Chemical Structure Validator

Motivation and Goal

This document outlines the concept of a standard web service API for validating chemical structures. Machine readability of chemical structure information is a critical part of modern chemical publications, but there is sometimes a disconnect between how a chemist conceives and draws chemical structure versus how that structure is stored electronically and interpreted by other various chemical information systems. Imagined here is a system by which a chemist's application – like ChemDraw or some ELN software – could be connected to validation services provided by major databases/institutions. The chemist could then ask, with the push of a button in their application, "How will PubChem interpret my structure?" or "What will EPA CompTox make of this?" and finally "Do their representations match what I know of this chemical?"

Every major chemical information system has its own rules on how to process and validate chemical structures. The goal here is not to create a standard for those rules, but rather to provide a common way for each institution to provide their own validation feedback from a chemical structure as input in an industry standard format (like SMILES or MOL/SDF). If each institution uses the same web service API for this validation, then an application like ChemDraw could easily let the user chose from a number of institutions, then provide a "validate" button that would send their structure to that institution over the internet and get back a standard response with feedback on that chemical, that the application can interpret and present back to the user.

These validators would presumably answer basic chemical informatics questions like "Are all the atoms in this structure valid chemical elements and isotopes?" or "Are there proper valences for each element?" or "Are all the stereocenters present in this molecule fully recognized and defined?" If the validator can produce an image, then the chemist could check whether the automatically produced image matches what they have drawn.

Note that this document is intended more as a conceptual white paper, rather than a detailed technical specification. The implementation examples provided below are demonstrations only, not exemplars of a final product.

Prototype and Examples

PubChem has, for demonstration purposes, created a prototype of this chemical structure validation service. In its simplest use, it takes chemical structure input (e.g. SMILES) and produces a JSON message:

https://pubchem.ncbi.nlm.nih.gov/resolver/resolver.cgi?action=validate_structure&smiles=CCCC

```
{
  "Result": {
    "Message": "Structure is valid",
    "Statistics": [
      {
        "Type": "DefinedAtomStereo",
        "Value": "0"
      },
      {
        "Type": "UndefinedAtomStereo",
        "Value": "0"
      },
      {
        "Type": "DefinedBondStereo",
        "Value": "0"
      },
      {
        "Type": "UndefinedBondStereo",
        "Value": "0"
      },
      {
        "Type": "HeavyAtoms",
        "Value": "4"
      },
      {
        "Type": "IsotopeAtoms",
        "Value": "0"
      },
      {
        "Type": "CovalentUnits",
        "Value": "1"
      }
    ]
  }
}
```

The service provides some basic chemical informatics properties, such as number of atoms, stereocenters, and so on. Again, the goal is to make sure that this machine-interpreted result matches the chemist's expectations.

(Technical note: the data model used for this response is available as an XML schema here: https://pubchem.ncbi.nlm.nih.gov/resolver/resolver_data.xsd ; PubChem has tools that simplify XML and JSON input/output based on XML Schema, so it was more convenient to implement this prototype from XML Schema. If JSON is the default format of this service, it would probably be better ultimately to use JSON Schema if/when that, or something like it, becomes a more widely recognized standard. But it is important to have a fully defined data model so that any application that uses this service knows what to expect, and how to parse the results.)

Here is a slightly more complicated structure, with both atom (sp3) and bond (sp2) stereochemistry:

https://pubchem.ncbi.nlm.nih.gov/resolver/resolver.cgi?action=validate_structure&smiles=C/C=C/[C@H]1[C@H](O1)C

```
{
  "Result": {
    "Message": "Structure is valid",
    "Statistics": [
      {
        "Type": "DefinedAtomStereo",
        "Value": "2"
      },
      {
        "Type": "UndefinedAtomStereo",
        "Value": "0"
      },
      {
        "Type": "DefinedBondStereo",
        "Value": "1"
      },
      {
        "Type": "UndefinedBondStereo",
        "Value": "0"
      },
      {
        "Type": "HeavyAtoms",
        "Value": "7"
      },
      {
        "Type": "IsotopeAtoms",
        "Value": "0"
      },
      {
        "Type": "CovalentUnits",
        "Value": "1"
      }
    ]
  }
}
```

The chemist can easily see that the stereocenters have been perceived and fully defined.

If there is an error in the structure, there should be some feedback as to what the problem is, for example a pentavalent carbon:

https://pubchem.ncbi.nlm.nih.gov/resolver/resolver.cgi?action=validate_structure&smiles=CC(C)(C)(C)C

```
{
  "Fault": {
    "Code": "Invalid",
    "Message": "Structure is not valid",
    "Details": [
      "Record 0: Warning: \"pcData/pubchem_valence.cpp\", line 290: 
Detected illegal valence for element \"C\": 5 sigma bonds, 0 pi bonds, 
0 charge",
      "Exception: Valence validation failed"
    ]
  }
}
```

Here is another example, with an invalid isotope:

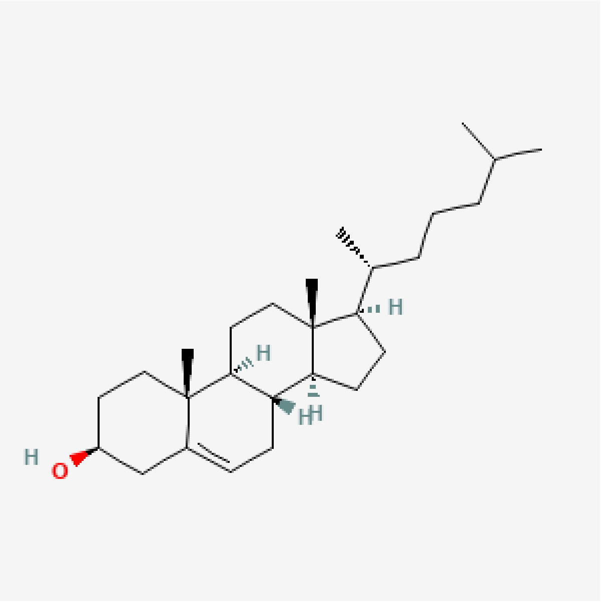
https://pubchem.ncbi.nlm.nih.gov/resolver/resolver.cgi?action=validate_structure&smiles=C[5H]

```
{
  "Fault": {
    "Code": "Invalid",
    "Message": "Structure is not valid",
    "Details": [
      "Record 0: Info: \"OpenEye/pubchem_compound.cpp\", line 3121: 
Atom ID \"2\" has illegal isotope (5) for atomic number 1 (\"H\")",
      "Exception: Element validation failed"
    ]
  }
}
```

While ^{5}H exists, it is PubChem's policy is to reject anything with isotopes with <1ms lifetime. Other institutions may have different policies. It is not the intention here to define the precise validation rules, but rather to provide a way for each organization to give feedback on the structure according to their own existing internal rules, but in a standard format.

Chemists are used to looking at chemical structure drawings, and so may prefer to see an image rather than the data fields shown above. For example:

https://pubchem.ncbi.nlm.nih.gov/resolver/resolver.cgi?action=validate_structure&format=png&smiles=C[C@H](CCCC(C)C)[C@H]1CC[C@@H]2[C@@]1(CC[C@H]3[C@H]2CC=C4[C@@]3(CC[C@@H](C4)O)C)C



Sample Web Application

The examples above show the actual HTTP URLs used by this proposed service. Of course, chemists aren't generally going to be typing in URLs into their web browser, but would instead be using this as part of some other application. PubChem has a (very simplistic) example of this as a web page service that uses a simple form input to the validation CGI:

https://pubchem.ncbi.nlm.nih.gov/resolver/resolver.cgi?action=input_form

This sample web interface lets the user select an input type, and choose whether to get validation details (as JSON) or an image. It also demonstrates the possibility of accepting MOL/SDF as input, which is a multi-line format not amenable to simple URL syntax as in the examples above.

(Technical note: PubChem has two different implementations of this service, one using PubChem's existing but internal standardization software, and another using RDKit – an open-source chemical information toolkit. At some point, example C++ code for the RDKit implementation may be shared. However, the details of the chemical validation rules will differ in some cases between PubChem and RDKit.)