OpenTox Deliverable Report 6.2

OpenTox Tutorials

Grant Agreement  Health–F5–2008–200787
Acronym  OpenTox
Name  An Open Source Predictive Toxicology Framework
Coordinator  Douglas Connect
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Document Type:</td>
<td>Deliverable Report</td>
</tr>
<tr>
<td>WP/Task:</td>
<td>WP6</td>
</tr>
<tr>
<td>Name</td>
<td>OpenTox Tutorials</td>
</tr>
<tr>
<td>Document ID:</td>
<td>OpenTox WP6 D6.2 report</td>
</tr>
<tr>
<td>Date:</td>
<td>Sept 30, 2010</td>
</tr>
<tr>
<td>Status:</td>
<td>Final Version</td>
</tr>
<tr>
<td>Organisation:</td>
<td>Douglas Connect</td>
</tr>
<tr>
<td>Contributors</td>
<td>Roman Affentranger (Author)</td>
</tr>
<tr>
<td></td>
<td>Barry Hardy (Review &amp; editing)</td>
</tr>
<tr>
<td>Distribution:</td>
<td>Final version</td>
</tr>
</tbody>
</table>

**Purpose of Document:**
This document describes the initial OpenTox tutorials.

**Document History:**
1 – RA (DC) authored
2 – BH (DC) reviewed and edited final version
Table of Contents

Table of Contents ............................................................................................................. 3
Table of Figures .................................................................................................................. 5
Acknowledgements ............................................................................................................. 7
Summary .............................................................................................................................. 8

1 Predict the Toxicity of a Compound .............................................................................. 9
  1.1 Introduction .................................................................................................................. 9
  1.2 ToxPredict Tutorial A: Predict the Toxicity of a Chemical in the Database .......... 9
  1.3 ToxPredict Tutorial B: Predict the Toxicity of a New Chemical (not in the Database) 12

2 Build a Predictive QSAR Model and Validate it .......................................................... 13
  2.1 Introduction ................................................................................................................ 13
  2.2 ToxCreate step 1: Create .......................................................................................... 13
  2.3 ToxCreate Step 2: Inspect ........................................................................................ 14
  2.4 ToxCreate Step 3: Predict ......................................................................................... 15

3 Build a Model Based on In Vitro Data ......................................................................... 16
  3.1 Introduction ................................................................................................................ 16
  3.2 Inspect US EPA’s ToxCast Data ............................................................................... 16
  3.3 Select in vitro Assays based on Correlations to an in vivo Endpoint ....................... 16
  3.4 Use the ToxCreate Web Application to Create a Model ......................................... 17
     3.4.1 Create the data file .............................................................................................. 17
     3.4.2 Build the Model .................................................................................................. 18
     3.4.3 Make a prediction .............................................................................................. 19
  3.5 Create a model using the OpenTox API .................................................................... 20
     3.5.1 Locate data on the ambit web service ................................................................. 20
     3.5.2 Create a model on the command line using cURL ........................................... 21
     3.5.3 Make a prediction .............................................................................................. 21

4 Validate your QSAR Model and Create a Report (for Developers) ......................... 23
  4.1 Introduction ................................................................................................................ 23
  4.2 Prerequisites .............................................................................................................. 23
  4.3 Validation Examples ................................................................................................. 23
     4.3.1 Validate an algorithm on a dataset via training–test–split ................................ 23
     4.3.2 Validate an algorithm on a dataset via bootstrapping ..................................... 24
  4.4 Validation Reports ................................................................................................... 24
4.4.1 Create validation report from validation................................................................. 24
4.4.2 Create a QMRF Report .............................................................................................. 25
4.5 Further validation techniques....................................................................................... 25

5 Query and Access Toxicity Data.................................................................................... 26
  5.1 Introduction...................................................................................................................... 26
  5.2 Inspect Data of the ISSMIC Dataset ........................................................................... 26
  5.3 Retrieve Data in Selected Formats ............................................................................. 27
  5.4 Find Similar Structures and Browse the Available Data for a Given Chemical............ 27

6 Drug Discovery Predictive Toxicology Application I: Prioritizing compounds........... 28
  6.1 Introduction...................................................................................................................... 28
  6.2 Step 1: Predicting Oral Toxicity..................................................................................... 28
  6.3 Step 2: Analyse Cytotoxicities of the Cramer Class I compounds............................... 30
  6.4 Step 3: Predicting the Mutagenicity of the Selected Compounds............................... 32
  6.5 Step 4: Predicting Sites of Cytochrome P450 Metabolism .......................................... 34

7 Drug Discovery Predictive Toxicology Application II: Building a Model to Predict
  Kinase Inhibitor Activity.................................................................................................. 37
  7.1 Introduction...................................................................................................................... 37
  7.2 Activity B: Selecting a subset to create a model with ToxCreate............................... 37

8 Set up an OpenTox Data Service .................................................................................... 40
  8.1 Introduction...................................................................................................................... 40
  8.2 Setting up a Windows System....................................................................................... 41
    8.2.1 Installing the Java Runtime Environment (JRE) and Java Development Kit (JDK)........ 41
    8.2.2 Installing MySQL..................................................................................................... 41
    8.2.3 Installing Tomcat ..................................................................................................... 41
  8.3 Setting Up a CentOS 5 Linux System ........................................................................... 43
    8.3.1 Installing MySQL..................................................................................................... 43
    8.3.2 Installing the Java Runtime Environment and Development Kit............................. 43
    8.3.3 Installing Tomcat ..................................................................................................... 43
  8.4 Download and Deploy AMBIT 2.0 .............................................................................. 48

9 Conclusions ...................................................................................................................... 50
**Table of Figures**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Starting page of ToxPredict</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Structure entering by name, identifier, SMILES, etc.</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>The structures found for the search string are presented for verification</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>The list of endpoints currently available in ToxPredict. The user can select any number of endpoints.</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>The “Run” page indicates which computations are currently running, and which have completed already.</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>The results are displayed for each of the selected endpoints.</td>
<td>11</td>
</tr>
<tr>
<td>7</td>
<td>The structure upload tab of ToxPredict</td>
<td>12</td>
</tr>
<tr>
<td>8</td>
<td>Starting page of ToxCreate</td>
<td>13</td>
</tr>
<tr>
<td>9</td>
<td>Specify a name for the model you are about to create. Unless you plan on choosing regression for the model building, i.e. for a classification model, a unit needs not be specified.</td>
<td>13</td>
</tr>
<tr>
<td>10</td>
<td>How to format a ToxCreate training data set in Excel</td>
<td>14</td>
</tr>
<tr>
<td>11</td>
<td>How to format a ToxCreate training data set as CSV file</td>
<td>14</td>
</tr>
<tr>
<td>12</td>
<td>The “Inspect” tab of ToxCreate provides information about the created model</td>
<td>15</td>
</tr>
<tr>
<td>13</td>
<td>Structures can be drawn or searched for in ToxCreate’s Predict tab</td>
<td>15</td>
</tr>
<tr>
<td>14</td>
<td>Extensive details are provided for predictions</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>The ToxCast correlation spreadsheet</td>
<td>17</td>
</tr>
<tr>
<td>16</td>
<td>Select the data of your chosen assay in one of the files containing the original data</td>
<td>17</td>
</tr>
<tr>
<td>17</td>
<td>Build your data file by combining assay data (column B) with chemical structure data (SMILES, column A)</td>
<td>18</td>
</tr>
<tr>
<td>18</td>
<td>Starting page of ToxCreate</td>
<td>18</td>
</tr>
<tr>
<td>19</td>
<td>Results page of ToxCreate</td>
<td>19</td>
</tr>
<tr>
<td>20</td>
<td>The “Predict” tab of ToxCreate allows to run predictions using the created model</td>
<td>19</td>
</tr>
<tr>
<td>21</td>
<td>ToxCreate provides detailed results on the predictions run with the created model</td>
<td>20</td>
</tr>
<tr>
<td>22</td>
<td>Find your chosen endpoint in the OpenTox data service</td>
<td>20</td>
</tr>
<tr>
<td>23</td>
<td>Searching for the ISSMIC dataset on <a href="http://apps.idealconsult.net:8080/ambit2/dataset">http://apps.idealconsult.net:8080/ambit2/dataset</a></td>
<td>26</td>
</tr>
<tr>
<td>24</td>
<td>Download icons for the various available formats</td>
<td>27</td>
</tr>
<tr>
<td>25</td>
<td>Searching for similar structures by clicking on the first magnifying glass icon.</td>
<td>27</td>
</tr>
<tr>
<td>26</td>
<td>Browse available data on a given compound using the “All available feature values” link</td>
<td>27</td>
</tr>
<tr>
<td>27</td>
<td>Launching a predictive model on a chosen dataset URL from within the AMBIT data service</td>
<td>28</td>
</tr>
<tr>
<td>28</td>
<td>The results of the predictive model are delivered as a URL</td>
<td>29</td>
</tr>
<tr>
<td>29</td>
<td>Distribution of the Cramer classes in the TCAMS dataset, depicted using the OpenTox chart generation service</td>
<td>29</td>
</tr>
<tr>
<td>30</td>
<td>Browsing the results</td>
<td>30</td>
</tr>
<tr>
<td>31</td>
<td>Retrieving the feature URL</td>
<td>31</td>
</tr>
</tbody>
</table>
Figure 32 A column is added to the compounds with the feature_uris[] directive ........................................ 31
Figure 33 Another column is added to the compounds with the feature_uris[] directive ................................. 32
Figure 34 Selecting the Benigni/Bossa rulebase for mutagenicity and carcinogenicity ................................. 33
Figure 35 Search results enhanced with columns for carcinogenicity predictions .................................... 34
Figure 36 Running SmartCYP on the malaria box ......................................................................................... 35
Figure 37 Example SmartCYP output ........................................................................................................... 35
Figure 38 SmartCYP color code .................................................................................................................. 36
Figure 39 Example SOME output ............................................................................................................... 36
Figure 40 SOME color code ....................................................................................................................... 36
Figure 41 The list of antimalarial datasets on http://pirin.uni-plovdiv.bg:8080/malaria/dataset ..................... 37
Figure 42 Search results for “Ser/Thr protein kinase” on the TCAMS antimalarial dataset ...................... 38
Figure 43 SourceForge page of AMBIT 2.0 ............................................................................................... 40
Figure 44 The Tomcat icon in the Windows7 taskbar .............................................................................. 42
Figure 45 Configuring the Tomcat startup type ......................................................................................... 42
Figure 46: Tomcat “home” page ............................................................................................................... 47
Figure 47: Tomcat “manager” page ........................................................................................................... 48
Figure 48: The AMBIT 2.0 page ................................................................................................................. 49
Acknowledgements

Research Funding

Project Partners
Douglas Connect (DC), In Silico Toxicology (IST), Ideaconsult (IDEA), Istituto Superiore di Sanita’ (ISS), Technical University of Munich (TUM), Albert Ludwigs University Freiburg (ALU), National Technical University of Athens (NTUA), David Gallagher (DG), Institute of Biomedical Chemistry of the Russian Academy of Medical Sciences (IBMC), Seascape Learning (SL), Jawaharlal Nehru University (JNU), Fraunhofer Institute for Toxicology & Experimental Medicine (ITEM).

Advisory Board

Correspondence
Dr. Barry Hardy, OpenTox Project Coordinator and Director, Community of Practice & Research Activities, Douglas Connect, Baermeggenweg 14, 4314 Zeiningen, Switzerland

Email: barry.hardy –(at)– douglasconnect.com
Summary

This document represents a collection of tutorial materials on several OpenTox topics including walk-throughs of the two end user prototype applications ToxPredict and ToxCreate, illustration of the use of validation and reporting services applied to predictive toxicology models, the application of OpenTox facilities in a drug discovery workflow, and detailed instructions on how to get a system set up to host an OpenTox data service.

The tutorial example on the prototype OpenTox application ToxPredict ([www.toxpredict.org](http://www.toxpredict.org)) accepts chemical structures and names as input from the user and generates toxicity reports based on various pre-calibrated toxicity models and existing toxicity data.

In the ToxCreate ([www.toxcreate.org](http://www.toxcreate.org)) tutorial, the user provides a dataset of chemical structures and target variable data. ToxCreate subsequently builds and validates a Quantitative Structure-Activity Relationship (QSAR) predictive toxicology model. The user receives a reporting on details of model results and model predictions which they may examine, and including using the model for new predictions.

In the *in vitro* data model building tutorial, a predictive model is built based on *in vitro* data using OpenTox web services. Several models can be built and inspected based on application to the US EPA ToxCast dataset.

The tutorial on web validation and reporting web services, which are also behind the end user applications ToxPredict and ToxCreate, shows how cURL calls can be used to validate a predictive model or an algorithm. A number of different validation methods are used, including K-fold split, training-test-split and bootstrapping. Furthermore, QMRF reports are generated and visualized using the QMRF Editor web start application.

The objective of the ISSMIC data analysis tutorial is to illustrate searching facilities and data visualization tools in the OpenTox framework, specifically in the context of *in vivo* micronucleus mutagenicity assays contained within ISSMIC, a curated database, containing critically-selected information on chemical compounds tested with the assay.

A tutorial example of a predictive toxicology application in drug discovery is provided using the data on anti-malarial compounds made available at the ChEMBL Neglected Tropical Disease (NTD) archive ([www.ebi.ac.uk/chemblntd/](http://www.ebi.ac.uk/chemblntd/)). The anti-malarial compounds are prioritized based on a strongly conservative model for predicting oral toxicity. Experimentally-determined cytotoxicities against human cells of the compounds predicted to be safe are further examined, and their mutagenicities predicted. Sites of cytochrome P450 metabolism are predicted for selected compounds with no mutagenicity alerts, low human cytotoxicity, but high anti-malarial activity.

A tutorial is provided to guide the user through the setup of an OpenTox data service based on the download of the AMBIT software and its subsequent installation either on Windows or Linux.

All tutorials and their updates are made available online under [www.opentox.org/tutorials](http://www.opentox.org/tutorials).
1 Predict the Toxicity of a Compound

1.1 Introduction
The objective of these tutorials is to demonstrate the prototype OpenTox application ToxPredict (www.toxpredict.org) that accepts chemical structures and names as input and automatically generates a toxicity report based on various precalibrated toxicity models.

ToxPredict provides a web-based interface for predicting the toxicity of individual chemicals. Users can either search for a compound in the OpenTox prototype database, which currently includes quality-labelled data for 163,122 chemicals (including all chemicals currently registered with REACH) grouped in 2,409 datasets, or can upload their own chemical structure in SDF format. ToxPredict runs the selected calculations automatically using a collection of distributed computational services. Its initial version currently includes eighteen validated models addressing 15 different endpoints, and is extensible.

ToxPredict has been designed to be easy-to-use and self-explanatory. Therefore, the need for extensive tutorials should not be great, and the examples presented here are kept brief.

1.2 ToxPredict Tutorial A: Predict the Toxicity of a Chemical in the Database
ToxPredict searches its database for chemical structures entered as chemical names, SMILES strings, CAS numbers or via an integrated 2D chemical structure drawing editor. The following example illustrates using the chemical name.

1. Using a web browser, navigate to the starting page of www.toxpredict.org (see Figure 1)
2. Click "NEXT"
3. Type a chemical name in the text box, e.g. "benzene" (or CAS number, SMILES string, …) (see Figure 2)
4. Click “NEXT” (to see if the search was successful. If not, go back to step 3) (see Figure 3)

5. Click “NEXT” again (to view the list of endpoints), given the search in step 4 was successful

6. Click to select any of the 18 boxes in the left column (multiple selections are possible) (see Figure 4)

7. Click “NEXT” to start the calculations. Calculations that are currently running are indicated with 🍋, those that have finished already with ✓ (see also Figure 5)
Figure 5 The “Run” page indicates which computations are currently running, and which have completed already.

8. Click “NEXT” again (to view the results) (see Figure 6)

Figure 6 The results are displayed for each of the selected endpoints.
1.3 ToxPredict Tutorial B: Predict the Toxicity of a New Chemical (not in the Database)

To predict the toxicity of a new chemical that is not in the current OpenTox data infrastructure, you will need to upload the structure file as an SDF (“SD File”) as follows:

1. Using a web browser, navigate to www.toxpredict.org (see Figure 1)
2. Click “NEXT”
3. Click the “Upload” tab (Figure 7)
4. Click “Browse…”
5. Navigate to and select your SD file
6. Click “Open”
7. Click “NEXT” (to validate your input structure, a 2D drawing of the structure is presented, see Figure 3)
8. Click “NEXT” again (to view the list of endpoints, see Figure 4)
9. Click to select any of the 18 boxes in the left column (multiple selections are possible, see Figure 4)
10. Click “NEXT” to start the calculations (see Figure 5)
11. Click “NEXT” again (to view the results, see Figure 6)
2 Build a Predictive QSAR Model and Validate it

2.1 Introduction

The objective of this tutorial is to demonstrate the prototype OpenTox application ToxCreate (www.toxcreate.org). Based on a provided dataset, ToxCreate builds and validates a Quantitative Structure–Activity Relationship (QSAR) model.

Similar to ToxPredict, ToxCreate has been designed to be easy-to-use and self-explanatory. Therefore, the need for extensive tutorials should now be necessary, and the example presented here is kept fairly brief. The requirements for this tutorial are: a web browser with Java script activated, Java installed and a working internet connection.

2.2 ToxCreate step 1: Create

1. Navigate to the ToxCreate starting page at www.toxcreate.org (see Figure 8) and follow the instructions on the web page. Click on bold and purple text to get further information on any topic.
2. Specify a name for the endpoint for which you plan to create a QSAR model. Specify the unit of your input data if you want to use regression for model building. (Figure 9)
3. Upload training data in Excel or CSV format (make sure you prepare your training data file according to the rules given at www.toxcreate.org/help/ (see also Figure 10 and Figure 11).
Figure 10 How to format a ToxCreate training data set in Excel

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC(=O)Nc1cc(O)cc1</td>
<td>1</td>
</tr>
<tr>
<td>O=c1[nH]cc2[nH]cccc12</td>
<td>1</td>
</tr>
<tr>
<td>CCCCCNc1ccc(cc1Oc2cccccc2)(S(=O)(=O)N)c(=O)O</td>
<td>1</td>
</tr>
<tr>
<td>CC(C)(C)NCC(O)COC1cccc2NC(=O)CCc12</td>
<td>1</td>
</tr>
<tr>
<td>CN(C)CCCCC1(OCc2cc(C#N)cccc21)c3cccc(F)c3c3</td>
<td>1</td>
</tr>
<tr>
<td>CCC(CC)CCN1c(=O)CN=C(C2CCCCCC2F)c3ccc(C)cccc13</td>
<td>0</td>
</tr>
<tr>
<td>CCN(CC)CC(=O)Nc1c(C)cccc1C</td>
<td>0</td>
</tr>
<tr>
<td>CC(C)(C)NCC(O)COC1cccc2CC(O)(O)Cc12</td>
<td>0</td>
</tr>
<tr>
<td>CN1CCCCC1c2ccccnc2</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 11 How to format a ToxCreate training data set as CSV file

4. Click the “Create Model” button to start the model building and validation calculations

2.3 ToxCreate Step 2: Inspect

On this page you get a brief summary of all models with validation results (see Figure 12). Find your model by name and click on bold and purple links to view or download more detailed information (e.g. the feature dataset as an XML file, or a detailed validation report as a PDF document).
2.4 ToxCreate Step 3: Predict

1. Click on the “Predict” tab

2. On the “Predict” the user can draw a structure, or enter a name (or InChI, SMILES, CAS number) (see Figure 13).

3. Select one or more prediction models (find the one just created by its name), and click on “Predict”

4. Check the results and click on “Details” for prediction details such as similar compounds, relevant substructures, etc. (see Figure 14).

5. Click on bold and purple topics to get further information.
3 Build a Model Based on In Vitro Data

3.1 Introduction

The objective of this tutorial is to illustrate searching facilities, data representation, and the API of the OpenTox framework, specifically in the context of using in vitro data for building predictive toxicity models. In the tutorial, a predictive model is built based on in vitro data with OpenTox web services using a graphical user interface (GUI) as well as via the command line. Several models can be built and inspected.

For this tutorial the following software is needed:
- Microsoft Office (preferably version 2003 or later) is needed, or a similar office suite
- A web browser (e.g. Mozilla Firefox, www.mozilla.com/firefox)
- An important command-line tool used in this tutorial is called “cURL” (curl.haxx.se). On many Linux systems, cURL can be easily installed from a package repository using a standard package manager. It can also be downloaded from curl.haxx.se/download.html. Under Windows, there are two options for using cURL: 1) installing cURL natively, preferable using this version: www.gknw.net/mirror/curl/win32/curl-7.21.1-devel-mingw32.zip, or 2) installing the VMWare Player (www.vmware.com/products/player/) and running a small Linux environment (www.maunz.de/opentox/dsl-4.1.zip) under Windows (after installing VMWare Player and unpacking the dsl-4.1.zip file, just double-click the dsl-4.1.vmx file).

In addition, the ToxCast Data Utilities need to be downloaded from www.maunz.de/opentox/ToxCastDataUtils.zip. The archive contains data in a convenient Excel sheet form. The .zip archive needs to be expanded. It contains the files referenced in this tutorial.

3.2 Inspect US EPA's ToxCast Data

Open the file “ToxCastDataUtils/10 Tong Liver Toxicity TDAS.pdf”. The presentation addresses several aspects of the liver toxicity data in ToxCast (www.epa.gov/ncct/ToxCast). Take 15 minutes time to inspect the characteristics of the in vitro assay data.

3.3 Select in vitro Assays based on Correlations to an in vivo Endpoint

Open the file “ToxCastDataUtils/ToxCastCorrelations_20100708.xlsx”. This Excel file analyses all correlations between in vivo endpoints and in vitro assays present in the ToxCast dataset (phase 1), for qualitative (nominal) as well as quantitative (continuous) values (see Figure 15).

1. Under Assay_2, select a liver toxicity endpoint in Rat or Mouse, e.g. CHR_Rat_LiverProliferativeLesions.
2. Under Assay_1, select an assay among the top correlated ones (red), e.g. ATG_PPARg_trans.

Note: You can find the original data in the files
- ToxCastDataUtils/ToxCastAssays_20100708.xlsx (Assays, LOEL values)
- ToxCastDataUtils/ToxCastAssays_discrete_20100708.xlsx (Assays, discretized active/inactive)
- ToxCastDataUtils/ToxCastEndpoints_20100708.xlsx (Endpoints, LOEL values)
- ToxCastDataUtils/ToxCastEndpoints_discrete_20100708.xlsx (Endpoints, discretized active/inactive)
3.4 Use the ToxCreat Web Application to Create a Model

3.4.1 Create the data file

Fill the data into an Excel Sheet: open an original data file, such as ToxCastDataUtils/ToxCastAssays_discrete_20100708.xlsx. Select the column with your assay. (see Figure 16).

Create a new Excel file and insert the column into the first sheet, second column. Remove the first row (header). Open ToxCastDataUtils/ToxCast_Chemicals_20100708.xlsx and select column E (SMILES codes). Copy it to the first column of the first sheet of your new workbook (Figure 17).
Build your data file by combining assay data (column B) with chemical structure data (SMILES, column A). Save the workbook.

### 3.4.2 Build the Model

Direct your browser to [www.toxcreate.org](http://www.toxcreate.org). Enter a name for your data (preferably a short form of the chosen assay’s name, such as `ATG_PPARG_trans`). In the file upload box, select the workbook you created in the previous step. (see Figure 18).

Create and evaluate models to predict toxicity.

Click on "Create model" and wait until computation has finished.

---

[1] You can also try [ot-test.in-silico.ch](http://ot-test.in-silico.ch) for the latest development versions.
The model is validated by 10-fold crossvalidation. Click on “Show” next to “Detailed report” for more detailed information provided by the OpenTox reporting service (see Figure 19). When you are done inspecting the detailed report, select the “Predict” tab.

### 3.4.3 Make a prediction

Draw a chemical or identify it in the text box. Hit predict.

On the next page, view the prediction results and related details on confidence (not a probability), nearest neighbours and relevant substructures (see Figure 21).
3.5 Create a model using the OpenTox API

3.5.1 Locate data on the ambit web service

The ToxCast data can be found online at apps.ideaconsult.net:8080/ambit2/dataset/.

- Search for the datasets analysed in step 2 (see section 3.2). Assay providers are spelled out here and are easily recognizable: ATG = Attagene, BSK = Bioseek, CLM = Cellumen, … For example, the dataset by ATG is available at apps.ideaconsult.net:8080/ambit2/dataset/104.

- Find the endpoint, e.g. ATG_PPARg_trans for ATG, in the columns. In OpenTox terms, it is represented as a feature. Point with the mouse to the columns header: The address is apps.ideaconsult.net:8080/ambit2/feature/22333.
3.5.2 Create a model on the command line using cURL

Open a command shell (under Windows, you will find it under "All Programs -> Accessories -> Command Prompt"). We will now learn a $k$-nearest neighbour model for the dataset from the above process. First, we will gather some information about the model (if you are using the Windows Command Prompt, replace "curl" in the following command lines by the path to the cURL executable on your system, e.g. C:\Users\Username\Programs\curl-7.21.0-win64\curl.exe):

```bash
curl http://opentox.informatik.tu-muenchen.de:8080/OpenTox-dev/algorithm/kNNregression
```

The output is rdf/xml, the OpenTox standard data format. This rich representation contains links to formal logical concepts and definitions, making the OpenTox framework explicit to machines. As a consequence, this enables logic inference and reasoning.

For example, the description tells you that this model is an exposure of WEKA’s kNN-model. Following that, it lists the parameters that the model can be configured with, e.g. whether crossvalidation should be used to select parameter $k$, dataset uri, nearest neighbour search algorithm, etc.

We will now feed it the data we just selected. Execute the following code (replace with your own data):

```bash
http://opentox.informatik.tu-muenchen.de:8080/OpenTox-dev/algorithm/kNNregression
```

Since model generation can take some time, this will return a URI for the running task. Use it to obtain the model URI, once it has finished (just keep trying).

```bash
curl -i -H "accept: text/uri-list" http://opentox.informatik.tu-muenchen.de:8080/OpenTox-dev/task/<task-id>
```

Once you have the model URI, locate the compound we had in the GUI example. To do so, direct your browser to [http://apps.ideaconsult.net:8080/ambit2/query/similarity?search=c1ccccc1Oc2ccccc2&threshold=0.9](http://apps.ideaconsult.net:8080/ambit2/query/similarity?search=c1ccccc1Oc2ccccc2&threshold=0.9) and search for your structure (here: "Clc2ccc(c1ccccc1)cc2", which you can find at [http://apps.ideaconsult.net:8080/ambit2/compound/16479](http://apps.ideaconsult.net:8080/ambit2/compound/16479)). If your structure is not already there select the most similar one.

3.5.3 Make a prediction

Use the model URI to predict the compound:

```bash
curl -i -X POST -d "dataset_uri=http://apps.ideaconsult.net:8080/ambit2/compound/16479"
http://opentox.informatik.tu-muenchen.de:8080/OpenTox-dev/model/TUMOpenToxModel_kNN_98
```

Once again, you will be returned a task uri. Keep querying it to receive a dataset URI:

```bash
curl -i -H "accept: text/uri-list" http://opentox.informatik.tu-muenchen.de:8080/OpenTox-dev/task/<task-id>
```

This will return a dataset URI:
Note that the predicted value is 1000000.0, which corresponds to *inactive* (as in the GUI example).

The above code gave you plain text information for purposes of this tutorial. You can also try the OpenTox standard format rdf/xml:

```
```
4 Validate your QSAR Model and Create a Report (for Developers)

4.1 Introduction

The objective of this tutorial is to demonstrate how to handle the validation and reporting services for predictive models using cURL calls. The web validation and reporting web services that are also behind the end user applications ToxPredict and ToxCreate are contacted using cURL calls to validate a model or an algorithm. A number of different validation methods are used, such as K-fold split, training–test–split or bootstrapping. Furthermore, QMRF reports are generated and visualized using the QMRF Editor web start application. This tutorial is mostly aimed at developers who want to get a picture of how these services work in detail.

4.2 Prerequisites

It is probably of advantage to have the web page with the OpenTox API definitions for the validation services (http://www.opentox.org/data/documents/development/validation/Validation/) open in a web browser.

An important command-line tool used in this tutorial is called “cURL” (http://curl.haxx.se). On many linux systems, cURL can be easily installed from a package repository using a standard package manager. It can also be downloaded from http://curl.haxx.se/download.html. Under Windows, there are two options for using cURL: 1) installing cURL natively, preferable using this version: http://www.gknw.net/mirror/curl/win32/curl-7.21.1-devel-mingw32.zip, or 2) installing the VMWare Player (http://www.vmware.com/products/player/) and running a small Linux environment (http://www.maunz.de/opentox/dsl-4.1.zip) under Windows (after installing VMWare Player and unpacking the dsl-4.1.zip file, just double-click the dsl-4.1.vmx file).

Also needed is Java 6, with web start enabled: (http://www.oracle.com/technetwork/java/javase/downloads/index.html).

4.3 Validation Examples

First we want to list all available validations. To do that, it is necessary, to execute the following command in a terminal window (under Windows, you will find it under “All Programs –> Accessories –> Command Prompt”; replace “curl” in the following command lines by the path to the cURL executable on your system, e.g. C:\Users\Username\Programs\curl-7.21.0-win64\curl.exe):

```
curl http://opentox.informatik.uni-freiburg.de/validation
```

4.3.1 Validate an algorithm on a dataset via training–test–split

This will create a new validation object. A model is constructed by splitting a dataset into two parts: one for learning a model and one for testing, i.e. predicting and estimating the performance of the constructed model. Splitting the dataset is done in random fashion. One can also define the ratio for splitting into training and test, the default is 67% training and 33% test.

```
curl -X POST
-d algorithm_uri="http://opentox.informatik.uni-freiburg.de/algorithmlazar"
-d dataset_uri="http://opentox.informatik.uni-freiburg.de/dataset/1"
-d prediction_feature="http://localhost/toxmodel/feature%23Hamster%2520Carcinogenicity%2520(DSSTOX/CPDB)"
-d algorithm_params="feature_generation_uri=http://opentox.informatik.uni-freiburg.de/algorithmfminer"
-d split_ratio=0.9 -d random_seed=2
http://opentox.informatik.uni-freiburg.de/validation/training_test_split
```
Validating algorithms can be a time consuming task. Therefore the result of the above cURL call is a task URI. To query the status of the task URI, enter the following command in the terminal (where the term `<TASK-ID>` should be replaced with the correct task ID):

```
curl http://opentox.informatik.uni-freiburg.de/task/<TASK-ID>
```

As soon as the task is completed, your validation is available. The validation URI can be found in the resultURI property of the task:

```
---
:uri: http://opentox.informatik.uni-freiburg.de/task/<id>
:hasStatus: Completed
:resultURI: http://opentox.informatik.uni-freiburg.de/validation/<VALIDATION-ID>
```

Use cURL to get a closer look at your validation result:

```
curl http://opentox.informatik.uni-freiburg.de/validation/<VALIDATION-ID>
```

Just like the task result, the validation result is formatted in YAML, a markup language that is human readable. You could look at the statistics such as area-under-roc or confusion matrix values.

### 4.3.2 Validate an algorithm on a dataset via bootstrapping

Bootstrapping is a machine learning technique that splits a dataset into training and test set via "sampling with replacement".

```
curl -X POST
-d algorithm_uri="http://opentox.informatik.uni-freiburg.de/algorithm/lazar"
-d dataset_uri="http://opentox.informatik.uni-freiburg.de/dataset/1"
-d prediction_feature="http://localhost/toxmodel/feature%23Hamster%2520Carcinogenicity%2520(DSSTOX/CPDB)"
-d algorithm_params="feature_generation_uri=http://opentox.informatik.uni-freiburg.de/algorithm/fminer"
-d random_seed=2
http://opentox.informatik.uni-freiburg.de/validation/bootstrapping
```

Again, this cURL call returns a task. As soon as the bootstrapping validation is finished, you validation is provided as before.

### 4.4 Validation Reports

Validation reports present validation results in a nice human readable format. This cURL call gives you a list of available validation reports:

```
curl http://opentox.informatik.uni-freiburg.de/validation/report/validation
```

#### 4.4.1 Create validation report from validation

This cURL call will create a report for the validation that you just performed before. Choose which validation you like (training–test split or bootstrapping).

```
curl -X POST
-d validation_uris="http://opentox.informatik.uni-freiburg.de/validation/<VALIDATION-ID>"
http://opentox.informatik.uni-freiburg.de/validation/report/validation
```

A report is created that is wrapped in a task URI as above.

You can access your report in YAML–format with the following cURL call (this time you have to specify YAML as requested format manually, as the default report format is ‘text/html’)

```
curl http://opentox.informatik.uni-freiburg.de/task/<TASK-ID>
```
4.4.2 Create a QMRF Report

QMRF (QSAR Model Reporting Format) is a harmonized template supported by the European Commission Joint Research Centre (EC JRC) for summarizing and reporting key information on QSAR models.

A QMRF is created for a particular QSAR model. To this end, you can build a model on the complete dataset we have been using so far with the following cURL call:

```bash
```

Use the new model to build a QMRF report via:

```bash
<<< http://opentox.informatik.uni-freiburg.de/validation/report/validation/id_i
```

This report can be accessed via cURL:

```bash
curl -X POST -d model_uri=http://opentox.informatik.uni-freiburg.de/model/<MODEL-ID> http://opentox.informatik.uni-freiburg.de/validation/reach_report/QMRF
```

Alternatively, use the QMRF editor to edit this report by visiting the address with your browser:

```bash
http://opentox.informatik.uni-freiburg.de/validation/reach_report/QMRF/<REPORT-ID>/editor
```

4.5 Further validation techniques

For additional technical description and further examples including:

- how to validate a model on a test dataset
- how to validate an algorithm on a training and test dataset
- how to create a validation object by comparing feature values
- how to validate an algorithm on a dataset via k-fold cross-validation

please consult the examples web page located at:

```bash
http://opentox.informatik.uni-freiburg.de/validation/examples
```
5 Query and Access Toxicity Data

5.1 Introduction

The objective of this tutorial is to illustrate searching facilities and data visualization tools in the OpenTox framework, specifically in the context of in vivo micronucleus mutagenicity assay in rodent’s data of the ISSMIC database. ISSMIC is a curated database, containing critically-selected information on chemical compounds tested with the in vivo micronucleus mutagenicity assay in rodents. In vivo mutagenicity testing is ranked three (well ahead of rodent carcinogenicity) as an animal consuming experimentation. Results in bone marrow cells, peripheral blood cells and splenocytes for male/female rat/mouse are reported. The data are collected from publicly available datasets (Toxnet, NTP) and from the Leadscape FDA CRADA Toxicity Database. ISSMIC provides both biological calls and chemical structures, and is the basis for establishing sound read-across and QSAR risk assessment. The ISSMIC data is available from OpenTox data services.

5.2 Inspect Data of the ISSMIC Dataset

1. Open the following URL in your web browser: http://apps.ideaconsult.net:8080/ambit2/dataset
2. Enter “ISSMIC_v2a_151_2Apr09” in the search box and press the “Search” button.
3. Click on the returned URL and take your time to browse the information available in the ISSMIC dataset.
4. Click on a given property value of interest to retrieve a list of chemicals with similar property values.
5.3 Retrieve Data in Selected Formats

Download and inspect a selected subset (or the entire dataset) in SDF, CML, SMILES, URI list, XML, PDF, CSV, plain text, ARFF and/or RDF format. These downloads are accessible through the following icons:

![Download Icons](image)

Figure 24 Download icons for the various available formats

5.4 Find Similar Structures and Browse the Available Data for a Given Chemical

Select a chemical of interest and perform a search for similar compounds (e.g. Tanimoto > 0.85) in the entire OpenTox database by clicking on the top magnifying glass icon on the left of the 2D drawing of the chemical (see Figure 25).

![Chemical Drawing](image)

Figure 25 Searching for similar structures in the OpenTox database (by clicking on the first magnifying glass icon).

Alternatively, you can search the OpenTox database using your selected chemical as a substructure by clicking on the second magnifying glass icon.

Browse available data for a selected compound by clicking on its 2D drawing, and on the following page on the "All available feature values" link.

![Compound Drawing](image)

Figure 26 Browse available data on a given compound using the "All available feature values" link.
6 Drug Discovery Predictive Toxicology Application I: Prioritizing compounds

6.1 Introduction

An example of a predictive toxicology application in drug discovery is provided here using the data on antimalarial compounds made available at the ChEMBL Neglected Tropical Disease (NTD) archive (www.ebi.ac.uk/chemblntd/). In this tutorial example, the antimalarial compounds are prioritized based on a strongly conservative model for predicting oral toxicity. Experimentally-determined cytotoxocities against human cells of the compounds predicted to be safe are further examined, and their mutagenicities predicted. Sites of cytochrome P450 metabolism are predicted for selected compounds with no mutagenicity alerts, low human cytotoxicity, but high anti-malarial activity.

6.2 Step 1: Predicting Oral Toxicity

Go to the list of antimalarial datasets at http://pirin.uni-plovdiv.bg:8080/malaria/dataset. We’ll first predict oral toxicity for the TCAMS dataset. Start by clicking on the TCAMS dataset link. The URL in the browser should read http://pirin.uni-plovdiv.bg:8080/malaria/dataset/12. You can browse the compounds (See Figure 27.)

In a new tab of your browser, go to the list of OpenTox models at http://pirin.uni-plovdiv.bg:8080/malaria/model (or follow the “Models” link at the top of the page listing the datasets). To predict oral toxicity we will use the “Toxtree Cramer rules” model. Clicking on the Cramer rules link will open its page. OpenTox models accept dataset URLs as input (instead of file names). Enter (or paste) the TCAMS URL (“http://pirin.uni-plovdiv.bg:8080/malaria/dataset/12”) into the text box. Click “Predict” which will launch calculations. You might click on the links to find out if the calculations are completed. When completed, clicking on the link will lead to a dataset with the results (Figure 28).
The results of the predictive model are delivered as a URL.


- Class I substances are simple chemical structures with efficient modes of metabolism suggesting a low order of oral toxicity;
- Class III substances are those that permit no strong initial presumption of safety, or may even suggest significant toxicity or have reactive functional groups; and finally,
- Class II are intermediate. This model is very conservative and places most of the compounds in Class III.

During this exercise, we'll look for compounds of low toxicity (Class I) and high antimalarial activity. There are a small number of Class I compounds, the distribution can be seen via the OpenTox chart generation service:

```
```

**Figure 29** Distribution of the Cramer classes in the TCAMS dataset, depicted using the OpenTox chart generation service

To filter for Class I compounds, click the "Chemical compounds" links on the top, and enter “Low (Class I)” in the search box. This results in the following web address:
http://pirin.uni-plovdiv.bg:8080/malaria/compound?type=smiles&property=&search=Low+%28Class+I%29
(which could be also used directly, instead of typing the search query in the text box). The results can be browsed as below (Figure 30).

6.3 Step 2: Analyse Cytotoxicities of the Cramer Class I compounds

From the previous step we ended up with a list of compounds considered safe according to Cramer rules. However, we would like to have some more information other than just the chemical structures. For example, we would like to know the antimalarial activity of these compounds.

To add such a column, we need to edit the URL by adding an entry denoting the antimalarial activity given in the TCAMS Dataset. All data columns in OpenTox have their unique URL, and in this example, the URL of the data indicating the percentage inhibition of the growth of the *P. falciparum* strain 3D7 (column "Percentage_Inhibition_3D7" in the TCAMS dataset) is

http://pirin.uni-plovdiv.bg:8080/malaria/feature/190
To add this column to our filtered list of compounds considered safe according to Cramer rules (Cramer class I), we simply add a `feature_uris[]` parameter to the URL of our filtered list:


Copy this address into the web browser. There will be a small number of nonempty entries in the second column (see Figure 32).

We’re not only interested in the antimalarial activity, but would also like to take into account the experimentally determined human cytotoxicity. To do so, we add a second data column to our filtered list, now with human cytotoxicity data from the TCAMS dataset (Percentage_inhibition_HEPG2).
http://pirin.uni-plovdiv.bg:8080/malaria/feature/179). The combination of the two features – antimalarial activity and human cytotoxicity – will result in the following URL (see Figure 33):


Figure 33 Another column is added to the compounds with the feature.uris[] directive

6.4 Step 3: Predicting the Mutagenicity of the Selected Compounds

To add a further criterion to be used when selecting our drug candidate, we predict the compounds’ mutagenicities. To do so, we’ll use the Toxtree Benigni/Bossa rules for mutagenicity and carcinogenicity (Benigni et al., Mechanistic QSAR of aromatic amines: new models for discriminating between mutagens and nonmutagens, and validation of models for carcinogens, Environ Mol. Mutag. 48:754–771 (2007)). The URL of this model is http://pirin.uni-plovdiv.bg:8080/malaria/model/12.

Analogously as you have done for the Cramer rules, follow the URL of the Benigni/Bossa model (http://pirin.uni-plovdiv.bg:8080/malaria/model/12), type or paste the URL or the TCAMS dataset (http://pirin.uni-plovdiv.bg:8080/malaria/dataset/12) in the text box and click “Predict”. Alternatively, the URL of the filtered list could be entered here, as well.

OpenTox models store the prediction results again under data columns with unique URL. These are available via http://host/model/{id}/predicted, which in our example corresponds to http://pirin.uni-plovdiv.bg:8080/malaria/model/12/predicted
The Toxtree mutagenicity and carcinogenicity model predicts whether there are structural alerts for genotoxic or nongenotoxic carcinogenicity, and also uses a linear discriminant model for specific classes of compounds.

For our purpose, we select the columns “Structural Alert for genotoxic carcinogenicity” (http://pirin.uni-plovdiv.bg:8080/malaria/feature/258) and “Structural Alert for nongenotoxic carcinogenicity” (http://pirin.uni-plovdiv.bg:8080/malaria/feature/259). As before, we add data columns for these structural alerts to our Cramer-class filtered list of compounds, again using the feature_uris[] method. The resulting URL is:


The resulting table (as well as any other) can be sorted according to the values in any column by clicking on the column header.

In the following examples, we'll consider the first compound in the image below as our antimalarial drug candidate. It is a Cramer class I compound that inhibits growth of *P. falciparum* 3D7 by 99% at the concentration tested (2µM), has a very low human cytotoxicity and no structural alerts for carcinogenicity. (You may choose a different compound).
Similarly to datasets and models, each compound in OpenTox services also has its unique URL. You can find the URL of a compound by clicking on its 2D structure, and stripping off the "?media=text/html" part at the end of the URL this brings you to.

The URL of the compound selected above is


6.5 Step 4: Predicting Sites of Cytochrome P450 Metabolism

The URL of our drug candidate will be used to submit this compound to two models predicting cytochrome P450 sites of metabolism, namely

SmartCYP ² (http://pirin.uni-plovdiv.bg:8080/malaria/model/10) and

SOME ³ (http://pirin.uni-plovdiv.bg:8080/malaria/model/21)

Model prediction is done analogously to the two models used in this exercise. Go to

http://pirin.uni-plovdiv.bg:8080/malaria/model/10 and copy the compound URL into the text box.

---


When completed, the results will be available at


and will consist of information on which atoms are of rank 1, 2, 3 or lower. Higher rank means a more labile site. This information will be best viewed graphically, which could be achieved by the following URL

http://pirin.uni-plovdiv.bg:8080/malaria/compound/458166/conformer/773441?model_uri=http://pirin.uni-plovdiv.bg:8080/malaria/model/10 (see Figure 37).

The colour code for the result can be found by clicking on the "legend" link on the model page.
Similarly, the SOME model predictions are visualized via


And the color code is

- Aliphatic Hydroxylation
- Aromatic Hydroxylation
- N-Dealkylation
- O-Dealkylation
- S-Oxidation

The color-coding of the metabolic sites according to the type of metabolic reaction taking place allows the user – with a little knowledge in organic chemistry – to work out the metabolites of the compound.

To obtain the predictions of the sites of metabolism for the entire dataset, one can use the following URL:

7 Drug Discovery Predictive Toxicology Application II: Building a Model to Predict Kinase Inhibitor Activity

7.1 Introduction

Using the data on antimalarial compounds made available at the ChEMBL Neglected Tropical Disease (NTD) archive (http://www.ebi.ac.uk/chemblntd/), in this exercise subsets of the antimalarials are extracted to be used in a model building exercise via the OpenTox prototype application ToxCreat. 857 of the 13'519 compounds contained in the TCAMS dataset are annotated with a protein (class) target. Of these 857 compounds, 233 are annotated as Ser/Thr kinase inhibitors. In this exercise we'll use this information to create a dataset that can be used to build a model that predicts whether or not a given compound is likely to be a kinase inhibitor. The dataset for the model building therefore needs to consist of two columns: the SMILES string of the compound and its classification (Ser/Thr kinase inhibitor = 1, otherwise 0).

7.2 Activity B: Selecting a subset to create a model with ToxCreat

To create the dataset required for model building go to http://pirin.uni-plovdiv.bg:8080/malaria/dataset

Click on "Tres Cantos Antimalarial TCAMS dataset (screening against GlaxoSmithKline's library)"

Browse the dataset and find the column "Target hypothesis". You will note that most entries are empty (only ~6% of the compounds have a target hypothesis annotated). In the 100 compounds displayed by default when following the link to the TCAMS data, you will only find one entry with value "Adrenergic receptor antagonist". You could click on the link, which would filter out only compounds with this potential target.

For our purpose, we want the list of compounds annotated to be kinase inhibitors. You could try to increase the number of displayed compounds until you find one, or you could enter "Ser/Thr protein kinase" in the searching text box at the top of the page and click the "Search" button. The results will be displayed as below (see Figure 42).
To build a model, it is not enough to have a list of Ser/Thr kinase inhibitors. We also need some "negatives". Although strictly speaking we don’t have any true negatives, we will use the compounds that do have a target hypothesis annotation – but one that is not "Ser/Thr kinase" – as negatives. So, we extract the whole list of compounds with non–empty target hypothesis, and replace "Ser/Thr kinase" with a "1", and all the other target hypotheses with "0".

To extract the list of compounds with non–empty target hypotheses, use the following URL:


This operation is not (yet) possible via the "Search" text field (it does not allow negation, e.g. something like Target_Hypothesis !=""), but only via the URL: briefly, the search for non–empty Target Hypothesis is done in the above URL, first with &search=+ (the "+" stands for empty) – thus searching for all the empties – and then negating the search by &condition=!%3D (%3D stands for the "=" sign, thus !%3D stands for !=, or "not equal").

When following the above URL you’ll get a table with compounds that have a non–empty Target_Hypothesis. The next step will be to export data. Click on the left one of the two little Excel icons (when moving the mouse pointer on top of it, a small text box "text/csv" should appear) to save the selected data as CSV.

For the model building, we will use the OpenTox application ToxCreate (www.toxcreate.org). Thus, first we need to format the data as explained at www.toxcreate.org/help. That is, we leave only the SMILES column and the Target_Hypothesis column.

Now you should have the Target_Hypothesis in column 1 (or A), and the SMILES in column 2 (or B). If you are using Excel, go to the cell C2. Type

=IF(A2="Ser/Thr protein kinase"; 1; 0)

and hit "Enter". Again click on cell C2 to activate it. Now double-click on the little black square at the bottom–right corner of the cell’s border to fill the column with this formula.

Now, copy the whole column C, and paste it (at the same place) using Excel’s “Paste Special” function, pasting only the values. Once that’s done, delete column A (holding the text entries for the Target_Hypothesis). Delete as well row 1 and save the resulting table as text CSV file to TCAMS–kinase_full.csv.

In your web browser, navigate to www.toxcreate.org. Read the instructions, and try to create a model using your dataset. As ToxCreate is currently a prototype, there are still some limitations. You might get an error in the model building, in which case you could try to reduce the number of compounds used to build the model.
to about 600. Just delete some rows until that table contains 600 rows or less. Save the resulting table to TCAMS-kinase-subset.csv.
8  Set up an OpenTox Data Service

8.1  Introduction

IDEA has developed a standalone application, AMBIT 2.0, that implements the OpenTox REST API. The application is made available as a web archive, to be downloaded at ambit.sourceforge.net. AMBIT 2.0 allows for the easy setup of one’s own OpenTox data service. Instructions are given at ambit.sourceforge.net on how to deploy the web archive and how to set up the data service:

Figure 43 Source Forge page of AMBIT 2.0

However, getting point 1 of the instructions (“Requirements: Java Runtime Environment 6.x, MySQL and Tomcat application server, or compatible servlet container”) satisfied can pose some problems under certain Linux distributions. The following sections describe how to get set up both on Windows and on a CentOS Linux system.
8.2 Setting up a Windows System

Getting the software set up that is necessary to install the AMBIT 2.0 application is fairly straightforward under Windows. The example given here is for a Windows 7 Home Premium 64bit system.

8.2.1 Installing the Java Runtime Environment (JRE) and Java Development Kit (JDK)

The 32-bit version of JRE 6 Update 21 was downloaded from www.java.com located at http://javadl.sun.com/webapps/download/AutoDL?BundleId=41723 and installed. Most likely, JRE will already be installed on a Windows system, and the installer will inform you about that.

The 64-bit version of JRE 6 Update 21 was downloaded from http://www.oracle.com/technetwork/java/javase/downloads/index.html. The direct download link is this (link is too long to be displayed).

JDK 6 Update 21 was downloaded from www.oracle.com located at http://www.oracle.com/technetwork/java/javase/downloads/index.html; the direct download link to the Windows 64-bit software is this (link too long to display). The installer will again inform if JDK 6u21 is already installed on your system, in which case the installation can be aborted.

8.2.2 Installing MySQL

The current version of MySQL is version 5.1.50. Unfortunately, the 5.1.* versions of MySQL do not work properly anymore with the current version of the AMBIT application as of September 2010. Therefore, if MySQL is not installed yet, download version 5.0.* from http://downloads.mysql.com/archives.php?p=mysql-5.0&o=-win (e.g., version 5.0.91 as Windows 64-bit installer from http://downloads.mysql.com/archives/mysql-5.0/mysql-essential-5.0.91-minix64.msi).

For this tutorial, the Setup Type “Typical” was chosen, and “Configure the MySQL Server now” was selected.

A “Standard Configuration” was chosen for the MySQL Server, the “Install As Windows Service” and the “Launch the MySQL Server automatically” checkboxes were left checked on, and the “Include Bin Directory in Windows PATH” checkbox was checked.

Security Settings were modified in the sense that a root password was defined, and root access from remote machines was enabled. No anonymous account was created.

8.2.3 Installing Tomcat

Apache Tomcat 6.0.29 was downloaded as a service installer file from http://apache.mirror.testserver.li//tomcat/tomcat-6/v6.0.29/bin/apache-tomcat-6.0.29.exe. A “Normal” type of install was selected. The connector port was left at its default value of 8080. A Tomcat Administrator Login was defined. The path to the installed 64-bit JRE was given as C:\Program Files\Java\jdk1.6.0_21\jre.

After installing Tomcat, the Release Notes explain that Tomcat 6.0 no longer needs to have the complete JDK, and that a JRE is now sufficient. So, probably it wasn’t necessary to install JDK to deploy AMBIT 2.0, but no tests were made if that’s true.

If everything went alright during the installation of Tomcat 6.0, the service should be running now, and navigating to http://localhost:8080 should display the common Tomcat starting page (see Figure 46). If this page cannot be found, some settings might still be wrong. There should be a small icon for Tomcat in the taskbar of Windows 7 (Figure 44). Right-clicking on this icon allows starting the service, selecting “Configure…” allows configuring an automatic start of the service (Figure 45). Try clicking the “Start” button under the “General” tab, to see if you can start the service manually. If not, take a look at the entries in the “Java” tab, in
particular the one under "Java Virtual Machine". If you’re running a 64-bit machine, you should point Tomcat to a 64-bit Java Virtual Machine. Revise your settings, and try if you can start the service.

Once Tomcat is started, continue with paragraph 8.4 “Download and Deploy AMBIT 2.0”.

Figure 44 The Tomcat icon in the Windows7 taskbar

Figure 45 Configuring the Tomcat startup type
8.3 Setting Up a CentOS 5 Linux System

Reminding ourselves of the requirements to deploy the AMBIT 2.0 web archive, “Requirements: Java Runtime Environment 6.x, MySQL and Tomcat application server, or compatible servlet container”, the following sections explain how to set up each one of these software. We assume a more or less “virgin” CentOS x86_64 system, that is, neither of these software is installed already. We also assume that root privileges are available.

8.3.1 Installing MySQL

One of the more easy parts is the installation of MySQL. It turns out that a MySQL installation using the YUM package manager is sufficient. Given that package repositories for YUM have been configured, the MySQL installation (as root) is as simple as

```bash
# yum install mysql.x86_64 mysql-server.x86_64
```

To start the MySQL daemon, run

```bash
# /etc/init.d/mysqld start
```

To have mysqld start automatically when the system is booted, go to the directory /etc/init.d and run

```bash
# chkconfig --add mysqld
```


8.3.2 Installing the Java Runtime Environment and Development Kit

Download the Java Runtime Environment (JRE) from http://javapl.sourceforge.net/webapps/download/AutoDL?BundleId=40911. In this case, it is the x64_64 Linux package jre-6u21-linux-x64.bin. Save it e.g. to /usr/java/ (create the directory if necessary). Also download the Java SE Development Kit (JDK) from http://www.oracle.com/technetwork/java/javase/downloads/jdk6-jsp-136632.html. The direct download link for the Linux x86_64 file is here (URL is too long to display). Also save the JDK (jdk-6u21-linux-x64.bin) to /usr/java/.

Install the two packages as simply as follows:

```bash
# cd /usr/java
# sh jre-6u21-linux-x64.bin
# sh jdk-6u21-linux-x64.bin
```

8.3.3 Installing Tomcat

8.3.3.1 Download, Unpack and Prepare Tomcat

Download apache-tomcat (http://apache.mirror.testserver.li/tomcat/tomcat-6/v6.0.29/bin/apache-tomcat-6.0.29.tar.gz) and apache-ant (http://apache.mirror.testserver.li/ant/binaries/apache-ant-1.8.1-bin.tar.gz). Should you have problems downloading, find the downloads from http://apache.org and change the mirror. Save the two archives wherever you want; here we assume they’re saved to /usr/share.

Unzip the archives to /usr/share:

```bash
# tar -xzf apache-tomcat-6.0.29.tar.gz
# tar -xzf apache-ant-1.8.1-bin.tar.gz
```
Create a symbolic link for ant to work:

```
# ln -s /usr/share/apache-ant-1.8.1/bin/ant /usr/bin
```

In case ant is already present in /usr/bin, it is probably from a previous package installation. Try to remove ant with

```
# yum remove ant
```

and set the symbolic link again.

Next, we need to tell Tomcat where to find the Java Runtime Environment. This can be done by setting the appropriate variable in the Tomcat run script catalina.sh.

```
# cd /usr/share/apache-tomcat-6.0.29/bin
```

Open catalina.sh with your favourite text editor

```
# emacs catalina.sh
```

and add the line `JAVA_HOME=/usr/java/jdk1.6.0_21` right after the large comment block at the top of catalina.sh.

```
...  
# $Id: catalina.sh 947714 2010-05-24 16:57:18Z markt $  
#  
JAVA_HOME=/usr/java/jdk1.6.0_21  
# OS specific support. $var_must_be_set to either true or false.  
cygwin=false  
...  
```

8.3.3.2 Start Up Tomcat

To test if the current set up is working, run

```
# /usr/share/apache-tomcat-6.0.29/bin/startup.sh
```

Take a look at the log file catalina.out (/usr/share/apache-tomcat-6.0.29/logs/catalina.out), and check for potential error messages. Even if there are no errors, most likely you’ll find an INFO stating

```
INFO: The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found on the java.library.path:  
/usr/java/jre1.6.0_21/lib/amd64/server:/usr/java/jre1.6.0_21/lib/amd64:/usr/java/jre1.6.0_21/../lib/amd64:/usr/lib64:/lib64:/lib:/usr/lib
```

This message can safely be ignored. If ever needed, though, one could install APR following [http://tomcat.apache.org/tomcat-6.0-doc/apr.html](http://tomcat.apache.org/tomcat-6.0-doc/apr.html).

To make starting up tomcat easier, create a symbolic link for startup.sh to /usr/bin/tomcat:

```
# ln -s /usr/share/apache-tomcat-6.0.29/bin/startup.sh /usr/bin/tomcat
```

8.3.3.3 Creating a Startup Script to run Tomcat as user tomcat

To be able to start Tomcat as user tomcat upon system start, we need to compile a small program that comes with the tomcat archive. In older versions of Tomcat this used to be the package jsvc.tar.gz, located in the /bin directory of the Tomcat installation.

In recent versions (see [https://issues.apache.org/biz_bugzilla/show_bug.cgi?id=49585](https://issues.apache.org/biz_bugzilla/show_bug.cgi?id=49585)), the program archive is commons-daemon-native.tar.gz, located still in the /bin directory of the Tomcat installation. To install the program (assuming gcc is installed. Otherwise, run `yum install gcc -y`) follow these steps:

```
# cd /usr/share/apache-tomcat-6.0.29/bin/
# tar -xzf commons-daemon-1.0.2-native.tar.gz
```
# cd commons-daemon-1.0.2-native-src/unix
# export CFLAGS=-m64
# export LDFLAGS=-m64
# ./configure --with-java=/usr/java/jdk1.6.0_21/
# make

Check whether you have set alternatives for java:
# ls -1 /etc/alternatives/java
lrwxrwxrwx 1 root root 35 Aug 31 01:50 /etc/alternatives/java -> /usr/lib/jvm/jre-1.6.0-sun/bin/java

Remove the alternative if anything is found
# alternatives -remove java /usr/lib/jvm/jre-1.6.0-sun/bin/java

Check again
# ls -1 /etc/alternatives/java
lrwxrwxrwx 1 root root 35 Aug 31 20:59 /etc/alternatives/java -> /usr/lib/jvm/jre-1.4.2-gcj/bin/java

Again, remove the alternative if anything is found
# alternatives -remove java /usr/lib/jvm/jre-1.4.2-gcj/bin/java

Repeat this procedure until no alternatives are set anymore. Then, install the following alternative:
# alternatives --install /etc/alternatives/java java /usr/java/jdk1.6.0_21/bin/java 90

Now, add a new user, tomcat, to the system:
# useradd -d /usr/share/apache-tomcat-6.0.29/ tomcat

Find the tomcat startup script to run tomcat as user tomcat:
# cd /usr/share/apache-tomcat-6.0.29/bin/commons-daemon-1.0.2-native-src/unix/native

Even though we installed Tomcat 6, the startup script is still called Tomcat5.sh. Copy Tomcat5.sh to Tomcat6.sh and open it with your favorite text editor:
# cp Tomcat5.sh Tomcat6.sh
# emacs Tomcat6.sh

In the script Tomcat6.sh change the lines
JAVA_HOME=/home2/java/j2sdk1.4.2_03
CATALINA_HOME=/home/tomcat5/tomcat5/jakarta-tomcat-5/build
DAEMON_HOME=/home/jfclere/daemon
TOMCAT_USER=tomcat5

to
JAVA_HOME=/usr/java/jdk1.6.0_21
CATALINA_HOME=/usr/share/apache-tomcat-6.0.29
DAEMON_HOME=/usr/share/apache-tomcat-6.0.29/bin
TOMCAT_USER=tomcat

In addition, find any occurrence (there are two: one for starting the daemon, one for stopping it) of
$DAEMON_HOME/src/native/unix/jsvc \

and change it to
$DAEMON_HOME/commons-daemon-1.0.2-native-src/unix/jsvc \


With these changes, I still had some problems. It turned out that the variable CATALINA_HOME in Tomcat6.sh was used, although the script says it is only used for multi-instances of Tomcat and I was running only one instance of it. So, in Tomcat6.sh I also changed the line

```
CATALINA_BASE=/home/tomcat5/tomcat5/jakarta-tomcat-5/build
```

to

```
CATALINA_BASE=/usr/share/apache-tomcat-6.0.29
```

If you do intend to run multiple instances of Tomcat, please make sure that this (having CATALINA_HOME and CATALINA_BASE equal) doesn’t cause any problems, otherwise adapt your set up.

Now, test to see if the startup script works:

```
# ./Tomcat6.sh start
# pgrep -u tomcat -l
```

The process number will likely be different on your system, but you should see something similar.

### 8.3.3.4 Having the Tomcat Daemon Start Automatically

To have the Tomcat daemon start automatically upon system start, copy the Tomcat6.sh script to /etc/init.d:

```
# cp Tomcat6.sh /etc/init.d/tomcat6
# chmod +x /etc/init.d/tomcat6
```

Go to /etc/init.d, open tomcat6 with your favourite text editor, and add the following lines right after `#!/bin/sh` (note that in the following two lines the "#" is not the root command prompt, but the shell-script comment symbol):

```
# chkconfig: 234 20 80
# description: Small shell script to start/stop Tomcat using jsvc.
```

You should already be in /etc/init.d now. Issue the following two commands:

```
# chkconfig --add tomcat6
# chkconfig --list tocat6
tomcat6 0:off 1:off 2:on 3:on 4:on 5:off 6:off
```

This tells you that tomcat6 is started in runlevels 2, 3 and 4.

Because we installed Tomcat as root, we need to give the user “tomcat” permission/ownership for the whole Tomcat installation directory:

```
# cd /usr/share/apache-tomcat-6.0.29/
# chown -R tomcat *
# chgrp -R tomcat *
```

Now, try to reboot your system, and see if Tomcat starts properly. Before you do so, empty the log file to find possible errors more easily:

```
# mv /usr/share/apache-tomcat-6.0.29/logs/catalina.out /usr/share/apache-tomcat-6.0.29/logs/catalina.out.l
```

You might get errors related to the boot sequence, that is, tomcat is started too early in the boot process. In that case, you can try to move tomcat to be started later. To do so, change the line `# chkconfig 234 20 80` in /etc/init.d/tomcat6 to `# chkconfig 2345 96 80`, followed by

```
# chkconfig --del tomcat6
# chkconfig --add tomcat6
```
8.3.3.5 Adding Tomcat Users

The next step is to add Tomcat users, in particular a manager user. Under the Tomcat configuration directory (/usr/share/apache-tomcat-6.0.29/conf), there is a template file to define users, tomcat-users.xml. I couldn’t get this file to work, so I created one of my own with the following content:

```bash
# cd /usr/share/apache-tomcat-6.0.29/conf
# cat tomcat-users.xml
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
    <role rolename="standard"/>
    <role rolename="manager"/>
    <user username="roman" password="*******" roles="standard,manager"/>
</tomcat-users>
```

Note that I replaced the password with stars above. In reality, the password is stored in the file in clear text.

Once you have the users defined, reboot your system, or restart tomcat:

```bash
# /etc/init.d/tomcat6 stop
# /etc/init.d/tomcat6 start
```

Open a web browser, and navigate to [http://localhost:8080/](http://localhost:8080/). You should see the following screen (Figure 46):

![Tomcat “home” page](image)

**Figure 46: Tomcat “home” page**
8.4 Download and Deploy AMBIT 2.0

Download the AMBIT 2.0 application (http://www.ideaconsult.net/downloads/ambit2/ambit2.war). Save the file ambit2.war anywhere you like. With your web browser, navigate to http://localhost:8080, and click on “Tomcat Manager” in the Administration box at the top-left of the screen (Figure 46). You’ll be prompted to enter the user name and password of the Tomcat manager/administrator you have set up. On the manager page, scroll to the bottom and find the box entitled “WAR file to deploy” (Figure 47).

Figure 47: Tomcat “manager” page

Under “WAR file to deploy”, click “Browse…”, find ambit2.war and click “Deploy”. You should now have successfully installed the AMBIT 2.0 implementation of the OpenTox REST API. If you next navigate to http://localhost:8080/abmit2, you should see the screen corresponding to the one shown in Figure 48. As explained in the installation instructions for AMBIT 2.0 (http://ambit.sourceforge.net/), this release (September 2010) comes without an embedded database.

Thus, create an empty database using cURL (http://curl.haxx.se). On many linux systems, cURL can be easily installed from a package repository using a standard package manager. It can also be downloaded from http://curl.haxx.se/download.html. Under Windows, there are two options for using cURL: 1) installing cURL natively, preferable using this version: http://www.gknw.net/mirror/curl/win32/curl-7.21.1-devel-mingw32.zip, or 2) installing the VMWare Player (http://www.vmware.com/products/player/) and running a small Linux environment (http://www.maunz.de/opentox/dsl-4.1.zip) under Windows (after installing VMWare Player and unpacking the dsl-4.1.zip file, just double-click the dsl-4.1.vmx file).

Under Linux, after installing cURL simply type the following command as root in a console:

```
# curl -X POST -d "dbname=ambit2" -d "user=mysqladminuser" -d "pass=mysqladminpass"
http://localhost:8080/ambit2/admin/database
```
(replace "mysqladminuser" and "mysqladminpass" with the username and password of a MySQL user that has sufficient privileges to create a database, see paragraph 8.3.1).

Under Windows, if you have downloaded the cURL .zip file and have extracted it e.g. to C:\Users\Username\PROGRAMS\curl\curl-7.21.0-win64, open a command prompt (cmd.exe, usually under All Programs -> Accessories -> Command prompt), and type the following command (without the ">" sign):

```
> C:\Users\Username\PROGRAMS\curl\curl-7.21.0-win64\curl -X POST -d "dbname=ambit2" -d "user=mysqladminuser" -d "pass=mysqladminpass" http://localhost:8080/ambit2/admin/database
```

(replace "mysqladminuser" and "mysqladminpass" with the username and password of a MySQL user that has sufficient privileges to create a database, see paragraph 8.2.2).

If your POST is successful, navigating to http://localhost:8080/ambit2/admin/database should yield something like:

```
Comparing 127.0.0.1 127.0.0.1 : true
ambit2
Version: 3.0
Created: 2010
Note: AMBIT2 schema
```

Figure 48: The AMBIT 2.0 page
9 Conclusions

Initial OpenTox applications can now be used to carry out several core fundamental activities in predictive toxicology. Tutorials have been developed to provide guidance to the user in using these services for model building, predicting toxicities and establishing their own data resources. Additionally tutorials for more complex applications such as examining the US EPA ToxCast dataset and evaluating the toxicity-related properties of anti–malarial inhibitor compounds have been provided. We expect that expansion of these tutorial materials as further applications are developed will provide a useful knowledge resource to the community.

All tutorials and their updates are made available online under [www.opentox.org/tutorials](http://www.opentox.org/tutorials)