Tox21 Enricher
User's Manual

(Last Updated 09/27/2021)

http://hurlab.med.und.edu/Tox21Enricher

Junguk Hur, Ph.D.

Department of Biomedical Sciences

University of North Dakota, School of Medicine and Health Sciences

Grand Forks, North Dakota 58202, USA

Email: InformaticsTools@gmail.com

http://hurlab.med.und.edu

Table of Contents

Table of Contents (Continued)

I. Introduction to Tox21 Enricher

Humans are exposed to tens of thousands of chemicals that are used in daily life, some at levels that may pose a health risk. There is limited toxicological information for many of these chemicals, which makes risk assessment difficult or impossible. The United States Toxicology Testing in the 21st Century (Tox21) program was established to develop more efficient and human-relevant toxicity assessment methods. The Tox21 program is currently screening a set of over 10,000 chemicals, the Tox21 10K library, using quantitative high-throughput screening (qHTS) of assays that measure effects on toxicity pathways. To date, more than 70 assays have yielded >12 million concentration-response curves by Tox21 researchers. To efficiently apply these data for identifying potential hazardous compounds and for informing further toxicological testing, the United States National Toxicology Program (NTP) has developed several web applications (Tox21 Toolbox: **http://ntp.niehs.nih.gov/tbox/**), including tools for data visualization (Tox21 Curve Browser) and exploration (Tox21 Activity Profiler).

One critical usage of this dataset is to perform chemical-relational analysis based on the patterns of activity across the Tox21 assays and then to use nearest neighbor-based prediction to infer the toxicological properties of untested chemicals via their association with tested chemicals. One approach to inferring the specific properties is to perform chemical annotation enrichment of chemical neighborhoods.

Here, we present Tox21 Enricher, a web-based chemical annotation enrichment tool for Tox21 assay data built using the R Shiny framework. Tox21 Enricher identifies significantly over-represented chemical biological annotations among sets of chemicals (neighborhoods), which facilitates the identification of the toxicological properties and mechanisms in the chemical sets.

II. Setup

II.1. Getting a Copy of Tox21 Enricher

The Tox21 Enricher code for both the server-side and client-side applications may be found at https://github.com/hurlab/tox21enricher. Assuming you have cloned the repository into a folder called "/home/user/tox21enricher/" on your machine, server-side code can be found in /home/user/tox21enricher/ and client-side code can be found in /home/user/tox21enricher/tox21enricher/.

II.2. Server Application Setup

There are three main components to the Tox21 Enricher server-side utilities: the Plumber API, the request queue, and the database. All installation guides assume you are using a *nix environment, preferably Ubuntu 20.04 LTE.

II.2.a. Server Configuration

Assuming you have cloned the code to /home/user/tox21enricher/, the server-side code's configuration file can be found at /home/user/tox21enricher/config.yml. This configuration file has two namespaces, `tox21enricher` and `tox21enricher-queue`, for the Plumber API and the queue, respectively.

The importance of each of the variables in this configuration file is explained below:

- **driver** – The database driver. This should always be set to `"Postgres"`.
- **server** – The location of the Tox21 Enricher database in the form `<host_ip>:<port>/<database_name>`. For example, if you are hosting the database on the same machine at the default Postgres port of 5432 with a database name of "tox21enricher", this should be set to `"127.0.0.1:5432/tox21enricher"`.
- **host** – The name or IP address of the machine where the database is hosted. If you are hosting the database on the same machine, this should be set to `"127.0.0.1"` or `"localhost"`.
- **uid** – The username of the Postgres user that owns the Tox21 Enricher database.
- **pwd** – The password of the Postgres user that owns the Tox21 Enricher database.
- **port** – The port that the database is listening on. If you set up PostgreSQL to use the default port, this should be set to `"5432"`.
- **database** – The name of the Tox21 Enricher database on the Postgres server. This will likely be set to `"tox21enricher"`.
- **appversion** – The current version of the application. This is used to know which version of the Tox21 Enricher manual to serve to users.
- **appdir** – The location of the server-side project files. This should be wherever you cloned the repository to (i.e., `"/home/user/tox21enricher/"`).
- **python** – The location of the Python installation **in the RDKit environment** (i.e., `"/home/user/anaconda3/envs/my-rdkit-env/bin/python3.6"`).

II.2.b. Setting Up the Plumber API

The Plumber API serves as the primary point of contact between the Tox21 Enricher client application and the other components of the Tox21 Enricher server-side utilities. Requests to perform enrichment and to access the application's database are processed through the Plumber API via HTTP requests. The Plumber API should ideally run as a daemon on the same machine as the queue and the database.

To set up the Plumber API:

1. If they do not already exist, create an Input/ and Output/ directory (note the capital letters) in the project's root directory. Assuming you have cloned the code to /home/user/tox21enricher, these directories should be created at /home/user/tox21enricher/Input and /home/user/tox21enricher/Output.
2. Download and install R from the appropriate CRAN mirror: https://cran.r-project.org/mirrors.html.
3. Download and install an OpenJDK Java distribution from https://openjdk.java.net/ or through SDKMAN! (https://sdkman.io/).
4. Install a few necessary packages using `apt-get update && apt-get install`:
    - libbz2-dev
    - libpcre2-dev
    - libpq-dev
    - libz-dev
    - r-base-dev
5. Use `install.packages('package_name',dependencies=TRUE)` to install the necessary R libraries for the API:
    - config
    - future
    - ggplot2
    - httr
    - parallel
    - plumber
    - plyr
    - pool
    - promises
    - reticulate
    - rjson
    - RPostgreSQL
    - stringr
    - tidyverse
    - uuid
    - xlsx
6. Ensure you have properly updated the server's configuration file.
7. Ensure the Tox21 Enricher PostgreSQL server is running and listening on an open port.
8. Open the startPlumberAPI.sh script in a text editor and set `port=9000` to

whatever port you want the API to listen on.

9. Ensure the startPlumberAPI.sh script has executable permissions by running `chmod +x /home/user/tox21enricher.startPlumberAPI.sh`.
10. Run the API with the command `/home/user/tox21enricher.startPlumberAPI.sh`.

II.2.c. Setting Up the Queue

The queue is an Rscript that continually observes the filesystem to see if any enrichment requests have been submitted and either performs enrichment analysis or fetches corresponding annotations for the inputted chemicals.

To set up the queue:

1. If it does not already exist, create a Queue/ directory (note the capital letter) in the project's root directory. Assuming you have cloned the code to /home/user/tox21enricher, this directory should be created at /home/user/tox21enricher/Queue.
2. In this Queue/ directory, create a blank file named "0__queue__default" if it does not already exist. This is necessary for the queue numbering to work correctly.
3. Download and install R from the appropriate CRAN mirror: https://cran.r-project.org/mirrors.html.
4. Download and install an OpenJDK Java distribution from https://openjdk.java.net/ or through SDKMAN! (https://sdkman.io/).
5. Install a few necessary packages using `apt-get update && apt-get install`:
    - libbz2-dev
    - libpcre2-dev
    - libpq-dev
    - libz-dev
    - r-base-dev
6. Use `install.packages('package_name',dependencies=TRUE)` to install the necessary R libraries for the API:
    - config
    - future
    - ggplot2
    - httr
    - parallel
    - plumber
    - plyr
    - pool
    - promises
    - rjson
    - RPostgreSQL
    - stringr
    - tidyverse
    - uuid
    - xlsx

7. Ensure you have properly updated the server's configuration file.
8. Ensure the Tox21 Enricher PostgreSQL server is running and listening on an open port.
9. Ensure the startQueue.sh script has executable permissions by running `chmod +x /home/user/tox21enricher.startQueue.sh`.
10. Run the queue with the command `/home/user/tox21enricher.startQueue.sh`.

II.2.d. Setting Up the Database

Tox21 Enricher uses a PostgreSQL database to store chemical and annotation data. Specifically, the application uses the PostgreSQL cartridge for RDKit (https://www.rdkit.org/docs/index.html).

To install RDKit and the PostgreSQL database:

1. Download and install RDKit, following the directions at https://www.rdkit.org/docs/Install.html.
2. This will require you to download and install the Conda package manager and Python.
3. Install and initialize the RDKit PostgreSQL cartridge, following the directions at https://www.rdkit.org/docs/Install.html#installing-and-using-postgresql-and-the-rdkit-postgresql-cartridge-from-a-conda-environment.
4. Make sure the PostgreSQL server is running by using <conda_folder>/envs/my-rdkit-env/bin/pg_ctl -D /<data_folder>/.
5. Obtain a backup or copy of the Tox21 Enricher database from Dr. Hur.
6. Create a database called "tox21enricher" and either:
    a. Log into it by using `<conda_folder>/envs/my-rdkit-env/bin/psql tox21enricher`. Run the command `\i /<path_to_database_backup>/` from within Postgres to restore the database from the backup.
    b. Use the pg_restore utility to restore the database from backup by running from the command line using `<conda_folder>/envs/my-rdkit-env/bin/pg_restore /<path_to_database_backup>/`.

II.3. Client Application Setup

The Tox21 Enricher client application provides a graphical user interface for performing enrichment analysis and interacting with the data stored in the Tox21 Enricher database. The client application may be run on the same machine as the server-side utilities and exposed to the internet so that other remote clients may use the application via a web browser. Otherwise, the client application may also be run directly on the client machine as a standalone executable and connect remotely to the server utilities running on a remote host. To access the Tox21 Enricher client application via an internet browser, your browser must have JavaScript enabled for the application to load properly.

II.3.a. Client Configuration

Assuming you have cloned the code to /home/user/tox21enricher/, the server-side code's configuration file can be found at /home/user/tox21enricher/tox21enricher/config.yml. This configuration file is notably simpler than the configuration file used for the Tox21 Enricher server. The client configuration file has only one namespace, `tox21enricher-client`.

The importance of the two variables in this configuration file is explained below:

- **host** – The name or IP address of the machine where the Tox21 Enricher Plumber API is hosted. If you are hosting the API on the same machine, this should be set to `"127.0.0.1"` or `"localhost"`.
- **port** – The port that the Tox21 Enricher Plumber API is listening on. By default, this is set to `"9000"`.

II.3.b. Setting Up the Client Application (Standalone)

The standalone client application will be available in a future update.

II.3.c. Setting Up the Client Application (Web Application)

The Tox21 Enricher client application is a Shiny (https://www.rstudio.com/products/shiny/) application and thus can be hosted on a machine and accessed from a remote client via a internet browser.

To set up the client application as a web application:

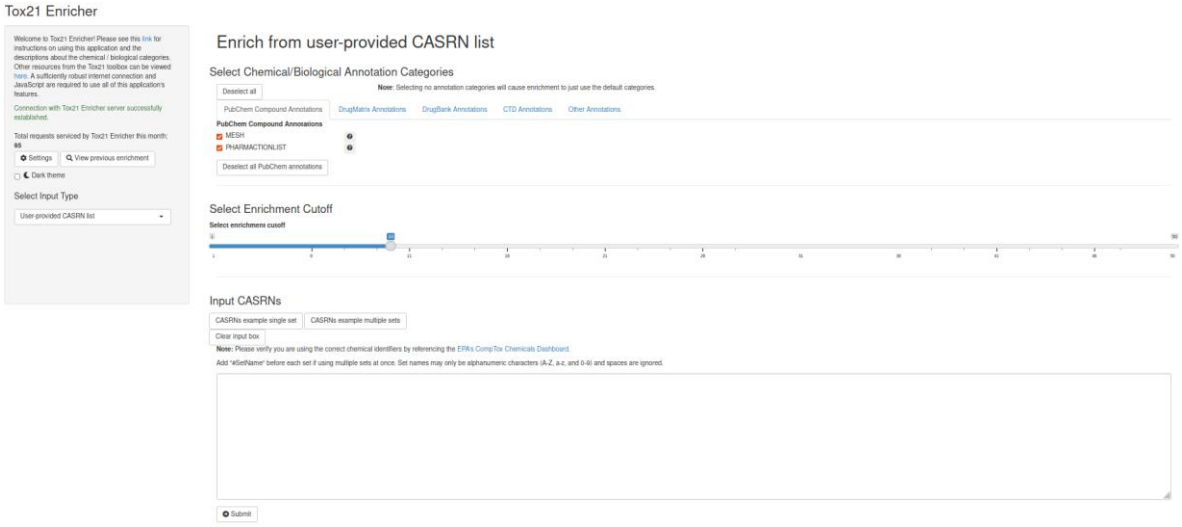
1. Download and install R from the appropriate CRAN mirror: https://cran.r-project.org/mirrors.html.
2. Download and install an OpenJDK Java distribution from https://openjdk.java.net/ or through SDKMAN! (https://sdkman.io/).
3. Install a few necessary packages using `apt-get update && apt-get install`:
    - libbz2-dev
    - libgdal-dev
    - libpcre2-dev
    - libpq-dev
    - libudunits2-dev
    - libz-dev

- r-base-dev

4. Use `install.packages('package_name',dependencies=TRUE)` to install the necessary R libraries for the API:
    - bslib
    - catmaply
    - CePa
    - config
    - dplyr
    - DT
    - future
    - ggVennDiagram
    - httr
    - igraph
    - igraphdata
    - parallel
    - plotly
    - pool
    - promises
    - rclipboard
    - rjson
    - RPostgreSQL
    - shiny
    - shinyBS
    - shinycssloaders
    - shinydashboard
    - shinyjs
    - shinythemes
    - stringr
    - uuid
    - visNetwork
    - xlsx
5. Ensure you have properly updated the client application's configuration file.
6. Launch the application according to the **Running in a Separate Process** section in the instructions here: https://shiny.rstudio.com/articles/running.html.
7. Open the port that the client application runs on. By default, this is a random port, but you can configure the application to run on the same port every time by specifying a specific port for the `port` argument of the `shiny::runApp()` function as described in the documentation here: https://shiny.rstudio.com/reference/shiny/1.0.1/runApp.html.

III. Landing Page

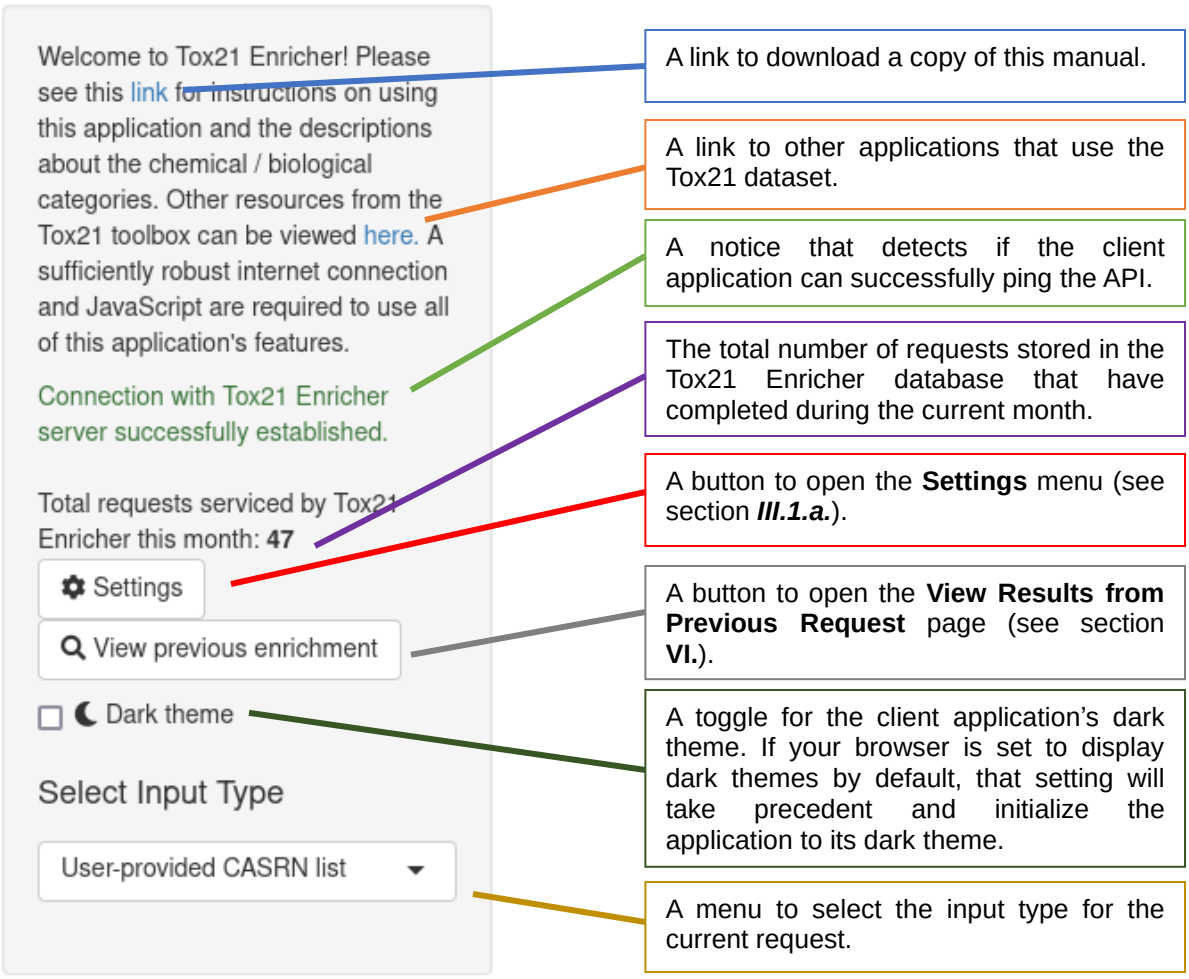
Upon first using Tox21 Enricher via an internet browser or directly through the client application, you will see the landing page.



The various utilities and options on the landing page will be discussed at length in this section.

III.1. Sidebar & Settings

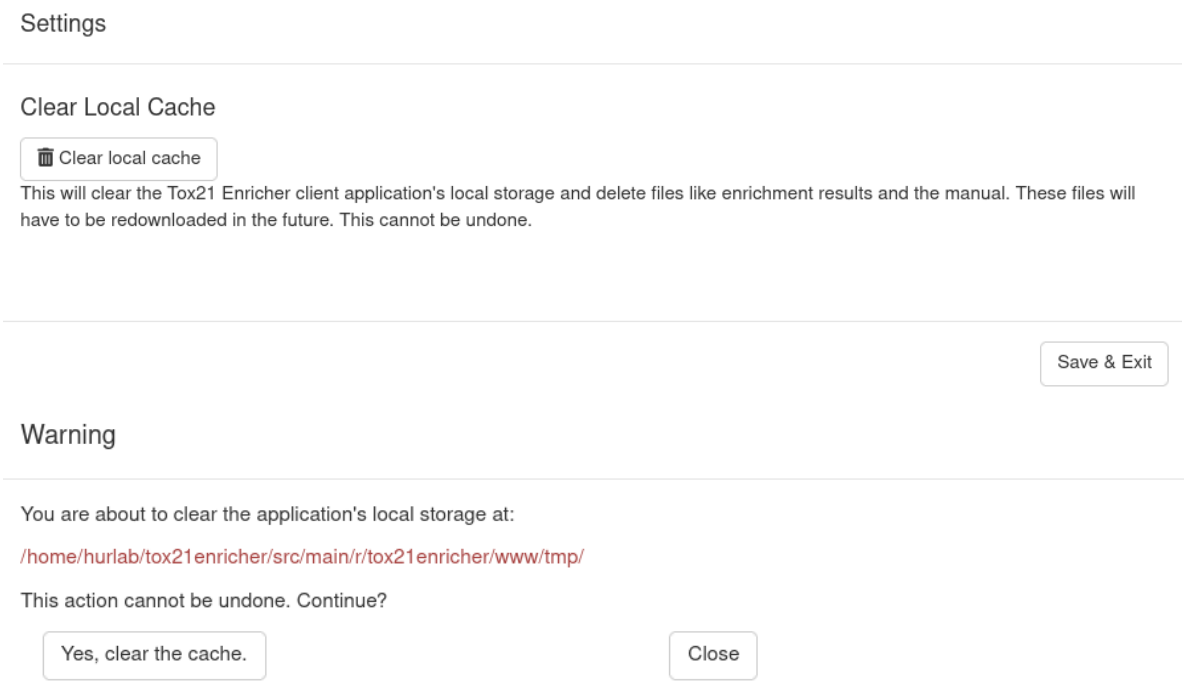
The sidebar of the application displays helpful information and various options and settings.



III.1.a. Settings Menu

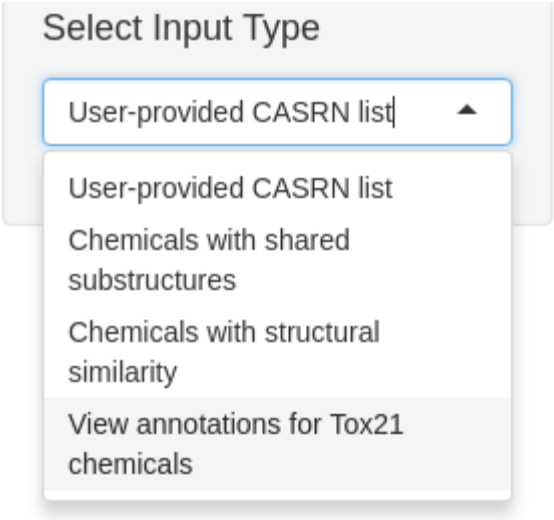
The settings menu allows you to clear the local cache. Whenever you submit a request from the client application to the API or whenever you download a file from the server through the API, the files are stored in a temporary directory at `<project_root>/tox21enricher/www/tmp/`. For example, if you cloned the code to /home/user/tox21enricher/, the temporary directory will be at `/home/user/tox21enricher/tox21enricher/www/tmp/`. Clicking the **Clear local cache** button will prompt you to delete the files present in the temporary directory.

Note that this **will not** delete enrichment record files. Those files are handled separately in the "View previous enrichment" menu. Also, any files deleted in this manner cannot be recovered and will have to be redownloaded from the Tox21 Enricher server in the future.



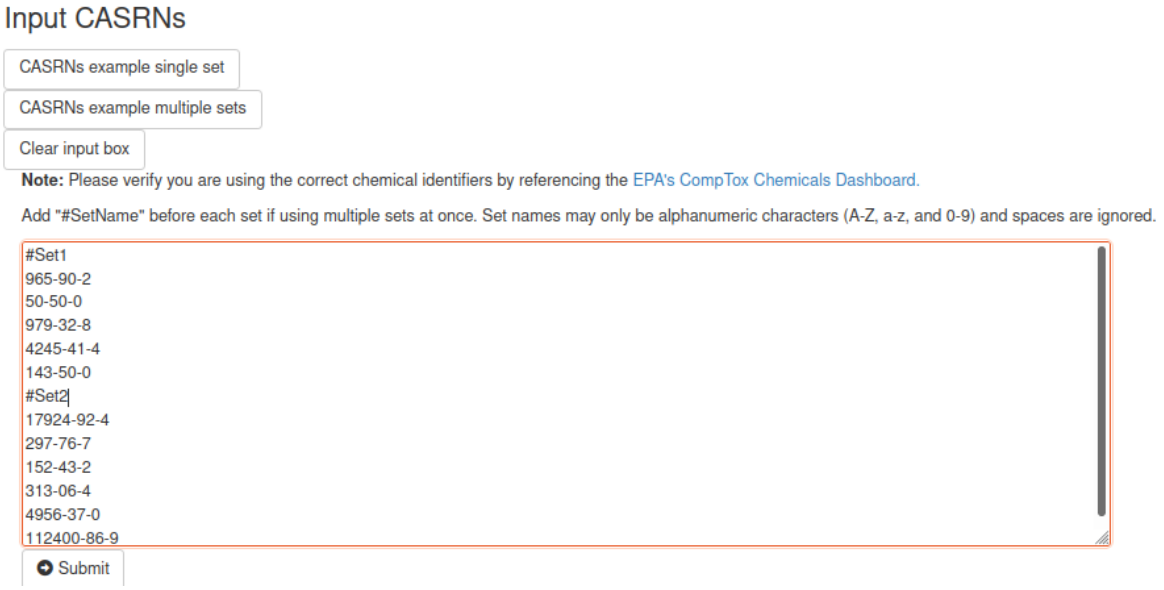
III.2. Selecting the Enrichment Mode and Input Type

The first step in using Tox21 Enricher is selecting the input type. There are three enrichment analysis modes: User-provided CASRN list, Chemicals with shared substructures, and Chemicals with structural similarity. A fourth mode, View annotations for Tox21 chemicals, does not perform enrichment but rather retrieves annotations from the Tox21 Enricher database that correspond to the inputted chemicals.



III.2.a. Enrich From User Provided CASRN List

This enrichment analysis type accepts only CASRN strings as input with one CASRN per line. Multiple sets may be independently enriched at the same time. To enrich multiple sets, separate lines of CASRN strings with **unique** set names of the form "#SetName" with a pound symbol (#) before the name. Only alphanumeric characters (A-Z, a-z and 0-9) may be used in set names. If you choose to provide names for input sets, **all** sets must be assigned a name. Spaces in set names or CASRN strings are ignored, and extra newline characters between lines are also ignored. If no set names are provided, all lines of the input box will be treated as one set and will be assigned the default name of "Set1."

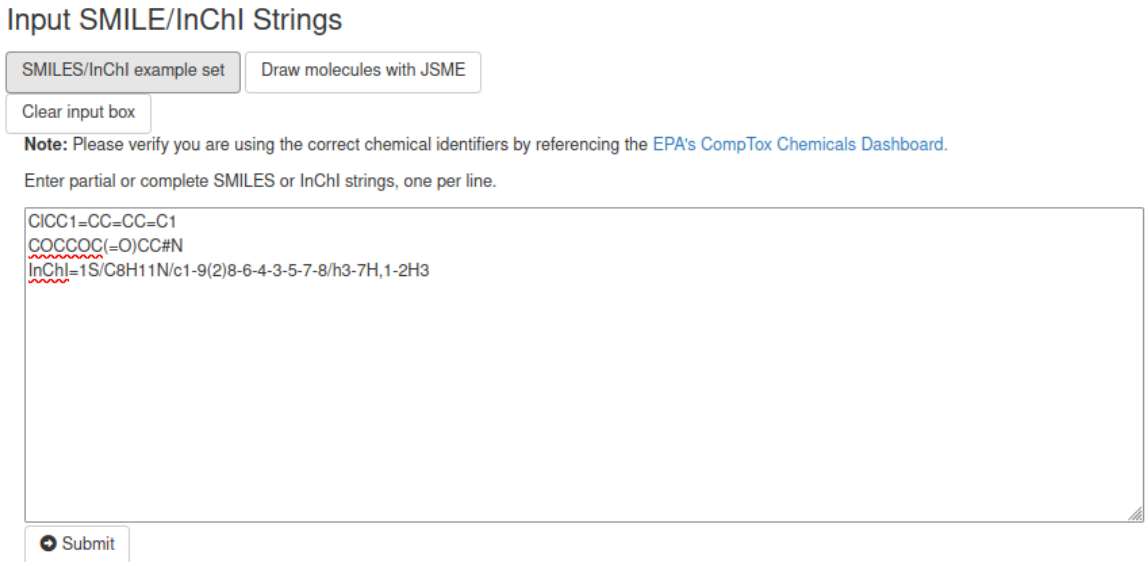


Before performing enrichment from a user provided CASRN list, please cross-reference your CASRN inputs with those in the EPA's CompTox Chemicals Dashboard (https://comptox.epa.gov/dashboard) to ensure accuracy during the enrichment analysis.

Clicking the **CASRNs example single set** button will fill the input box with one, unnamed set of CASRNs. Clicking the **CASRNs example multiple sets** button will fill the input box with three sets of CASRNs. Clicking the **Clear input box** button will reset the input box to a blank state.

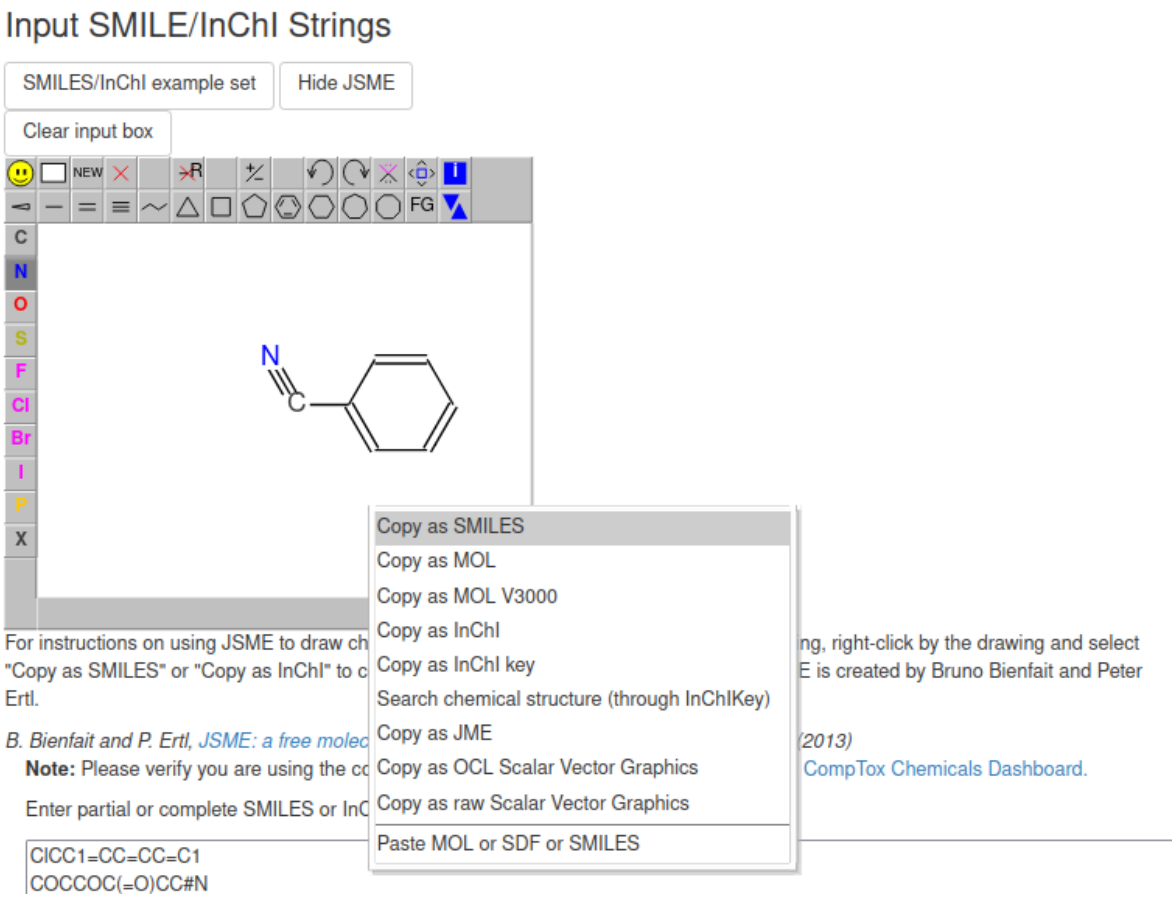
III.2.b. Enrich From Chemicals With Shared Substructures

This enrichment analysis type accepts either SMILES or InChI strings (one per line). Each string is treated as its own data set and will be enriched independently of any other sets/strings provided. SMILES and InChI strings may be mixed: you may perform enrichment with a SMILES string on one line and an InChI string on another.



Clicking the **SMILES/InChI example set** button will fill the input box with three chemicals: two SMILES strings and one InChI string. Clicking the **Clear input box** button will reset the input box to a blank state.

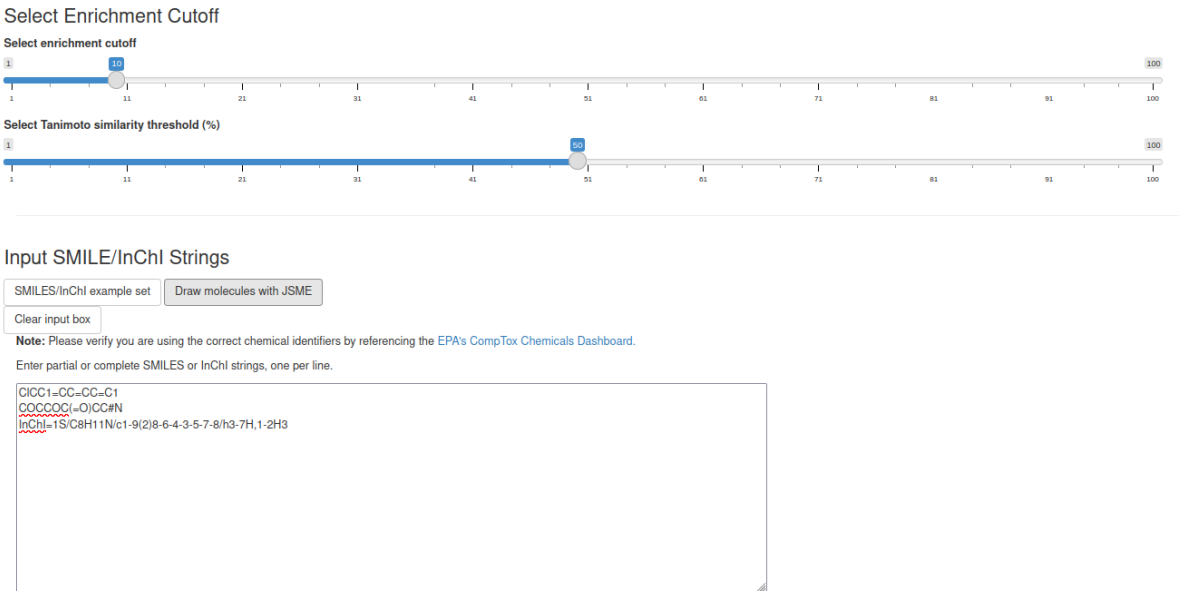
Clicking the **Draw molecules with JSME** button will open the JSME Molecule Editor by Peter Ertl and Bruno Bienfait. Here, you can draw a molecule using the applet's tools. When you are finished drawing, we can right-click in the whitespace of the canvas and select "Copy as SMILES" or "Copy as InChI" and paste the copied text in the input box.



III.2.c. Enrich From Chemicals With Structural Similarity

This enrichment analysis type accepts either SMILES or InChI strings (one per line). Each string is treated as its own data set and will be enriched independently of any other sets provided. SMILES and InChI strings may be mixed: you may perform enrichment with a SMILES string on one line and an InChI string on another.

The Tanimoto similarity threshold may be specified using the slider controls above the input box to set the strictness of the search. By default, this is set to 50%.



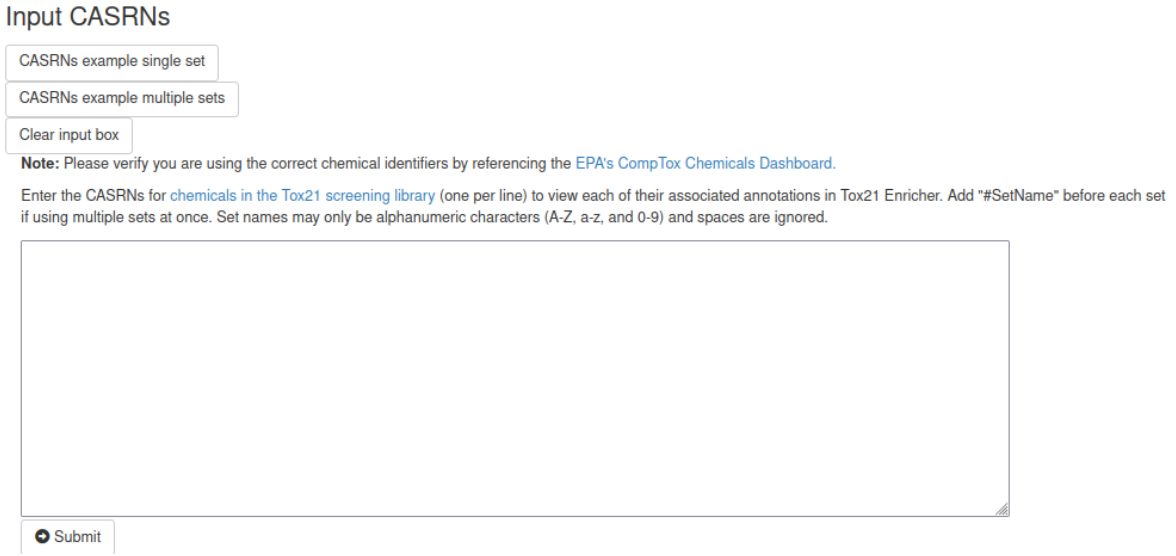
Clicking the **SMILES/InChI example set** button will fill the input box with three chemicals: two SMILES strings and one InChI string. Clicking the **Clear input box** button will reset the input box to a blank state.

Clicking the **Draw molecules with JSME** button will open the JSME Molecule Editor by Peter Ertl and Bruno Bienfait. Here, you can draw a molecule using the applet's tools. When you are finished drawing, we can right-click in the whitespace of the canvas and select "Copy as SMILES" or "Copy as InChI" and paste the copied text in the input box.

III.2.d. View Annotations for Tox21 Chemicals

This function does not actually perform enrichment analysis; rather, it returns a list of every annotation in the Tox21 database that is associated with each of the supplied chemicals. Like the **User-provided CASRN list** enrichment type, this enrichment analysis type accepts only CASRN strings as input (one CASRN per line). You may find annotations for multiple sets of chemicals at the same time. To use multiple sets, separate lines of CASRN strings with **unique** set names of the form "#SetName" with a pound symbol (#) before the name. Only alphanumeric characters (A-Z, a-z and 0-9) may be used in set names. If you choose to provide names for input sets, **all** sets must be assigned a name. Spaces in set names or CASRN strings are ignored, and extra newline characters between lines are also ignored. If no set names are provided, all lines of the input box will be treated as one set and will be assigned the default name of "Set1."

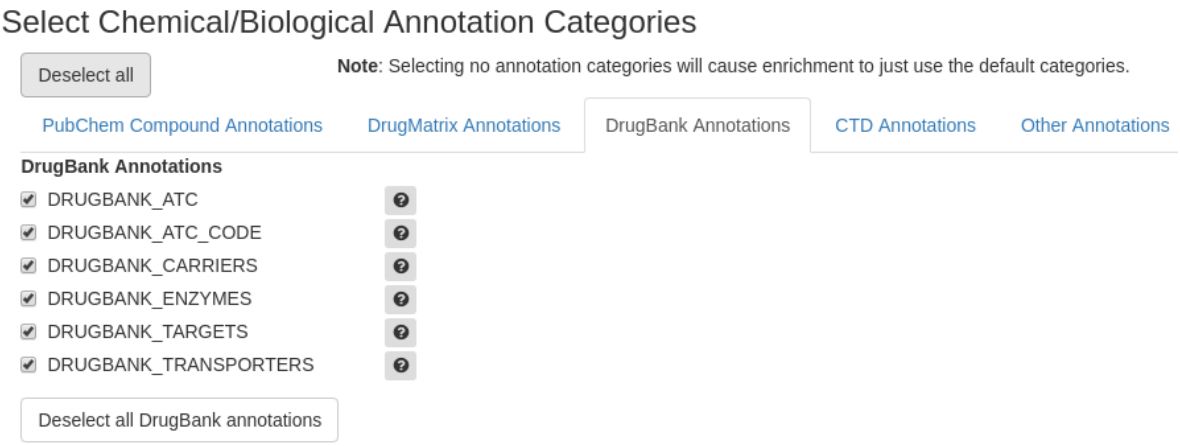
This mode will not provide any annotations for a CASRN if that chemical is not present in the Tox21 screening library. To view all the chemicals in the Tox21 screening library, click the **chemicals in the Tox21 screening library** link which opens the page https://comptox.epa.gov/dashboard/chemical_lists/TOX21SL in a new window.



Clicking the **CASRNs example single set** button will fill the input box with one, unnamed set of CASRNs. Clicking the **CASRNs example multiple sets** button will fill the input box with three sets of CASRNs. Clicking the **Clear input box** button will reset the input box to a blank state.

III.3. Selecting Annotation Categories

The top of the landing page displays lists of annotation categories, separated into five category types**: PubChem Compound Annotations**, **DrugMatrix Annotations**, **DrugBank Annotations**, **CTD Annotations**, and **Other Annotations**. By default, the **PubChem Compound Annotations** tab is selected. You can click on the tabs to view each tab's corresponding annotation categories. Each category name has a checkbox to its left that, when checked, means that the application will attempt to find matching annotations from the given category. The application will ignore any potential results from categories left unchecked. By default, all categories are checked except for the CTD_GOFAT_BIOPROCESS category: using this category can potentially slow down enrichment by a large amount, so it is not recommended to be used in most scenarios. Each category name has a question mark button to its right that, when hovered over, displays additional information about each category. If no categories are selected, the application defaults to just using all the annotation categories (except for CTD_GOFAT_BIOPROCESS).

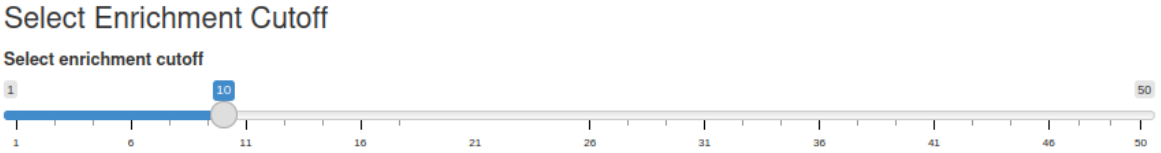


Clicking the **Deselect all/Select all** button will either deselect or select all the annotation categories across all tabs. Note that clicking the **Select all** button will select the CTD_GOFAT_BIOPROCESS category too.

Each tab also has a corresponding button to deselect or select all the annotation categories within the tab. An example can be seen in the image above: clicking the **Deselect all DrugBank annotations** button will deselect only the annotations in the **DrugBank Annotations** tab. Note that clicking the **CTD** tab's corresponding **Select all CTD annotations** button will select the CTD_GOFAT_BIOPROCESS category too.

III.4. Selecting Enrichment Cutoff Value

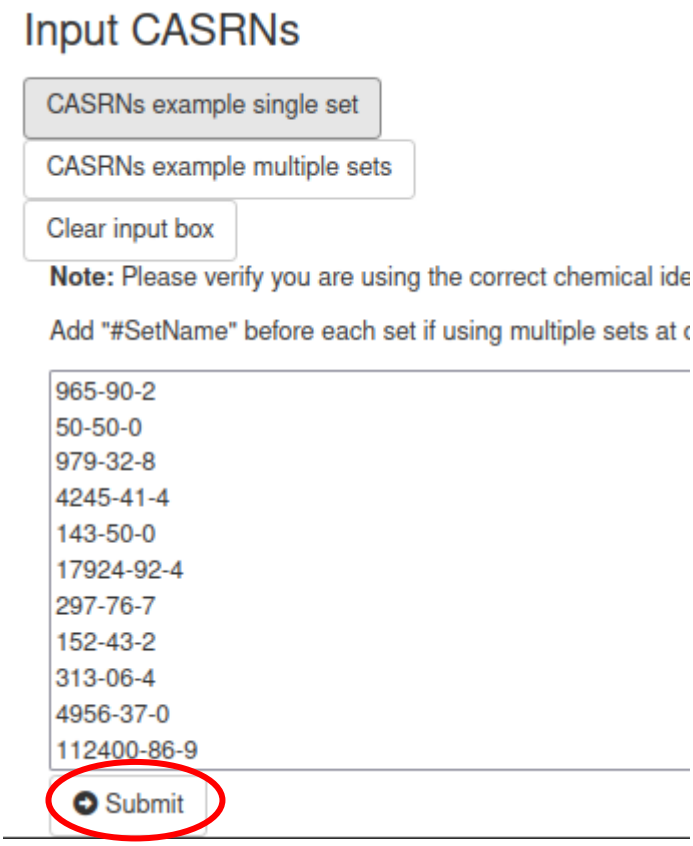
Under the tabs for selecting the appropriate annotation categories, the landing page displays a slider input for specifying the enrichment cutoff value. This value determines the maximum number of enrichment results displayed per data set submitted. By extension, this will also determine the maximum number of nodes displayed during network generation. By default, this is set to 10 and can be increased to a maximum of 50 or decreased to a minimum of 1.



III.5. Performing Enrichment

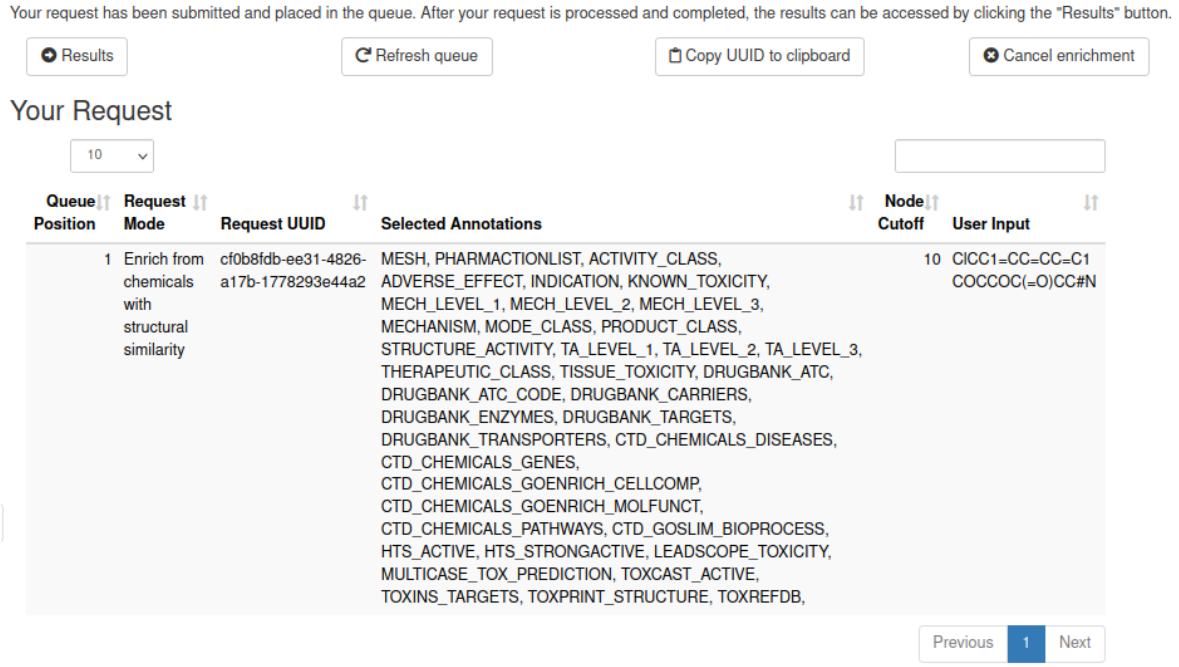
After input has been entered, it is time to begin enrichment. This can be done by clicking on the **Submit** button under the input box. If there are no problems validating your input, you will be sent to the waiting page.

Validation errors may occur if you input a string that is not a properly formatted CASRN, SMILES, or InChI identifier, if you forget to name a set, or if you try to submit a blank input. If a validation error occurs, Tox21 Enricher will display a warning and the request will not proceed.



IV. Waiting Page

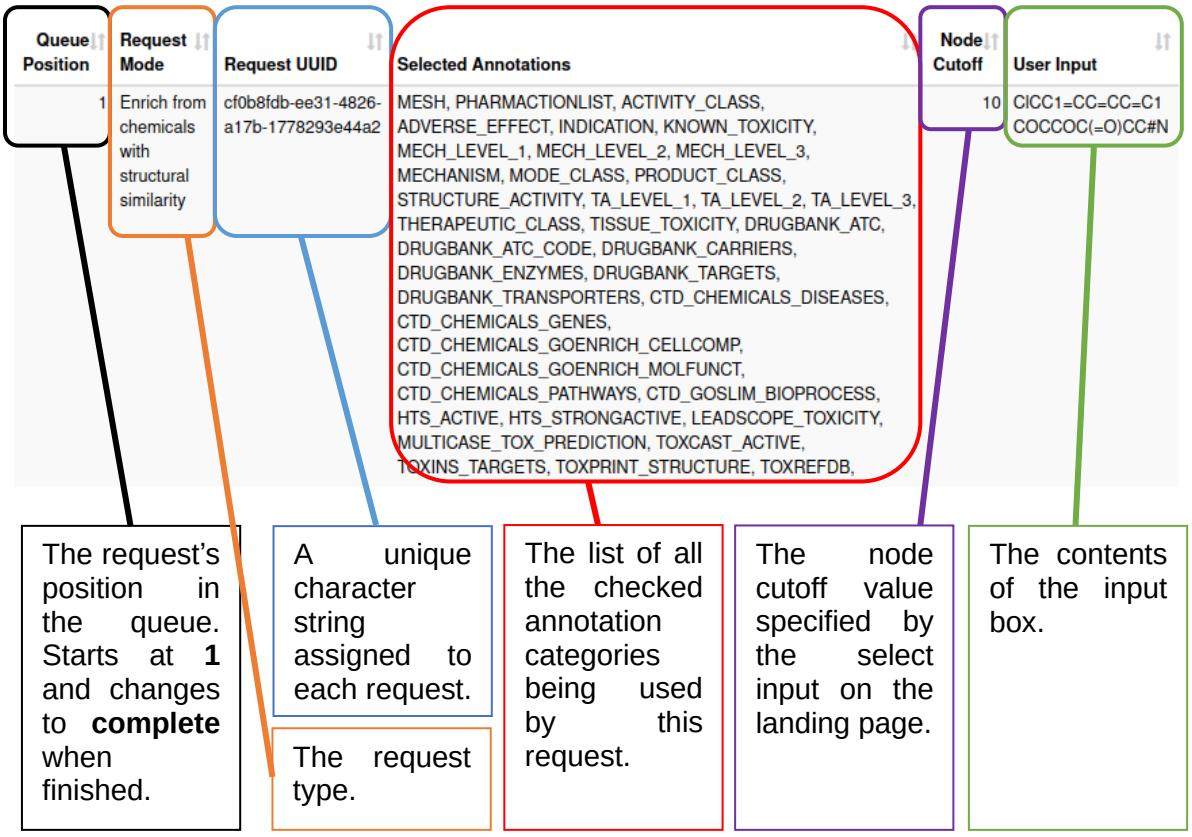
After submitting your request, you will be taken to the waiting page.



There are four buttons at the top of the waiting page that perform different tasks when clicked:

- **Results** – This will take you to the request's results page if the request has finished. If the request has not yet finished, the waiting page will display a warning.
- **Refresh queue** – This will refresh the **Your Request** table, accordingly updating the **Queue Position** column.
- **Copy UUID to clipboard** – This will copy your request's UUID to the clipboard so it may be pasted elsewhere.
- **Cancel enrichment** – This will open a prompt to cancel the request. Cancelled requests cannot be recovered.

The **Your Request** table contains some helpful information related to the parameters of your request:



V. Results Page

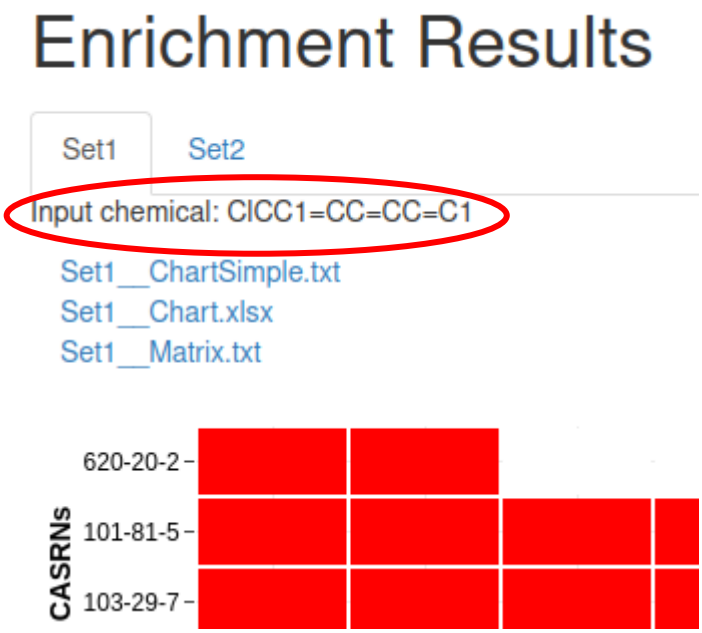
V.1. Enrichment Results

Once your request has completed, you may click the **Results** button to view the request's results.

V.1.a. Result Files

The top of the results page displays relevant files generated by the enrichment analysis process. If multiple sets were submitted, the files corresponding to each set are organized into separate tabs.

Additionally, if you submitted SMILES or InChI strings for enrichment, the original SMILES/InChI string is displayed with the set's results.



If one could be generated, an interactable heatmap using the Plotly library for R (https://plotly.com/r/) will be displayed under the result file links. A red square signifies that the annotation on the x-axis belongs to the CASRN(s) on the y-axis, while a white square signifies that the annotation is not associated with the CASRN(s) on the y-axis.

Clicking any of the result file links will download a copy of the corresponding file from the Tox21 Enricher server to the client application's temporary cache so that it may be viewed at any time in the client application.

The result files are as follows:

- **<set_name>__ChartSimple.txt** – A list of the top 10 most significant annotations for each annotation class (.txt format).
- **<set_name>__ChartSimple.xlsx** – A list of the top 10 most significant annotations for each annotation class (.xlsx format).
- **<set_name>__Chart.txt** – A list of all significant annotations (.txt format).
- **<set_name>__Chart.xlsx** – A list of all significant annotations (.xlsx format).
- **<set_name>__Cluster.txt** – A list of significant terms in which functionally similar annotations are grouped together to remove redundancy. This is performed with respect to the whole annotation set rather than to individual annotation classes (.txt format).
- **<set_name>__Cluster.xlsx** – A list of significant terms in which functionally similar annotations are grouped together to remove redundancy. This is performed with respect to the whole annotation set rather than to individual annotation classes (.txt format).
- **<set_name>__Matrix.txt** – A plaintext representation of the heatmap (.txt format).
- **<set_name> Input** – A plaintext list of the input chemicals for this set (.txt format).

V.1.b. Result Chemicals

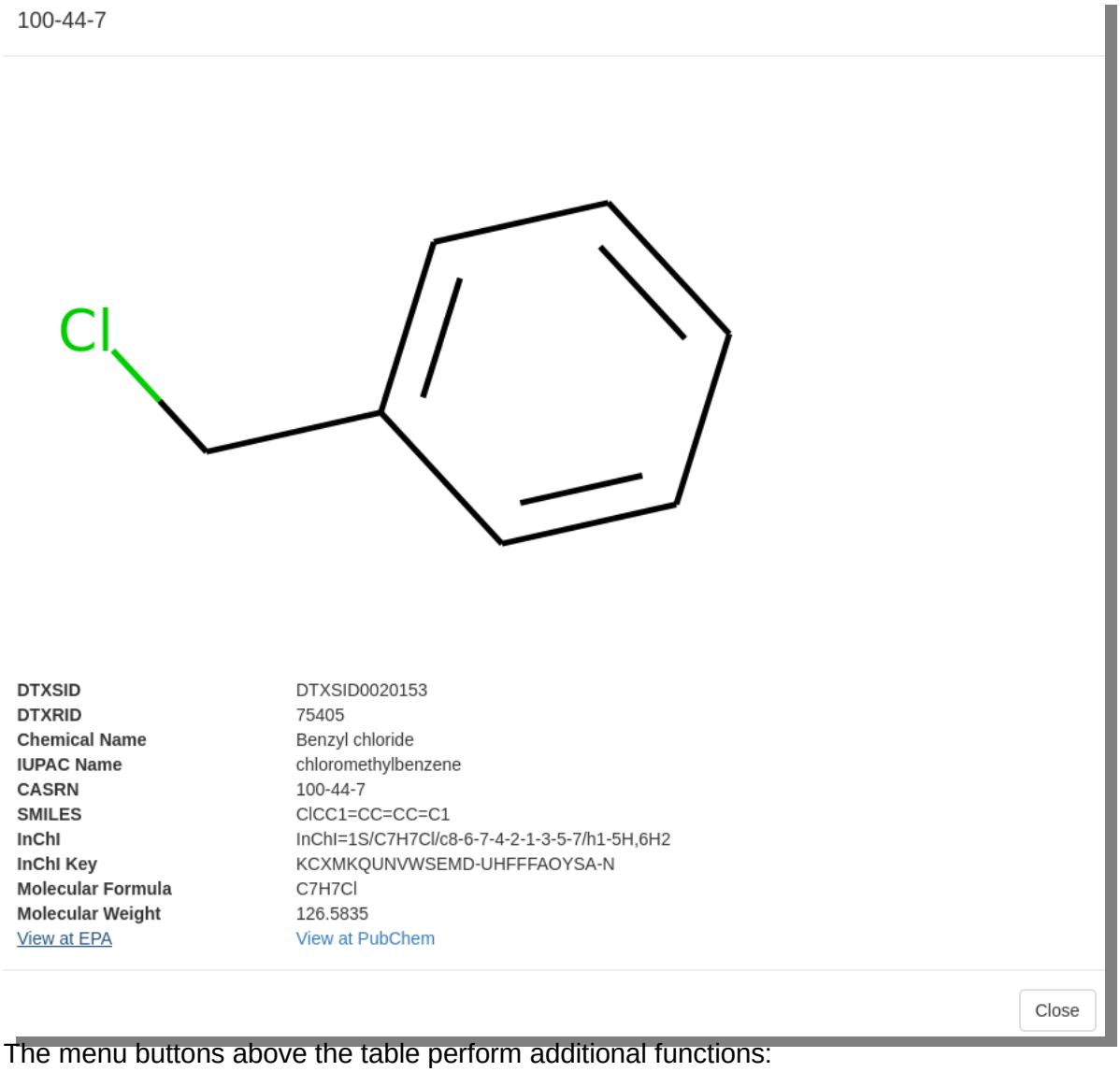
The **Result Chemicals** table is only visible when enriching from chemicals with shared substructures or structural similarity. This table displays information about all of the related chemicals to the original SMILES/InChI input strings.

Result Chemicals

Copy | CSV | Excel | PDF | Print | Column visibility

Select	Chemical Structure	DSSTox Substance ID	CASRN	IUPAC Name	SMILES	InChI	InChI Key	Molecular Formula	Molecular Weight	Similarity
☑		DTXSID0020153	100-44-7	chloromethylbenzene	ClCC1=CC=CC=C1	InChI=1S/C7H7Cl/c8-6-7-4-2-1-3-5-7/h1-5H,6H2	KCXMKQUNVWSEMD-UHFFFAOYSA-N	C7H7Cl	126.5835	1
☑		DTXSID5021839	100-46-9	phenylmethanamine	NCC1=CC=CC=C1	InChI=1S/C7H9N/c8-6-7-4-2-1-3-5-7/h1-5H,6,8H2	WGQKYBSKWIADBV-UHFFFAOYSA-N	C7H9N	107.1531	0.56

Additionally, if you performed enrichment from chemicals with structural similarity, an additional column is shown displaying a Tanimoto value describing the similarity of each result chemical to the original input chemical. Clicking the chemical structure image for a given chemical will open a modal that displays a larger image of the chemical structure, additional information about the chemical, and links to view the chemical's entries on the EPA's CompTox Chemicals Dashboard website and the PubChem database.

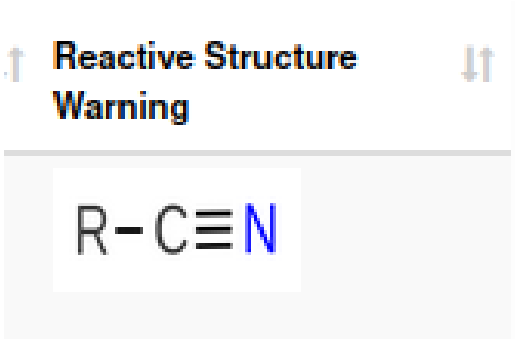


The menu buttons above the table perform additional functions:

- **Copy** – Copies the entire table to the clipboard.
- **CSV** – Allows you to save a copy of the table as a .csv file.
- **Excel** – Allows you to save a copy of the table as an Excel spreadsheet.
- **PDF** – Allows you to save a copy of the table as a .pdf file.
- **Print** – Allows you to print a copy of the table.
- **Column visibility** – Allows you to selectively show or hide certain columns of the table.

Each chemical has an associated checkbox that marks the chemical for re-enrichment. By default, all chemicals' checkboxes are checked.

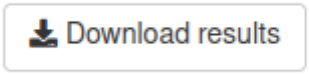
If an originally submitted SMILES/InChI string contains a certain reactive structure, Tox21 Enricher will check to see if each of its related chemicals also contain that structure. If a related chemical does not contain the structure, an additional **Reactive Structure Warning** column is added to the **Result Chemicals** table displaying a picture of the suspect structure. A warning will also be displayed if the original input string does not contain a reactive structure, but one of its related chemicals does. Currently, Tox21 Enricher displays warnings if the following structural groups are detected in either (but not both) the input chemical and one of its related chemicals:



- **Nitrile group (cyanide)**
- **Aldehyde**
- **Epoxide**
- **Isocyanate**

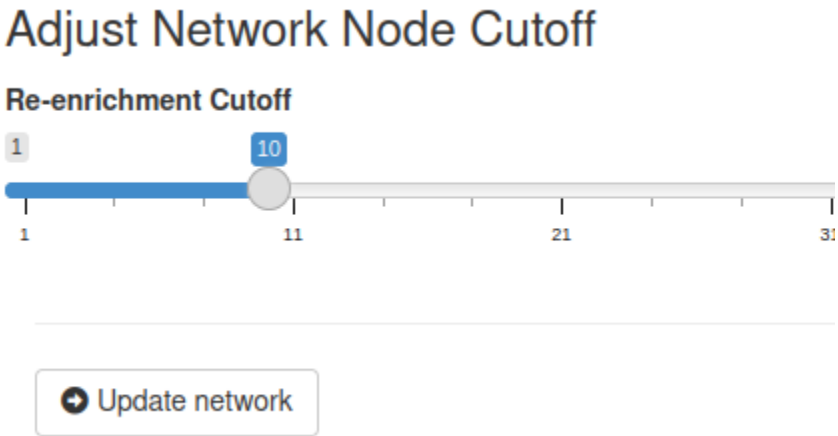
V.1.c. Download Results Button

Clicking the **Download results** button will prompt you to download a .zip archive of all of the results files for all the sets for a given request.

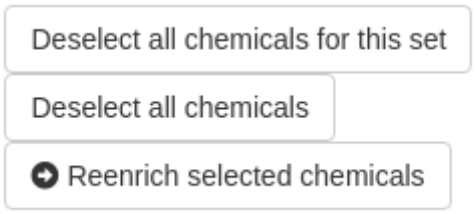


V.1.d. Adjust Network Node Cutoff & Perform Re-enrichment

Below the result files and chemicals is a series of controls for rerunning enrichment on the result set. First is a slider control for adjusting the node cutoff value so that you may change the number of nodes without having to completely perform enrichment again from scratch.



Under this slider input, there is a button to either update the network (if enriching from a user-provided CASRN list) or perform enrichment again with only the selected result chemicals (if enriching from chemicals with structural similarty or shared substructures). Additionally, if enriching from chemicals with structural similarity or shared substructures, more buttons are present that interact with the **Result Chemicals** table:



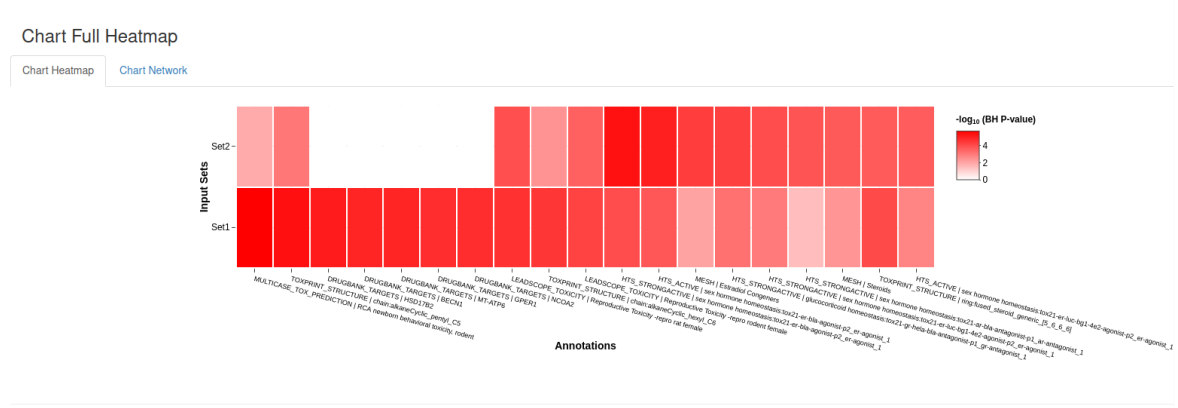
- **Deselect all chemicals for this set** – Deselects all of the result chemicals only for the currently selected tab (result set).
- **Deselect all chemicals** – Deselects all of the result chemicals across all tabs.

Additionally, if any result sets have chemicals with reactive structure warnings, a third 

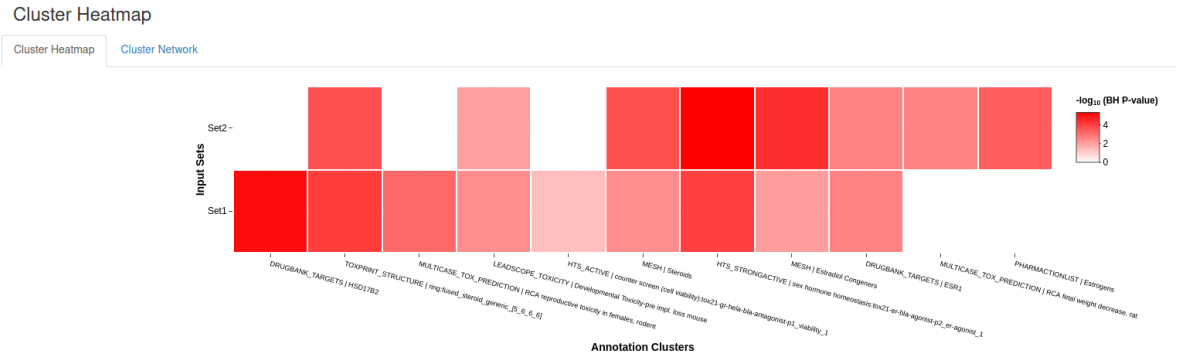
Deselect all chemicals with warnings button will appear. Clicking this button will deselect all chemicals with reactive structure warnings across all tabs.

V.1.e. Chart and Cluster Heatmaps

Below the re-enrichment and node cutoff controls are interactive heatmaps using the Plotly library for R. The chart heatmap displays the most significant annotations across all the chemicals in each set.



The cluster heatmap displays the most significant annotations across all the chemicals in each set as well, but reduces these annotations down to groups of functionally similar annotations.



V.1.f. Chart and Cluster Networks

Clicking on the **Chart Network** tab next to the **Chart Heatmap** tab will open the chart network display. Similarly, clicking on the **Cluster Network** tab will open the cluster network display. Both network displays have identical layouts. The nodes correspond to significantly enriched annotations and the edges indicate that there is significant overlap between the two annotations in terms of chemical contents. The edge color gradient indicates the degree of overlap based on a Jaccard index. Note that the network may take some time to load depending on the number of nodes represented.

On the left side, there is a number input for changing the q-value. This is the cutoff value for the statistically significant overlap between pairs of nodes. There are also checkboxes for changing which input sets should be represented in the network. Under the **Other Options** section, there is the option to enable network physics, which will make the nodes physically react with each other. There is also the option to toggle curve smoothing for the network edges. This is enabled by default, but disabling it may improve performance. Clicking the **Update network** button will regenerate the network, reflecting any changes made to selected input sets, physics, or the q-value.

In the middle of the network display, there are two drop-down menus. The top menu allows you to select certain annotations in the network without having to click on the nodes in the network. The bottom menu allows you to hide all annotations not of the selected category, making the network easier to navigate and read. Below these menus is a legend that shows each category's unique color.

You may interact with the network by clicking and dragging nodes around the canvas. The mouse wheel can be used to zoom in and out of the network view as well. Additionally, the green buttons overlaid on the network may also be clicked to manipulate the network view. Under the network, there is the **Export as png** button which allows you to save an image of the network as a .png file.

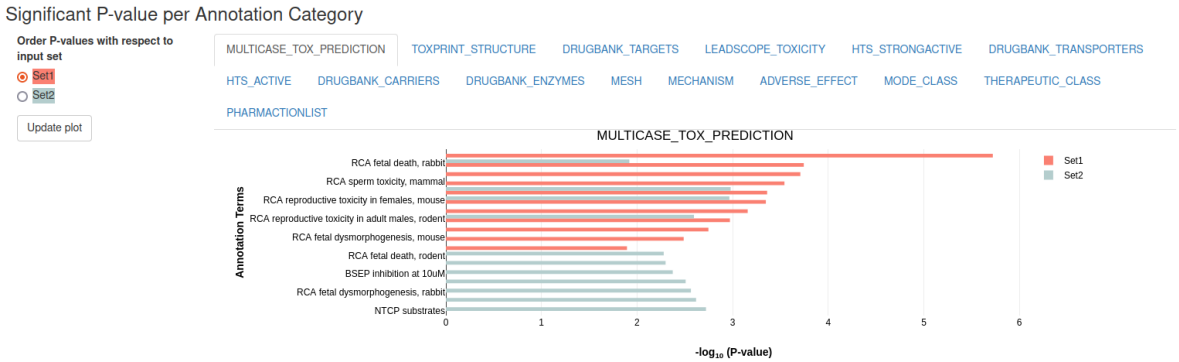
Under the **More Information for Selected Annotation** section, you will see a prompt to select any node in the network. Doing this will display a link to an external website or database with more details about the annotation in the selected node.

Under the **Overlapping Chemicals** section, you will see a prompt to select any edge in the network. Doing this will generate a Venn diagram under the network detailing the number of chemicals shared and unique to the annotations displayed in the connecting nodes of the selected edge. Clicking an edge will also generate buttons that will allow you to view a list of all the chemicals (as CASRNs) associated with either of the two given annotations. A third button will also be generated that will allow you to view a list of all the chemicals (as CASRNs) that are associated with both given annotations. Underneath these buttons are links that allow you to save the Venn diagram as a .png image or .pdf document when clicked.


Chart Full Heatmap
Chart Heatmap
Chart Network
Edge Selection Criteria
Q-value
0.05
Selected Input Sets
Set1
Set2
Other Options
Enable physics?
Smooth curve for edges?
Update network
More Information for Selected Annotation
Click on any node to view additional information for the annotation in the selected node.
Overlapping Chemicals
Click on any edge to view a Venn diagram of the chemicals associated with the annotations in its two nodes.
Select by id
Select by group
LEADSCOPE_TOXICITY
MESH
MULTICASE_TOX_PREDICTION
TOXPRINT_STRUCTURE
chain:aromaticAlkane_Ph-C1_acyclic_connect_noDblBd
Hydrocarbons, Cyclic
chain:aromaticAlkane_Ph-C1_acyclic_generic
Benzene Derivatives
ring:aromatic_phenyl
chain:aromaticAlkane_Ph-C1_acyclic_connect_H_gt_1
Neurotoxicityy-pup behavior rodent
Export as png
View chemicals for MESH | Hydrocarbons, Cyclic
View chemicals for TOXPRINT_STRUCTURE | chain:aromaticAlkane_Ph-C1_acyclic_generic
View shared chemicals
Download plot as .png
Download plot as .pdf
MESH | Hydrocarbons, Cyclic
TOXPRINT_STRUCTURE | chain:aromaticAlkane_Ph-C1_acyclic_generic
469
620
2033
Chart Full Heatmap
Chart Heatmap
Chart Network
Edge Selection Criteria
Q-value
0.05
Selected Input Sets
Set1
Set2
Other Options
Enable physics?
Smooth curve for edges?
Update network
More Information for Selected Annotation
Click on any node to view additional information for the annotation in the selected node.
Overlapping Chemicals
Click on any edge to view a Venn diagram of the chemicals associated with the annotations in its two nodes.
Hydrocarbons, Cycli
Select by group
LEADSCOPE_TOXICITY
MESH
MULTICASE_TOX_PREDICTION
TOXPRINT_STRUCTURE
chain:aromaticAlkane_Ph-C1_acyclic_connect_noDblBd
Hydrocarbons, Cyclic
chain:aromaticAlkane_Ph-C1_acyclic_generic
Benzene Derivatives
ring:aromatic_phenyl
chain:aromaticAlkane_Ph-C1_acyclic_connect_H_gt_1
Neurotoxicityy-pup behavior rodent
Export as png
Click here to view more details about Hydrocarbons, Cyclic.

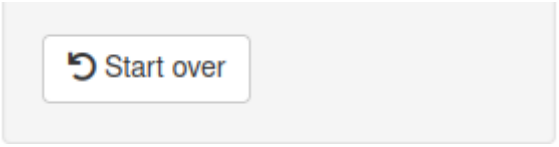
V.1.g. Significant P-value per Annotation Category Bar Graphs

At the bottom of the results page, you can find a series of bar graphs that display all the significant annotations across all result sets and each set's inverse p-value for each annotation, if applicable. Separate bar graphs are created for each annotation category and are put into each category's respective tab. A color legend appears on the righthand side of each bar graph if multiple sets are represented. Clicking on a set name in this legend will hide its trace from the graph, while double-clicking a set name will hide all other sets' traces. To the left of the bar graphs is a menu that allows you to select one set to order values by. Clicking the **Update plot** button will update all the bar graphs to be ordered with respect to the selected set. Trace colors are unique for each set and are randomly selected from a predefined list each time enrichment is performed. Bar graphs are always ordered with the most significant value for the selected set at the top and the least significant value for the selected set at the bottom.



V.1.h. Start Over Button

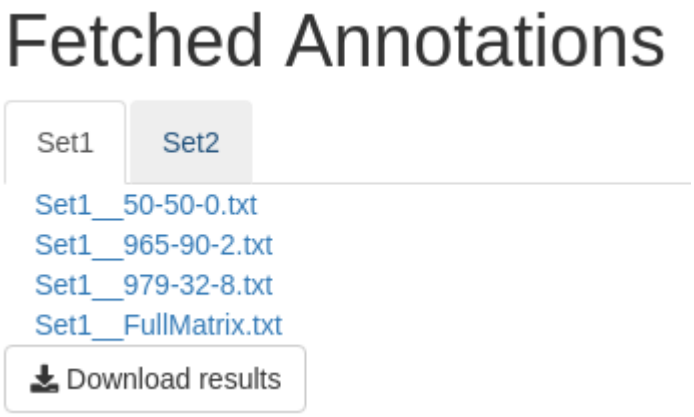
After enrichment is successfully performed, the **Start over** button is added to the bottom of the sidebar. Clicking this button will take you back to the landing page. This button is functionally identical to refreshing the application from within an internet browser.



V.2. View Annotation Results

V.2.a. Result Files

The top of the results page displays relevant files generated when fetching annotations. If multiple sets were submitted, the files corresponding to each set are organized into separate tabs.

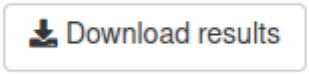


Each submitted CASRN will have its own text file with a name of the form **<set_name>__<casrn>.txt**. This file will contain all of the annotations in the Tox21 Enricher database that are associated with the submitted chemical. The annotations in this file are organized so that each line contains one complete annotation: the category the annotation belongs to is on the left and the annotation's name is on the right, and they are separated by a tab. If Tox21 Enricher could not find any associated annotations for a submitted CASRN, the CASRN will be ignored and a corresponding result file will not be generated. The annotation-fetching process will only search through the annotation categories that were checked on the landing page.

An additional text file of the **name <set_name>__FullMatrix.txt** will be generated for each submitted set of CASRNs. This file contains a plaintext representation of a matrix of all the fetched annotations in the set and if they are associated with each of the submitted CASRNs.

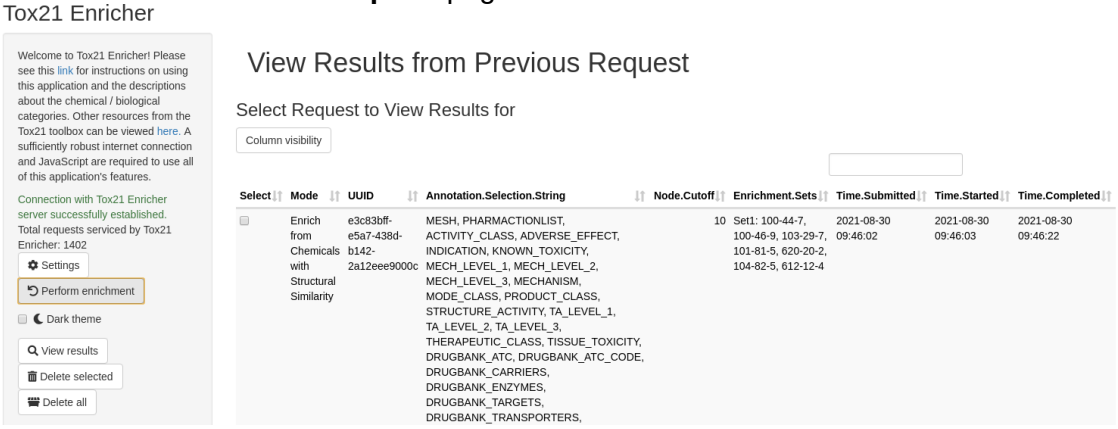
V.2.b. Download Results Button

Clicking the **Download results** button will prompt you to download a .zip archive of all of the results files for all the sets for a given request.



VI. View Results From Previous Request Page

Clicking the **View previous results** button on the sidebar will take you to the **View Results from Previous Request** page.



There are three buttons in the sidebar menu that perform different functions:

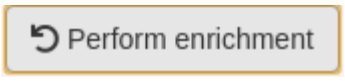
- **View results** – This button will take you the results page of whatever previous request you have selected in the table below.
- **Delete selected** – This button will prompt you to delete the records of all the checked requests in the table below.
- **Delete all** – This button will prompt you to delete the records of all previous requests.

There is also the **Column visibility** button above the table. This button allows you to selectively show or hide certain columns of the table.

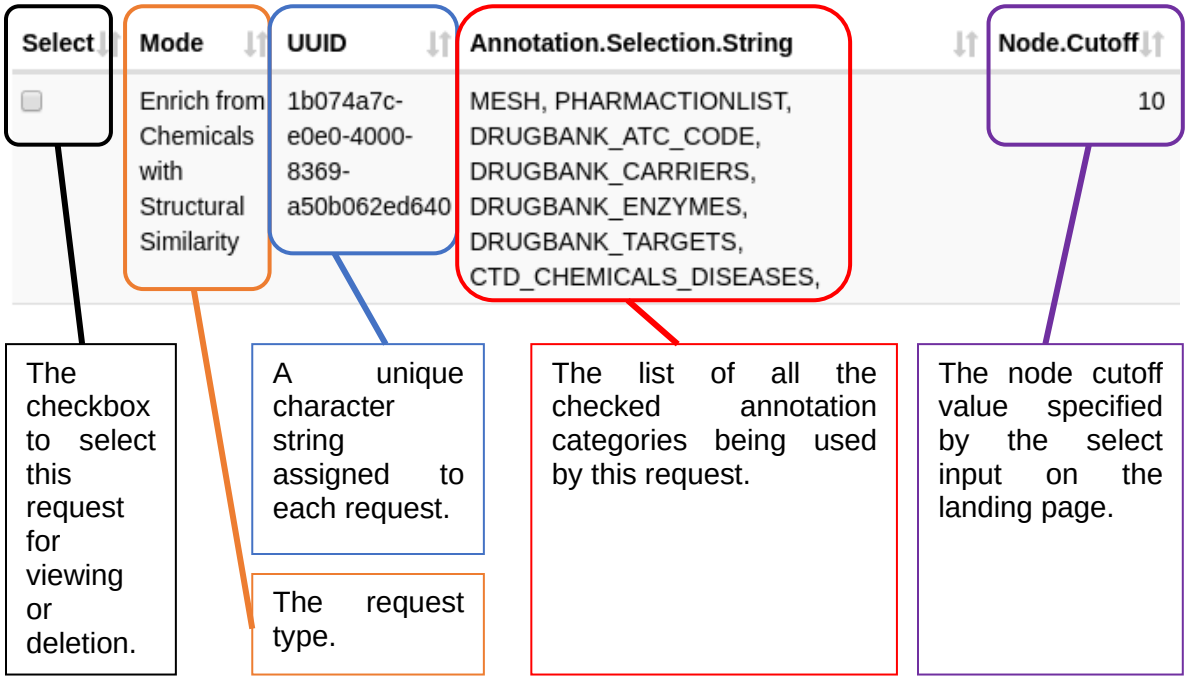
Whenever a request is submitted to the Tox21 Enricher API, whether it be to perform enrichment or view annotations, a record file detailing the details and parameters of the request is created in a temporary directory at `<project_root>/tox21enricher/www/tmp/transaction/`. For example, if you cloned the code to /home/user/tox21enricher/, the temporary directory will be at `/home/user/tox21enricher/tox21enricher/www/tmp/transaction/`. The records that are displayed in the table on the **View Results from Previous Request** page are pulled from the files in this directory. When a record is deleted, the corresponding record file in the temporary directory is deleted. Note that record files deleted in this manner cannot be recovered. Additionally, the files in the temporary directory's transaction folder are not deleted when clearing the local cache from the **Settings** menu.

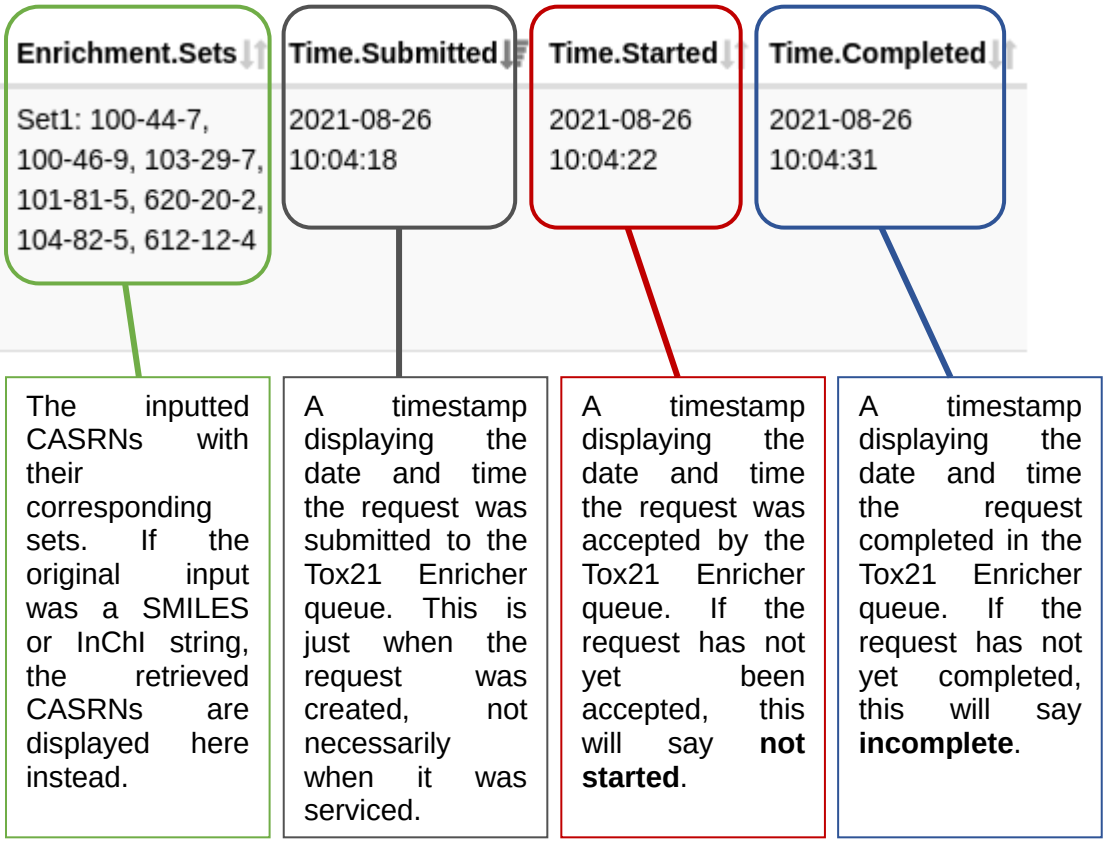
By default, none of the checkboxes are checked. Selecting more than one checkbox and clicking the **View results** button will present a warning, as you may not load the results for multiple requests at the same time.

On this page, the **View previous results** button in the sidebar will change to say **Perform enrichment**. Clicking this button will take you back to the landing page.



The columns of the table are as follows:





VII. Common Issues

Both the client and server code for Tox21 Enricher are still in continued development, so there are likely issues that may arise when trying to either perform certain tasks or set up the application. Known bugs in the code and planned features are documented on the project's GitHub repository at: https://github.com/hurlab/tox21enricher/issues. Other known issues not necessarily related to the code itself are documented in this section of the manual.

VII.1. Setup Issues

These are known issues that you may experience when attempting to set up Tox21 Enricher.

- **Problem installing "xlsx" R package** – The **xlsx** package requires a Java installation on your computer. Specifically, the package requires that you install the **rJava** R package as well. Even if you have a correctly set-up installation of Java, the installation of the rJava package may fail and ask you to run the `R CMD javareconf` command. Running this command may also fail. To fix this, you must explicitly include the path of your JDK installation when running `R CMD javareconf`, like so: `R CMD javareconf JAVA_HOME=/<path_to_your_java_installation>/`.
- **Problem installing RDKit PostgreSQL cartridge** – Even if you successfully install RDKit using Conda, Conda may fail to install the RDKit PostgreSQL cartridge due to package version conflicts. A workaround for this is to create a new Conda environment using `conda create -n <environment_name>`, delete or move the contents of the new environment's `/bin/` directory, install the RDKit PostgreSQL cartridge to that environment, then finally move the newly installed files from the new environment's `/bin/` directory into the RDKit environment's /bin/ directory.
- **Potential conflict with Python versions when using the "reticulate" R package** – The Tox21 Enricher API (plumber.R) sets the environment variable **RETICULATE_PYTHON** to the **python** path defined in the application's config.yml file. This may cause a conflict if you have another application running on the same machine using the **reticulate** package with a different version of Python. A fix or workaround for this may be implemented in a future update.

VII.2. Runtime Issues

This section will be updated in the future.

VIII. References

B. Bienfait and P. Ertl, JSME: a free molecule editor in JavaScript, J. Cheminformatics 5:24 (2013)

END OF THE USER'S MANUAL