# Instructions for the proteomaps workflow - for internal use

## 1 Overview of the proteomaps workflow

The fastest and easiest way to generate proteomaps is to use the website

http://www.protecs.uni-greifswald.de/bionic-vis/index.php

Instructions (in particular, regarding the choice of protein identifiers in the input file) can be found on the website. However, the website works only for standard cases (organisms for which proteomaps have been established already), and does not provide all the output files that we need to include new maps into our main website http://www.proteomaps.net/. In all other cases, a number of python and matlab scripts ("proteomaps workflow") need to be run. This text gives some basic information about how to prepare data files for the workflow.

Currently, generating proteomaps involves the following (partially manual) steps:

1. Only for new organisms: prepare organism files (Wolf, with input data provided by user - see section 3 and 4)
2. Prepare processed files for paver (Wolf, with input data from user - see section 2)
3. Use the paver software to make proteomaps (Dan, using the processed files)
4. If desired, put proteomaps on website (Wolf, using the paver output files)

Files used in the workflow are stored in different places:

1. General method description: http://www.proteomaps.net/methods.html
2. Selection of data files: http://www.proteomaps.net/download.html
3. Github project proteomaps-workflow with public data files: https://github.com/liebermeister/proteomaps-workflow

## 2 How to prepare data files for the proteomaps workflow

In this section, we assume that proteomaps for an organism have already been established, and that we want to generate a proteomap with a new data set for this organism. Proteomaps are generated by the paver software, and input files for paver can be generated by a python script that preprocesses the original proteome data. This script expects the original data to be given in a fixed table format. The input file is a table in tab-separated .csv format. It contains two columns, one containing the protein identifiers (see below), and one containing the abundance data (numbers). For clarity, the file can also contain a header row with extra information, and a row with the column headers. These two rows are marked, respectively, by "!!" and "!". Here is an example (for *E. coli* proteins):

| !!SBtab TableType='Proteomaps' | Organism='eco' |
|---|---|
| !ProteinIdentifier | !Abundance |
| b0878 | 7 |
| b0948 | 12 |
| b3933 | 12 |
| .. | .. |

The file attributes in the first row (such as "Organism" in the example) contain additional information. Recommended possible attributes are "Organism", "Condition", "Tissue", "ValueType", "Unit", and "Title". The leading rows starting with "!" are optional and are currently not evaluated by the python script; thus, it is also possible to simply write

| b0878 | 7 |
|---|---|
| b0948 | 12 |
| b3933 | 12 |
| .. | .. |

In any case, it is the order of columns that counts. The protein identifiers (in the first column) must be chosen according to the instructions given in "How to add a new organism to the proteomaps workflow". The abundance data (second column) must represent protein molecule count numbers. Do not use data that correspond to mass-weighted count numbers (mass weighting will be done anyway in the workflow). Do *not* use logarithmic values.

Several data sets can be provided together, but must be given as separate files. For each file, the following information needs to be given:

1. The organism (three-letter) shortname
2. A short name of the experiment or paper corresponding to the data set, to be used within file names (e.g., `Valgepea_2013_48`)
3. A human-readable name of the experiment or paper corresponding to the data set (e.g. "Escherichia coli ($\mu$=0.48/h) - Valgepea et al. (2013)")

# 3 How to add protein annotations

In this section, we assume that proteomaps for an organism have been established, but that some proteins have wrong or missing annotations, If you notice that proteins in proteomaps are wrongly placed in the functional hierarchy or not mapped at all, you can change this by adding new protein annotations. To do this, you need to find the proteins to be annotated (or reannotated, which does not make a difference here), their protein identifiers, protein names, and KO numbers, as well as the desired pathway annotations, and put all this information into a table file. The file format for your additional annotations follows the format of the file genomic_data/KO_gene_hierarchy/KO_gene_hierarchy_changes.csv in the github project proteomaps-workflow. The columns are as follows:

Organism shortname
Protein identifier
Protein name
KO number
Pathway name
Comment or provenance information

Example (human hemoglobin Hba2, assigned to the pathway "Hemoglobin"):

| hsa | P69905 | Hba2 | K13822 | Hemoglobin | http://ghr.nlm.nih.gov/gene/HBA2 |
|---|---|---|---|---|---|

The first three entries are mandatory. The "KO number entry" can be left blank. Finding the right pathway name is a bit involved. You may start by checking the pathway given on KEGG's website for that gene.

Since our pathway definitions differs from the original KEGG pathway definitions, please make sure you adhere to our (not KEGG's) naming scheme. To see our list of pathways, you can browse the hierarchy tree file (at `http://www.proteomaps.net/download.html`, link " Hierarchy tree (levels 1-3)").

When your file is ready, please send it to Wolf. If you think that a pathway should be added to the hierarchy or that the hierarchy should be restructured, please contact Wolf.

A practical way to proceed is to focus on not-mapped, highly expressed proteins. To do so, have a look at the preprocessed version of your proteome data file (which contains, for every protein, the current pathway annotation). If you sort the proteins by abundance, you will see which proteins deserve new annotations.

In some cases, you will notice that a protein has an entry in KEGG, but does not carry an annotation in the proteomaps files. Usually, the reason is that in KEGG (i) no gene name is assigned or (ii) not pathway is assigned to this protein (the entry in KEGG's field "Definition" does not serve as a gene name nor pathway).

# 4 How to add a new organism to the proteomaps workflow

In this section, we assume that for a certain organism, proteomaps have not been established yet. Currently the workflow can only handle the organisms from the following list. For each of them, a specific type of protein identifiers must be used (the examples shown refer to enolase proteins):

| Organism | org | ID type | Example | Example URL | Estimated protein count |
|---|---|---|---|---|---|
| Mycoplasma pneumoniae | mpn | Locus tag | MPN606 | www.ncbi.nlm.nih.gov/gene/?term=MPN606 | 50000 |
| Mycobacterium tuberculosis | mtu | Locus tag | Rv1023 | www.genome.jp/dbget-bin/www_bget?mtu: | 1500000 |
| Escherichia coli | eco | Locus tag | b2779 | www.genome.jp/dbget-bin/www_bget?eco:b2779 | 3000000 |
| Synechococcus elongatus sp. PCC7942 | syf | CyanoBase | Synpcc7942_0639 | genome.microbedb.jp/cyanobase/SYNPCC7942/genes/slr0752 | 3000000 |
| Synechocystis sp. 6803 | syn | CyanoBase | slr0752 | genome.microbedb.jp/cyanobase/Synechocystis/genes/slr0752 | 3000000 |
| Saccharomyces cerevisiae | sce | Locus tag | YHR174W | identifiers.org/sgd/YHR174W | 100000000 |
| Schizosaccharomyces pombe | spo | PomBase | SPBC1815.01 | www.pombase.org/spombe/result/SPBC1815.01 | 300000000 |
| Arabidopsis thaliana | ath | Tair | AT1G74030.1 | www.arabidopsis.org/servlets/TairObject?type=locus&name=AT1G74030.1 | 10000000000 |
| Drosophila melanogaster | dme | FlyBase | CG17654 | www.genome.jp/dbget-bin/www_bget?dme:Dmel_CG17654 | 10000000000 |
| Mus musculus | mmu | NCBI Gene Id | 13806 | www.ncbi.nlm.nih.gov/gene/13806 | 10000000000 |
| Pan troglodytes | hsa | UniProt | P06733 | identifiers.org/uniprot/P06733 | 10000000000 |
| Homo sapiens | hsa | UniProt | P06733 | identifiers.org/uniprot/P06733 | 10000000000 |

The list is given as a file in github project proteomaps-workflow (file genomic_data/KO_gene_hierarchy/organisms.csv) and on the proteomaps website (http://www.proteomaps.net/download.html, link "Organism information") To add a new organism to the list, the necessary information need to be prepared, and a file with protein lengths must be provided.

As an example, assume that you wanted to add *E. coli* as a new organism (which, in fact, is already included). For the example, consider the KEGG page `http://www.genome.jp/dbget-bin/www_bget?eco:b3124` for enolase in *E. coli*.

1. Find out KEGG organism shortname (three letters)). On the KEGG page, it appears in the field `Organism`: (in the example, `eco` for Escherichia coli K-12 MG1655). Make sure that not only the organism, but also the strain matches the one for which you want to generate proteomaps. Choose the organism name (e.g., "Escherichia coli") to be used in proteomaps (which can, but does not have to coincide with the name used in KEGG).

2. Find out the type of protein identifiers used in KEGG for this organism. On the KEGG page, it appears first in the field `Entry`: (in the example, `b2779` for enolase. Make sure that your protein data carry the same type of protein identifiers.

3. Find the protein identifier for enolase (which we use as an example case)

4. Find a reference database that provides URLs for these protein identifiers (see table above). In the case of E. coli, we simply use KEGG itself: `www.genome.jp/dbget-bin/www_bget?eco:`; in other cases, we use an organism-specific database, e.g. `genome.microbedb.jp/cyanobase/` for cyanobacteriae).

5. Find out an (estimated) protein count number per cell; in case of doubt, contact Ron.

6. Prepare a table with protein lengths for these proteins. Note that length information refers to PROTEINS and not to mRNA! Important: unlike other tables (which use protein identifiers as keys), this table must use protein shortnames (e.g., "eno") as keys (in a column called **Protein:Name**). The format is:

| Protein:Name | Protein:Size |
|---|---|
| eno | 350 |
| .. | .. |

   Information about protein lengths can be obtained from uniprot, but due to gene name conversions, there may be some loss of information.

7. Send all this information, as well as a test protein data set (see instruction "How to prepare input data"), to Wolf; he prepares the input files for paver, and Dan makes the first picture.

8. Be aware that heavy reannotating is usually necessary after the first test pictures.

# 5 Contact

wolfram.liebermeister@gmail.com