

# Programming in the Life Sciences

*In the Maastricht Science Programme*

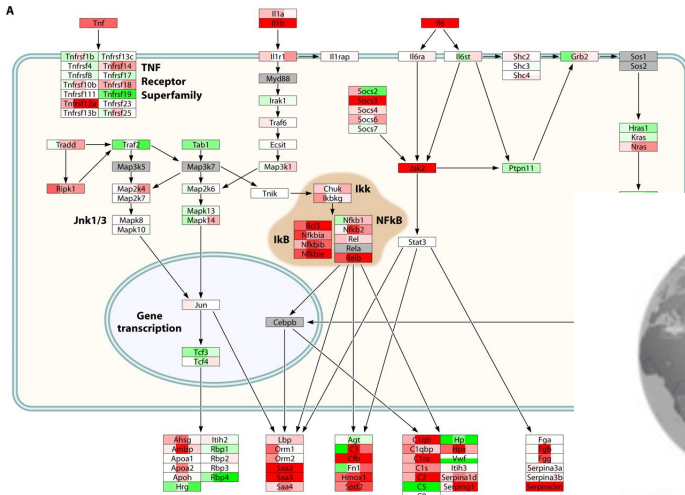
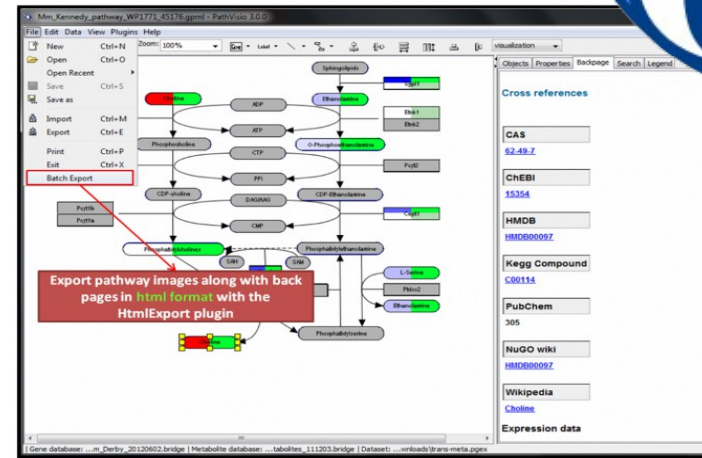
***Open PHACTS Community Workshop***

***London, 26 June 2014***

# Who am I?

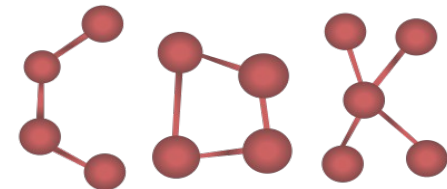
- Teacher at Dept. Bioinformatics – BiGCaT, NUTRIM, FHML, UM.
- <http://chem-bla-ics.blogspot.com/>
- @egonwillighagen
- ORCID:0000-0001-7542-0286
- <http://www.linkedin.com/in/egonw>
- <http://egonw.github.com/>

# BiGCaT? → Systems Biology



WIKIPATHWAYS  
Pathways for the People

# BridgeDb



## Portal:Drug Metabolism

### Welcome to the Drug Metabolism Portal on WikiPathways

### Browse, Draw, Analyze, Download, Publish and Share Drug Metabolism Pathways

This portal aggregates pathways summarizing the action and metabolism of drugs. For new users interested in adding new and editing existing pathways we recommend following the [HELP](#) and [USER](#) guide.

#### All Pathways

edit

#### Mode of Action

- fluoropyrimidine
- nicotine (chromaffin cells and dopaminergic neurons)
- nifedipine

#### Biotransformations

- aripiprazole
- cocaine
- codeine
- colchicine
- dantrolene
- diclofenac
- felbamate
- heroin
- lidocaine
- methapyrilene
- morphine
- moxicylyte
- nicotine
- sulindac (rat and human)
- tamoxifen

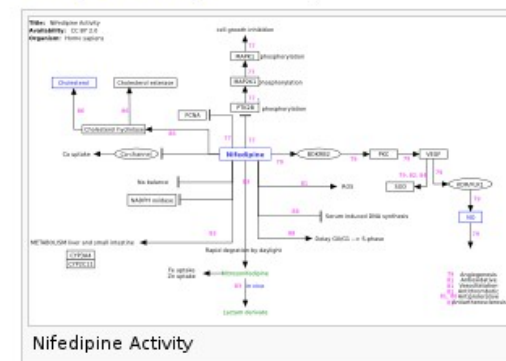
#### Receptor binding

- bradykinin
- dopamine
- histamine
- melatonin
- prostaglandin
- serotonin

#### Featured Pathway

edit

#### Nifedipine Activity (Homo sapiens)



Nifedipine Activity



# Description

*In the life sciences the interactions between chemical entities is of key interest. Not only do these play an important role in the regulation of gene expression, and therefore all cellular processes, they are also one of the primary approaches in drug discovery. Pharmacology is the science studies the action of drugs, and for many common drugs, this is studying the interaction of small **organic molecules and protein targets**.*

*And with the increasing information in the life sciences, **automation** becomes increasingly important. Big data and small data alike, provide challenges to integrate data from different experiments. The **Open PHACTS platform provides web services** to support pharmacological research and in this course you will learn how to use such web services from programming languages, allowing you to **link data** from such knowledge bases to other platforms, such as those for data analysis.*

# Schedule

- **Day 1** theory, template, exercises
- **Day 2** choose a use case, presentation in the afternoon
- **Day 3** work, work
- **Day 4** more work, informal progress report on prototype
- **Day 5** work
- **Day 6** final clean up, end presentation
  - Dec 5, presentations start at 14:00

# What do I expect?

- HTML+JavaScript “programs” (or...)
  - Using the Open PHACTS API (ops.js)
  - Visualization of data, preferable graphically (d3.js)
- If you feel ambitious: towards data analysis
  1. Mashup with “other” data
  2. Solving a biological question

# JavaScript

## ops.js client to the Open PHACTS API

– Ian Dunlop,  
Manchester

<http://github.com/openphacts/ops.js>

## d3.js visualization

<http://d3js.org/>

EBI > Databases > Small Molecules > ChEMBL Database

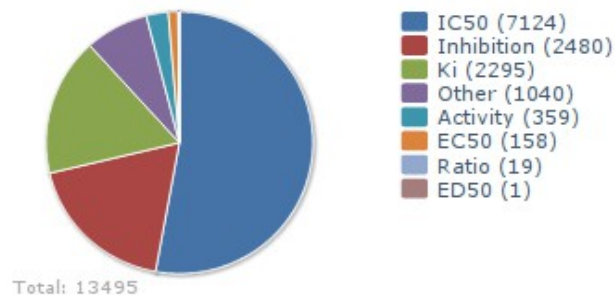
### Target Report Card

#### Target Name and Classification

Target ID	CHEMBL240
Target Type	SINGLE PROTEIN
Preferred Name	HERG
Synonyms	ERG   ERG-1   ERG1   Eag homolog   Eag-related protein 1   Ether-a-go-go related gene potassium channel 1   Ether-a-go-go-related protein 1   H-ERG   KCNH2   Potassium voltage-gated channel subfamily H member 2   Voltage-gated potassium channel subunit Kv11.1   hERG-1   hERG1
Organism	Homo sapiens
Species Group	No
Protein Target Classification	ion channel vkc vkc vlt cationic k kcnh, kv10-12.x (ether-a-go-go)

#### Target Associated Bioactivities

ChEMBL Activity Types for Target CHEMBL240



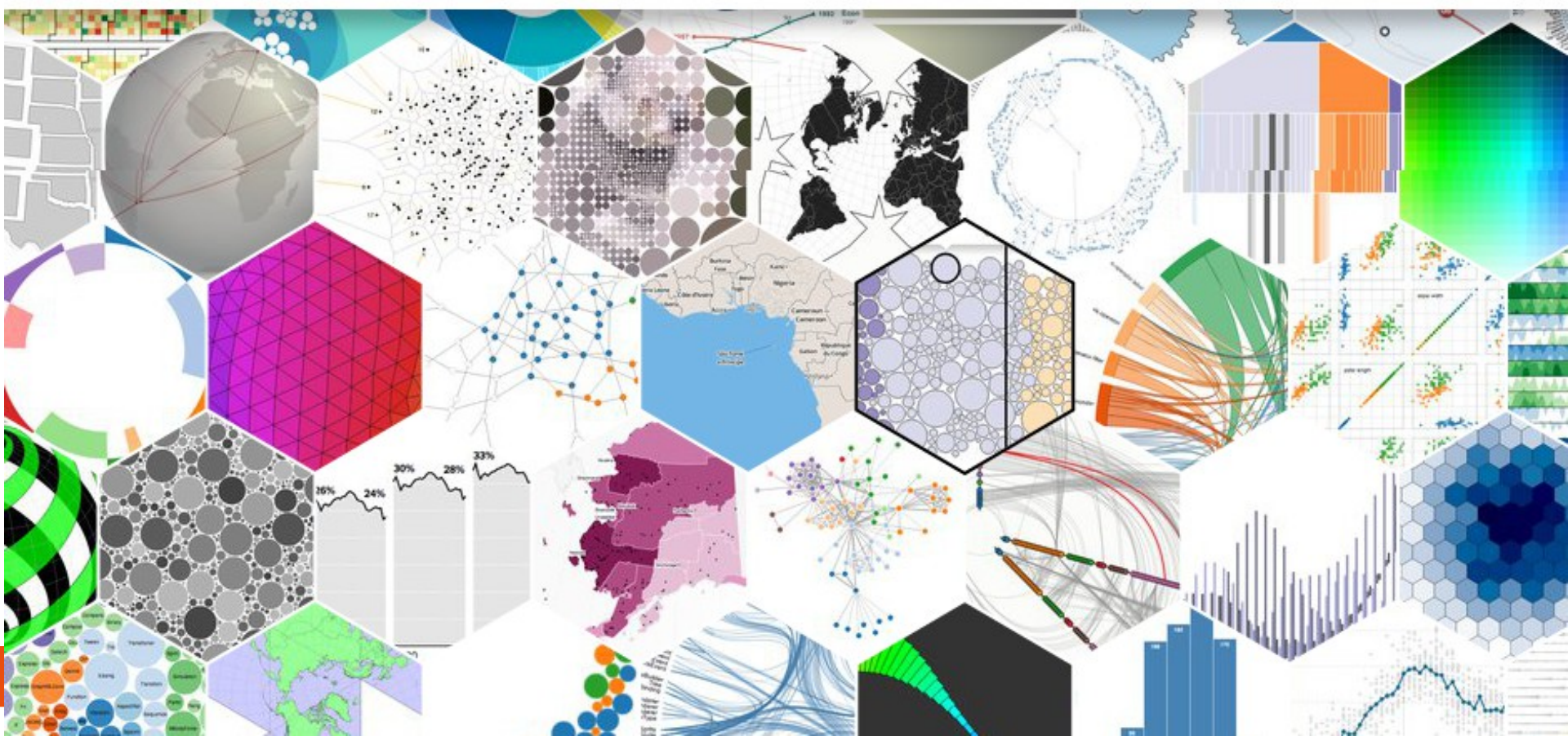
#### Target Associated Assays

ChEMBL Assays for Target CHEMBL240





# Data-Driven Documents



# OPS Linked Data API: pharmacology

[Developer Home](#)[Want help?](#)[Documentation](#)[Get my API keys!](#)[Featured Apps](#)[Workflow](#)

## OpenPHACTS API Active Docs

The response template for each operation colour coded as follows:

- Required elements that always return a **single value**.
- Required elements that return **either a single value or an array**.
- Optional elements that always return a **single value**.
- Optional elements that return **either a single value or an array**.

## Operations

### OpenPHACTS API

Chemical Structure Exact Search

`/structure/exact` GET

InchiKey to URL

`/structure` GET

Inchi to URL

`/structure` GET

Chemical Structure Similarity Search

`/structure/similarity` GET

SMILES to URI

`/structure` GET

# The Open PHACTS API keys

	A	B	C	
1	Student1 MSC	MSC_stud1	<u>egon.willighagen+msc1@gmail.com</u>	f518a30600c4
2		MSC_stud2	<u>egon.willighagen+msc2@gmail.com</u>	e6f05e59a384
3		MSC_stud3		1f12d1501ea2
4		MSC_stud4		95d1792955a8
5		MSC_stud5		010865b2acff
6		MSC_stud6		426700c06731
7		MSC_stud7		85cc58eaf6bc
8		MSC_stud8		e22b49534ed1
9		MSC_stud9		e5573e68bb42
10		MSC_stud10		05440e43b255
11		MSC_stud11		25b78cc409b5
12		MSC_stud12		990a8b38d14b
13		MSC_stud13		695cc4126938
14		MSC_stud14		d33055b83901

# Theory

- **Data Types**

- Variable (type), List, Map, List of Lists, Map of Maps

- **Automation**

- For-loop, operator, function/parameter/libraries, languages

- **Web Services / Clients**

- API, URI, dynamic HTML

- **Serialization formats**

- RDF, JSON

- **Other**

- Open Science, coding standards, notebooks (e.g. blog)

# Top-down programming

- Use web service
  - Call web service
    - Set up the specific call I want to make
    - Make the call
    - Record the returned answer
  - Output HTML
    - Convert the returned answer to a table, figure, etc
    - Dynamically insert this output somewhere in the HTML

# Template: Source Code #1

```
<html>
<head>
  <title>OpenPHACTS Jasmine Spec Runner</title>
  <script src="lib/jquery-1.9.1.min.js"></script>
  <script type="text/javascript" src="lib/purl.js"></script>

  <!-- include source files here... -->
  <script type="text/javascript" src="src/OPS.js"></script>
  <script type="text/javascript" src="src/ConceptWikiSearch.js"></script>

  <!-- setup -->
  <script type="text/javascript">
var prmstr = window.location.search.substr(1);
var prmarr = prmstr.split("&");
var params = {};

for ( var i = 0; i < prmarr.length; i++) {
  var tmparr = prmarr[i].split("=");
  params[tmparr[0]] = tmparr[1];
}
var showDetails = function(dataJSON){
  data = JSON.parse(unescape(dataJSON));
  // document.getElementById("details").innerHTML = JSON.stringify(data);
  document.getElementById("details").innerHTML = data._about;
};
</script>
```



# Template: Source Code #2

```

<body>
  <h3>Output</h3>
  <!-- p>app_id: <span id="appID"></span>, app_key: <span id="appKey"></span -->
  <h3>Search Results</h3>
  <p><div id="table"></div></p>
  <h3>Compound Details</h3>
  <p><div id="details"></div></p>
  <h3>JSON reply</h3>
  <p><div id="json">Nothing yet</div></p>
  <script type="text/javascript">
// document.getElementById("appID").innerHTML = params["app_id"];
// document.getElementById("appKey").innerHTML = params["app_key"];
var searcher = new Openphacts.ConceptWikiSearch("https://beta.openphacts.org", params["app_id"], params["app_key"]);
var callback = function(success, status, response){
  document.getElementById("json").innerHTML = JSON.stringify(response);
  html = "<table>";
  for (var i=0; i<response.length; i++) {
    html += "<tr>";
    html += "<td>";
    dataJSON = JSON.stringify(response[i]);
    // dataJSON.replace(/"/g, "");
    html += "Name: <span onClick=\"showDetails(' + escape(dataJSON) + '\")\">\" + response[i].prefLabel + "</span>";
    html += "</td>";
    html += "</tr>";
  }
  html += "</table>";
  document.getElementById("table").innerHTML = html;
};
searcher.byTag('Aspirin', '5', '4', '07a84994-e464-4bbf-812a-a4b96fa3d197', callback);
</script>
</body>

```

# OK... now what??

## **Ex. #1: The HTML framework**

1. Take the HTML/JavaScript template
2. Remove all JavaScript
3. Open the HTML in FireFox or Chrome
4. Change one of the headers in the HTML

## Ex. #2: The HTML/JS framework

1. Take the HTML/JavaScript template
2. *Keep* the JavaScript
3. Open the HTML in FireFox or Chrome
4. Explain in what linear order your browser visualizes this HTML page

## **Ex. #3: The HTML/JS framework**

1. Take the HTML/JavaScript template
2. What are the variables used?
3. What types have those variables?
4. Which libraries are used?
5. What methods are defined?
6. Validate the HTML against the W3C validation service.

## Ex. #4: the Open PHACTS LDA

1. Which bio-/chemical entities can you get information on?
2. Which service would you use to find the biological pathways alcohol is involved in?
3. What are the data serialization formats you can get your answers returned in?



## Ex. #5: JavaScript steps

1. Take the HTML/JavaScript template
2. *Keep* the JavaScript
3. Search for alcohol instead of aspirin
4. Have the template return 10 instead of 5 search results

## Ex. #6: JSON

1. Take the HTML/JavaScript template
2. *Keep* the JavaScript
3. Open the HTML in your browser
4. Explore the returned JSON
5. What keys are found in the JSON?

## Ex. #7: Another Open PHACTS method

1. Take the HTML/JavaScript template
2. *Keep* the JavaScript
3. Open the HTML in your browser
4. Select a different Open PHACTS LDA service
5. Update the JavaScript to call that service (use the ops.js examples on GitHub)
6. Show the results in a HTML table

You can find these examples here:

<https://github.com/egonw/mscpils/>

# What were the results?

# Difficulties: Asynchronous calls

- Multiple callbacks
    1. Get URI for a compound
    2. Look up interactions with proteins
    3. Get protein information
    4. ...
- So, need to keep track of intermediate information



# API Usage

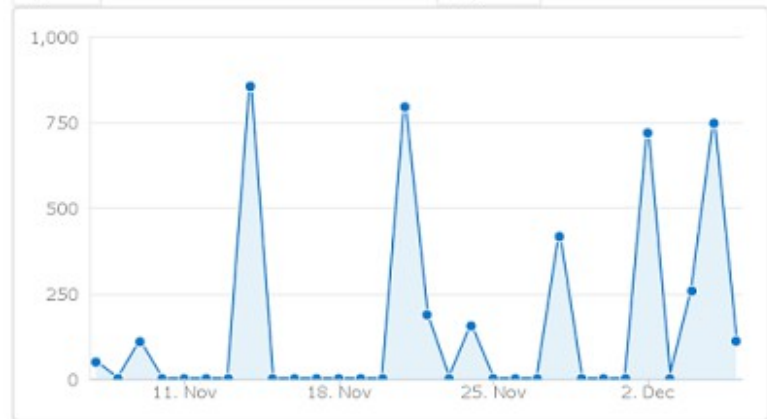
## API Usage 1,428 Hits

Metrics Methods  
 Hits  
 Year Month Week Day  
 7 Nov



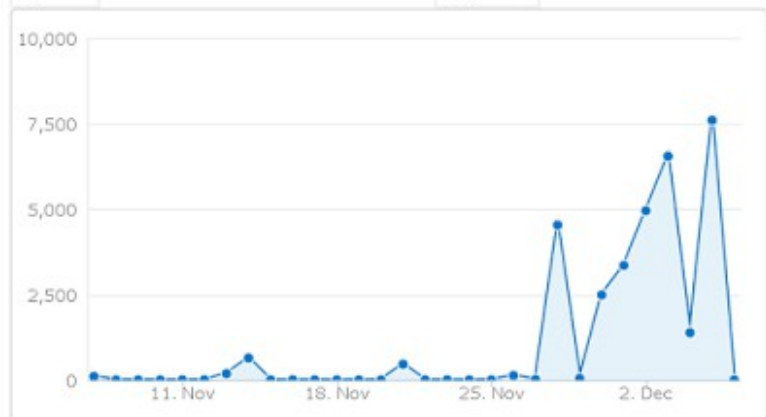
## API Usage 4,387 Hits

Metrics Methods  
 Hits  
 Year Month Week Day  
 7 Nov



## API Usage 32,373 Hits

Metrics Methods  
 Hits  
 Year Month Week Day  
 7 Nov



## Alzheimers Disease

Gene	Link Coding Protein	Coding Protein
Name: <a href="http://identifiers.org/ensembl/ENSG00000100030">http://identifiers.org/ensembl/ENSG00000100030</a>	<a href="http://identifiers.org/uniprot/A8CZ64">http://identifiers.org/uniprot/A8CZ64</a> <a href="http://identifiers.org/uniprot/Q1HBJ4">http://identifiers.org/uniprot/Q1HBJ4</a> <a href="http://identifiers.org/uniprot/P28482">http://identifiers.org/uniprot/P28482</a> <a href="http://identifiers.org/uniprot/B4DHN0">http://identifiers.org/uniprot/B4DHN0</a> <a href="http://identifiers.org/uniprot/MAPK1">http://identifiers.org/uniprot/MAPK1</a>	Mitogen-activated protein kinase 1 (Homo sapiens)
Name: <a href="http://identifiers.org/ensembl/ENSG00000002330">http://identifiers.org/ensembl/ENSG00000002330</a>	<a href="http://identifiers.org/uniprot/F5H1R6">http://identifiers.org/uniprot/F5H1R6</a> <a href="http://identifiers.org/uniprot/Q6FH21">http://identifiers.org/uniprot/Q6FH21</a> <a href="http://identifiers.org/uniprot/Q92934">http://identifiers.org/uniprot/Q92934</a> <a href="http://identifiers.org/uniprot/A8MXU7">http://identifiers.org/uniprot/A8MXU7</a> <a href="http://identifiers.org/uniprot/F5GYS3">http://identifiers.org/uniprot/F5GYS3</a> <a href="http://identifiers.org/uniprot/F5H3B1">http://identifiers.org/uniprot/F5H3B1</a> <a href="http://identifiers.org/uniprot/BAD">http://identifiers.org/uniprot/BAD</a> <a href="http://identifiers.org/uniprot/F5GY71">http://identifiers.org/uniprot/F5GY71</a>	Bcl-XL/Bcl-2-associated death promoter,Bcl-2-binding component 6,Bcl-2-like protein 8
Name: <a href="http://identifiers.org/ensembl/ENSG00000176749">http://identifiers.org/ensembl/ENSG00000176749</a>	<a href="http://identifiers.org/uniprot/Q15078">http://identifiers.org/uniprot/Q15078</a> <a href="http://identifiers.org/uniprot/CDK5R1">http://identifiers.org/uniprot/CDK5R1</a>	Cyclin-dependent kinase 5/CDK5 activator 1

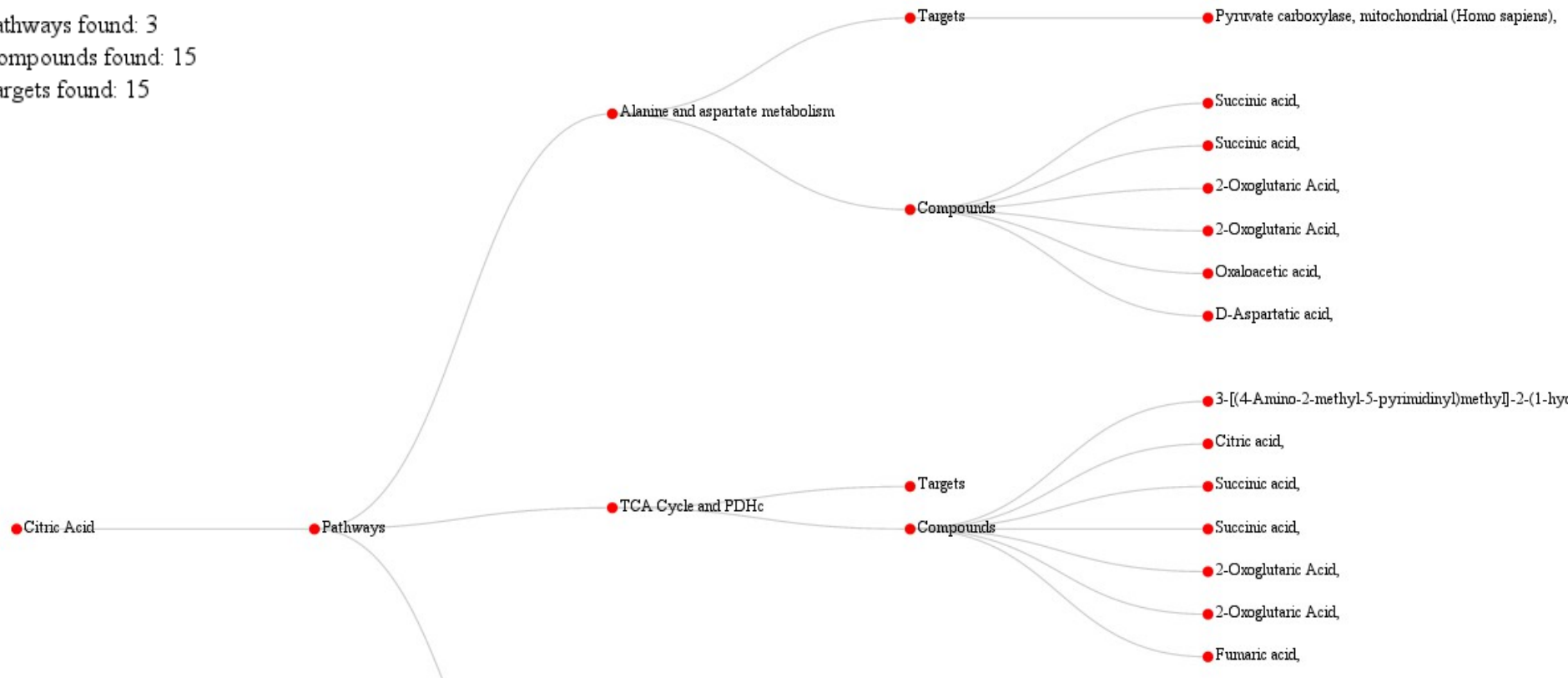
#### Status

33% complete

Pathways found: 3

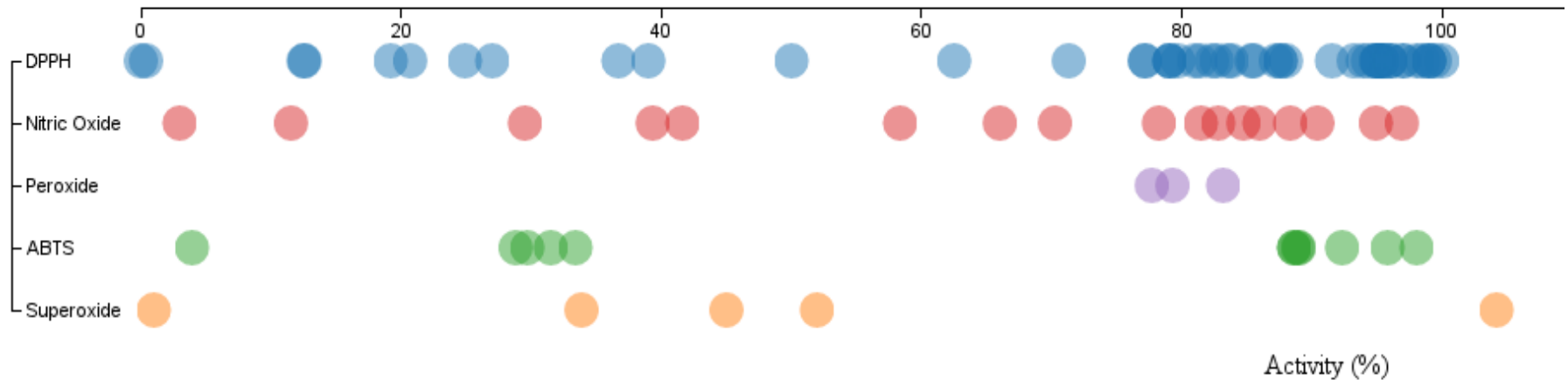
Compounds found: 15

Targets found: 15



# Antioxidant activity against free radicals

Antioxidant:    
 Radical target: ☒ DPPH ☒ Nitric Oxide ☒ Superoxide ☒ ABTS ☒ Peroxide  
 Activity unit: ☒ % ☐ uM ☐ ug/mL  
 Visualisation method: ☒ Graph ☐ Table



Description: DPPH radical scavenging activity at 100 uM

testosterone

Search

Click on desired compound

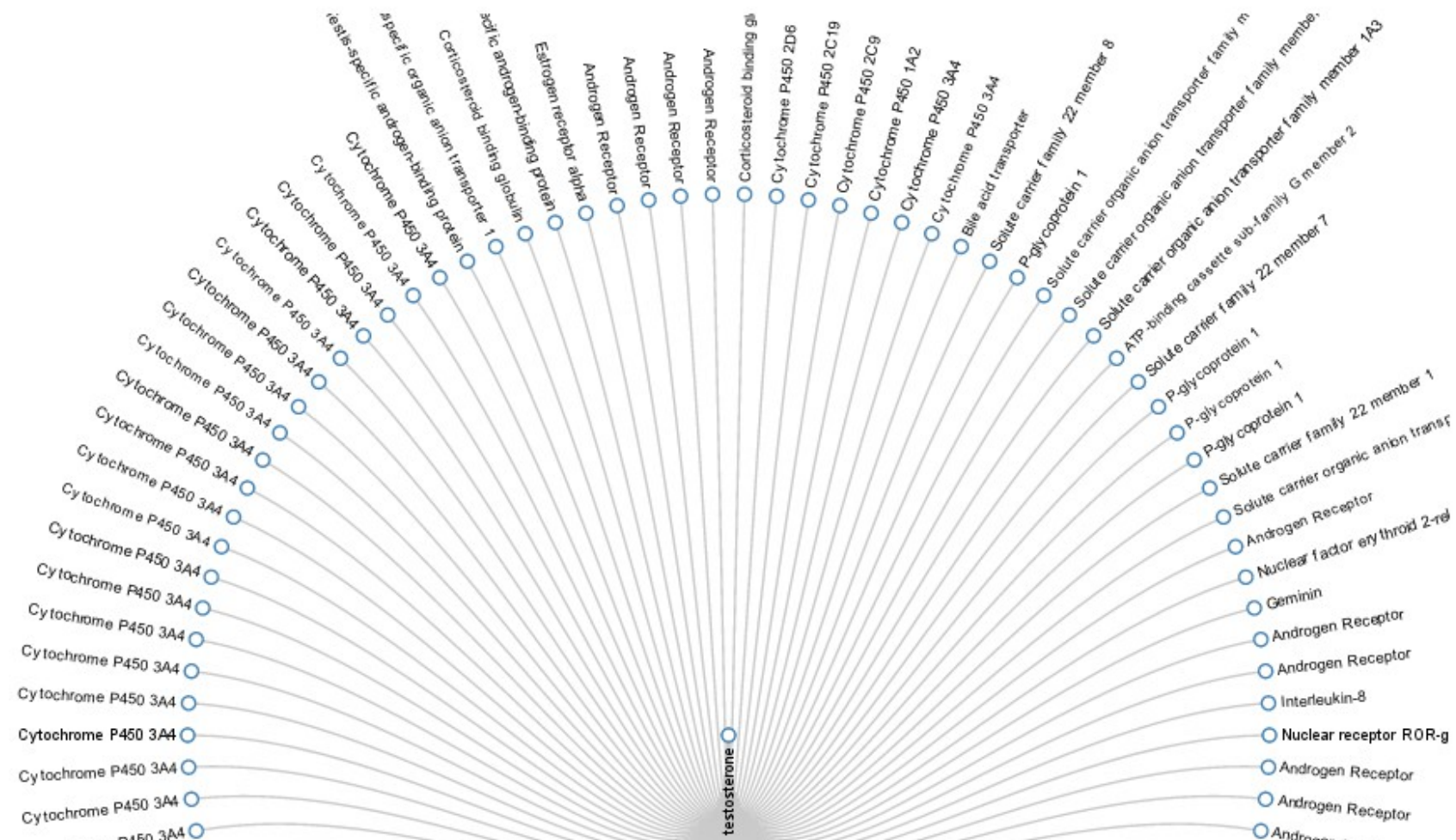
Name: Testosterone

Name: tp

Name: Cloxotestosterone

Name: Methyltestosterone

Name: testosterone acetate



# Conclusion

- A lot of languages to learn
  - HTML, CSS, JSON, JavaScript
- Many programming aspects
  - Basic stuff
  - Concurrency
  - Call-back functions
  - Complex data types (maps)
- OPS LDA allows monitoring of students



# Conclusion

- Hard for students without programming experience
- Impressive results
- Again in 2014-2015
  - But with some preliminary programming advised