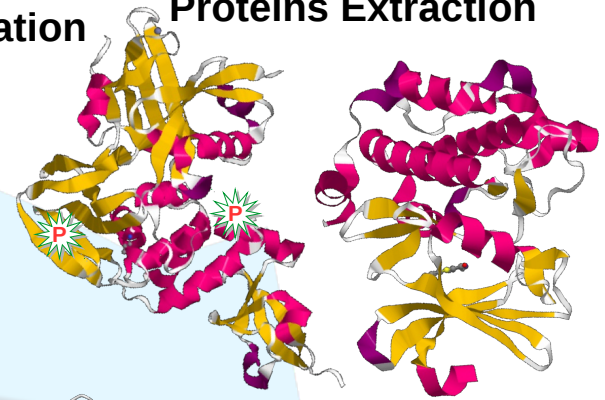
## Human T Lymphocytes



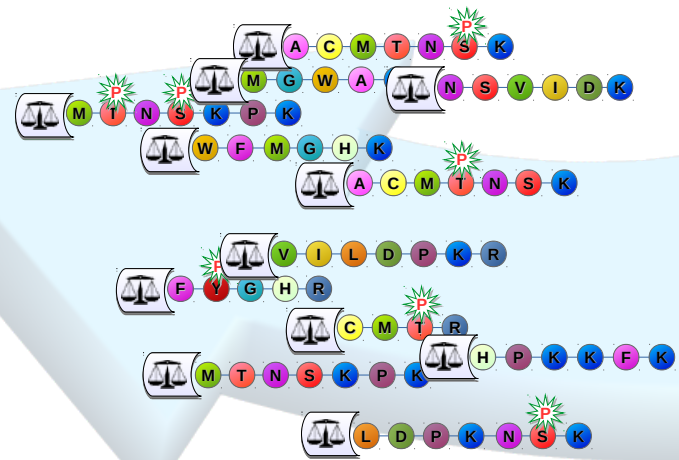
## Activation



## Proteins Extraction



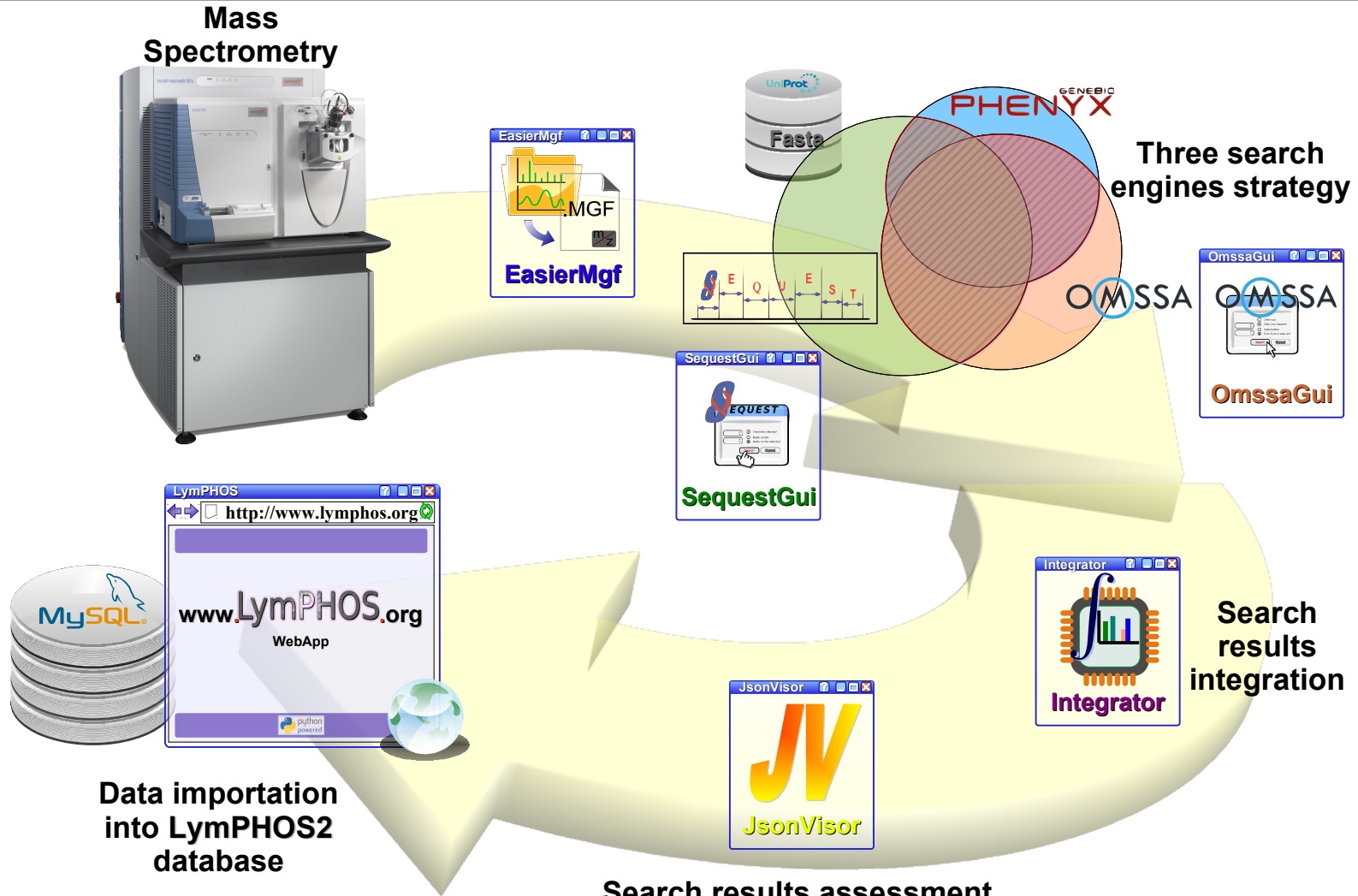
## Quantitative Reporter Tagging



Mass Spectrometry

Digestion Peptides

# LYMPHOS WORKFLOW – DRY LAB



# LYMPHOS WORKFLOW – DRY LAB

Mass



EasierMgf

Mass Spectrometry



Thermo  
 RAW files  
 (proprietary)

**Call backend software to extract spectra data**

raw files selection  
 Select Remove Clear

scan sequence  
 1 PQD/HCD 1 CID 1 MS3

coprocessing actions  
 Remove +1 scans  Split MS2-MS3

Convert

ReAdw4Mascot2 extract\_msn\_classic extract\_msn\_com ReAdw

centroid  precursorFromFilterLine  verbose  gzip  
 compress

keep readw mzXML  Do mzML (only)

```

===== Starting Process =====
exe_readw/ReAdw.exe --mzXML --centroid C:\temp\rai_demo\29_IP
D:\temp\PT_0_15_120_20120110_Fr16.mzXML
out-Xca
(Thermo
file vers
file should con
(got comput er
Calculating : na
--done (mzX /L
===== Proc
    
```

**backend**

**XDK**

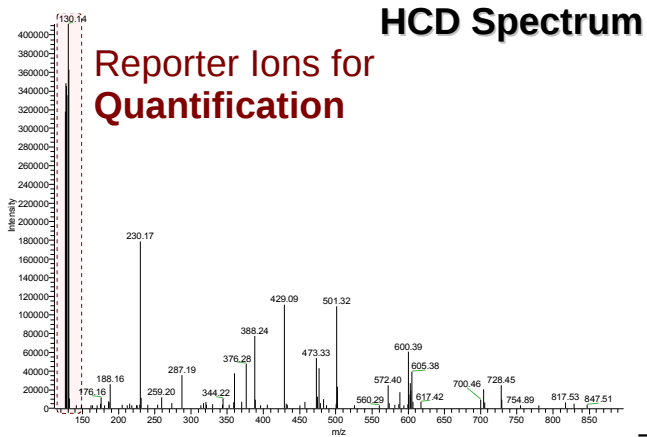


# LYMPHOS WORKFLOW – DRY LAB

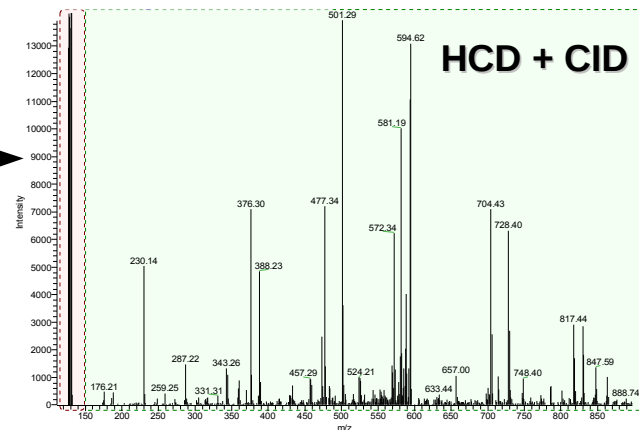
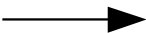
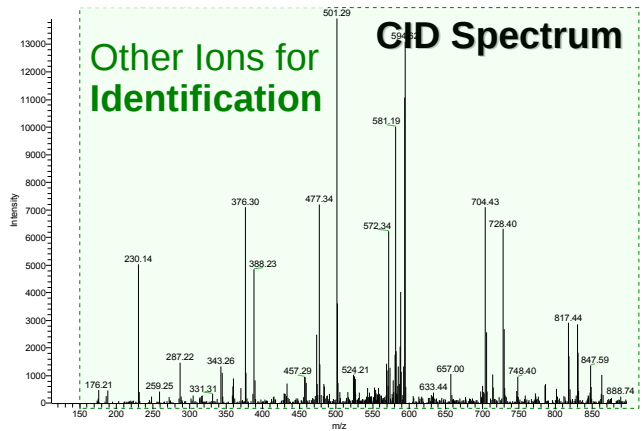
Mass



EasierMgf



Mix quantification and identification data from different kinds of spectra of the same peptide into a *single spectrum*





# LYMPHOS WORKFLOW – DRY LAB

Mass



## EasierMgf

Mass Spectrometry



Thermo  
RAW files  
(proprietary)

```
==== Starting Process =====
exe_readw/ReAdW.exe --mzXML --centroid C:\temp\raw_demo\29_I
D:\temp\29_IMAC_PI_0_15_120_20120110_Fr16.mzXML
out- Xcalibur 2.0 interface initialized.
(Thermo lib opened file C:\temp\raw_demo\29_IMAC_PI_0_15_120_
file version is 63, interface version is 2.0 or greater
file should contain scan numbers 1 through 6955
(got computer name: BIOINFO01)
Calculating sha1-sum of mzXML
--done (mzXML sha1):07084a9db1a377f0c7158ec3e8694ba
===== Process Terminated =====
```



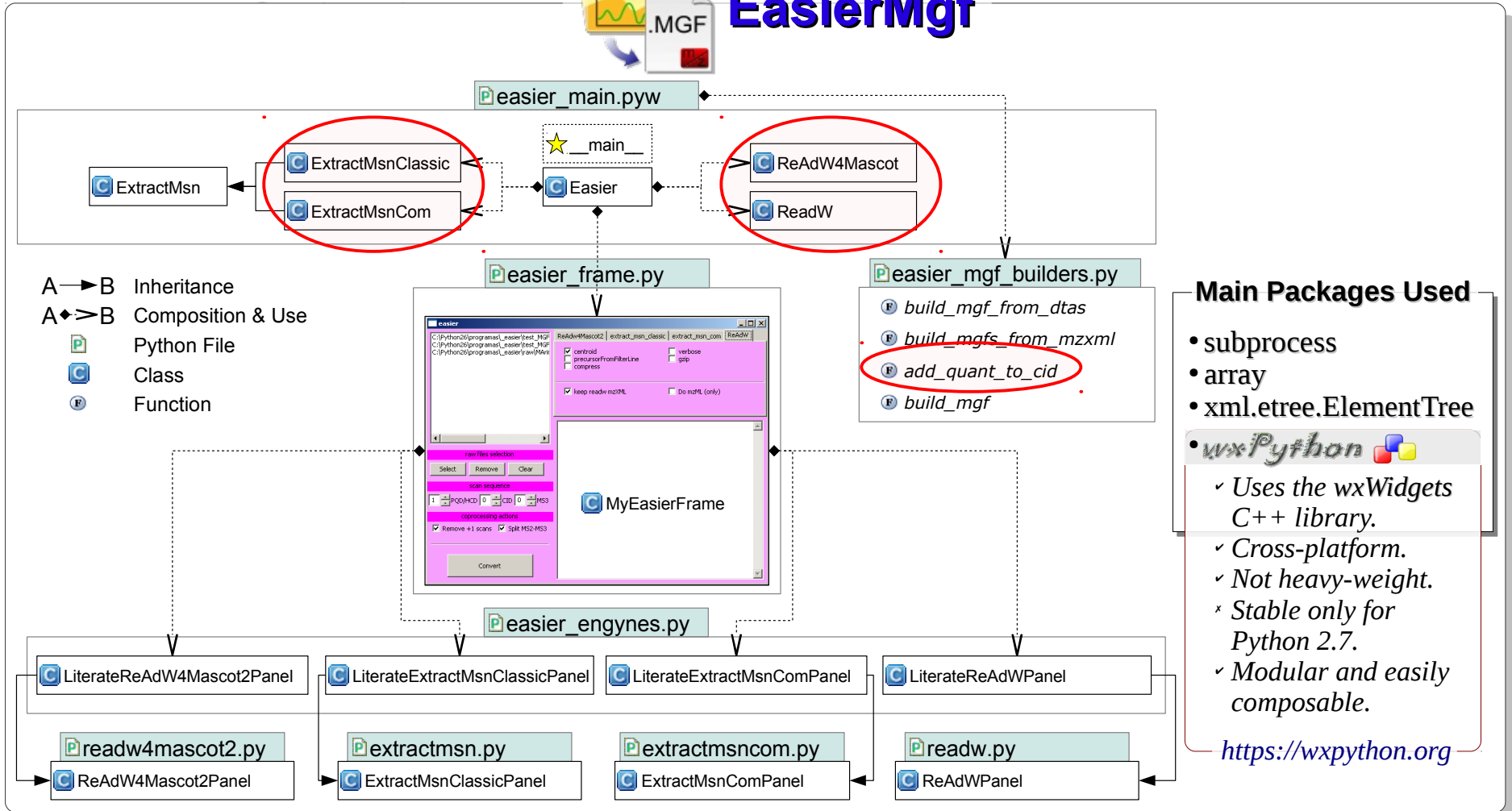
MGF files  
(standard)  
Directory

# LYMPHOS WORKFLOW – DRY LAB

Mass



## EasierMgf



### Main Packages Used

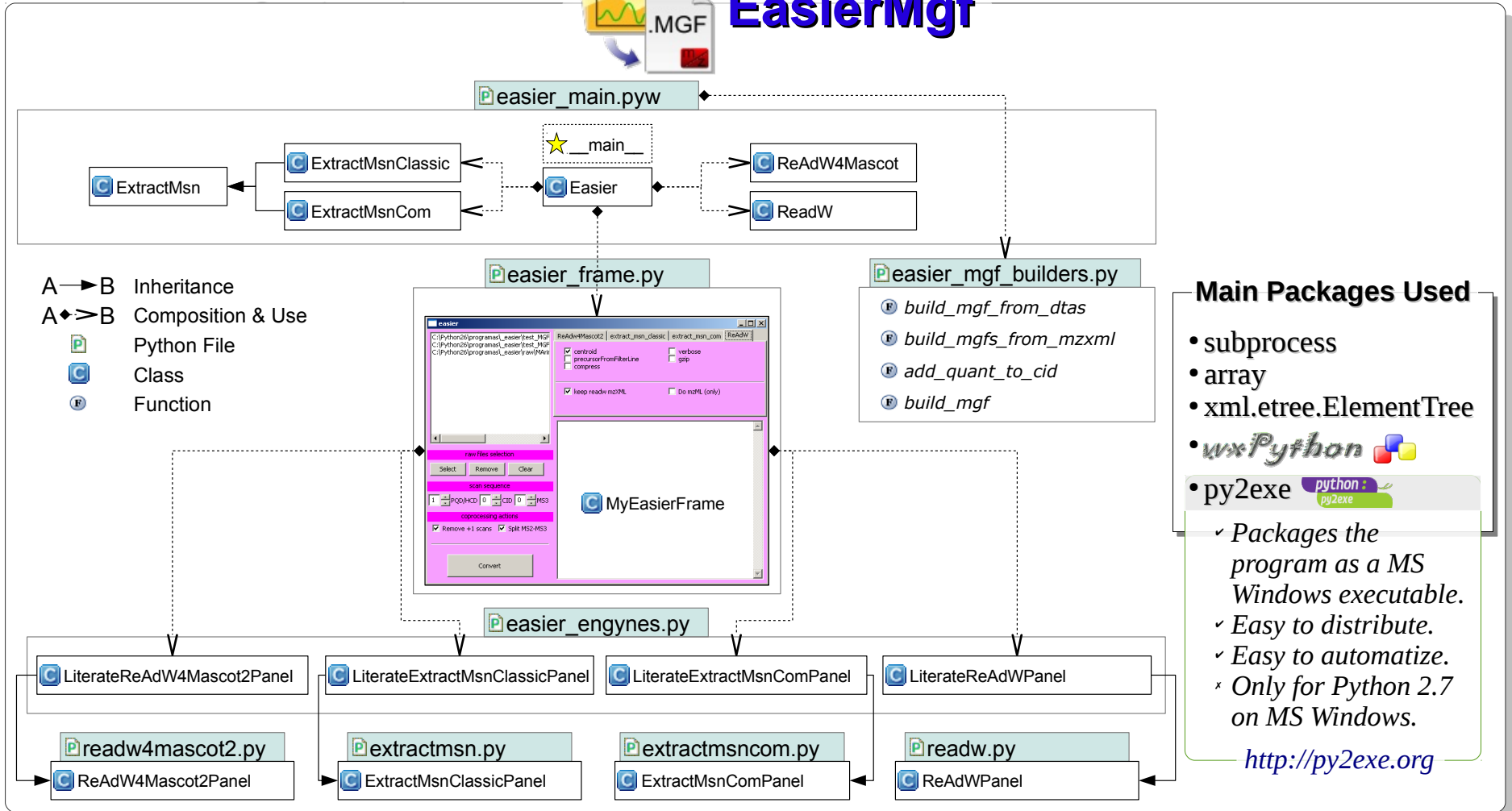
- subprocess
- array
- xml.etree.ElementTree
- wxPython
  - ✓ Uses the wxWidgets C++ library.
  - ✓ Cross-platform.
  - ✓ Not heavy-weight.
  - ✗ Stable only for Python 2.7.
  - ✓ Modular and easily composable.

<https://wxpython.org>

# LYMPHOS WORKFLOW – DRY LAB



## EasierMgf



### Main Packages Used

- subprocess
- array
- xml.etree.ElementTree
- wxPython
- py2exe

✓ Packages the program as a MS Windows executable.  
 ✓ Easy to distribute.  
 ✓ Easy to automatize.  
 ✗ Only for Python 2.7 on MS Windows.

<http://py2exe.org>

# LYMPHOS WORKFLOW – DRY LAB

Mass Spectrometry



SequestGui



MGF files Directory



Target + Decoy

**OMSSA GUI**

Group 1  
 Mgf DIR:  select

Met OX  
 STY Phos  
 ST DH

Quantitation:  select

Group 2  
 Mgf DIR:  select

Met OX  
 STY Phos  
 ST DH

Quantitation:  select

Files

OMSSA executable:  select

Blast formatted DB:  select

Report file:  select

Also save results as XML (.omx files)

Precursor Tolerance (Da):   
 Product Tolerance (Da):   
 Minimum charges:   
 Maximum charges:

Enzyme:  select  
 Missed cleavages:

**Run Omssa**



SequestGui TSV Result file

**SequestGui 0.5**

Path mgfs:  .params File

Path mgfs:  .params File

Path mgfs:  .params File

Sequest Path:  Search

Charges:  +1  +2  +3  +4  +5

Temporal Dir:  Dir

Output file:  File

XCorr Threshold:  made at bp.csic@uab.cat

temporal files  
 keep dtaa  
 keep .outs

**Run Sequest**



OmssaGui TSV Result file

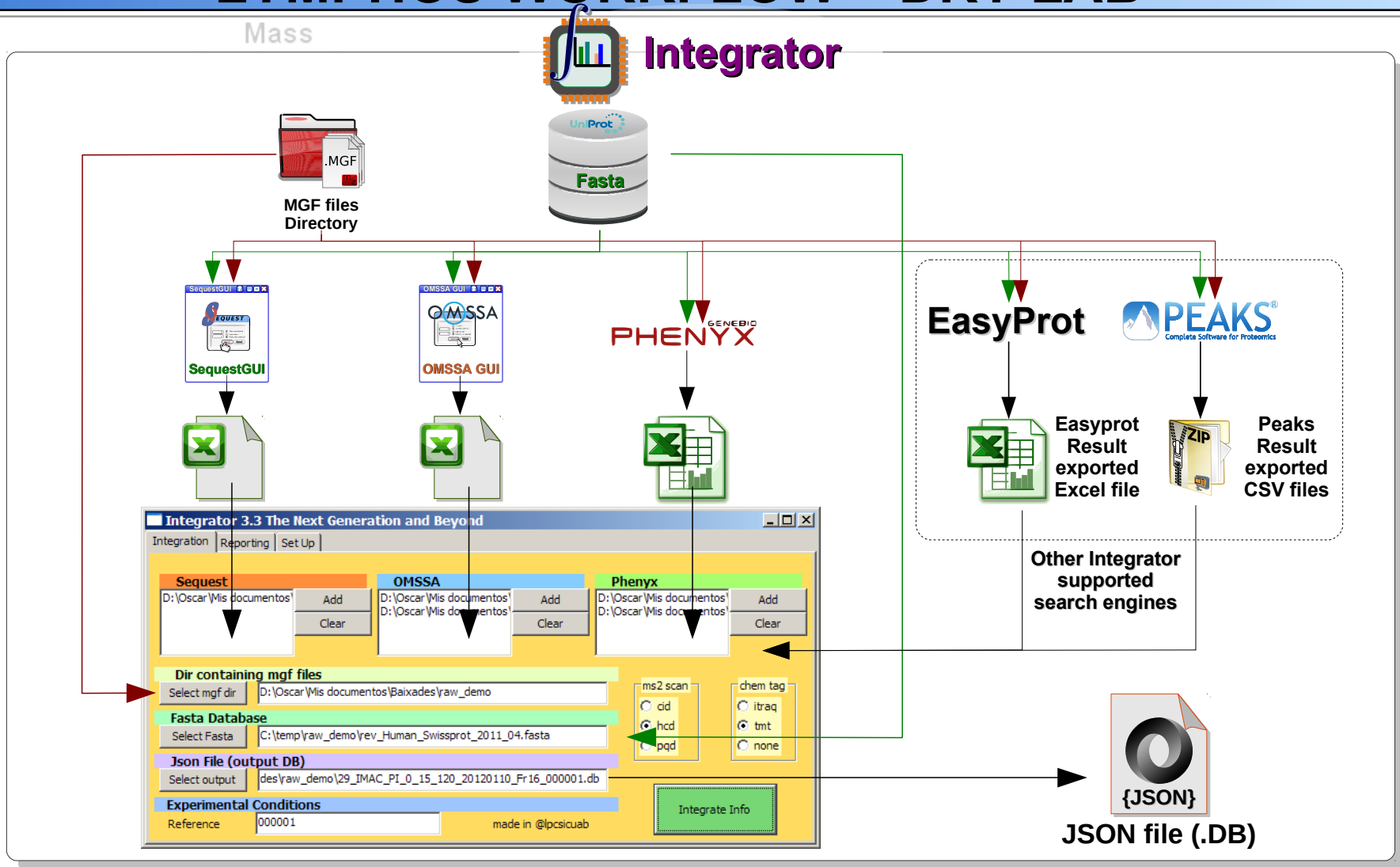
OMSSA

OmssaGui





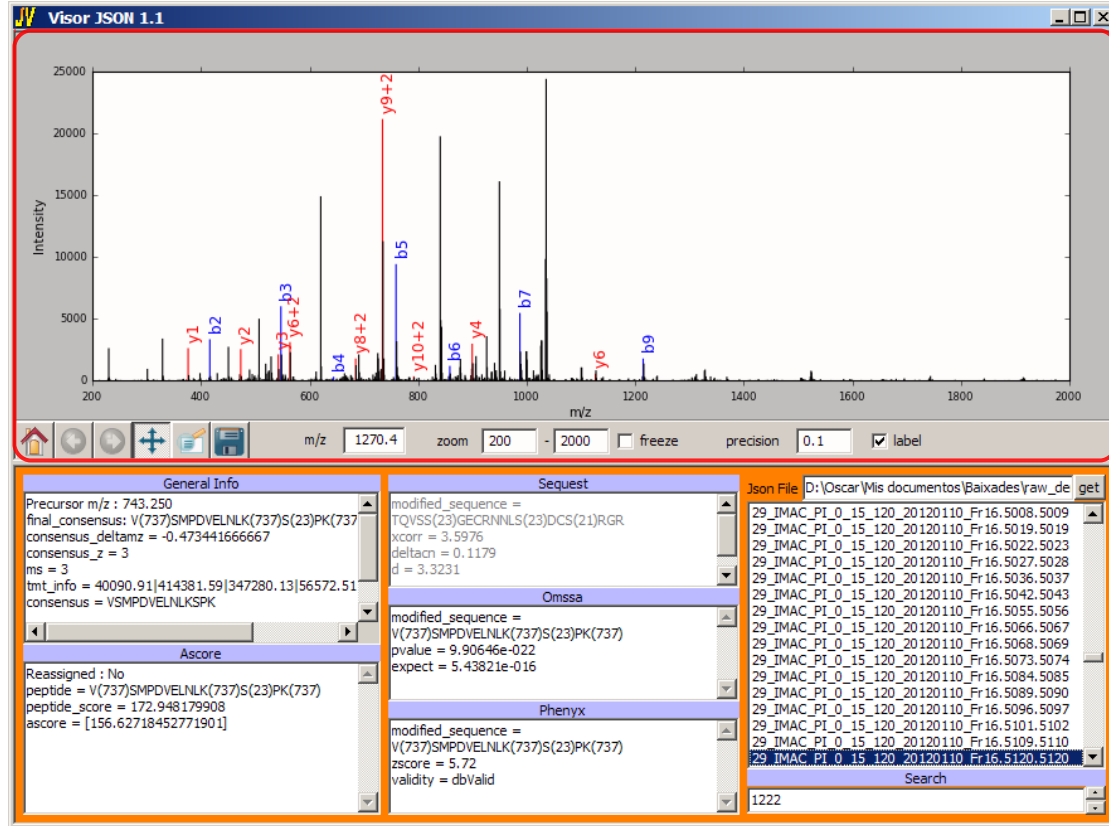
# LYMPHOS WORKFLOW – DRY LAB





# LYMPHOS WORKFLOW – DRY LAB

Mass



## Main Packages Used

- **matplotlib**

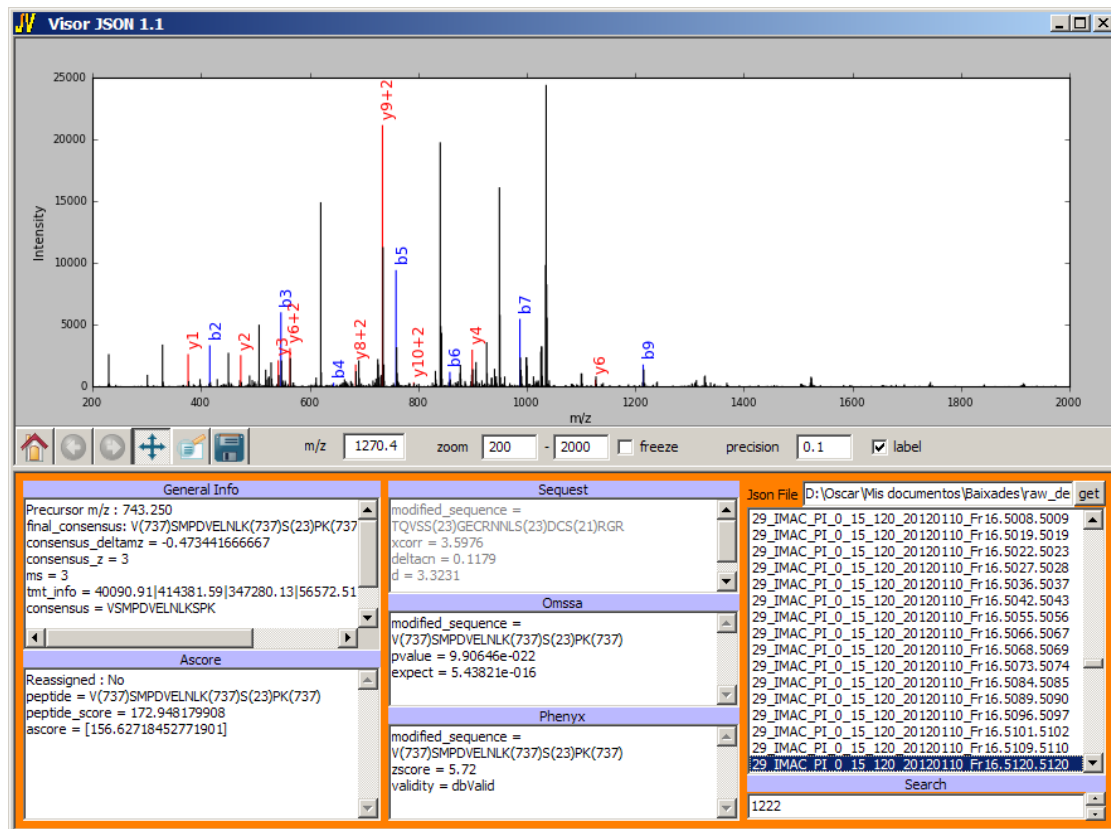
- ✓ 2D plotting library with many styles.
- ✓ A full object-oriented interface, and also a simpler procedural one (the matplotlib.pyplot module).
- ✓ Support for many GUI toolkits, with on screen interactive manipulation tools.
- ✗ Can be difficult to manipulate some plot details.

<https://matplotlib.org>



# LYMPHOS WORKFLOW – DRY LAB

Mass



## Main Packages Used

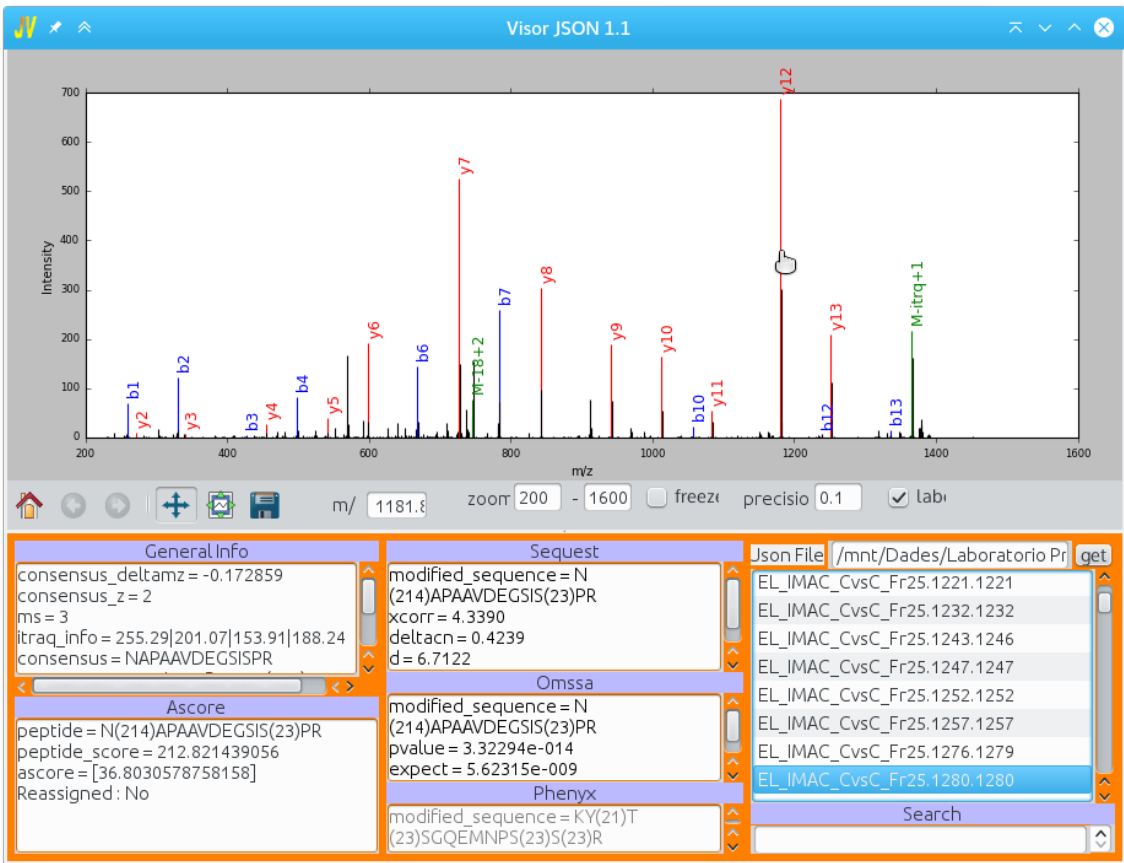
- **matplotlib**
- **NumPy**
  - ✓ Support for large, memory efficient, multi-dimensional arrays (ndarrays) and matrices.
  - ✓ Fast efficient math functions and operators for those arrays (vectorization).
  - ✗ The type of objects stored in NumPy arrays is constrained and homogeneous.

<https://www.numpy.org>



# LYMPHOS WORKFLOW – DRY LAB

Mass



## Main Packages Used

- matplotlib
- NumPy
- wxPython
- py2exe



# LYMPHOS WORKFLOW – DRY LAB

Mass

## Generated Data



- Experimental **conditions** and procedures metadata.
- The **spectra** data and metadata obtained from the mass spectrometers.
- Qualitative and **identification** data and metadata from the multiple search engines used to link the spectra with possible peptides (phosphorylated or not) from human proteins.
- **Semi-quantitative** data about the abundance of each possible identified peptide in the different assayed conditions.

### Additional complications:

- Missing values in quantitative data.
- Identification scores for multiple search engines.
- Phosphorylation Ascores, and phosphorylation reassignments.
- A lot of relationships and inter-linked data.

## What to do with this Data?

- **Process and evaluate quantitative data.**
- **Store** it in a dynamical and accessible way.
- **Link** together qualitative peptide identification data, processed semi-quantitative data, and proteins and experimental information.
- Make all the information easily **accessible** to researchers in our lab and worldwide, so they can study it to obtain biomedical knowledge.

Do **statistics**



Use a **database**



Make a **web application**





# LYMPHOS – VERSION 1

The screenshot shows the LymphOS web application interface. At the top, there are navigation tabs: Start, Experimental, Search, and Contribute. Below the tabs, the main content area displays information about the database, including a description of the phosphoproteome and details about the database's origin and design. On the right side of the interface, there is a sidebar with a list of phosphorylation sites, each with a corresponding mass spectrum plot. At the bottom right, there is an 'Admin' button.

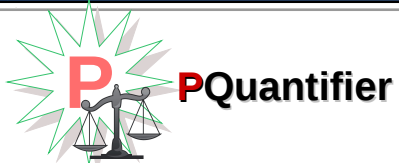
Overlaid on the screenshot is a MySQL Database schema diagram. The diagram shows the following tables and their relationships:

- spectra**: spectrum id, mass array, intensity array. It has a 1-to-1 relationship with the **data** table.
- conditions**: condition id, name, description. It has a 1-to-8 relationship with the **data** table.
- data**: data id, phos-file name, raw file name, scan. It has a 1-to-∞ relationship with the **phosphosites** table.
- phosphosites**: phosphosite id, protein id, phosphorylation position inside peptide, phosphorylation position inside protein. It has a 1-to-1 relationship with the **data** table.

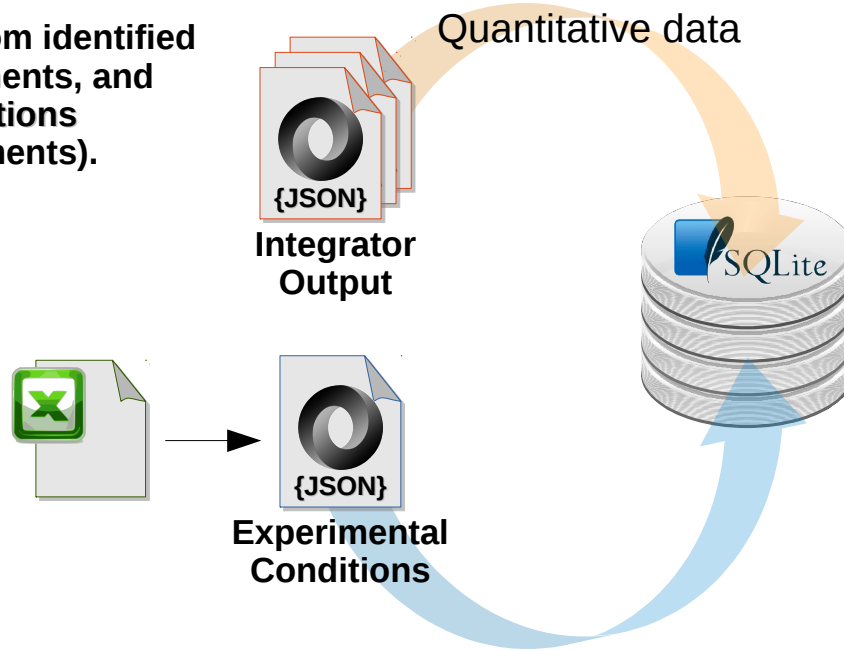
## Problems

- Constrained database **scheme**: only for limited qualitative data, and very little metadata.
- A MySQL **configuration** that does not ensures referential integrity and limits scalability.
- Too many **languages** used for the web application backend.
- Outdated language versions, libraries and **technologies**.
- Outdated web **design**.
- Poor **maintainability** and **scalability**, of both the code and the data.

# LymPHOS<sup>2</sup> – PQUANTIFIER



- Gets *minimal* amount of data from identified peptides of quantitative experiments, and their related experimental conditions (experiments and supra-experiments).



## Main Packages Used

- **SQLAlchemy**
    - ✓ 2 levels of abstraction: a “low level” SQL toolkit (the **Core**) and a high level Object Relational Mapper (**ORM**).
    - ✓ **Mature, fast and efficient**, specially when using the Core.
    - ✓ Highly flexible, configurable, modular, and extensible.
    - ✓ Powerful **query** system.
    - ✗ Sometimes too expressive and **complex**.
- <https://www.sqlalchemy.org/>



## PQuantifier

- Gets *minimal* amount of data from identified peptides of quantitative experiments, and their related experimental conditions (experiments and supra-experiments).
- Assigns missing values to minimum preset values, or low values calculated from each experiment's corresponding data.
- For each identified spectrum, averages quantitative intensities of duplicated reporter ions, and calculates the ratios for the different time points relative to the control.

$$\log_2 \left( \frac{\text{Intensity at time X}}{\text{Intensity at time 0}} \right)$$

### Main Packages Used

- **SQLAlchemy**

- NumPy

- **uncertainties**

- ✓ Provides data types and functions to handle values with uncertainties (ex.:  $7.23 \pm 0.15$ ).

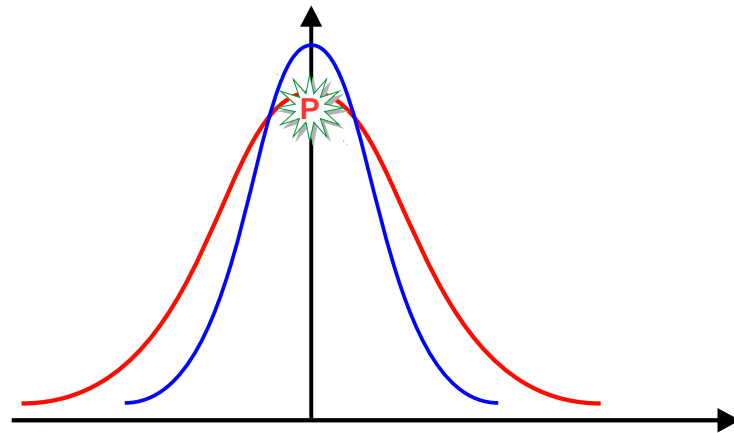
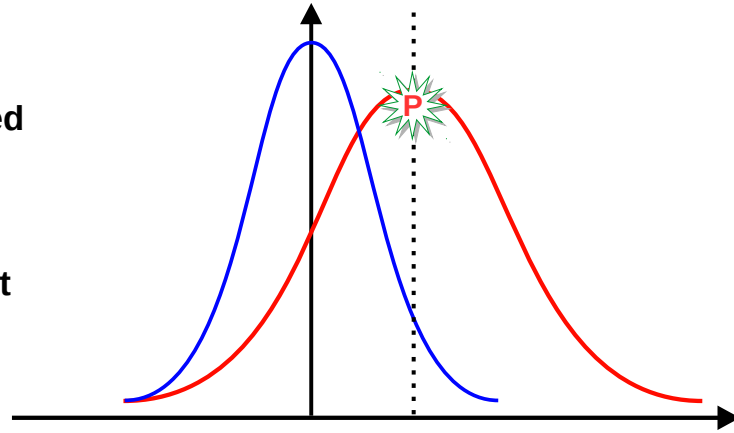
- ✓ Transparently allows **error propagation** in values.

<https://pythonhosted.org/uncertainties>





## PQuantifier

- Gets *minimal* amount of data from identified peptides of quantitative experiments, and their related experimental conditions (experiments and supra-experiments).
- Assigns missing values to minimum preset values, or low values calculated from each experiment's corresponding data.
- For each identified spectrum, averages quantitative intensities of duplicated reporter ions, and calculates the ratios for the different time points relative to the control.
- For each experiment, normalizes these ratios relative to the *distribution of non-phosphorylated peptides*.



### Main Packages Used

- **SQLAlchemy**
- NumPy 
- **uncertainties**
- SciPy 
  - ✓ *Calculations* package for mathematics, science, and engineering.
  - ✓ Includes modules for statistics, optimization, integration, linear algebra, Fourier transforms, .....
  - ✓ A **great amount** of numerical functions.
  - ✓ **Efficient**, based on NumPy.

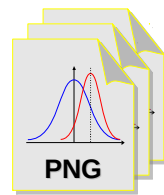
<https://scipy.org>





## PQuantifier

- Gets *minimal* amount of data from identified peptides of quantitative experiments, and their related experimental conditions (experiments and supra-experiments).
- Assigns missing values to minimum preset values, or low values calculated from each experiment's corresponding data.
- For each identified spectrum, averages quantitative intensities of duplicated reporter ions, and calculates the ratios for the different time points relative to the control.
- For each experiment, normalizes these ratios relative to the *distribution of non-phosphorylated peptides*.
- Calculates the peptide average ratios along the activation experiments of the same supra-experiment (*data aggregation*).
- Applies a Student's *t*-test to determine which peptides show significant changes in relative quantity according to average ratios.



PQuantifier  
Plots



PQuantifier  
Output

### Main Packages Used

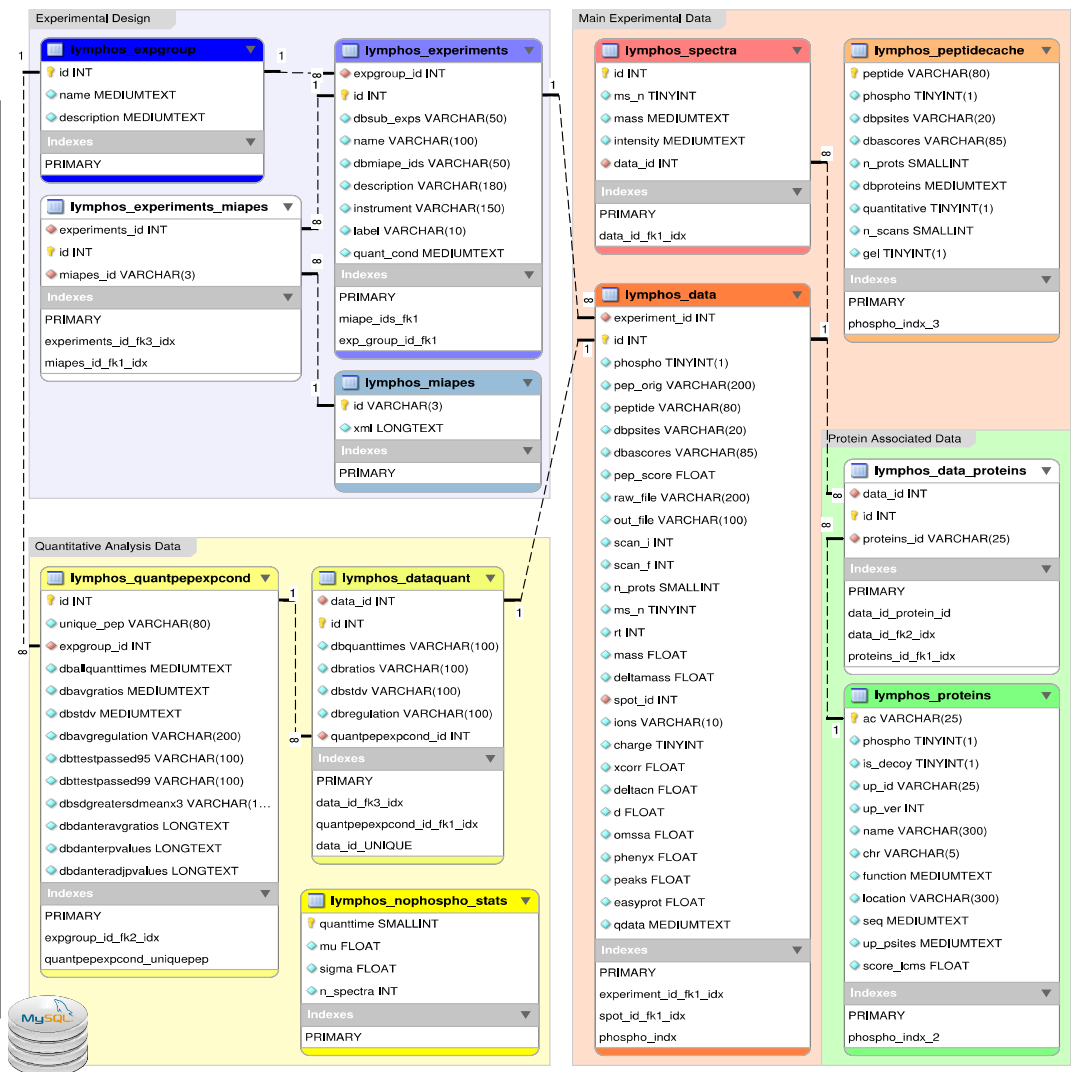
- **SQLAlchemy**
- NumPy
- **uncertainties**
- SciPy
- **matplotlib**
- json

# LymPHOS2 – NEW DATABASE DESIGN



## Information in Tables

- Red and Orange:** Experimental data obtained by mass spectrometry and resulting from proteomics search engines.
- Green:** Protein data associated with the experimental data.
- Yellow:** Quantitative processed data related to the experimental data, obtained after statistical analysis.
- Blue:** Experimental conditions data (experiments, experimental groups and experimental procedures).



But changing the *storage engine* for tables:

MyISAM → **InnoDB**

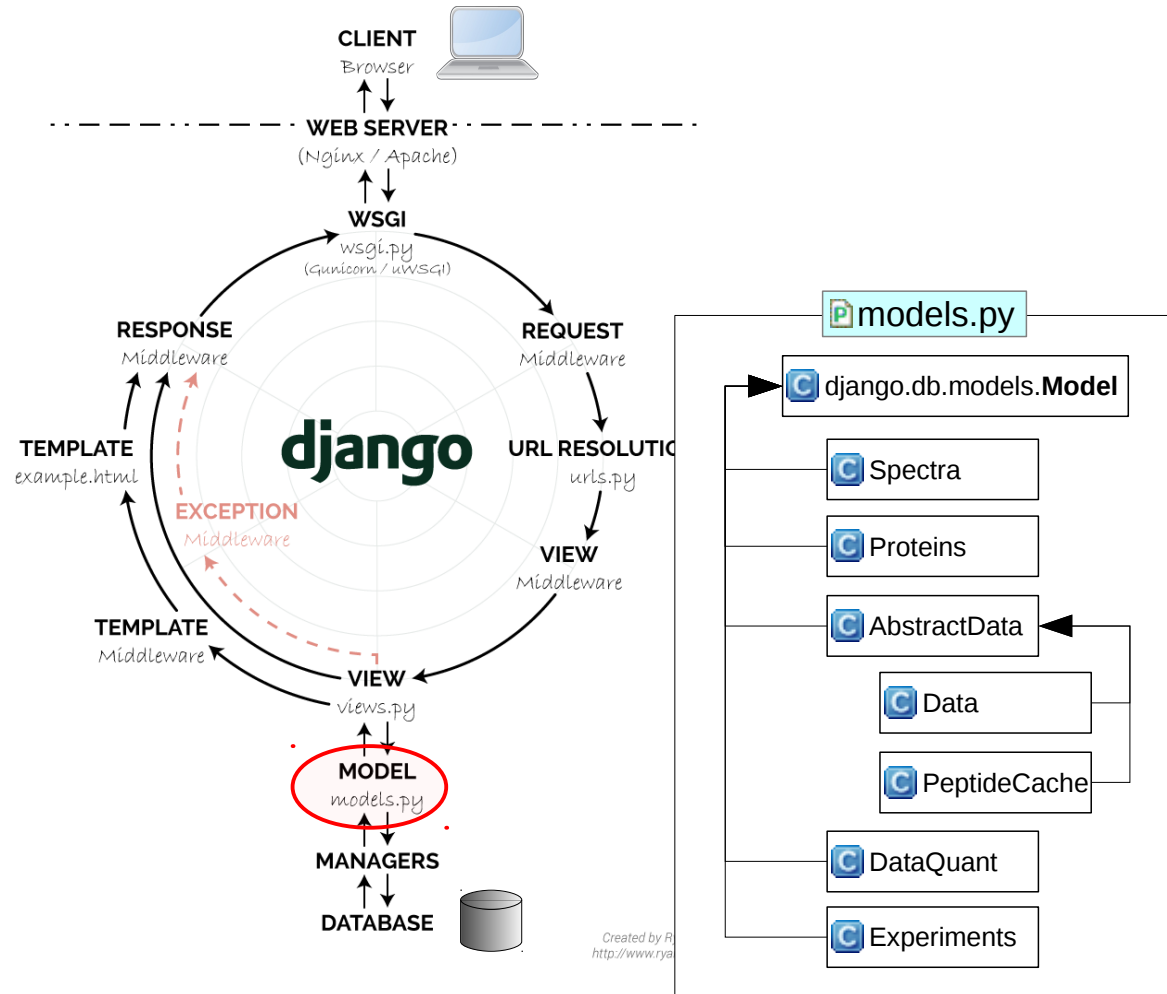
- Referential Integrity (Foreign Key constraints).
- Transactions (*commit* and *rollback*).
- Row-level locking instead of table-level locking (higher *concurrency*).
- More *resistant* to table corruption.
- Maximum performance when processing high volume of data (*scalable*).



## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python **Model** of the database, the Python View code that describes which data is presented, and django Template tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

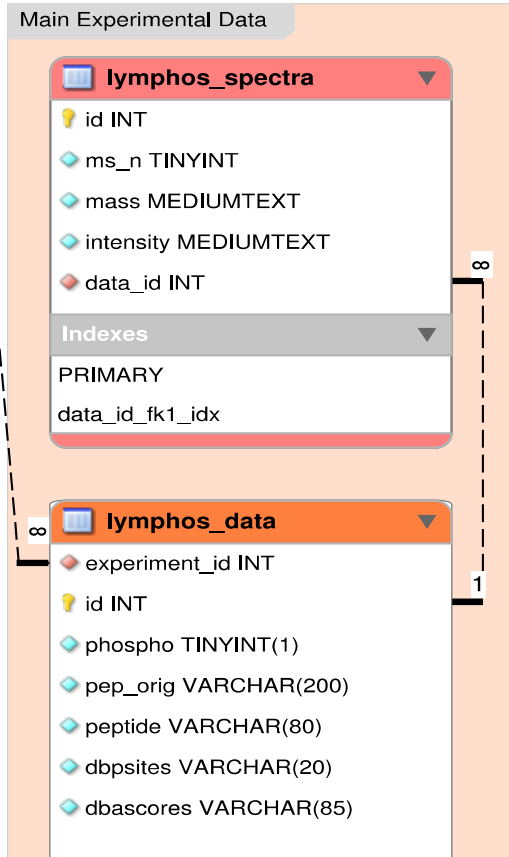
<https://www.djangoproject.com>





# LymPHOS2 – PYTHON AND THE DATABASE

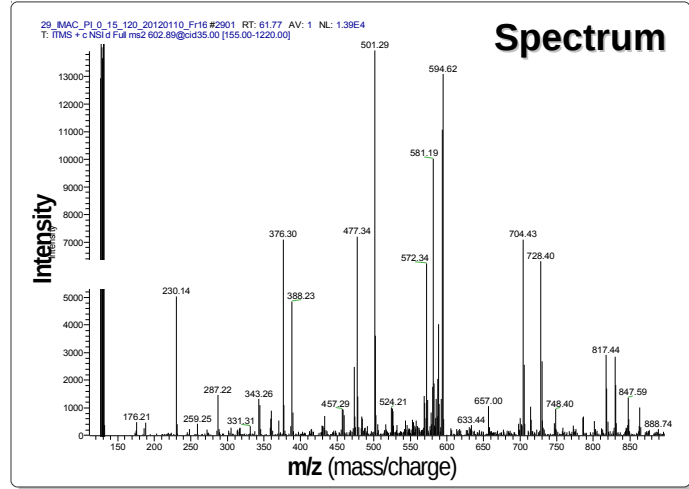
django



models.py

```
# Django import:
from django.db import models

# Models definitions:
class Spectra(models.Model):
    ms_n = models.PositiveSmallIntegerField(default=2)
    mass = models.TextField(default='')
    intensity = models.TextField(default='')
    data = models.ForeignKey(Data)
```



# LymPHOS2 – PYTHON AND THE DATABASE

## django

Main Experimental Data

**lymphos\_spectra**

- id INT
- ms\_n TINYINT
- mass MEDIUMTEXT
- intensity MEDIUMTEXT
- data\_id INT

Indexes

PRIMARY

data\_id\_fk1\_idx

**lymphos\_data**

- experiment\_id INT
- id INT
- phospho TINYINT(1)
- pep\_orig VARCHAR(200)
- peptide VARCHAR(80)
- dbpsites VARCHAR(20)
- dbascores VARCHAR(85)

Diagram showing a one-to-many relationship between lymphos\_data (1) and lymphos\_spectra (8).

models.py

```
# Django import:
from django.db import models

# Models definitions:
class Spectra(models.Model):
    ms_n = models.PositiveSmallIntegerField(default=2)
    mass = models.TextField(default='')
    intensity = models.TextField(default='')
    data = models.ForeignKey(Data)
```

### Examples

```
# Create a new Spectra object:
spectrum = Spectra(ms_n=2, mass='200.723|605.051|...',
                    intensity='23.456|1929.385|...')

# Associate it with an existing Data object:
spectrum.data = existing_data_obj

# Save it to database:
spectrum.save()

# Query the database for a spectrum:
spectrum201 = Spectra.objects.get(id=201)

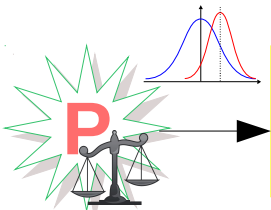
# Get all spectra matching a condition:
ms2_spectra = Spectra.objects.filter(ms_n=2)
```



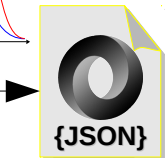
# LymPHOS2 - DATA IMPORT



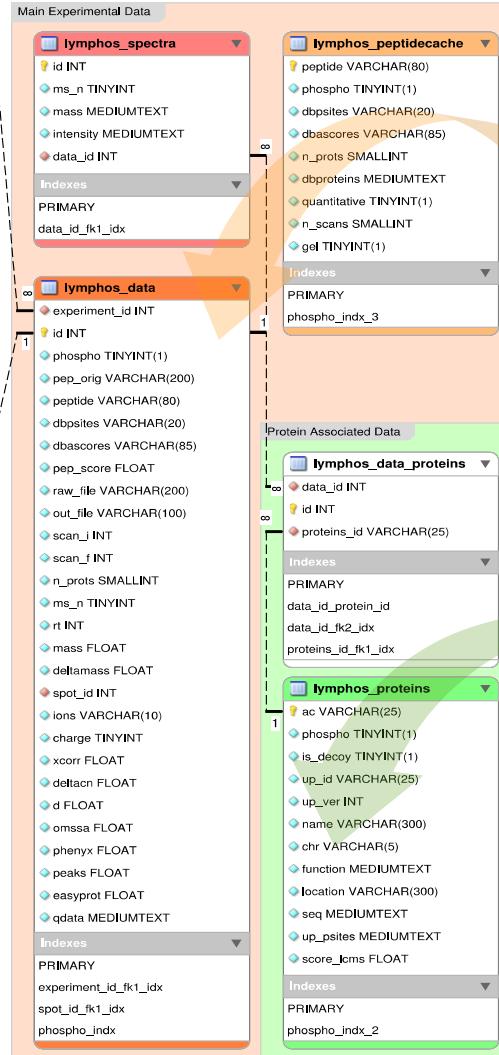
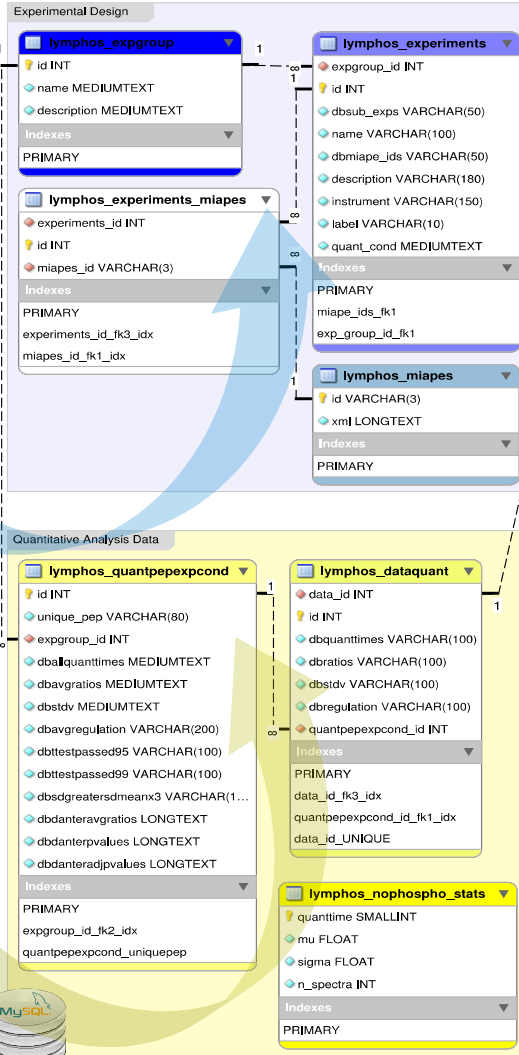
Experimental  
Conditions



PQuantifier



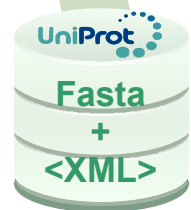
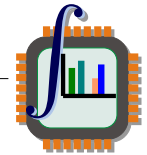
PQuantifier  
Output



filters.py



Integrator  
Output





After all data importation, the LymPHOS2 database nowadays (2017) contains:

- **131.908** *mass spectra*.
- **15.566** *phosphorylation sites*.
- **8.273** *unique phospho-peptides*.
- **4.937** *proteins*.
- Aside from the new **quantitative** data for **1.975** of the identified *phospho-peptides*.



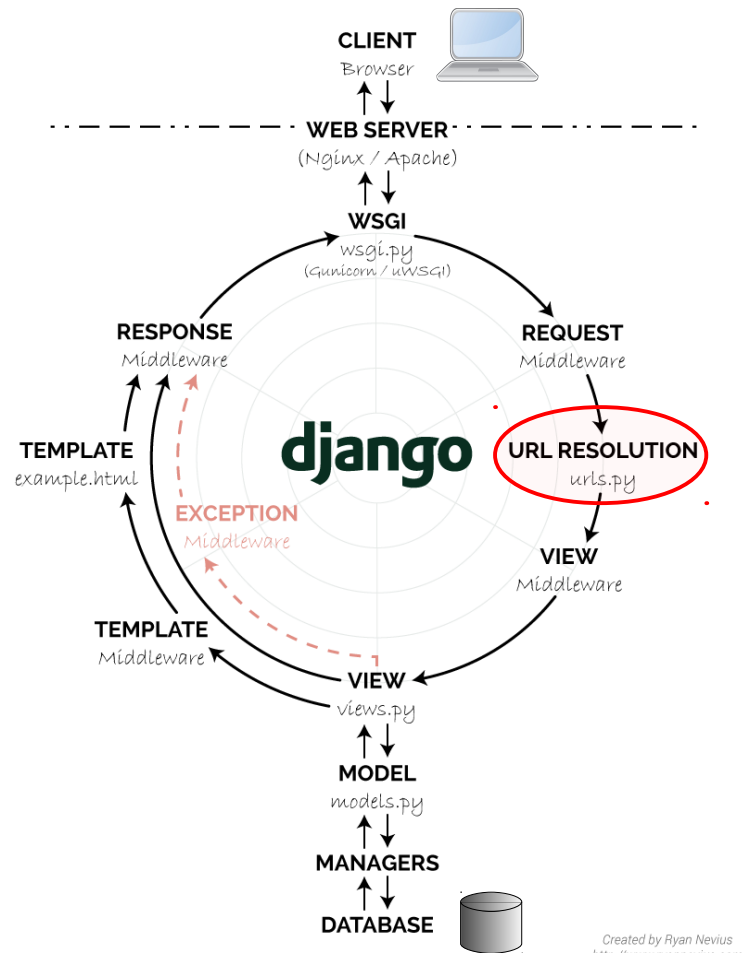


# LymPHOS2 – NEW WEB APPLICATION

## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python Model of the database, the Python **View** code that describes which data is presented, and django **Template** tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

<https://www.djangoproject.com>



Created by Ryan Nevis  
<http://www.ryannevis.com>





# LymPHOS2 – NEW WEB APPLICATION

## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python Model of the database, the Python **View** code that describes which data is presented, and django **Template** tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

<https://www.djangoproject.com>

## urls.py

```
# Django import:
from django.conf.urls import patterns
# Import the LymPHOS2 views:
import views

# Define URL patterns -> view callables:
urlpatterns = patterns('',
    # Home Page (/):
    (r'^$',
     views.FormsHtmlPage.as_view(template_name='home_section.html',
     {'extra_context': {'section_title': 'Home Page',
     'sidebar': {'template': 'side_bar.html'},
     'class_home': r'current'}}
    ),
    # Main Search Page (/search/):
    (r'^search/$',
     views.FormsHtmlPage.as_view(template_name='search_section.html',
     {'extra_context': {'section_title': 'Search LymPHOS DataBase',
     'css_sheet': 'default_nosidebar.css',
     'class_search': r'current'}}
    ),
    # Peptide Search Results Page (/peptide_search_results/):
    (r'^peptide_search_results/$',
     views.PeptideResultPage.as_view()),
    # Peptide Quantitative Details Page (/quant_view/LSLtDPLVAER-4):
    (r'^quant_view/(\w*)-(\d*)$',
     views.QuantViewPage.as_view()),
    # Spectrum Image (/spectrum_image/23097.png):
    (r'^spectrum_image/.*\.png$',
     views.Spectra2File.as_view(file_type='PNG')),
    ... )
```

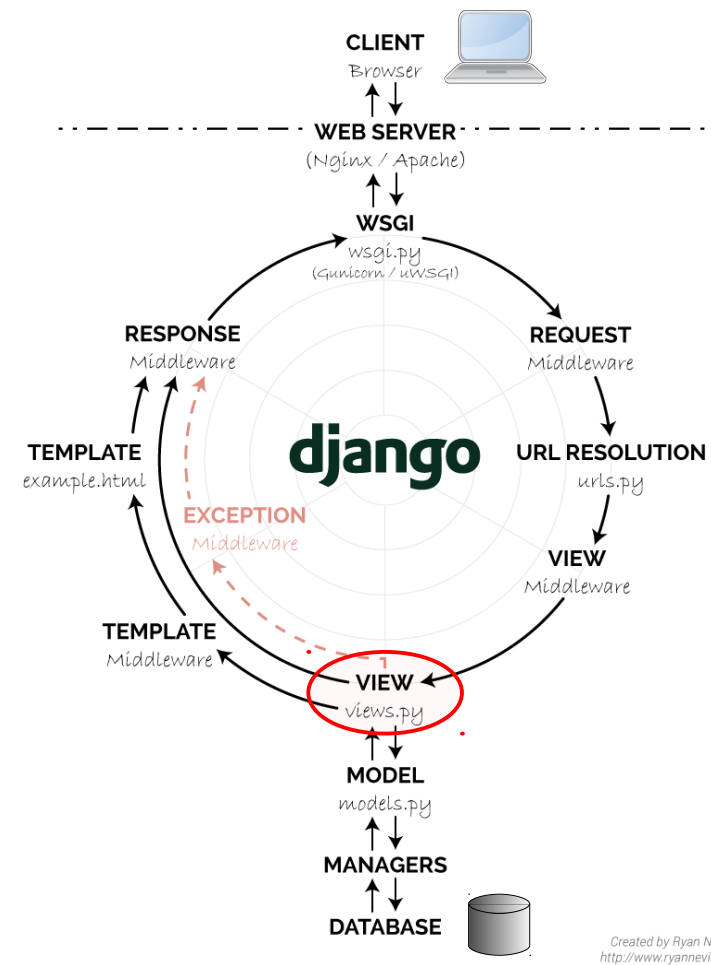


# LymPHOS2 – NEW WEB APPLICATION

## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python Model of the database, the Python **View** code that describes which data is presented, and django **Template** tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

<https://www.djangoproject.com>



Created by Ryan Nevis  
<http://www.ryannevis.com>

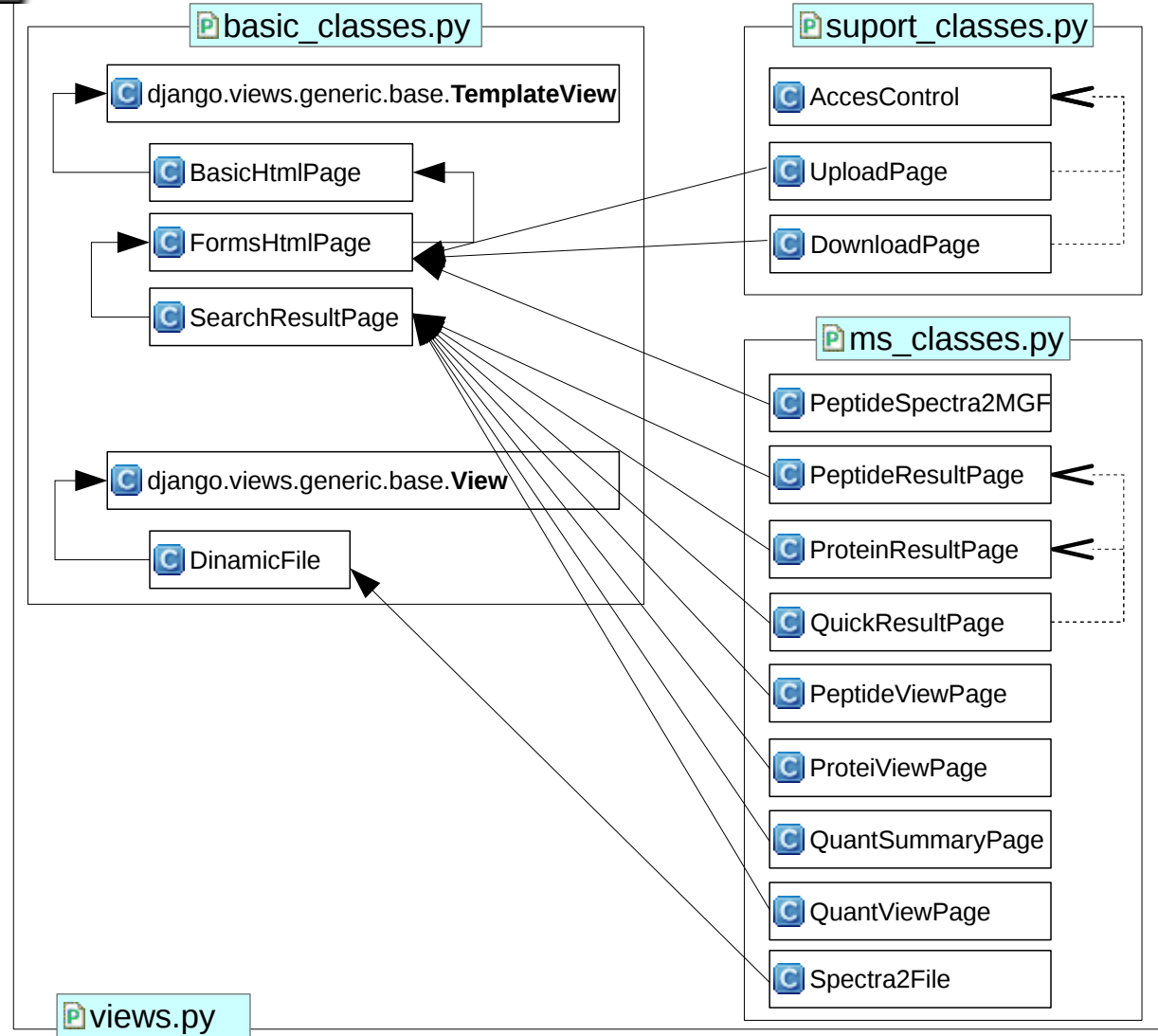


# LymPHOS<sup>2</sup> - NEW WEB APPLICATION

## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python Model of the database, the Python **View** code that describes which data is presented, and django **Template** tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

<https://www.djangoproject.com>



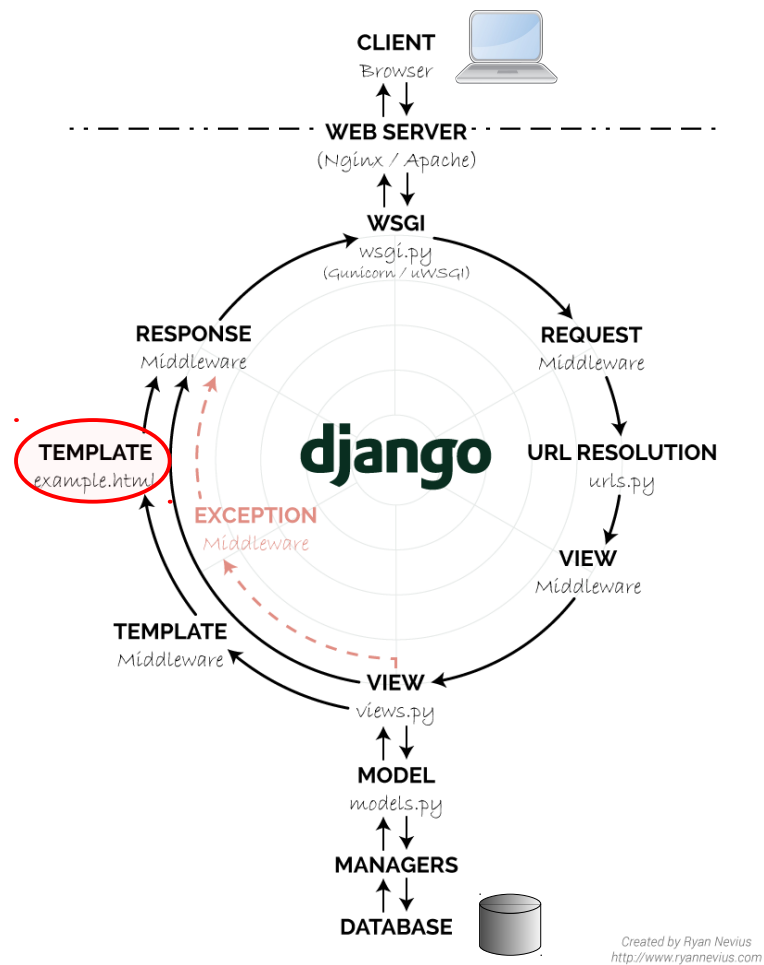


# LymPHOS2 – NEW WEB APPLICATION

## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python Model of the database, the Python **View** code that describes which data is presented, and django **Template** tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

<https://www.djangoproject.com>



Created by Ryan Nevius  
<http://www.ryannevius.com>



# LymPHOS<sup>2</sup> - NEW WEB APPLICATION

## django

- ✓ Web framework for perfectionists with deadlines (**fast** development).
- ✓ Full featured (batteries included).
- ✓ Extensible and scalable.
- ✓ **Model-Template-View** architecture: clean separation between the Python Model of the database, the Python **View** code that describes which data is presented, and django **Template** tags embedded in HTML + CSS, images, ... that defines how the data is presented to the final client/user.

<https://www.djangoproject.com>

peptide\_result.html

```
{% for pep in peptides.object_list %}
<tr align="center">
<td width="24%">
  <a href="/peptide_view/{{pep.peptide}}">
    <span class="sec">{{pep.htmlpeptide|safe}}</span>
  </a>
</td>
<td width="4%">{{pep.n_scans}}</td>
<td width="4%">
  {% if pep.quantitative %}
    <a href="/quant_summary/{{pep.peptide}}">
      
    </a>
  {% else %}
    <span title="No statistical quantitative data"></span>
  {% endif %}
</td>
<td width="4%">{{pep.n_prots}}</td>
<td width="62%">
  {% for protroot, isofnums in pep.grpprots.items %}
    {% if not protroot.2 %}
      <a href="/protein_view/{{protroot.0}}"
        title="{{protroot.1}}">
        {{protroot.0}}
      </a>
    {% else %}
      <a href="/protein_view/{{protroot.0}}-{{protroot.2}}"
        title="{{protroot.1}} - Isoform {{protroot.2}}">
        {{protroot.0}}-{{protroot.2}}
      </a>
    {% endif %}
    {% if isofnums|length > 0 %}
      <nobr>
      (
      <small>
      {% for isofnum in isofnums %}
        <a href="/protein_view/{{protroot.0}}-{{isofnum}}"
          title="{{protroot.1}} - Isoform {{isofnum}}">
          {{isofnum}}
        </a>
        {% if not forloop.last %},{% else %}</small>{% endif %}
      {% endfor %}
      </nobr>
    {% endif %}
    {% if not forloop.last %};{% endif %}
  {% endfor %}
</td>
</tr>
{% endfor %}
```



# LymPHOS2 – NEW WEB APPLICATION

L LymPHOS
 
Laboratori de Proteòmica CS
PyData Barcelona 2017
+

←
i
www.lymphos.org/search/

Cerca
-
100%
+

(cc) 2010-2017 Laboratori de Proteòmica CSIC/UAB

## LymPHOS2

HOME SEARCH EXPERIMENTAL UPLOAD CONTRIBUTE DOWNLOAD ABOUT HELP

### Phospho-peptide Search

Phospho-peptide Sequence Search:

Search Phospho-peptides

All Phospho-peptide Set from LymPHOS:

Browse All Phospho-peptides

You can search into the database on a phospho-peptide driven way: write a sequence in the | **text field** | up here, and then press the | **Search Phospho-peptides** | button. You must use the aminoacid *1-letter symbols*, and the maximum allowed chain is *25 characters* long. You can also use the *"\*"* wildcard for 'one or more' unknown aminoacids, and/or the *"?"* wildcard for 'only one' unknown aminoacid. The non-phosphorylated aminoacid symbols must be entered using *upper case letters*. You must *only use lower case letters* when you want to specify a *phosphorylated residue*: *y, s, t* (for phosphorylated forms of Tyrosine, Serine and Threonine).

You can also browse the whole peptide set, without entering any sequence, by pressing the | **Browse All Phospho-peptides** | button.

### Protein Search

Text Protein Search:

Search Phosphorylated Proteins

All Phosphorylated Protein Set from LymPHOS:

Browse All Phosphorylated Proteins

You can search into the database for a phosphorylated protein or sub-set of phosphorylated proteins. Write your search text in the | **text field** | up this sub-section, and then press the | **Search phosphorylated Proteins** | button. For the text-search you can use the UniProt/TrEMBL Accession Numbers (ex.: 'P16070'; 'Q709C8'), the UniProt/TrEMBL ID (ex.: 'MEFV\_HUMAN'; 'KPCB\_HUMAN') or some part of the UniProt/TrEMBL Description Line (ex.: 'Protein kinase C beta type'; 'Cyclin-K'; 'Retinoblastoma-associated protein'). The maximum allowed text size is *35 characters* long. You can also use the *"\*"* wildcard for 'one or more' unknown letters, and/or the *"?"* wildcard for 'only one' unknown letter. Some useful examples are: 'P16?70', 'P16\*', '\*kinase\*'. Remember that all phosphorylated proteins are from **human origin**.

You can also browse the whole protein set, without entering any text, simply pressing the | **Browse All phosphorylated Proteins** | button.

LymPHOS Web Application is designed using the [Django Framework](#) and the [Python Programming Language](#).

LymPHOS DataBase and Web-App by Laboratori de Proteòmica CSIC/UAB are licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).

# LymPHOS<sup>2</sup> - NEW WEB APPLICATION

L LymPHOS x

Laboratori de Proteòmica CSIC x
PyData Barcelona 2017 x
+

← | i | [www.lymphos.org/peptide\\_view/AAARLSLtDPLVAER](http://www.lymphos.org/peptide_view/AAARLSLtDPLVAER)
↻
☆
📁
↓

🔍 Cerca

-
100%
+
🔊
☰

## LymPHOS<sup>2</sup>

(cc) 2010-2017 Laboratori de Proteòmica CSIC/UAB

HOME SEARCH EXPERIMENTAL UPLOAD CONTRIBUTE DOWNLOAD ABOUT HELP

**Search Results : Peptide View**

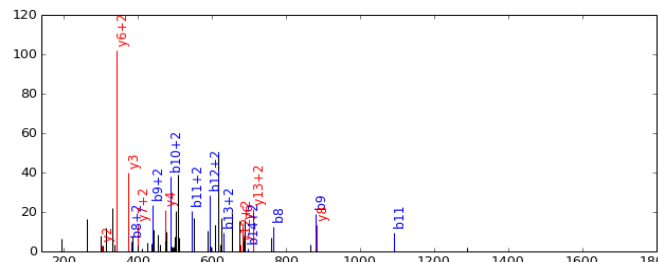
Peptide [AAARLSLtDPLVAER](#) [Download all spectra in MGF format](#)

**Proteins containing this peptide:** [P13224](#)  
[P13224-2](#)

**Spectra 11 - 15 of 62 spectra found for this peptide:** [<< previous](#) Page 3 of 13 [next >>](#)

Spectrum ID: 26296  
Original Peptide: AAARLSLT(23)DPLVAER  
Peptide: [AAARLSLtDPLVAER](#)

| Mass(amu)  | Charge  | DeltaMass(amu) | MSn          | Scan          | Rt (s)                                     | Instrument        |
|--|---------|----------------|--------------|---------------|--|-------------------|
| 1564.90  | 3       | -0.11          | MS3          | 4323 - 4323   | 0.00                                       | LTQ XL - Orbitrap |
| Sequest XCorr  | D-value | OMSSA Score    | Phenyx Score | Protein Count | Raw file                                   |                   |
| 3.10   | 4.29    | 1.06734e-08    | 0.00         | 2             | 32_IMAC_CC_20121210_extract121219_Fr03.raw |                   |
| MS Experiment  |         |                |              |               |  | Q-Ascore 1        |
| Qualitative analysis (Resting T-cells) (Exp. ID: 32) |         |                |              |               |  | 34.30             |



[Download spectrum 26296 in MGF format](#)

# LymPHOS<sup>2</sup> – NEW WEB APPLICATION



LymPHOS<sup>2</sup> x Laboratori de Proteòmica CSIC x PyData Barcelona 2017 x +

www.lymphos.org/quant\_summary/AAARLSLTDPLVAER

Cerca

(cc) 2010-2017 Laboratori de Proteòmica CSIC/UAB

## LymPHOS<sup>2</sup>

HOME SEARCH EXPERIMENTAL UPLOAD CONTRIBUTE DOWNLOAD ABOUT HELP

### Results : Quantitative Summary

Peptide [AAARLSLTDPLVAER](#)

*p*-sites: 1

**Condition:** Activation PMA-ionomycine. 15min and 2h. TMT labeling. (Cond. ID: 4)

**Number of experiments:** 2  
**Number of spectra:** 3

**State:**

- At 15 min.: ↑ (↓: 0, ✖: 0, ↑: 3)
- At 120 min.: ↑ (↓: 0, ✖: 0, ↑: 3)

| min.                   | 15                              | 120                |
|------------------------|---------------------------------|--------------------|
| $\log_2(\text{ratio})$ | PQuantifier: $1.24 \pm 0.17$ ** | $0.58 \pm 0.10$ ** |
|                        | DanteR: $1.38$ +                | $0.48$             |

[Detailed Quantitative data](#)

\*\* Significant change according to a *t*-student test with a 99% significance; for PQuantifier results.  
\* Significant change according to a *t*-student test with a 95% significance; for PQuantifier results.  
+ Significant change according to an ANOVA test with an adjusted *p*-value lower than 0.05; for DanteR results.

## CREDITS

**Data analysis, bioinformatics and informatics:**  
*Joaquín Abián and Óscar Gallardo.*

**Mass Spectrometry, experimental design and implementation:** *Montserrat Carrascal, Nguyen Tien Dung and Oriol Vidal-Cortés.*

**Sample preparation:** *Montserrat Carrascal, Nguyen Tien Dung, Oriol Vidal-Cortés and Vanessa Casas.*

**Past collaborators:** *David Ovelleiro and Marina Gay.*

**Direction:** *Joaquin Abian.*



Joaquín Abián



Óscar Gallardo



Montserrat Carrascal



Nguyen Tien Dung



Oriol Vidal-Cortés



Vanessa Casas



David Ovelleiro



Marina Gay



Albert Casanovas



Ester Sánchez

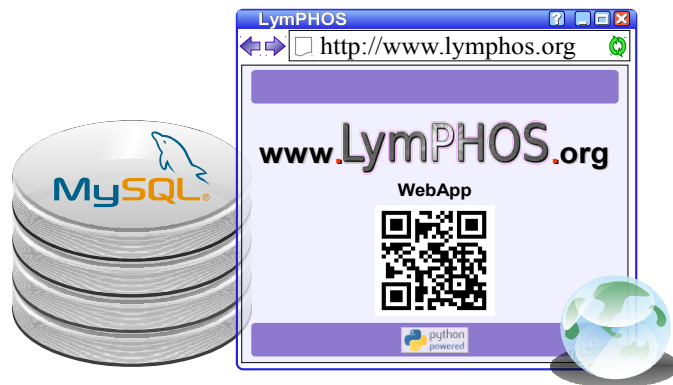


Roberto Pinto

Nguyen, TD., Vidal-Cortés, O., Gallardo, Ó., Abian, J., Carrascal, M., LymphOS 2.0: an update of a phosphosite database of primary human T cells. Database 2015  
doi: 10.1093/database/bav115



**Thank you for  
your attention.**



*Any questions?*

# LymPHOS<sup>2</sup>

**Organizing and Sharing Biological Data of the  
Human Lymphocyte Proteome using Python**

 <https://bitbucket.org/lp-csic-uab/>

 **PyData** Barcelona, 19-21th April 2017