Using Patent2Net to Obtain Patent Data

In this chapter we look at using open source Patent2Net software to access patent data from the European Patent Office Open Patent Service (OPS)[[1]](#footnote-21). Patent2Net can be used either through a Windows Client or using Python (across platforms) by downloading the programme and code from the Patent2Net Github Repository[[2]](#footnote-23). We will focus on using the Windows Client as this is the simplest method.

One important strength of Patent2Net is that it handles the parsing of patent data that is returned from OPS for you and allows for downloads of summary tables in .csv format as well as sections of patent documents. Because the XML or JSON that is returned from the OPS is quite complex, it is considerably easier to use Patent2Net than to parse the data for yourself in say Python or R.

The OPS service provides access to raw XML files from the EPO bibliographic EPODOC database, the Worldwide Legal Status Database, full-text (EPOQUE) and images (BNS). As the (FAQs)\*[[3]](#footnote-25) explain, these are the same datasources as espacenet but access is programmatic through a RESTful API webservice. Note that at the time of writing access to full text descriptions and claims at OPS is restricted to EP, WO, AT, CH, GB and CA and is therefore less complete than espacenet or data from Patentscope. However, the OPS allows you to access the following data:

* bibliographic
* family
* legal status
* full text (EP, WO, AT, CH, ES, GB, CA)
* facsimile images
* The EPO Register
* CPC classification

It is important to emphasise that the OPS service is not intended for bulk downloads of patent data and the EPO Fair Use Charter[[4]](#footnote-27) sets a weekly limit of 2.5GB per week for free downloads for registered users. Queries per IP address are limited to 10 search related actions per minute and the EPO also requests that data retrieval by robots be scheduled between 19.00 and 07.00 hours GMT or at weekends to limit the pressure on the service.

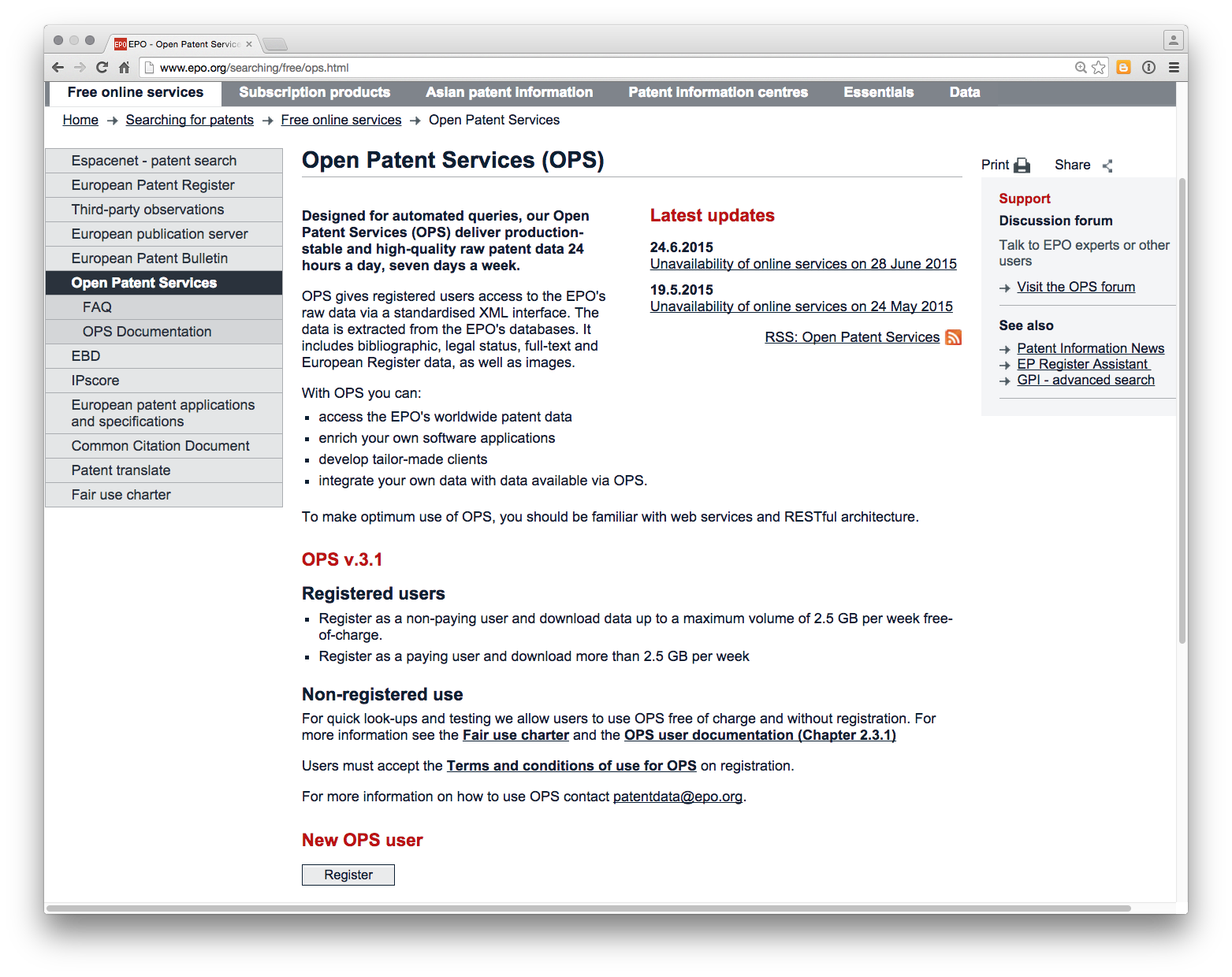
### Downloading Patent2Net

To install the Patent2Net Client in Windows (7 in this case) download the software from either the Patent2Net website[[5]](#footnote-30) (preferred) or the Github repository here[[6]](#footnote-32) for a direct download of the Windows executable. The Patent2Net website provides a zipped executable that is easier to work with than the Github version. For licence information read this file[[7]](#footnote-34).

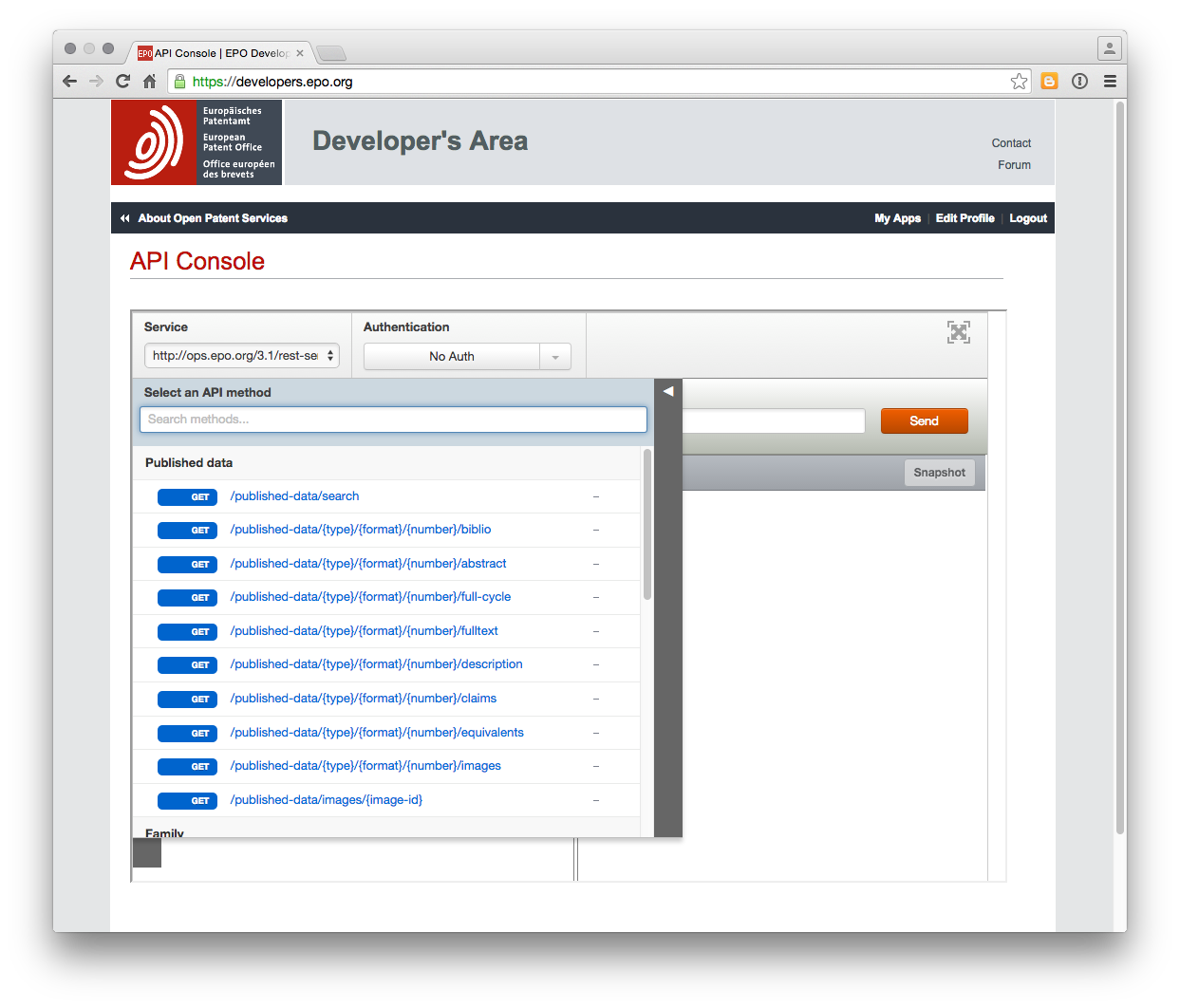
### Installing Patent2Net

We will repeat the instructions that are available in this file[[8]](#footnote-37) but add screenshots to make it as simple as possible.

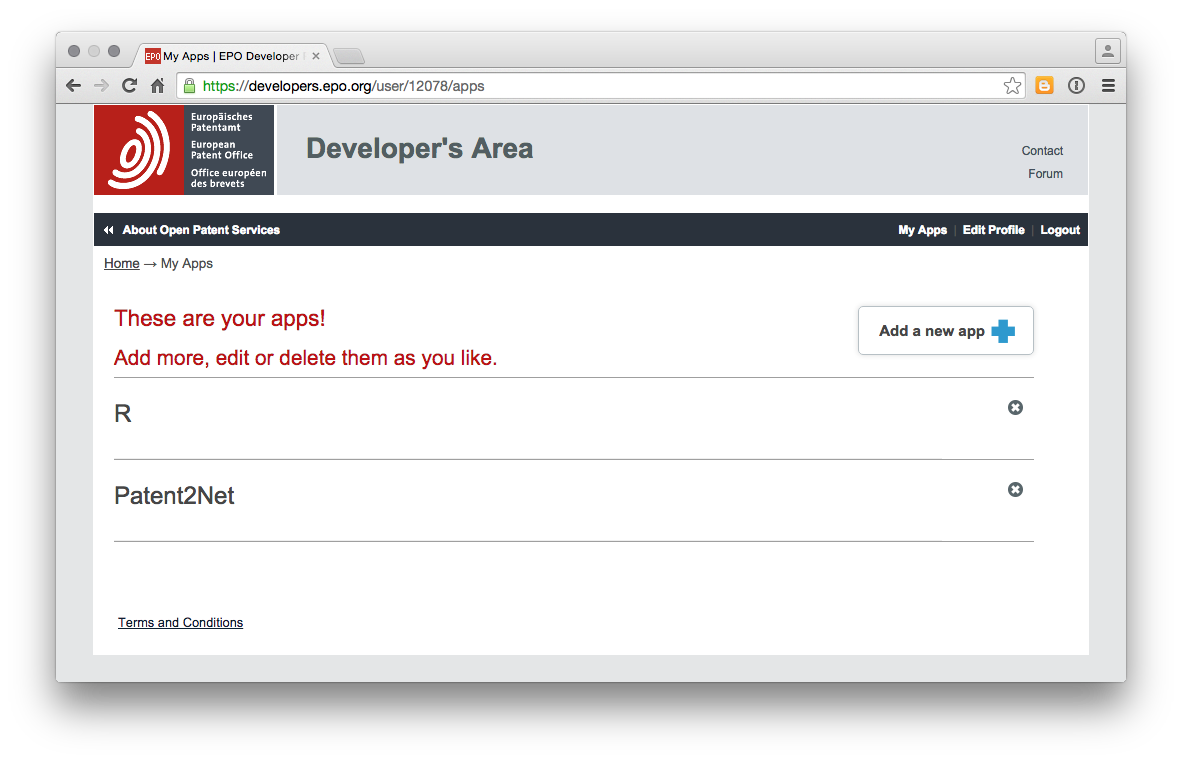
1. Unzip the P2N.zip file where you would like to locate the file (the Desktop is fine).
2. Register with OPS here[[9]](#footnote-39) by selecting New OPS user.



1. Once you have registered you need to create a new App called Patent2Net. First go to My Apps.



Then click on the large Add a new app icon and add an app called Patent2Net



Click on the Patent2Net app and copy down the Consumer Key and Consumer Secret in a text file and separate them by a comma. For example xyz1234,abc4567. Save the text file in the P2N folder as cles-epo.txt (for "keys epo"") and then close the file.

1. Open the requete.cql file and then under request enter `TA="pizza box" and in DataDirectory pizzabox.

Now choose Save As and save the file into the REQUESTS folder (removing the existing Lentville file). Note that if your file is not in the REQUESTS folder then the next step will default to a request in that folder (which can be very confusing). Basically, the search request you actually want to run should be located in the REQUESTS folder.

Then double click on CollectETraite.bat.

The terminal window will open and should begin to download the data from EPO. This can take some time and so it is important to be patient. To see the data downloading head to the DONNEES folder and you will see a new folder with the name pizzabox. Click inside that folder and (depending on your choices in the requet file) you will see an Abstract folder. If you click into this then you should see the number of items progressively increasing.

# Results

When the query has run you should see something like this in your file directory.

INSERT SCREENSHOT

# Strengths and weaknesses

Patent2Net is an important initiative that has involved very significant work. It is probably best used in Python with the Windows interface providing access to those without knowledge to learn Python. This is very welcome. However, as with many open source projects the code is subject to quite regular change (we used three different windows versions in writing this chapter) and so be prepared to spend time trying to make Patent2Net work for you with minimal documentation. That is the price of open source software.

On the plus side it is possible to make Patent2Net work in Windows and the data is retrieved in a variety of formats with summaries in .csv for use in Excel or other tools, html and Json. The html version allow you to view the data in a browser and include maps and there are also network visualisation options using GraphViz.

# Round Up

Patent2Net is an important resource for accessing OPS data in Python and also provides the only Windows interface for OPS access that we know of. The documentation for the Windows version is limited (and the interface may have changed by the time you read this chapter) but this is a common feature of open source software.

On the plus side this is the only functioning windows interface that we know of for accessing OPS patent data and an important free tool for overcoming the existing limitations with data downloads from tools such as espacenet, WIPO Patentscope and The Lens, notably the ability to obtain abstracts, descriptions and claims as well as family and other data. This initiative is therefore to be applauded. Hopefully with time other programmers will contribute to development of this resource to make it more widely available to others.

# DELETE BELOW AFTER REVIEW

Double click on the file CollectETraite.bat which will generate a message in the terminal

1. Open the .exe file in the P2N folder MAYBE LEAVE THE REQUET.cql blank]

Enter a search query for a title and abstract search use TA="pizza box"

enter a directory name where you would like to store the files and if you wish save the query at the bottom of the screen.

INSERT SCREEN SHOT TA="pizza box"

If all goes well the terminal will open in Windows and you will see the following:

!\_config.yml[[10]](#footnote-48)

It may well appear that nothing is happening. However, if you navigate to the pizzabox folder in Donnes and Abstract folder it should start to populate with the abstracts for the data as we can see in the background of the image above.

When the data run is complete the terminal will indicate any problems encountered and end.

If we open the pizzabox folder we will now see the following files and folders.

A nice feature of Patent2Net is that it generates data in multiple file formats

* .csv for work in Excel or other software
* html for displaying the data and maps online
* json for programmatic use

Most p

"requet.cql"

Note that .cql stands for Contextual Query Language [[11]](#footnote-49) as developed at the US Library of Congress Change as described here[[12]](#footnote-50). Change the request to something that you want. In this case we have used a title and abstract search TA="pizza box" because we love pizza boxes, particularly musical ones.

!\_config.yml[[13]](#footnote-52)

The default folder for downloads is entered in the DataDirectory field and is called Maroc3. In this case we have changed it to pizza box. Change this if you want to. Note that upon completion the data will be stored in a folder called DONNES.

Leave the other settings as they are but note that we could change from TRUE to FALSE etc as needed. The new beta interface simplifies this and is recommended [[14]](#footnote-53).

For other cql queries see the OPS Documentation here[[15]](#footnote-54) and try the Developers testing area here[[16]](#footnote-56). The Developers testing area is the best place to try out different types of queries and to assess the results.

!\_config.yml[[17]](#footnote-58)

When testing Patent2Net we recommend using quite a restrictive query to begin with. For example, "pizza box" returns 203 results in the Titles and Abstracts while pizza returns +3000 and will attempt to download them. We managed to download 2,000 abstracts in this way. However, we would suggest using smaller queries until you have a better understanding of the amount of data that is downloaded relative to the weekly limit to avoid being blocked by the OPS system.

When we ran a query on pizza we ended up with a set of folders as follows.

!\_config.yml[[18]](#footnote-59)

The patent biblios folder contained two files with a basic list that was difficult to interpret that looked like this.

!\_config.yml[[19]](#footnote-60)

A Families file was also created. It was a little difficult to interpret this data.

In the PatentContents folder we saw these results [[20]](#footnote-61).

!\_config.yml[[21]](#footnote-62)

The abstracts folder contained the abstracts and some basic identifier details

!\_config.yml[[22]](#footnote-63)

What is interesting here is that relatively quickly we were able to obtain a significant number of abstracts [[23]](#footnote-64) although we were less successful with the biblio information and piecing the data together would take time.

## The Beta Windows User Interface

Following this experiment with the original interface we switch over to the new beta interface. This can be opened by double clicking WindowsApplication1 below

!\_config.yml[[24]](#footnote-66)

We will then see the following interface.

We entered the details of the query and then the directory to save the results. We also specified a queries folder to save the query.

We ran in to problems that the interface was using the original pizza query as we had specified in the requete[[25]](#footnote-67) .cql file rather than our new query. We found that we needed to enter the query into that file rather than our saved file [[26]](#footnote-68)

!\_config.yml[[27]](#footnote-69)

The query then runs in the terminal and displays a range of messages [[28]](#footnote-70). We should then see a folder in DONNES with the following contents

!\_config.yml[[29]](#footnote-71)

The data includes a .csv file containing a summary of results that can be opened in Excel or Open Office as below.

!\_config.yml[[30]](#footnote-72)

We also have access to a Families table [[31]](#footnote-73). Note here that while the delimiter in the first table is a semi-colon, in the Families case it appears to be mixed and is likely to require further work to clean up.

The results folder also includes JSON files containing the data and links to an online javascript Pivot table library [[32]](#footnote-74).

In theory, although not in practice in this case, the interface will generate Gephi network files, although we did not see any files generated with a gexf or gephi extention.

### Round Up

In this article we have introduced the open source Patent2Net interface for retrieving patent data from the European Patent Office Open Patent Service. Patent2Net is particularly valuable because it takes the pain out of parsing the results from the OPS service and is particularly useful for gathering family and related data in a readily usable way.

The beta Windows interface for Patent2Net enjoys the advantage the no programming knowledge is required. However, bear in mind that the data is raw and will need to be cleaned in tools such as Open Refine, Python R or other tools.

1. <http://www.epo.org/searching-for-patents/technical/espacenet/ops.html> [↑](#footnote-ref-21)
2. <https://github.com/Patent2net/Patent2Net> [↑](#footnote-ref-23)
3. <http://www.epo.org/searching-for-patents/technical/espacenet/ops/faq.html#faq-64> [↑](#footnote-ref-25)
4. <https://www.epo.org/searching/free/fair-use.html> [↑](#footnote-ref-27)
5. <http://patent2netv2.vlab4u.info/P2N-V2.zip> [↑](#footnote-ref-30)
6. <https://github.com/Patent2net/Patent2Net/raw/master/ApplicationPatent2net.exe> [↑](#footnote-ref-32)
7. <https://github.com/Patent2net/Patent2Net/blob/master/Licence_CeCILL_V2.1-en.txt> [↑](#footnote-ref-34)
8. <http://patent2net.vlab4u.info/readme.txt> [↑](#footnote-ref-37)
9. <http://www.epo.org/searching/free/ops.html> [↑](#footnote-ref-39)
10. images/p2net\_figures/processing.png [↑](#footnote-ref-48)
11. previously known as Common Query Language [↑](#footnote-ref-49)
12. <http://www.loc.gov/standards/sru/cql/spec.html> [↑](#footnote-ref-50)
13. images/p2net\_figures/query.png [↑](#footnote-ref-52)
14. see below [↑](#footnote-ref-53)
15. <http://documents.epo.org/projects/babylon/eponet.nsf/0/7AF8F1D2B36F3056C1257C04002E0AD6/$File/OPS_v3.1_documentation_version_1.2.14_en.pdf> [↑](#footnote-ref-54)
16. <https://developers.epo.org/>? [↑](#footnote-ref-56)
17. images/p2net\_figures/developerquery.png [↑](#footnote-ref-58)
18. images/p2net\_figures/contents.png [↑](#footnote-ref-59)
19. images/p2net\_figures/result1.png [↑](#footnote-ref-60)
20. note that we retained the Maroc3 name for our pizza results when testing [↑](#footnote-ref-61)
21. images/p2net\_figures/result2.png [↑](#footnote-ref-62)
22. images/p2net\_figures/abstracts.png [↑](#footnote-ref-63)
23. bearing in mind the constraints noted above [↑](#footnote-ref-64)
24. images/p2net\_figures/application.png [↑](#footnote-ref-66)
25. request [↑](#footnote-ref-67)
26. although that may originate from an error on our part [↑](#footnote-ref-68)
27. images/p2net\_figures/query2.png [↑](#footnote-ref-69)
28. including error and problem messages [↑](#footnote-ref-70)
29. images/p2net\_figures/resultssynbio.png [↑](#footnote-ref-71)
30. images/p2net\_figures/resultssynbio1.png [↑](#footnote-ref-72)
31. as Familiessynbio in this case [↑](#footnote-ref-73)
32. see the online links when previewing a file [↑](#footnote-ref-74)