Seamless and uniform access to chemical data and tools: experience gained in developing the OpenTox framework

5th Meeting on U.S. Government Chemical Databases and Open Chemistry, 25-26 Aug 2011

Dr. Nina Jeliazkova
Ideaconsult Ltd., Bulgaria
www.ideaconsult.net

OpenTox

SEVENTH FRAMEWORK PROGRAMME

Overview

- What is OpenTox (Crash course - 2 slides)
- Motivation
- Architecture
- Technology
- Current status (OpenTox API implementation matrix)
- Experience: the devil is in the detail
  - REST web services
  - Resource Description Framework
- Implementations
  - AMBIT services
  - Dataset service examples: Chemical structures QA, data query and datasets comparison examples
- Conclusions

OpenTox crash course (1)
(with the help of cURL http://curl.haxx.se/)

1) Find a compound by an identifier, structure, similarity, substructure

curl -X GET http://host/query/compound/search/all?search=caffeine

Returns the URI of the compound http://host/compound/328

curl -X GET http://host/query/smarts?search=c1cccnc1-c2ncccc2

Returns URIs of the hits http://host/compound/456

2) Find a predictive model

curl -X GET http://host/model

Returns URI of the available models, e.g. http://host/model/8

3) Apply the model to the compound

curl -X POST http://host/model/8 -d "dataset_uri=http://host/compound/328"

Returns URI of the results, e.g. http://host/dataset/999

The results can be retrieved in all chemical MIME formats, as well RDF/XML, N3, CSV, ARFF

OpenTox crash course (2)

4) Find a dataset

curl -X GET http://host/dataset?search=TOXCST

Returns the URI of the dataset(s) http://host/dataset/78

5) Launch a descriptor calculation algorithm on this dataset

curl -X POST http://host/algorithm/8 -d "dataset_uri=http://host/dataset/78"

Returns URI of the results, e.g. **http://host/dataset/new**

6) Train a model

curl -X POST http://host/algorithm/LinReg -d dataset_uri=http://host/dataset/new

Returns URI of the model, e.g. http://host/model/newLRmodel

7) Apply the model to the compound from the previous slide

curl -X POST http://host/model/newLRmodel -d dataset_uri=http://host/compound/328

Motivation

- Predictive Toxicology applications need **common components**:
  - Access to datasets
  - Algorithms for descriptor calculation and model building
  - Validation routines
- The state-of-the-art involves re-implementation of these components in every new application
- If we had these components readily available we could
  - Quickly build new applications for specific purposes
  - Experiment with new combinations of algorithms
  - Speed up method development and testing
  - …

OpenTox Components

- **Compounds**: Structures, names, …
- **Features:** Chemical and biological (toxicological) properties, substructures, …
- **Datasets:** Relationships between compounds and features
- **Algorithms**: Instructions for solving problems
- **Models:** Algorithms applied to data yield models, which can be used for predictions
- **Validation**: Methods for estimating the accuracy of model predictions
- **Reports:** Report predictions and models, e.g. to regulatory authorities
- **Tasks:** Handle long running calculations
- **Authentication and Authorization**: Protect confidential data
- **Service registration** and querying : Finding services of specific type

Requirements & Technological choices

- Platform independence
- Interoperability with external programs and data sources
- Transparency for scientific and regulatory credibility
- Open for future extensions
- Web services (REST)
- Communication through well defined interfaces
- Ontologies for the exchange of knowledge and data
- Use and promote open standards
- Open source components

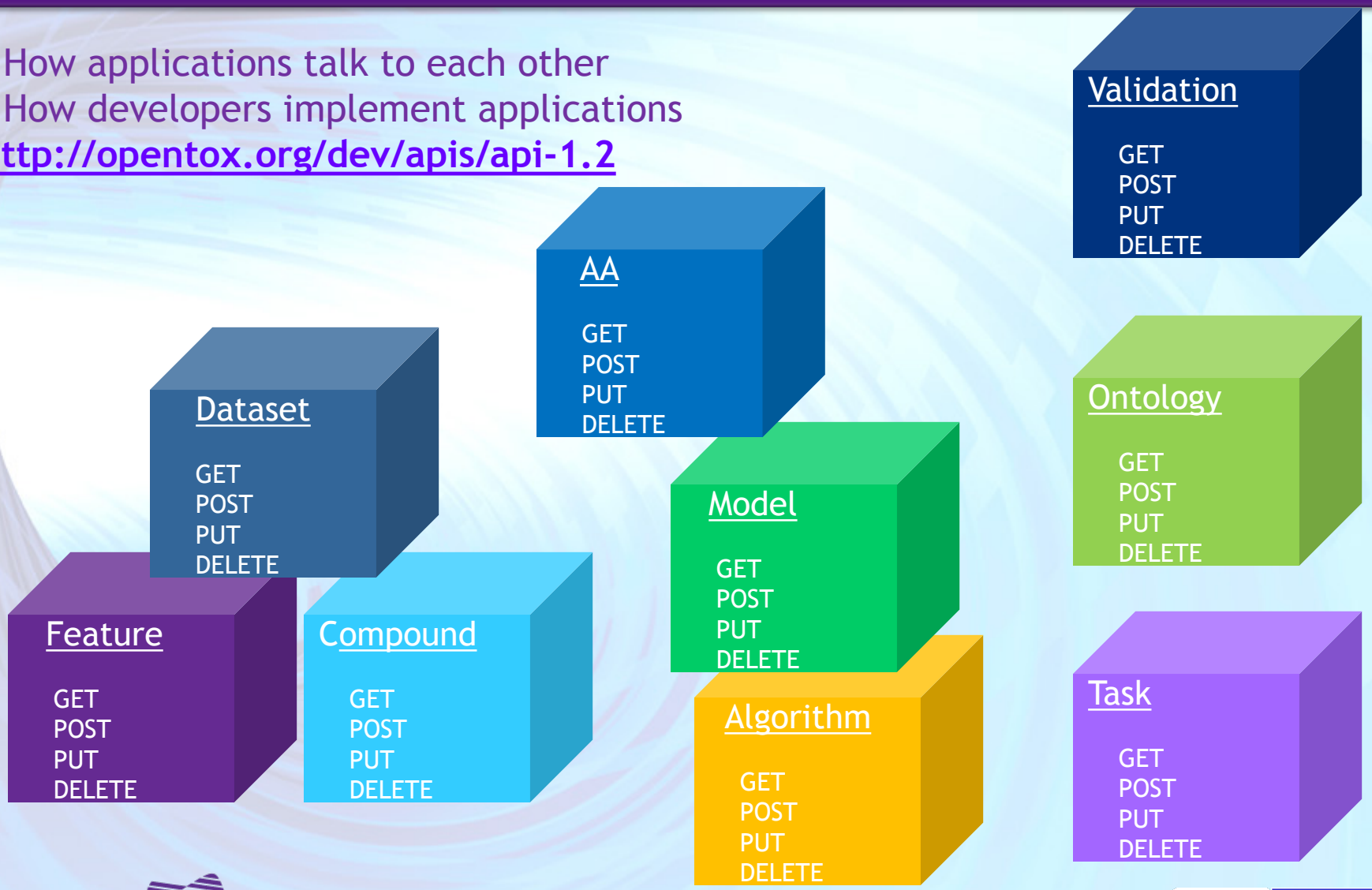
REpresentational State Transfer (REST)

- What?
  - Architectural style for distributed information systems on the Web
  - Simple interfaces, data transfer via **hypertext transfer protocol (HTTP)**, stateless client/server protocol
    - GET, POST, PUT, DELETE
  - Each **resource** is **addressed** by its own **web address**
  - **Multiple representations** per resource
- Why?
  - **Lightweight** approach to **web services**
  - **Simplifies/enables** development of **distributed and local** systems
  - Cacheability, scalability (inspired from the successful WWW architecture)
  - Language independent

OpenTox API (Application Programming Interface)

- How applications talk to each other
- How developers implement applications

http://opentox.org/dev/apis/api-1.2



OpenTox API Implementation Matrix

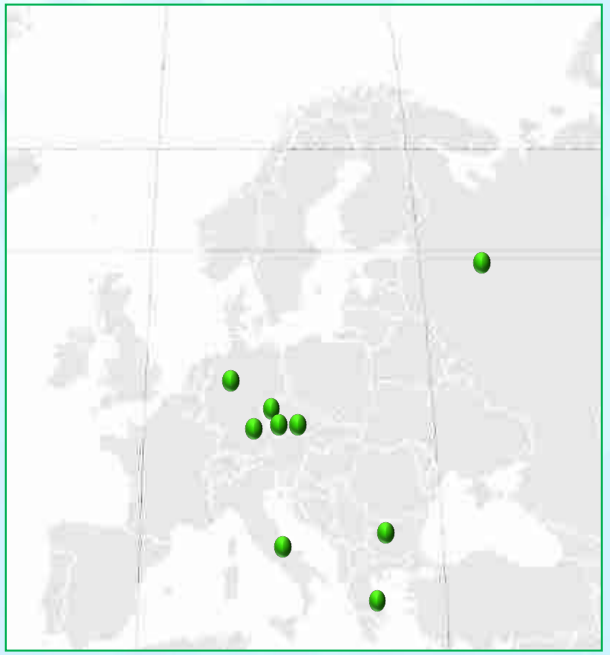
All components are implemented as REST web services.
There could be multiple implementations of same type of components.
(A subset of) services could be hosted by the same provider, or by multiple providers at separate locations.

OpenTox API Implementation Matrix

(list of existing independent implementations and pointers to deployed services and applications)

	ALU-FR (DE)	IBMC (RU)	IDEA (BG)	IST (CH)	NTUA (GR)	SL-JNU (IN)	TUM (DE)	3rd parties
Compound			✓	✓				
Dataset			✓	✓				
Feature			✓					
Algorithm		✓	✓	✓	✓	✓	✓	
Model		✓	✓	✓	✓	✓	✓	
Ontology			✓					
Validation & Reporting	✓							
Task		✓	✓	✓	✓	✓	✓	
AA & Policy Management				OpenAM				
	X	X	X	X	X	X	X	X
End-User Apps & Workflows	QMRF Editor	IBMCDesc	ToxPredict	ToxCreate	Q-Edit	MaxToxMCSS	ToxDesc	Bioclipse
	CheS-Mapper						Taverna	Bioclipse
Client Libraries			OT-client	Opentox-ruby	ToxOtis			
			OT-aa					

Last update: June 9, 2011



Implementation first, an API later

- The most common approach to scientific software and databases
  - Identify the data model and functionality
  - Translate the data model into a database schema
  - Implement the database and user interface functionality
  - (Optionally) provide libraries or expose (some) of the functionality as web services
- Advantages
  - Use one's favourite technology and jump directly into implementation
  - Attract end-users with nice GUI relatively quickly
  - Relatively easy to persuade funding organisations this will be a useful resource ☺
- Disadvantages
  - Proliferation of incompatible services, providing similar functionality, but incompatible programming interface
  - Difficult to extract and collate data automatically

What end users really need:

The user profile: organic chemistry background, working in industry, uses computational/modelling tools, but not a developer/programmer

- *I can do web search in the following databases and look for a compound (and perhaps later for some toxicity endpoint)*
  - *SciFinder - http://www.cas.org/products/sfacad/index.html*
  - *Toxnet - http://toxnet.nlm.nih.gov/*
  - *ChemID - http://chem.sis.nlm.nih.gov/chemidplus/chemidlite.jsp*
  - *SCCS* - http://ec.europa.eu/health/index_en.htm
  - *NTP - http://ntp.niehs.nih.gov/*
  - *Google - http://www.google.com/*
  - *Pubmed - http://www.ncbi.nlm.nih.gov/sites/entrez?tool=cdl&otool=cdlotool*
  - *PubChem - http://pubchem.ncbi.nlm.nih.gov/*

  (Disclaimer: the list is not comprehensive)
- *But how can I retrieve results for multiple compounds and endpoints automatically, without going manually to all the web pages?*
  - *And if technically possible, is it legal?*

The Internet provides a unique example of what society can achieve by adopting common standards

- Internet Engineering Task Force (IETF) working groups have the responsibility for developing and reviewing specifications intended as Internet Standards. The standardisation process starts by **publishing a Request for Comments (RFC)** – a discourse prepared by engineers and computer scientists for peer review or to convey new concepts or information.
- IETF accepts some RFCs as Internet standards via its **three step standardisation process**. If an RFC is labelled as a *Proposed Standard*, it needs to be implemented by at **least two independent and interoperable implementations**, further reviewed and after correction becomes a *Draft Standard*.
- With a sufficient level of technical maturity, a Draft Standard can become an **Internet Standard**. Organisations such as the World Wide Web consortium and OASIS support collaborations of open standards for software interoperability.
- While recently some authors argue the standardisation process is less than ideal and does not always endorse the best technical solutions, **the existence of the Internet itself**, based on *compatible hardware and software* components and services is a demonstration of the **opportunities** offered by collaborative innovation, flexibility, interoperability, cost effectiveness and freedom of action.

Standards in Life Sciences

- Cheminformatics:
  - Historically, the cheminformatics world has been driven by *de facto* standards, developed and proposed by different vendors;
  - A number of initiatives (relatively recent), have adopted open standardisation procedures (most notably InChI, CML, BlueObelisk initiatives), but there are no requirements for independent interoperable implementations so far;
- Bioinformatics:
  - Many (relatively recent) open data initiatives / standardisation efforts.
- Comparison with the network engineering practices
  - Network hardware and software have to work together by its very essence.
  - Reviewers in the network engineering world are likely comfortable with reviewing computer code of the implementations
  - Chemistry/Biology software and databases can live in their own worlds, unless we want data shared and tools interoperable
  - Interoperability / standards may affect business models

A common API first, (multiple independent interoperable) implementations later

What we have done differently in OpenTox

- Identified the data model and functionality
- Defined the (REST web service) application programming interface (API), which covers the data model and functionality
- **Developed six independent interoperable implementations** of the API, in 3 different languages
  - **test whether different implementations work together**
  - **if not, identify whether the API spec is ambiguous, leading to different interpretations, or just the implementation is buggy**
- Provided API libraries, developed end-user applications (web UI, standalone GUI, command line tools, workflow components)

A common API first, (multiple independent interoperable) implementations later

- Advantages:
  - Recall the use case
  - Avoid proliferation of incompatible resources (this however only makes sense if the API is adopted beyond a single implementation)
  - Easy to develop multiple GUI applications, once the API/library functionality is in place
- Disadvantages
  - Think first, then implement ☺
  - GUI comes last
  - Harder to persuade funding organisations (because reviewers usually look for and are pleased by nice GUIs)

OpenTox API implementation challenges

- Distributed team – 7 out of 11 partners developing software.
  - Different experience and background
- Distributed system
  - Efficient algorithms are the key
  - Multi-threading is important
  - The amount of data transferred may affect the performance severely
  - Network connections may fail or be slow
  - Optimizing the perceived latency (response time)

Troubleshooting a set of interoperating web services is hard!

Requires close cooperation and devotion of multiple developers!

OpenTox API implementation challenges
Steep learning curve

- Many API changes at the early stage
- Learning REST web services
  - A new technology for all but one partner
  - REST frameworks and browser peculiarities, bugs, instabilities
  - What is RESTful, what is not and why/whether it matters
  - Learning HTTP (who said it is a simple protocol…)
    - Pay attention to stream encoding, headers, error codes, redirection, many HTTP spec details
  - Selecting a solution for REST security
- Learning RDF
  - All OpenTox components are defined by **http://opentox.org/api/1.1/opentox.owl**
  - A new technology for all partners!

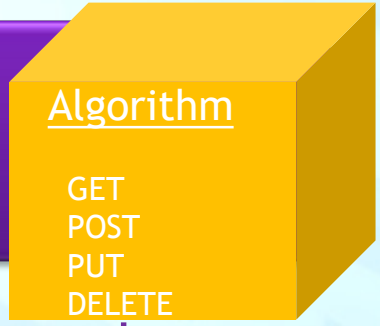
RDF: Lessons learned

- OpenTox specific
  - it hasn't started as Linked data/RDF project!
  - OpenTox uses RDF for serialization only, without mandatory backend triple storage
  - New resources and new triples are generated dynamically
- REST and RDF mix was not a popular choice back in 2009

  … but is natural for enabling retrieval of (partial) resource representation, described by triples

  … some issues discussed in http://www.jcheminf.com/content/3/1/18
- RDF: verbose; libraries: memory hungry, lack of streaming parsers
- Steep learning curve: some hard topics:
  - Data model vs. format
  - The subject-predicate-object concept vs. tabular/hierarchical structure
  - The recognition of the added value? (XML, JSON, YAML, plain text etc. vs. RDF)

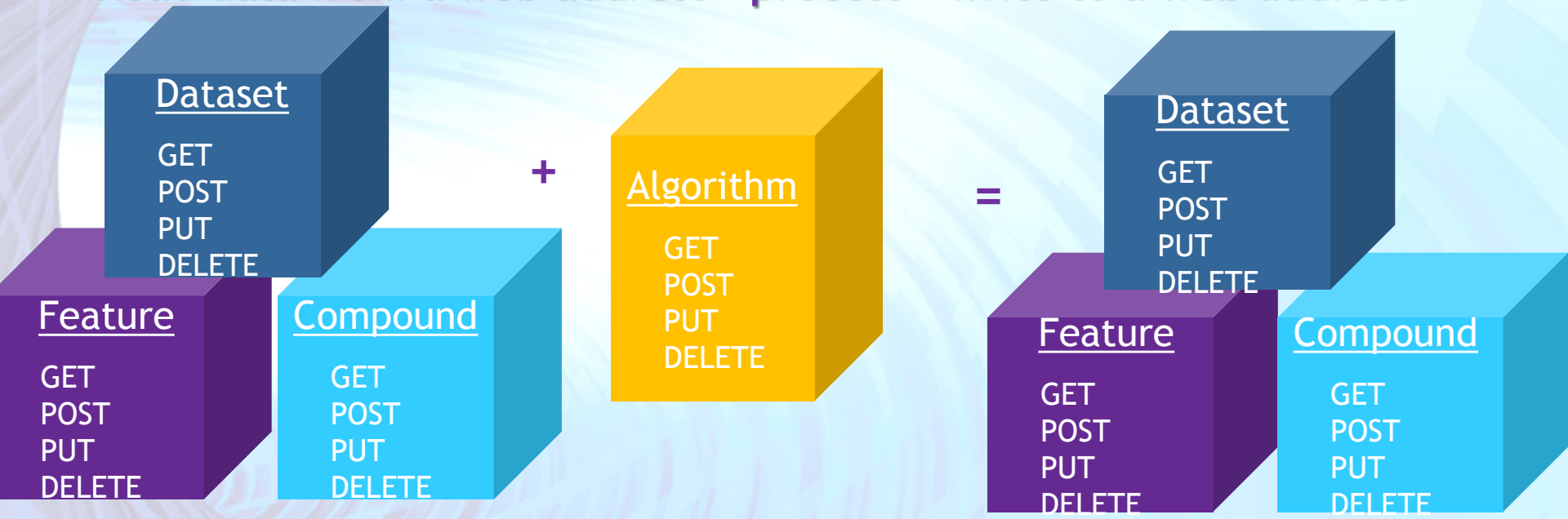
Algorithms



- Algorithms for **descriptor calculation**: **generation** and **selection** of **features** for the representation of chemicals (structure based features, chemical and biological properties);
- **Classification** and **regression** algorithms for creation of (Q)SAR models;
- **Rule based** algorithms;
- Algorithms for the **aggregation** of predictions from multiple (Q)SAR models and endpoints, and aggregation of predictions;
- **General purpose algorithms** (e.g. for visualization, similarity and substructure queries, applicability domain, read across, …)

Uniform approach to data processing

Read data from a web address - process - write to a web address

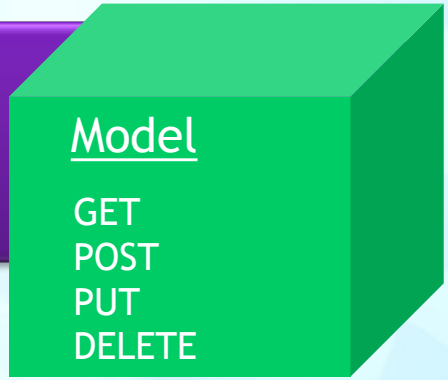


http://myhost.com/algorithm/{myalgorithm}

http://myhost.com/dataset/trainingset1

http://myhost.com/dataset/results

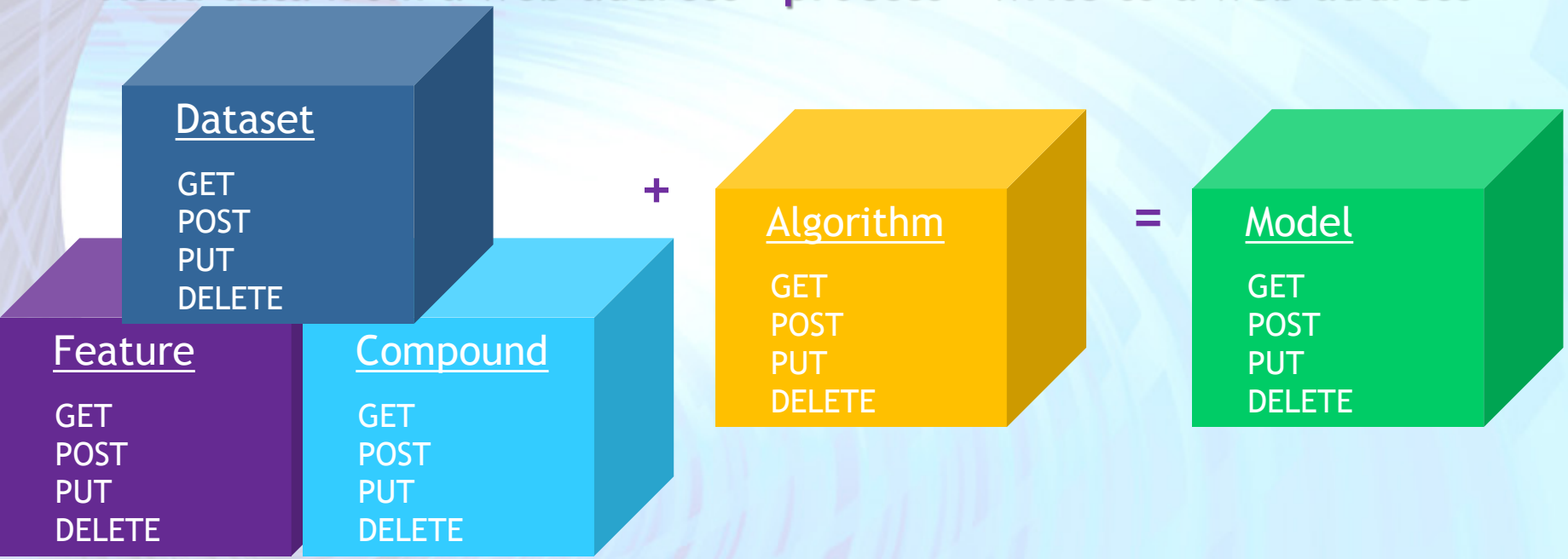
Models



- **Models:** Models are generated by respective algorithms, given specific parameters and data
  - Statistical models are generated by applying statistical/machine learning algorithms to specific dataset and parameters
  - Models can be other than statistical, e.g.
    - expert defined rules,
    - quantum mechanical calculations,
    - metabolite generation, etc.
- The intention of the framework is to be generic enough to accommodate varieties of predictive models.
- Model services provide facilities to inspect, store and delete models. Every model is identified by unique web address.

Uniform approach to models building

Read data from a web address - process - write to a web address



http://myhost.com/algorithm/neuralnetwork

http://myhost.com/dataset/trainingset1

http://myhost.com/model/predictivemodel1

OpenTox

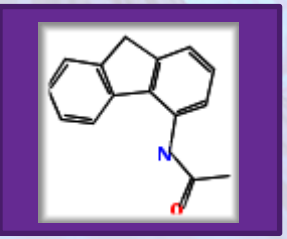
OpenTox API is database technology agnostic

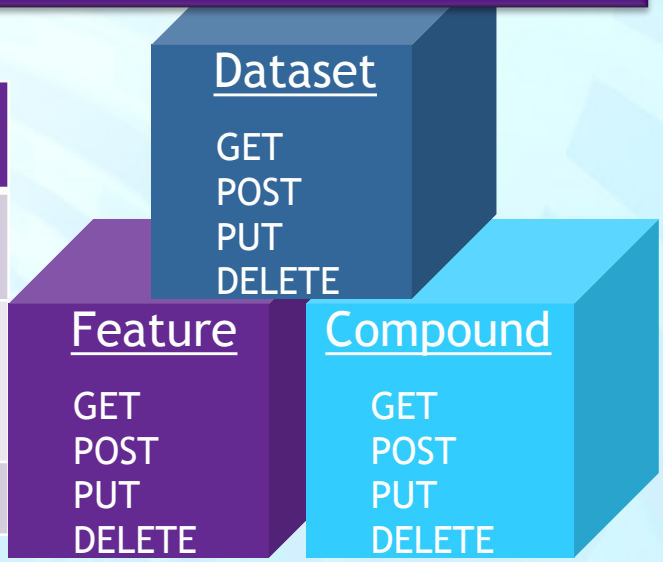
- No database schema mentioned so far. The API is database engine and database schema agnostic.
- The representation of resources is well defined by an OWL ontology.
- Resources are identified by dereferencable URIs and links between them established via RDF properties (instead of database relations)
- The data storage mechanism may vary: Memory, files, SQL, NoSQL, RDF triple storage, wrapper for remote services, etc.
- Data storage mechanism can be changed at any time (to fit emerging requirements), without affecting the API and end-user applications.

OpenTox datasets: Uniform data access

Table analogy (but defined as OWL)

Compound	http://myhost/feature/21580	http://myhost/feature/21589	http://myhost/feature/22114
http://myhost/compound/413	N,N-dimethyl-4-aminoazobenzene	CN(C1=CC=C(C=C1)N=NC2=CC=CC=C2)C	3.123
http://myhost/compound/4497	4-acetamidofluorene	O=C(Nc3c2c1ccccc1Cc2ccc3)C	2.085
...	...	...	...





http://myhost.com/feature/22114

a ot:Feature , ot:NumericFeature ;
dc:title "XLogP" ;
ot:hasSource <http://myhost/algorithm/XLogPDescriptor> ;
= otee:Octanol-water_partition_coefficient_Kow .

All columns have explicit and machine readable links to originating algorithms, models or data

Data publishing

1)HTTP POST a file with chemical structures and properties to an OpenTox dataset service.

Download as

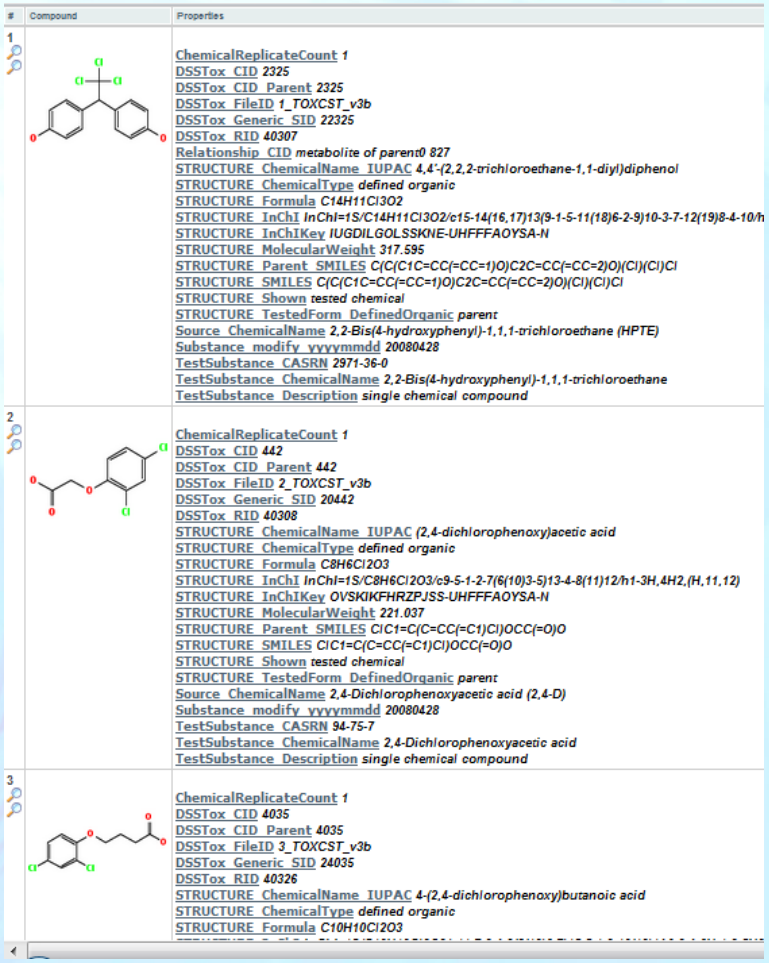
The structures and data are assigned a dataset URI and become available by multiple formats (RDF, Chemical MIME, CSV, Weka ARFF)

2)Assign metadata

PUT /dataset/{id}/metadata

3) Annotate any of dataset features /dataset/{id}/feature by assigning links to relevant ontologies

PUT /feature/{id}



OpenTox

AMBIT implementation of OpenTox API http://ambit.sourceforge.net

Our open source AMBIT REST software package implements a large subset of the OpenTox API (data and processing), and is available both as online services and as a downloadable archive

The dataset web service provides generic means to query, retrieve and upload chemical compounds and aggregate various data.

Built-in heuristics for automatic discovery of 2D chemical structure inconsistencies

Uploading a file with chemical structures and properties makes it automatically available in several formats.

Algorithm & model service: Descriptor calculation (CDK, MOPAC, more), machine learning methods (Weka), expert rules (Toxtree),applicability domain, wrapper for remote services, etc.

Journal of Cheminformatics 2011, **3**:18, http://www.jcheminf.com/content/3/1/18

OpenTox dataset service (content)

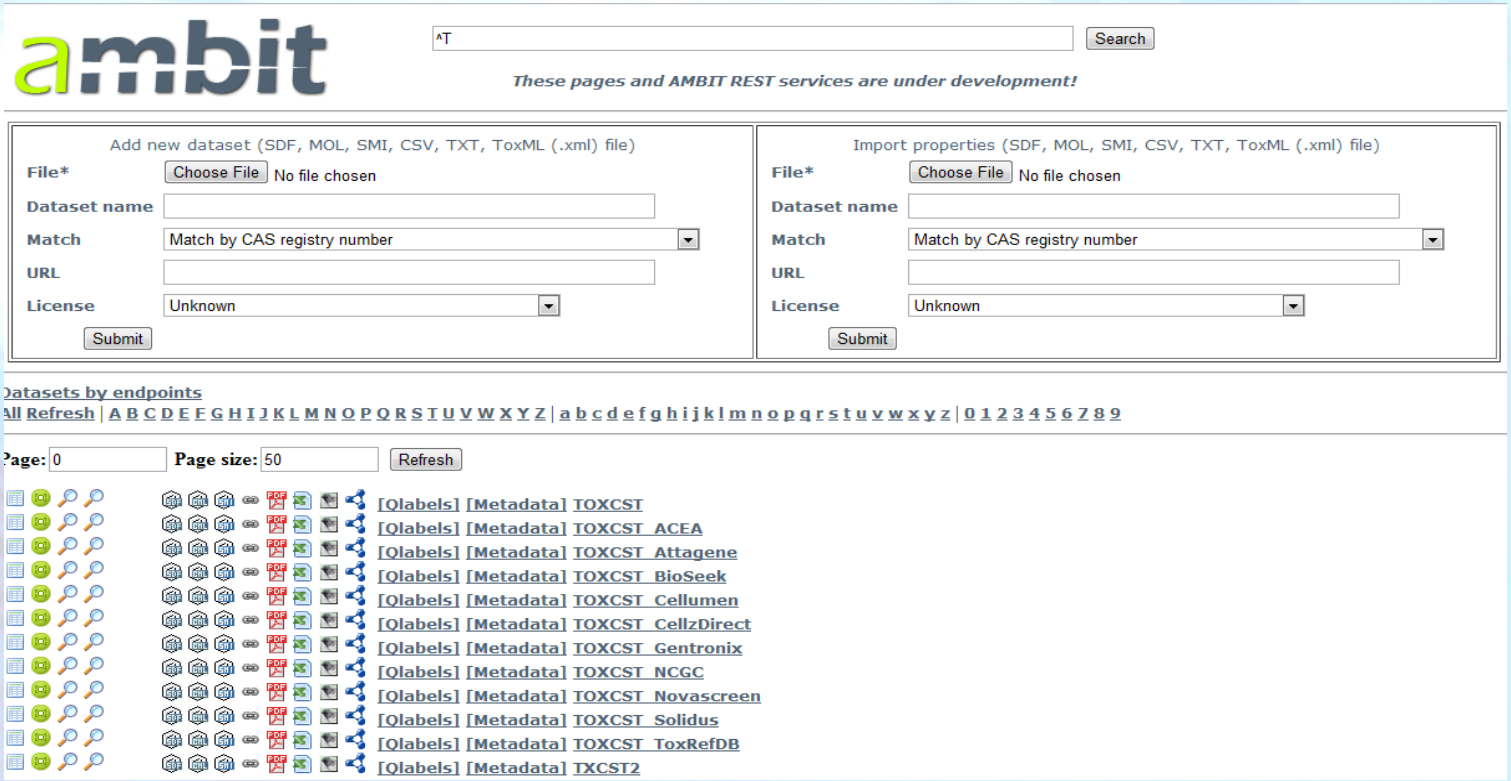
- **ECHA list of pre-registered substances (PRS)** : 143835 entries – names, CAS and EINECS numbers. Using the identifiers, structures were retrieved from
  - Chemical Identifier Resolver (structures retrieved by CAS)
  - ChemIDplus (structures retrieved by CAS)
  - OPSIN (Name to structure converter)
  - ChemDraw (Name to structure converter + partially manual expert inspection)
  - JRC PRS list (subset of ECHA PRS, generated by ACD/Name to structure converter)

…and imported into the dataset service via the OpenTox API

- The following datasets have been also imported (structures + data)
  - EPA DSSTox , ECETOC skin irritation , LLNA skin sensitisation , ,Bioconcentration factor (BCF) Gold Standard Database , EPA ToxCast , Benchmark Data Set for pKa Prediction of Monoprotic Small Molecules the SMARTS Way , Benchmark Data Set for In Silico Prediction of Ames Mutagenicity , Bursi AMES Toxicity Dataset , EpiSuite data , PubChem , Leadscope* data and Pharmatrope* data

OpenTox dataset service (an example)

ToxCast Phase I datasets have been uploaded and fields annotated



OpenTox dataset service (an example)

http://ambit.sourceforge.net/api_ontology.html

Retrieve list of features, representing Estrogen receptor gene (http://bio2rdf.org:geneid:2099) related assays from ToxCast Phase 1 datasets:

Ontology service 32472 triples

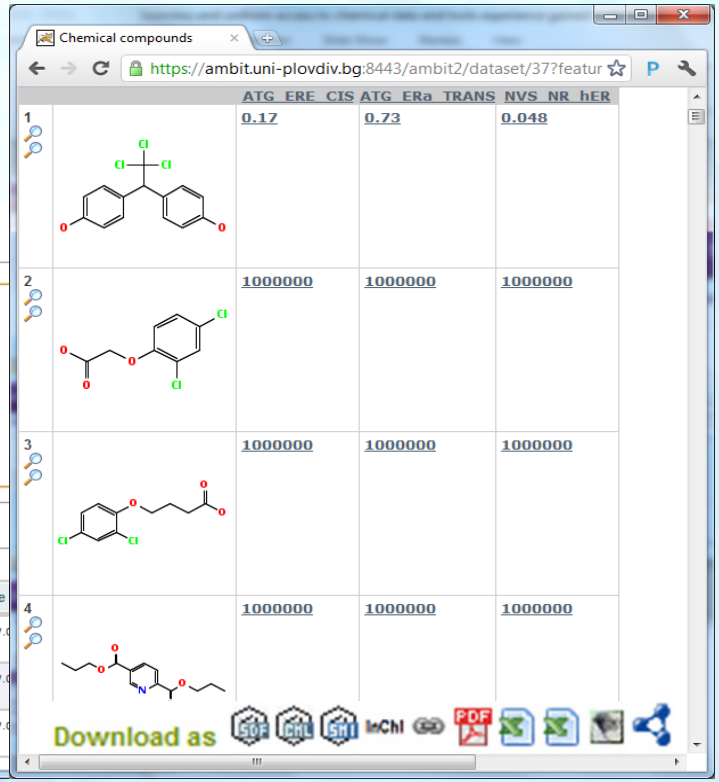
SPARQL

```
select ?Feature ?title ?OpentoxDataset ?assay ?genename
where {
        ?Feature rdf:type ot:Feature.
       {?Feature ot:hasSource ?OpentoxDataset}.
       {?Feature dc:title ?title}.
       {?Feature owl:sameAs ?assay}.
       {?assay toxcast:gene <http://bio2rdf.org/geneid:2099>}.
       {?assay toxcast:hasProperty ?genename}.
       {?genename rdf:type toxcast:GENE_NAME}.
}
```

Submit

Results [found in 6 ms]

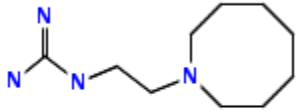
Feature	title	OpentoxDataset	assay	genename
https://ambit.uni-plovdiv.bg:8443/ambit2/feature/2643	'NVS_NR_hER'	https://ambit.uni-plovdiv.bg:8443/ambit2/dataset/4563	http://www.opentox.org/toxcast#NVS_NR_hER	http://www.c
https://ambit.uni-plovdiv.bg:8443/ambit2/feature/2285	'ATG_ERE_CIS'	https://ambit.uni-plovdiv.bg:8443/ambit2/dataset/723	http://www.opentox.org/toxcast#ATG_ERE_CIS	http://www.c
https://ambit.uni-plovdiv.bg:8443/ambit2/feature/2214	'ATG_ERa_TRANS'	https://ambit.uni-plovdiv.bg:8443/ambit2/dataset/723	http://www.opentox.org/toxcast#ATG_ERa_TRANS	http://www.c



Once we know which features are representing the Estrogen receptor related studies, the three columns can be collated with the ToxCast dataset and data retrieved in various formats (all via the OpenTox API)

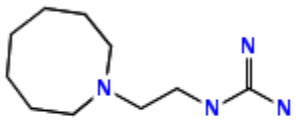
OpenTox database Quality Assurance

55-65-2
guanethidine



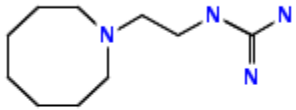
Consensus Quality
Label **Majority[1:3]**
Structure Quality
Label **ProbablyOK**

55-65-2
guanethidine



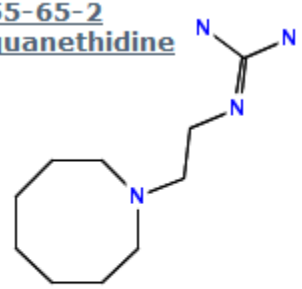
Consensus Quality
Label **Majority[1:3]**
Structure Quality
Label **ProbablyOK**

55-65-2
guanethidine



Consensus Quality
Label **Majority[1:3]**
Structure Quality
Label **ProbablyOK**

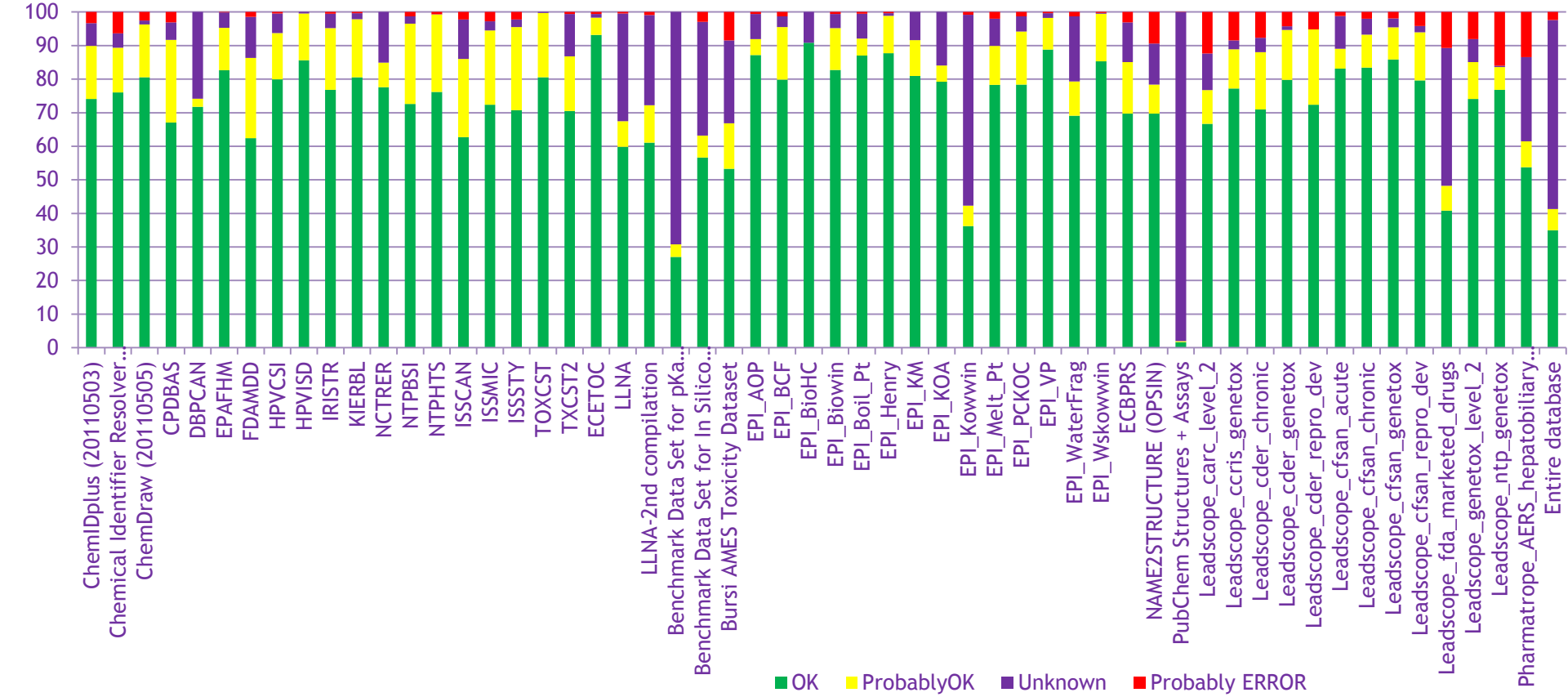
55-65-2
guanethidine



Consensus Quality
Label **Majority[1:3]**
Structure Quality
Label **ProbablyERROR**

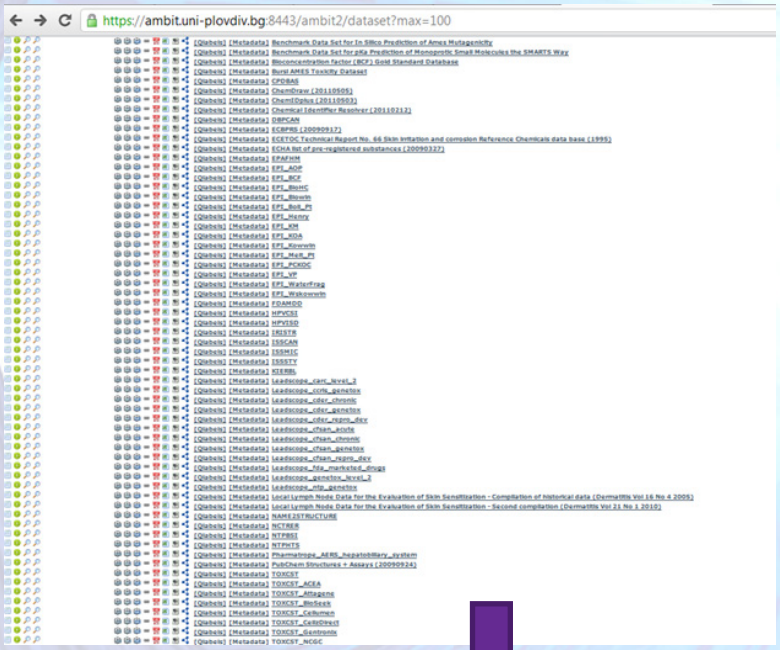
Automatic Classification	Initial Quality Label Assigned
Consensus	OK
Majority	ProbablyOK for the structure that belongs to the majority ProbablyERROR for the structure(s) that belong(s) to the minority
Ambiguous	Unknown (multiple sources)
Unconfirmed	Unknown (single source)

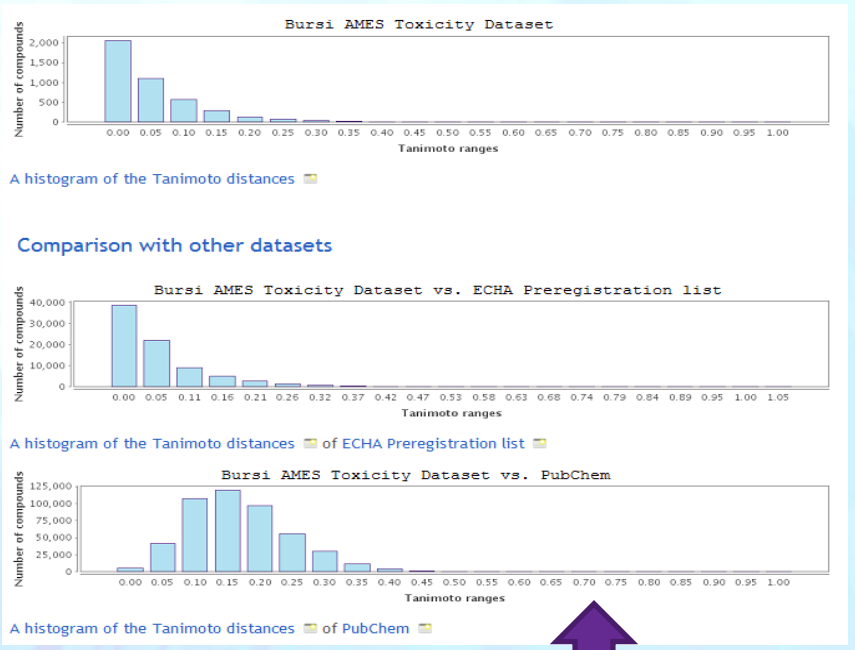
OpenTox database
Quality labels distribution

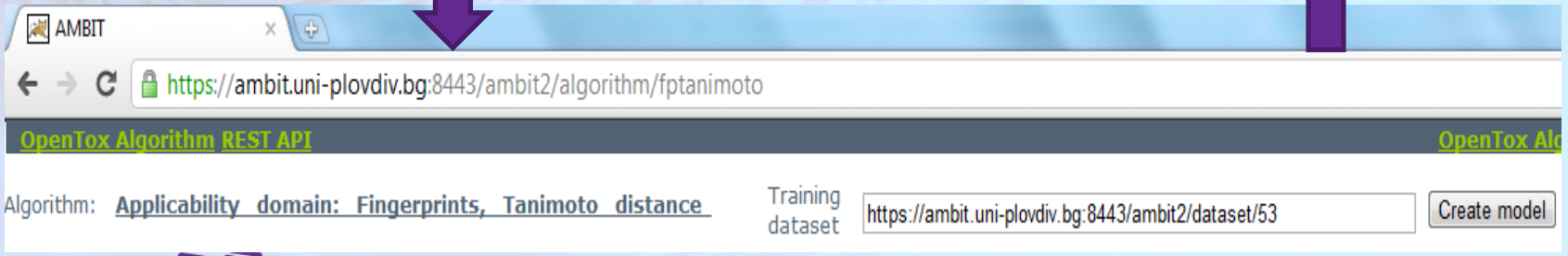


Dataset comparison
Use OpenTox algorithms and models

http://ambit.sourceforge.net/demo_datacomparison.html







Conclusions

- OpenTox partners have designed and built a framework of distributed resources, linking together chemical compounds, data, algorithms, models, model validation and reporting
  - Anybody can install and run separate instances of the services, either on Intranet or publicly on the Internet.
  - Anybody could publish own data, own algorithms and models
  - Dynamically generated RDF data via resolvable identifiers, can be crawled in a similar way as search engines crawl the web.
  - Mimics the WWW architecture (but read/write and makes use of structured data)
- The uniform interface to chemical compounds, data and models:
  - Helps reducing the diversity of processing to the simple paradigm “read data from a web address, perform processing, write to a web address”
  - Resembles conceptually the standard input and output streams in UNIX operating systems
  - Serves as a basis for web mashups, GUI applications, workflow components

Conclusions

- It is possible to define an implementation independent protocol (Application Programming Interface), covering the common chemoinformatics functionality:
  - Data access
  - Data upload
  - Chemical structures and data queries
  - Encapsulate various calculation and prediction methods under an uniform interface
- A common API may be implemented by any existing database or web service, without the need to change the underlying implementation

Quote

"Twenty to thirty years ago, most applications were written to solve a particular problem and were bound to a single database. The application was the only way data got into and out of the database. Today, data is much more distributed and data consistency, particularly in the face of extreme scale, poses some very interesting challenges"

http://www.zdnet.com/blog/microsoft/microsoft-big-brains-dave-campbell/1749

Acknowledgements

OpenTox project - An Open Source Predictive Toxicology Framework
http://www.opentox.org/
EU FP7 HEALTH-2007-1.3-3 Promotion, development, validation, acceptance and implementation of QSARs for toxicology
Project Reference Number
Health-F5-2008-200787 (2008-2011)

This presentation includes slides made by
Barry Hardy (DC),
Christoph Helma (IST)
Nina Jeliazkova (IDEA)
Olga Tcheremenskaya (ISS),
Stefan Kramer (TUM)
Andreas Karwath (ALU)
Haralambos Sarimveis (NTUA)

Thank you!

Try OpenTox demo applications
http://www.toxpredict.org **http://www.toxcreate .org**

Build an application with OpenTox REST Web Services API
http://opentox.org/dev/apis

Download OpenTox software http://opentox.org/downloads
(services, libraries, applications)

Download AMBIT Implementation of OpenTox API and launch your own OpenTox service
http://ambit.sourceforge.net

OpenTox