Accessing the PubChem Database as a Virtual Structure File

Wolf-D. Ihlenfeldt, Xemistry GmbH, Lahntal, Germany

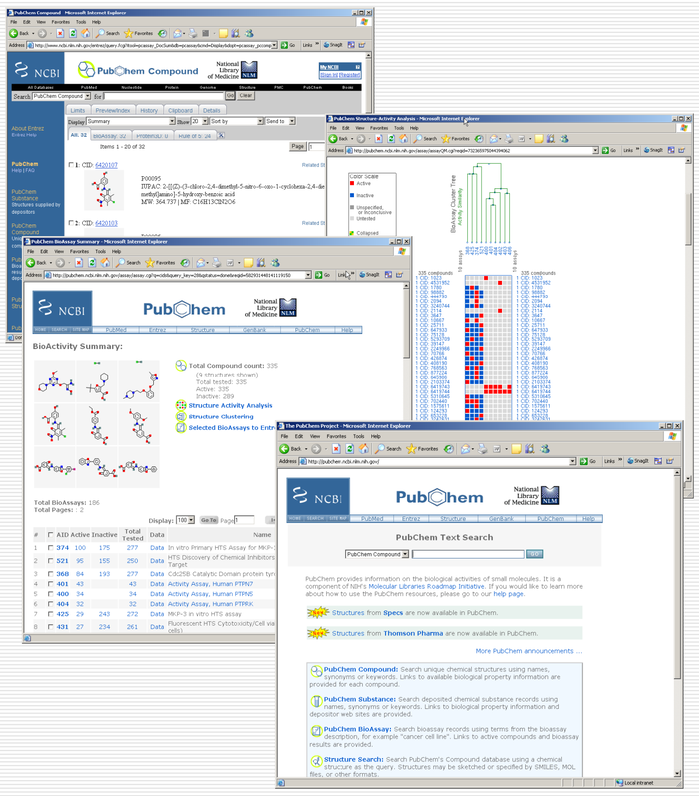
The PubChem Database Project Mission

Provide comprehensive public access to screening data generated by NIH Roadmap Initiative and other public research projects

Link assay results, structures screened, literature references, basic computed properties, external information sources

Powerful, convenient and free queries and download of filtered structure and assay data for further research

Wait a moment - they call it *convenient and powerful*?! Is it really?



Programmatic Access Routes to PubChem

Some disconnected building blocks exist:

<u>Entrez e-utils</u>
Basic access to Entrez text databases, get status, retrieve ID sets, some record data or set history via simple text-based queries

<u>PubChem structure display pages</u>
Can be abused for direct download of single records in ASN.1 format, bypassing the FTP wait queue

<u>PubChem Power User Gateway (PUG)</u>
Recently released ASN.1 specification for executing simple structure queries and getting ID sets or history handle from PubChem servers

No, no direct SQL database access for anybody!

Problems with PubChem

Separation between text/data (Entrez) and structure query systems with inconsistent interfaces

Intentionally dumbed-down structure query interface, but overengineered text query tools

Obscure Entrez syntax for combining queries

Quirky Entrez approaches on numerical queries, quoting, field names, output formats, history titles, auto query expansion…

History of history problems

Very limited customization of downloadable data content

Complete structure data record available only as ASN.1 blob, or as XML dump not adhering to any standard

Downloadable SD-file does not contain full data, is a structure approximation and still not compatible with pedantic interpretation of MDL standards

Pubchem has been engineered for human browsing, not for computerized minining, scripted lookup and custom dataset selection!

The PubChem Virtual File Driver – A Cactvs Toolkit I/O Extension Module Providing:

• Improved, transparent access to the PubChem database, ideally making it indistinguishable from accessing a local, read-only structure file in the Cactvs scripting environment

• Input functions that read structures and *all* their data from PubChem

• Query functions for the convenient scripting of custom lookup, data selection and download tasks that exceed the capabilites of the Web interfaces and PUG, while adhering to the standard, source-neutral Cactvs chemical objects query and retrieval syntax that is applicable to any structure file

PubChem and the Cactvs Toolkit

Cactvs is an universal, extensible scripting environment for structure-oriented chemical information processing.

It provides a comprehensive framework of chemical objects (ensembles, reactions, tables, …), dynamically defined object properties with associated computation methods, and extension modules (I/O modules for dozens of file formats, database access, data type handlers, command extensions,…)

The Toolkit has been licensed by NCBI as integral component of the PubChem software suite

It is used in PubChem for file I/O, syntax verification, property computation, structure depiction, structure identification via hashcodes, interface to NIST InChI suite, fingerprints, the full/sub/superstructure, similarity & formula search system and Web-based structure sketching

It is the only commercially available chemistry toolkit that understands the native PubChem data structures (ASN.1 specifications for substances, compounds, assays, and PUG) – including literature references, special bond types, conformers, etc.

It is maintained and distributed by Xemistry GmbH.

Tight Integration of PubChem Data with Cactvs Toolkit

Sample Operations:

Ensemble object creation via PubChem CID:

```
set eh [ens create $cid]
```

Operation behind the scenes: Direct download and parsing of binary ASN.1 record via display page. Also supported as file I/O module.

Computation of CID and SIDs from structure:

```
set cid [ens get $eh E_CID]
set sidlist [ens get $eh E_SIDSET]
```

Parsing of Entrez E-utils output from submission of InChI string as text search

Compound name lookup from PubChem

```
set iupacname [ens get $eh E_NAME]
```

Direct download and parsing of XML CID display record via Eutils, extracting OpenEye computed name

CAS number lookup via PubChem

```
set casno [ens get $eh E_CAS]
```

Direct download and parsing of XML SID set display records which contain depositor-supplied names, using pattern recognition

CAS number file I/O module

```
set eh [molfile read $casfile]
```

Look up CID as generic term via E-utils, download ASN.1 record via CID. Also supported as object creation command:

```
set eh [ens create $cas]
```

Scripted PubChem Batch Query Operations

Simple code sample - straightforward substructure search with CID extraction:

```
set fh [molfile open <pubchem>]
set cidlist [molfile scan $fh „structure >= $smarts“ \
    {proplist E_CID}]
```

Operations behind the scenes:
Set-up of PUG record
Post PUG, monitor return status
Cache CID result data
Direct access to result set, no structure download

Medium complexity code sample – parallel multi-structure search with structure object download for matches:

```
set fh [molfile open <pubchem>]
set enslist [molfile scan $fh \
„or {structure = $smiles1} {structure = $smiles2}\
{structure = $smiles3}“ enslist]
```

Operations behind the scenes:
Create and post PUG records, get history keys
Perform server-side e-utils result merge via history keys
Retrieve CID set
Download structures as ASN.1 blobs via CID

Advanced code sample – generate an Excel table with 1000 stereo-defined PubChem single-component non-metal structures which are similar (>= 0.95 Tanimoto score) to any structure from a local structure collection in an SD file:

```
set myfh [molfile open $mysdfile]
set fh [molfile open <pubchem>]
set th [molfile scan $fh \
„and {structure ~>= $myfh 95} {formula >= \[M\]0} \
{E_NMOLECULES = 1} {E_STEREO_COUNT(1) >= 1}“ \
{table E_CID score E_SMILES E_FORMULA record image} \
    {} {maxhits 1000}]
table write $th similar_in_pubchem.xls
```

The Excel table has CIDs, scores, SMILES strings, and structure images. It can be written on any platform.

Bioassay data access is unfortunately not yet part of PUG and therefore currently not supported.