Chalk *J Cheminform* (2016) 8:54
DOI 10.1186/s13321-016-0168-9

Journal of Cheminformatics

RESEARCH ARTICLE Open Access

SciData: a data model and ontology for semantic representation of scientific data

Stuart J. Chalk*

Abstract

With the move toward global, Internet enabled science there is an inherent need to capture, store, aggregate and search scientific data across a large corpus of heterogeneous data silos. As a result, standards development is needed to create an infrastructure capable of representing the diverse nature of scientific data. This paper describes a fundamental data model for scientific data that can be applied to data currently stored in any format, and an associated ontology that affords semantic representation of the structure of scientific data (and its metadata), upon which discipline specific semantics can be applied. Application of this data model to experimental and computational chemistry data are presented, implemented using JavaScript Object Notation for Linked Data. Full examples are available at the project website (Chalk in SciData: a scientific data model. http://stuchalk.github.io/scidata/, 2016).

Keywords: Science data, Semantic annotation, Ontology, JSON-LD, RDF, Scientific data model

Background

For almost 40 years, scientists have been storing scientific data on computers. With the advent of the Internet, research data could be shared between scientists, first via email and later using web pages, FTP sites, and online databases. With the advancement of Internet technologies and online and local storage capabilities, the options for collecting and stored scientific information have become unlimited.

Yet, with all these advancements science faces an increasingly important issue of interoperability. Data are commonly stored in different formats, organized in different ways, and available via different tools/services severely impacting curation [2]. In addition, data is often without context (no metadata describing it), and if there is metadata it is minimal and often not based on standards. Though the Internet has promoted the creation of open standards in many areas, scientific data has, in a sense, been left behind because of its inherent complexity. The strange part about this scenario is that scientific data itself is not the biggest problem. The problem is the contextualization of the scientific data—the metadata that describes system that it applies to, the way it was investigated, the scientists that determined it, and the quality of the measurements.

So, what is scientific data and where is the metadata? Peter Murray-Rust grappled with these questions in 2010 and concluded that it is "factual data that shows up in research papers" [3]. When writing scientific articles, researchers add most (in most cases not all) of the valuable metadata in the description of the research they have performed. The motivation of course is open sharing of knowledge for the advancement of science, with appropriate attribution and provenance of research work. As we move toward the fourth paradigm [4], where large aggregations of data are the key to discovery, it is imperative that the context of the data are articulated completely (or as completely as possible), not only to identify it's origin and authenticity, but more importantly to allow the data to be located correctly on the "scientific data map".

To address these issue, this paper describes a generic scientific data model (SDM)/framework for scientific data derived from (1) the common structure of scientific articles, (2) the needs of electronic notebooks to capture scientific research data and metadata, and (3) the clear need to organize scientific data and its contextual descriptors (metadata). The SDM is intended to be data format/software agnostic and extremely flexible, so that

*Correspondence: schalk@unf.edu
Department of Chemistry, University of North Florida, Jacksonville, FL 32224, USA

it can be implemented as the scientific research dictates. While the SDM is abstract in nature, it defines a concrete framework that can be easily implemented in any database and does not constrain the data and metadata that can be stored. It therefore serves as a backbone upon which data and its associated metadata can be 'attached'.

In addition, this paper describes an ontology that defines the terms in the SDM, which can be used to semantically annotate the structure of the data reported. In this way, scientific data can be integrated together by storage in Resource Description Framework (RDF) [5] triple stores and searched using SPARQL Protocol and RDF Query Language (SPARQL) queries [6].

The use of the ontology in the generation of RDF is demonstrated in examples of scientific data saved in JavaScript Object Notation (JSON) for Linked Data (JSON-LD) [7] format using the framework described by the SDM. From these examples it is shown how useful a hybrid structured (relational)/graph (unstructured) approach is to the representation of scientific data.

JSON-LD is a recent solution to allow transfer of any type of data via the web's architecture—Representational State Transfer (REST) [8]—using a simple text-based format—JSON. [9] JSON-LD allows data to be transmitted with meaning, that is, the "@context" section of a JSON-LD document is used to provide aliases to the names of data reported and link them to ontological definitions using a Uniform Resource Identifier (URI)—often a Uniform Resource Locator (URL). In addition, the structure/data of the JSON-LD file can be automatically be serialized to Resource Description Format (RDF) using a JSON-LD processor, e.g. the JSON-LD Playground [10]. This capability makes JSON-LD files not only useful as a data format but also a compact representation of the meaning of the data.

Methods

Aim, design and setting of the study

The aim of this work was to develop a serialization of scientific data and its contextual metadata. The design was encoded using the JSON-LD specification [7] because it is both a human readable and editable format and can easily be converted to RDF [5] triples for ingestion into a triple store and subsequent SPARQL searching [6]. The intent was that the data model, developed to afford the serialization, would be able to structure any scientific data (see examples).

Description of materials

Data were taken from different data sources and encoded in the proposed serialization. Items 5, 6, and 7 were created using XSLT files.

1. laboratory notebook data
2. research article data
3. spectral data (NMR)
4. computational chemistry data
5. PubChem download as XML
6. Dortmund Data Bank webpage as HTML
7. Crystallographic Information Framework (CIF) file as text

Description of all processes and methodologies employed

In this work different pieces of scientific data were selected and an analysis performed of the required metadata that was necessary to completely describe the context of how the data were obtained. After looking at the data and its context, reading a number of research articles on what scientific data is, and reviewing journal guidelines for submission of research, a preliminary generic structure of scientific data and metadata was developed. This was iteratively improved by encoding the data of higher and higher complexity into the framework and adding/deleting/adjusting as necessary to make the model fit the needs of the data.

Statistical analysis

Statistical analyses were not performed.

Results and discussion

Considerations for a scientific data model

What is scientific data?

In order to appreciate what scientific data is we took a step back and looked at the scientific process to abstract the important aspects that underpin the framework of what scientists do and how they do it. When we teach students to think and act like scientists we start with the general scientific method [11]:

- *Define a research question* What is the scope of the work? What area of science is the investigation in? What phenomena are we investigating?
- *Formulate a hypothesis* What parameters/conditions do we control or monitor in order to evaluate the effect on our system?

- *Design experiments* What instrumentation/equipment do we use? What are the settings and/or conditions? What procedures are used?
- *Make observations* What are the values of the controlled parameters, experimental variables, measured data, and/or observations?
- *Generate results* How is data aggregated? What calculations are used? What statistical analysis is done?
- *Make conclusions/decisions* What are the outcomes? Is the data good quality? Do they help answer the question(s) asked? How does the data influence/impact subsequent experiments?

The process above defines the types of information scientists collect as they perform science and once a project is complete they aggregate all of the important details (data, metadata, and results) from the process and synthesize one or more research papers to inform the world of their work. Thus, scientific papers can be considered a pseudo data model for science. Yet, this format has significant flaws as, in general, it is not typically setup uniformly, often has only a subset of all the metadata of the research process, and is influenced by the biases of authors and the constraints of publication guidelines.

How is scientific data structured?

Scientists have grappled with structuring scientific data since its inception. Communication of scientific information in easy to understand formats is extremely important for comprehension and hypothesis development, especially as the size and complexity of data grows. Its representation is also highly dependent on the research area both in terms of size/complexity of captured data and common practices of the discipline.

In chemistry the best example of data representation is the periodic table [12], the fundamental organization of data about elemental properties, structure and reactivity, and it is impossible to be chemist without appreciating the depth of knowledge it represents. The same is true in biology about the classification of species [13, 14], or in physics the data model underlying the grand unification of forces [15].

Data representation/standardization in chemistry has since evolved primarily in two areas: Chemical structure representation and analytical instrument data capture [16].

Chemical structure representation

Communication of chemical structure has been an area of significant development since John Dalton introduced the idea that matter was composed of atoms in 1808, and developed circular symbols to represent known atoms [17]. It wasn't long before Berzelius wrote the first text based chemical formula, H_2SO_4, showing the relative number of atoms of each element. Since these early steps chemists have found need to create representations of molecular structure for many different applications. In the Twentieth century this has brought us text string notations such as Wiswesser Line Notation (WLN) [18], simplified molecular-input line-entry system (SMILES) [19], and most recently the International Chemical Identifier (InChI) [20] in addition to the classical condensed molecular formula. Both SMILES and InChI are elegant solutions to encoding structural information in text where the string to structure conversion (and vice versa) can be done accurately by computer for small molecules. Solutions for large molecules, crystals and polymers are still needed, as are definitive representation of stereocenters.

Chemical structure representation on computers, using standard file formats, has been a challenge many have attempted to solve. Currently, there are over 40 different file formats (see [21]) for 2D, 3D, and reaction representation. Of these, the.mol file (MOL) V2000 [22] is the most widely available even though the V3000 format has been out for many years. The MOL file, like many others contains a connection table that defines the positions of, and bonds between, the atoms (Fig. 1).

In addition to MOL files, the Chemical Markup Language (CML) [23], an Extensible Markup Language (XML) [24] format, is a more recent development allows the content and structure of the file (through use of an XML schema) to be validated. This is an important feature for reliable storage and transmission of chemical structural information and provides a mechanism, through digital signatures, to ensure integrity of the files. Figure 2 shows the equivalent, valid CML file for the MOL file in Fig. 1. While the CML is larger (1931 vs. 721 bytes) it is easier to read by humans (and computers) and contains information about the hydrogen atoms where the MOL file does not.

Finally, the exemplar chemical structure representation standard for data reporting is the Crystallographic Information Framework (CIF) developed in 1991

```
Benzene, ID: C001
  SJC    20160623 1   1.00000     0.00000
Example Benzene mol file.
  6  6  0     0  0              1 V2000
   -1.3961    0.0013   -0.0504 C   0  0  0  0  0  0           0  0  0
   -0.7402   -0.3516    1.1313 C   0  0  0  0  0  0           0  0  0
    0.6556    -0.344    1.1844 C   0  0  0  0  0  0           0  0  0
    1.3956    0.0123    0.0546 C   0  0  0  0  0  0           0  0  0
    0.7398    0.3611   -1.1284 C   0  0  0  0  0  0           0  0  0
    -0.656    0.3577   -1.1803 C   0  0  0  0  0  0           0  0  0
  2  1  2  0     0  0
  1  3  1  0     0  0
  4  2  1  0     0  0
  3  6  2  0     0  0
  5  4  2  0     0  0
  6  5  1  0     0  0
M  END
```

Fig. 1 Example MOL file format for benzene

```
<molecule title="benzene" xmlns="http://www.xml-cml.org/schema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.xml-cml.org/schema
    http://www.xml-cml.org/schema/schema3/schema.xsd">
    <atomArray>
        <atom id="a1" elementType="C" x3="-1.3961" y3="0.0013" z3="-0.0504"/>
        <atom id="a2" elementType="C" x3="-0.7402" y3="-0.3516" z3="1.1313"/>
        <atom id="a3" elementType="C" x3="0.6556" y3="-0.344" z3="1.1844"/>
        <atom id="a4" elementType="C" x3="1.3956" y3="0.0123" z3="0.0546"/>
        <atom id="a5" elementType="C" x3="0.7398" y3="0.3611" z3="-1.1284"/>
        <atom id="a6" elementType="C" x3="-0.656" y3="0.3577" z3="-1.1803"/>
        <atom id="a7" elementType="H" x3="-2.4975" y3="-0.0029" z3="-0.0919"/>
        <atom id="a8" elementType="H" x3="-1.3241" y3="-0.6376" z3="2.0213"/>
        <atom id="a9" elementType="H" x3="1.1731" y3="-0.6224" z3="2.1169"/>
        <atom id="a10" elementType="H" x3="2.4971" y3="0.0152" z3="0.0956"/>
        <atom id="a11" elementType="H" x3="1.3239" y3="0.6376" z3="-2.0214"/>
        <atom id="a12" elementType="H" x3="-1.1735" y3="0.6347" z3="-2.1132"/>
    </atomArray>
    <bondArray>
        <bond atomRefs2="a1 a2" order="A"/>
        <bond atomRefs2="a1 a6" order="A"/>
        <bond atomRefs2="a1 a7" order="S"/>
        <bond atomRefs2="a2 a3" order="A"/>
        <bond atomRefs2="a2 a8" order="S"/>
        <bond atomRefs2="a3 a4" order="A"/>
        <bond atomRefs2="a3 a9" order="S"/>
        <bond atomRefs2="a4 a5" order="A"/>
        <bond atomRefs2="a4 a10" order="S"/>
        <bond atomRefs2="a5 a6" order="A"/>
        <bond atomRefs2="a5 a11" order="S"/>
        <bond atomRefs2="a6 a12" order="S"/>
    </bondArray>
</molecule>
```

Fig. 2 Example CML file format for benzene

[25–27] as an implementation of the Self-defining Text Archive and Retrieval (STAR) format [28]. The CIF/STAR format uses a similar approach to JCAMP-DX (see below) in that a number of text strings are defined to identify specific metadata/data items. The use of well-defined labels is not only more extensive in CIF but the format also includes the option to create pseudo tables of any size using the loop_ instruction, whereas JCAMP is limited to two columns (XY data or peak tables). The format has evolved significantly from its inception due to community input and support and is now integrated into the publishing of crystallographic data in journal articles through the Cambridge Crystallographic Data Centre (CCDC). Figure 3 shows an example CIF file for NaCl.

Analytical instrument data capture

Since the introduction of microcomputers in the early 1970's, chemists have used a number of formats to deal with the large amounts of data produce by scientific instruments. The significant initial limitation, that of available storage space, resulted in two different approaches (1) the use of a ASCII text file format (JCAMP-DX) [29] with options for text based compression of data and (2) binary file format (netCDF) [30] where the file structure is inherently more space efficient. Both the Analytical Data Interchange (ANDI) format [31, 32] (built using netCDF) and JCAMP-DX are still in use today with the JCAMP-DX specification more prevalent because of its text-based format.

The Joint Committee on Atomic and Molecular Physical Data (JCAMP) under the International Union of Pure and Applied Chemistry (IUPAC) has published a number of versions of the data exchange (DX) standard for near-infrared, infrared, and ultraviolet–visible spectrophotometry, mass spectrometry, and nuclear magnetic resonance. JCAMP-DX is a file specification consisting of a number of LABELLED-DATA-RECORDs or LDRs. These are defined to allow reporting of spectral metadata and raw/processed instrument data. Figure 4 shows an example mass spectrum in JCAMP-DX format.

Although the JCAMP-DX file format is widely used for export and sharing of spectral data, the specification has not been updated for over 10 years and as a result has limitations in terms general metadata support (static set of LDRs), technique coverage, and is prone to errors/alteration for unintended uses—which breaks compatibility with readers. As a result, an effort was started in 2001 to develop an XML format to replace the suite of JCAMP-DX specifications. The Analytical Information Markup Language (AnIML) [33] is an effort to 'develop a data standard that can be used to store data from any analytical instrument'. This lofty goal has led to a long development process that will be completed in 2016, and result in a formal standard through the American Society for Testing and Materials (ASTM).

AnIML defines a core XML schema for basic elements that will contain data and then uses an additional metadata dictionary, and AnIML Technique Definition Document (ATDD) to prescribe the content of an AnIML file for a particular instrumental technique [33]. This approach makes the format flexible so that it can be used to represent data of all types, from a single datapoint, to a complex array of three-dimensional data. In addition, information about samples, sample location (relative to introduction into an instrument), analytes and instrumental parameters are stored with the raw instrument data. Figure 5 shows an example AnIML file.

How is scientific data stored?

In addition to knowing what scientific data is and how it is represented, it is important to consider how it is stored (and hopefully annotated). Outside of scientific articles, scientific data is published in many databases where the data can be compared with other like data in order to show trends/patterns and afford a higher-level of knowledge mining. Commonly, these are implemented using Structured Query Language (SQL) based relational databases such as MySQL [34], MS SQL Server [35], or Oracle [36]. These software store data in tables and link them together via fields that are unique keys. SQL based software is very good for well-structured information that can be represented in a tree format (rigid schema).

```
data_1000041
loop_
_publ_author_name
'Abrahams, S C'
'Bernstein, J L'
_publ_section_title              Accuracy of an automatic diffractometer. ...
_journal_coden_ASTM              ACCRA9
_journal_name_full               'Acta Crystallographica (1,1948-23,1967)'
_journal_page_first              926
_journal_page_last               932
_journal_paper_doi               10.1107/S0365110X65002244
_journal_volume                  18
_journal_year                    1965
_chemical_formula_structural     'Na Cl'
_chemical_formula_sum            'Cl Na'
_chemical_name_systematic        'Sodium chloride'
_space_group_IT_number           225
_symmetry_cell_setting           cubic
_symmetry_Int_Tables_number      225
_symmetry_space_group_name_Hall  '-F 4 2 3'
_symmetry_space_group_name_H-M   'F m -3 m'
_cell_angle_alpha                90
_cell_angle_beta                 90
_cell_angle_gamma                90
_cell_formula_units_Z            4
_cell_length_a                   5.62
_cell_length_b                   5.62
_cell_length_c                   5.62
_cell_volume                     177.5
_refine_ls_R_factor_all          0.022
_cod_database_code               1000041
loop_
_symmetry_equiv_pos_as_xyz
x,y,z
y,z,x
...
1/2+z,y,1/2-x
1/2+z,1/2+y,-x
loop_
_atom_site_label
_atom_site_type_symbol
_atom_site_symmetry_multiplicity
_atom_site_Wyckoff_symbol
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
_atom_site_occupancy
_atom_site_attached_hydrogens
_atom_site_calc_flag
Na1 Na1+ 4 a 0. 0. 0. 1. 0 d
Cl1 Cl1- 4 b 0.5 0.5 0.5 1. 0 d
loop_
_atom_type_symbol
_atom_type_oxidation_number
Na1+ 1.000
Cl1- -1.000
```

Fig. 3 Example CIF file for NaCl

However, large sets of research data do not fit rigid data models, as by its very nature scientific data is high variable in structure.

Advances in the area of big data have attempted to address the non-uniformity in aggregate datasets by using different data models. Recently, there has been a major shift toward graph databases in support of big data applications across a variety of disciplines. Storing and searching large, often heterogeneous, datasets in relational databases creates problems with speed and scale up [37]. As a result, many companies with large amounts of data have turned to graph databases (one of

```
##TITLE= 2-Chlorophenol
##JCAMP-DX= 5.00   $$ ISAS JCAMP-DX program (V.1.0)
##DATA TYPE= MASS SPECTRUM
##DATA CLASS= PEAKTABLE
##ORIGIN= H. Mayer, ISAS Dortmund
##OWNER= COPYRIGHT (C) 1993 by ISAS Dortmund, FRG
##SPECTROMETER/DATA SYSTEM= Finnigan MAT Magnum
##INSTRUMENTAL PARAMETERS= LOW RESOLUTION
##.SPECTROMETER TYPE= TRAP  $$ ion trap spectrometer
##.INLET= GC                $$ gas chromatograph as inlet
##.IONIZATION MODE= EI+     $$ electron impact ionization with positiv polarity
##.BASE PEAK= 128
##.BASE PEAK INTENSITY= 687729 COUNTS
##.RIC= 3063043
##XUNITS= M/Z
##YUNITS= RELATIVE ABUNDANCE
##NPOINTS= 26
##PEAK TABLE= (XY..XY)
50, 5.84
51, 9.55
52, 4.19
53, 1.12
54, 12.67
60, 3.80
61, 10.16
62, 13.47
63, 58.30
64, 60.43
65, 33.02
66, 4.32
72, 1.70
75, 1.62
91, 1.03
92, 24.95
93, 4.20
94, 1.25
99, 7.20
100, 19.83
101, 3.45
102, 6.47
128, 100.00
129, 6.52
130, 32.45
131, 2.13
##END=
```

Fig. 4 JCAMP-DX format mass spectrum file for 2 chlorophenol

many NoSQL type databases where 'NoSQL' stands for 'Not only SQL') where data is stored as RDF subject-object-predicate 'triples'. In comparison to relational databases, graph databases are considered schema-less where the organization of the data is more natural and not defined by a rigid data model. Essentially, any set

```
<?xml version="1.0" encoding="UTF-8"?>
<AnIML xmlns="urn:org:astm:animl:schema:core:draft:0.90" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    version="0.90" xsi:schemaLocation="urn:org:astm:animl:schema:core:draft:0.90
    http://animl.cvs.sourceforge.net/viewvc/animl/schema/animl-core.xsd">
    <SampleSet>
        <Sample name="Test Sample" sampleID="sample1">
            <Category name="Description">
                <Parameter name="Descriptive Name" parameterType="String">
                    <S>Pond water sample from retention pond by the arena parking garage</S>
                </Parameter>
            </Category>
        </Sample>
    </SampleSet>
    <ExperimentStepSet>
        <ExperimentStep name="Analysis" experimentStepID="step1">
            <Method>
                <Category name="Common Method">
                    <Category name="Instrument Settings">
                        <Parameter name="Measurement Type" parameterType="String">
                            <S>Single reading</S>
                        </Parameter>
                    </Category>
                </Category>
            </Method>
            <Result name="Spectrum">
                <SeriesSet name="Spectrum" length="1">
                    <Series name="Wavelength" dependency="independent" seriesID="wavelength1" seriesType="Float32">
                        <IndividualValueSet>
                            <F>253.2</F>
                        </IndividualValueSet>
                        <Unit label="nm">
                            <SIUnit factor="1e-9" exponent="1">m</SIUnit>
                        </Unit>
                    </Series>
                    <Series name="Absorbance" dependency="dependent" seriesID="absorbance1" seriesType="Float32">
                        <IndividualValueSet>
                            <F>0.2486</F>
                        </IndividualValueSet>
                        <Unit label="A">
                            <SIUnit exponent="1" factor="1">1</SIUnit>
                        </Unit>
                    </Series>
                </SeriesSet>
                <Category name="Measurement Description">
                    <Parameter name="Experiment Duration" parameterType="Float32">
                        <F>20.0</F>
                        <Unit label="ms">
                            <SIUnit exponent="1" factor="1e-3">s</SIUnit>
                        </Unit>
                    </Parameter>
                </Category>
            </Result>
        </ExperimentStep>
    </ExperimentStepSet>
</AnIML>
```

Fig. 5 Example AnIML file—a single reading of absorbance

of RDF subject-predicate-object triples can be thought of as a three-column table in a relational database. Software used to store RDF data is called triple stores [38]—or quad stores [39] if an additional column for a named graph identifier is added. Data in these databases can be searched using the World Wide Web consortium (W3C) defined SPARQL query language [6].

In chemistry there are many websites that show the power of using a database to store large amounts chemical data made available for free or via paid access. Increasingly these sites are being used for basic research and industrial applications as they provide a way to; identify property trends; search for the existence of compounds; show property-structure relationships; and create datasets to build system models. Some highlights are:

- PubChem [40]—chemical, substance, and assay data available with over 91 million compounds. Has user API to downloading data and RDF querying.
- ChemSpider [41]—chemicals, instrument data, and property data for over 56 million compounds. Links to suppliers, literature articles, patents. Has limited API and RDF/XML download
- Dortmund Data Bank [42]—curated property data for over 53,000 compounds. Limited set can be searched for free.
- Cambridge Crystallographic Data Centre [43]—over 833,000 crystal structures (CIF files). Limited set can be searched for free.

What is the best way to communicate context?

Given that the global aggregation of research data is the goal, an important component that is needed relative to any type of framework is a formal definition of the meaning of the data and metadata (contextual data). As mentioned above, current scientific practices are lacking in the generation/reporting of contextual data as researchers are only considering their audience to be human (where meaning is either implicit or can be inferred). If data/metadata is migrated to computers systems, some mechanism to articulate the meaning of the data and metadata is required as storing text in a database is just that—text—to a computer. Through the development of the semantic web this can be achieved through the use of an ontology, or a suite of ontologies. Ontologies are the 'formal explicit description of concepts in a domain of discourse' [44], or an agreed standard for describing the concepts within a field of study. In the recent move toward the semantic web, the importance of ontologies and their unified representation cannot be understated. In 2004 (and updated in 2009) the W3C released the Web Ontology Language (OWL) [45] as a standard way to represent ontologies in RDF.

How best to save, organize, archive, and share data?

Even with all the developments mentioned above there are still challenges that have not been solved. In a nutshell, the problem is that the solutions currently available have been built in isolation (by necessity limiting the scope makes projects more tractable), have little/no machine actionable semantic meaning, are too rigid, are not easy to extend (without breaking existing systems), and are tied heavily to their implementation. As a result, although data is available from many sources it is difficult and time consuming to integrate that data. It is also difficult to search across this heterogeneous pool of information as everyone identifies things differently—there is no broad use of agreed ontological definitions of terms.

A solution to these problem requires abstracting the scenario to a higher level where the structure of the data is normalized in the broadest sense such that any data/metadata can be placed in that structure. This is the essence of the SDM. It does not try to define the data/metadata needed to accurately record and contextualize the scientific data, rather it defines its metaframework, and via an ontology its meaning.

The task of defining the meaning of data and metadata that is placed in any metaframework is the purview of the discipline, where standard ontologies should be developed/refined and implemented. Although this might

```
{
    "@context": "http://stuchalk.github.io/scidata/contexts/scidata.jsonld",
    "uid": "STRENDA_ID",
    "title": "STRENDA data",
    "author": [{
        "@type": "dc:creator",
        "name": "Stuart Chalk"
    }],
    "description": "Example of how STRENDA data might be mapped to the SDM",
    "toc": {
        "sections": [
            "enzyme (1A)",
            "preparation (1A)",
            "assay conditions (1A)",
            "activity data (1A)",
            "methodology (1A)",
            "metrics (1B)",
            "inhibition data (1B)",
            "activation data (1B)"
        ]
    },
    "scidata": {
        "@type": "sci:scidata",
        "type": ["property value"],
        "property": ["reaction rate"],
        "kind": ["time based array"],
        "methodology": {
            "@type": "sci:methodology",
            "evaluation": "experiment",
            "aspects": [
                {"@type": "preparation (1A)"},
                {"@type": "methodology (1A)"}
            ]
        },
        "system": {
            "@type": "sci:system",
            "discipline": "biology",
            "subdiscipline": "enzymology",
            "facets": [
                {"@type": "enzyme (1A)"},
                {"@type": "assay conditions (1A)"},
                {"@type": "metrics (1B)"},
                {"@type": "substrate (1A)"}
            ]
        },
        "dataset": {
            "@type": "sci:dataset",
            "name": "STRENDA dataset",
            "datagroup": [
                {"@type": "activity data (1A)"},
                {"@type": "inhibition data (1B)"},
                {"@type": "activation data (1B)"}
            ]
        }
    }
}
```


Fig. 6 STRENDA Data Categories [52] mapped into the SDM structure

```
{
    "@context": [
        "http://stuchalk.github.io/scidata/contexts/scidata.jsonld",
        {"dc": "http://purl.org/dc/terms/" },
        {"@base": "http://stuchalk.github.io/scidata/"}
    ],
    "@id": "identifier",
    "uid": "dc:identifier (string)",
    "title": "dc:title (string)",
    "author": [{
        "@id": "author/1/",
        "name": "foaf:name (string)",
        "organization": "foaf:organization (string)",
        "email": "foaf:mbox (string)",
        "orcid": "dc:identifier (string)"
    }],
    "description": "dc:description (string)",
    "publisher": "dc:publisher (string)",
    "keywords": "dc:subject (string)",
    "version": "dc:hasVersion (integer)",
    "date": "dc:subject (string)",
    "permalink": "dc:identifier (uri)",
    "related": ["one or more external links (uri)"],
    "toc": {
        "@id": "toc/",
        "sections": ["one or more internal links (uri)"]
    },
    "scidata": {
        "@id": "scidata/",
        "type": ["type of data (from enum list e.g. 'property value')"],
        "property": ["list of one to many properties (string)"],
        "kind": ["list of one to many kinds of data (set list)"],
        "methodology": {
            "@id": "methodology/",
            "evaluation": "how the data was obtained (enum list e.g. 'experimental')",
            "aspects": ["measurement","procedure","resource","calculation","basisset","software"]
        },
        "system": {
            "@id": "system/",
            "discipline": "science area",
            "subdiscipline": "subdiscipline area",
            "facets": ["organism","compound","molecularsystem","chemicalsystem","material","condition"]
        },
        "dataset": {
            "@id": "dataset/",
            "uid": "unique identifier (string)",
            "name": "name of dataset (string)",
            "source": "which aspect was used to obtain data (internal uri)",
            "scope": "which facet is this measured on (internal uri)",
            "attribute": ["one or more attributes about the dataset (parameter)"],
            "datapoint": [{ [55 lines]
            "dataseries": [{ [22 lines]
            "datagroup": [{ [6 lines]
        }
    },
    "references": [{
        "@id": "reference/1/",
        "citation": "textual citation (string)",
        "url": "dc:identifier (uri)"
    }],
    "rights": {
        "@id": "rights/1/",
        "license": "dc:rights (uri)",
        "holder": "dc:rightsHolder (string)"
    }
}
```

(See figure on previous page.)
Fig. 7 The top-level structure of the SciData Data Model (information in [] indicates the number of lines of hidden code, "dc" stands for "Dublin Core")

seem a significant challenge, previous work to standardize the reporting of chemical data can be repurposed to fit this need. For instance, metadata on safety would logically come the new Globally Harmonizes System (GHS) of Classification and Labeling [46], metadata for functional groups of organic compounds would come from the IUPAC Blue book on organic compound nomenclature [47], or for inorganic naming from the IUPAC Red Book [48]. In the biosciences existing work on 'minimal information standards' such as the Minimal Information About a Microarray Experiment (MIAME) [49], Minimal Information Required for a Glycomics Experiment (MIRAGE) [50], and Standards for Reporting Enzymology Data (STRENDA) [51] could be reused in the SDM without much alteration. Figure 6 shows an example of how categories of STRENDA data/metadata could logically be mapped to the SDM.

In order to reinvent how science saves, searches, and re-uses data the implemented solution must have a low barrier to adoption by scientists. While the individual researcher may be excited to use a globally searchable dataset(s), they do not want to be burdened with IT related issues in order to access or implement it. Although the SDM is designed to be format/implementation agnostic, the JSON-LD standard is perfect for representation of the data model as it is a simple text-based encoding, that can handle the types of data needed for the model, and is built to translate to RDF. Examples below that use the SDM are formatted in JSON-LD.

The goal of science is to share research data such that the community can search and use it to advance science. Based on the discussion above, initially one might think that a system for this should be based on a graph database because of its inherent flexibility (anything can be linked to anything) as opposed to relational databases (where data is in tables and linked via unique keys). However, implementing a graph database without any kind of structure would be equivalent to trying to search the current heterogeneous landscape of research data—impossible because nothing is standardized (for example, think about how many ways a scientist could indicate that they used spectrophotometry in their work). What is needed is a hybrid model where a framework for the data and metadata from scientific experiments is used to provide organization (separate from the scientific data/metadata), yet allows flexibility in the types of data put on the framework via creation of discipline specific descriptions and/or ontologies. This is the premise behind the development of the SDM.

Description of the SciData scientific data model

Detailed below is an initial attempt to create a framework upon which to organize scientific data and its metadata. It is by no means a definitive or complete framework and serves only as a starting point to demonstrate the potential of this idea, and act as a catalyst to encourage other scientists to contribute to its development. None of the elements described below are required, other elements can be added (as long as they have a semantic definition and logically fit the scope), and all elements are open to revision (readers are encouraged to provide feedback). Readers are also encouraged to visit the project website [1] for the current version of the data model.

Figure 7 shows a JSON-LD file that outlines the data model framework. The root level of the structure (everything other than 'scidata') contains general metadata to describe the "data packet", i.e. attribution and provenance. The 'toc' attribute is use to articulate the kinds of methodology 'aspects', system 'facets', and 'dataset' elements the report contains. This is an important feature relative to the federated search of data as mechanisms to limit the size/scope of searches will be important if a global search of such data is to be realized.

The generic container for the data and metadata in the model is 'scidata'. This contains metadata descriptors for the types and formats of data, as well a list of the properties for the data that is being reported. What follows are the three main sections that describe the research undertaken: 'methodology', 'system', and 'dataset'.

```
"dataset": {
    "@id": "dataset",
    "uid": "unique identifier (string)",
    "name": "name of dataset (string)",
    "source": "which aspect was used to obtain data (internal uri)",
    "scope": "which facet is this measured on (internal uri)",
    "attribute": ["one or more attributes about the dataset (parameter)"],
    "datapoint": [{
        "@id": "datapoint",
        "uid": "","name": "","source": "","scope": "",
        "quantity": "quantity name (string)",
        "quantityref": "external reference to a quantity (uri)",
        "property": "property name (string)",
        "propertyref": "external reference to a property (uri)",
        "value": {
            "@id": "numericValue",
            "exact": "exact value or not (boolean)",
            "number": {"@value": 6.022E23,"@type": "xsd:float"},
            "sigfigs": 4,
            "error":  {"@value": 1.23E20,"@type": "xsd:float"},
            "errortype": "what type of error is reported (string)",
            "unit": { [31 lines]
            "unitstr": "unit of measure (string)",
            "unitref": "internal or external reference to a unit of measure (uri)"
        },
        "textstring": {
            "@id": "textString",
            "text": "the textual value (string)",
            "texttype": "the text type (enum list) e.g. plain, JSON, etc...",
            "language": "the language of the text string (string)"
        }
    }],
    "dataseries": [{
        "@id": "dataseries",
        "type": "the type of series (string from enum)",
        "axis": "the axis on a graph the series represents (independent or dependent)",
        "label": "the axis label (string)",
        "parameterArray": [
            {
                "@id": "parameter/1",
                "uid": "","name": "","source": "",
                "scope": "","quantity": "","property": "","propertyref": "",
                "valuearray": {
                    "@id": "valuearray",
                    "numberarray": [0.1042,0.1037]
                }
            },
            {
                "@id": "parameter/1",
                "textarray": {
                    "@id": "textarray",
                    "textarray": "an array of textual values (strings)"
                }
            }]
    }],
    "datagroup": [{
        "@id": "datagroup",
        "uid": "","name": "","source": "","scope": "","attribute": [""],
        "datagroup": [{}],
        "dataseries": [{}],
        "datapoint": [{}]
    }]
}
```


Fig. 8 The dataset structure of the SciData Data Model

Methodology

Similar to the 'Experimental' section in a research paper the 'methodology' section contains metadata about how laboratory experiments, computer calculations, or theoretical analysis was done to arrive at the data in the packet. The 'evaluation' term indicates which of these approaches was used to obtain the data. Each of the different 'aspects' of the methodology is reported as a separate section (JSON object) and any/all metadata that are relevant to the methodology of the research can and should be included. Although in the diagram above the aspects of 'measurement', 'procedure', 'resource', 'calculation', 'basisset', and 'software' are shown, these are just examples of aspects that might be reported here. The SDM defines only 'methodology', 'evaluation' and 'aspects' here with metadata under 'aspects' included as needed/available, be it semantically annotated or not. It is envisioned that both general and discipline specific 'aspects' will be developed based on domain specific agreement on best practices for inclusion of minimal metadata and/or default ontological definitions (as discussed above).

System

The 'system'section contains data that is normally reported in different places in a research article. For any scientific research that is performed there is a system that the research is working with/on. A description of the compound(s), organism(s), or material(s) that the data is about needs to be articulated so that the scope of what the data describes can be characterized. It is also important to be able to report the condition(s) under which the data was recorded, the time-point or timeframe at or over which it was performed, etc. Several example facets are listed in the schema below, but again none are required and others can be included as needed to characterize the data. Just like the 'methodology' sections' 'aspects', the 'system' sections' 'facets' is a flexible part of the framework that can hold metadata about one to many 'facets' in addition to general descriptive terms about the discipline that the data is nominally from. Again the data model defines only 'system', 'discipline', 'subdiscipline', and 'facets' here with metadata under 'facets' included as needed/available, be it semantically annotated or not.

```
"unit": {
    "@id": "unitOfMeasure",
    "name": "the name of the unit",
    "unitsystem": "the unit system that the unit is part of (string)",
    "unitsystemref": "external reference to the unit system that the unit is part of (uri)",
    "quantity": "","quantityref": "",
    "abbrev": "unit abbreviation (string)",
    "symbol": "unit symbol (string)",
    "unitinsi": "description or definition of a unit in scientific notation (siunit)",
    "siunit": {
        "@id": "siUnit",
        "name": "the name of the SI unit (string)",
        "prefix": "the SI prefix (enum list)",
        "power": "the unit power (integer)"
    },
    "siunitstr": "the SI unit (string)",
    "siunitref": "external reference to a SI unit (uri)",
    "factor": {
        "@id": "conversionFactor",
        "fromunit": "unit from which the conversion is done",
        "tounit": "unit to which the conversion is done",
        "addend": "value to be added to the 'from unit' value (float)",
        "minuend": "value that the 'from unit' value is to be subtraced from (float)",
        "subtrahend": "value to be subtraced from the 'from unit' value (float)",
        "multiplier": "value to multiply the 'from unit' value by (float)",
        "dividend": "value that the 'from unit' value is to be divided into (float)",
        "divisor": "value to be divided into the 'from unit' value (float)",
        "exact": "indication that the conversion factor is an exact number or not (boolean)"
    },
    "factorref": "external reference to a conversion factor (uri)"
},
"unitstr": "unit of measure (string)",
"unitref": "internal or external reference to a unit of measure (uri)"
```

Fig. 9 Metadata for units in the SciData data model

Dataset

The final section of 'scidata', the 'dataset', is of course the most important (Fig. 8). Dataset is used as a descriptor here to indicate that it is a generic container for data that can logically be reported as a set. The level and scope of the aggregation for a 'dataset' can be at any scale (and is at the discretion of the researcher) and thus it can be used to report a single piece of data or all of the data from a large research study. Within 'dataset' data can be organized/reported in multiple ways. Individual pieces of data are added to the 'datapoint' section and it is implied that there is no relationship between values included. Data that is logically related to other data, either as a time or property series or correlated data such as a spectrum (multiple correlated arrays) are stored in the 'dataseries' section, either directly under 'dataset' or as part of a 'datagroup'. Here the array of data that is recorded can be reported as a JSON array, or as a JSON array of internal links (IRI's) to 'datapoint' data. This allows logical arrays of data to be efficiently stored while also allowing series of datapoints that are collected at different times to also be represented.

A 'datagroup' section is used where there is a need to aggregate data together based on a higher-level structure, and the ability to nest a 'datagroup' inside another 'datagroup' makes for hierarchical organization of data that fits the researcher need. It is also important to point out here that the use of 'datagroup', 'dataseries', and 'datapoint' on their own do not provide a semantic meaning of how the data relates to anything else, rather they are a way to compartmentalize data so that it can be related to other things through the use of the 'source' (methodology

Fig. 10 Terms defined in the Scientific Data Model Ontology (SDMO)

```
PREFIX
sci: <http://stuchalk.github.io/scidata/ontology/scidata.owl#>
chem: < http://semanticscience.org/resource/>
SELECT ?report
WHERE
  { ?report <sci:scientificData> ?scidata.
    ?scidata <sci:property> "Refractive Index".
    ?scidata <sci:context> ?context.
    ?system <sci:discipline> "chemistry".
    ?system <sci:systemFacet> ?facet.
    ?facet <sci:compound> ?compound.
    ?compound <chem:CHEMINF_000059> "VEXZGXHMUGYJMC-UHFFFAOYSA-N".
  }
```

Fig. 11 SPARQL query of scientific data

'aspect') and 'scope' (system 'facet') metadata elements. In reports that contain large datasets, across many experiments, this structure provides the maximum flexibility to report data yet still affords the structure that the data model provides.

```
{
    "scope": "molecule/1",
    "quantity": "magnetic field strength",
    "property": "Anisotropic NMR Shielding",
    "value": {
        "number": 3.2,
        "unitstr": "ppm_Hz"
    }
}
```

Fig. 12 Metadata for calculated parameter value

Also, in the dataset sub-framework there are references to 'parameter' types for certain elements. The 'parameter' object is a generic structure that is used as the basis for metadata in 'datapoints', 'attributes', 'conditions' (under 'system'), 'settings' (within 'methodology') and many more. A parameter is a report of a property (with/without its quantity) and its value and thus can be used to describe a wide variety of data/metadata in the data model. Parameter values may either be numeric ('value') or textual ('textstring'), single values or arrays of values. Numeric values are described by metadata that indicates its type (decimal, integer, float, etc.), significant figures, error, and if the value is an exact number or not (useful for calculations). Text values are described by their type (plain, JSON, Extensible Markup Language (XML), etc.)

```
{
    "@context": {
        "quantity": {
            "@id": "https://en.wikipedia.org/wiki/Quantity",
            "@type": "string"
        }
    },
    "scope": "molecule/1",
    "quantity": "magnetic field strength",
    "property": "Anisotropic NMR Shielding",
    "value": {
        "number": 3.2,
        "unitstr": "ppm_Hz"
    }
}
```

Fig. 13 Addition of a JSON-LD context element to the parameter value

and language. Implementers of the SDM can use the 'parameter' type in the definition of 'aspects' or 'facets' instead of having to invent their own data structures. This makes implementation easier and more consist and other parts of the SDM can be re-used in the same manner.

The final, and most fundamental piece of the data model is the representation of units of measure (Fig. 9). Unit metadata is designed to accurately represent any unit likely to be used in the context of scientific research as well as reference other representations of units (via 'unitref'). The user can report a unit without defining it (using 'unitstr'), define it in place using the metadata shown in Fig. 9 ('unit'), or reference a unit defined elsewhere (internally or externally) in the report via 'unitref'. The specification has been written to integrate online representations of units, quantities, prefixes and unit conversion factors, such as those currently available in the QUDT ontologies [53].

```
{
    "@context": {
        "scope": {
            "@id": "http://stuchalk.github.io/scidata/ontology/scidata.owl#dataScope",
            "@type": "@id"
        },
        "quantity": {
            "@id": "http://stuchalk.github.io/scidata/ontology/scidata.owl#quantity",
            "@type": "http://www.w3.org/2001/XMLSchema#string"
        },
        "property": {
            "@id": "http://stuchalk.github.io/scidata/ontology/scidata.owl#property",
            "@type": "http://www.w3.org/2001/XMLSchema#string"
        },
        "value": "http://stuchalk.github.io/scidata/ontology/scidata.owl#numericValue",
        "number": {
            "@id": "http://stuchalk.github.io/scidata/ontology/scidata.owl#number",
            "@type": "http://www.w3.org/2001/XMLSchema#float"
        },
        "unitstr": {
            "@id": "http://stuchalk.github.io/scidata/ontology/scidata.owl#unitOfMeasure",
            "@type": "http://www.w3.org/2001/XMLSchema#string"
        }
    },
    "scope": "molecule/1",
    "quantity": "magnetic field strength",
    "property": "Anisotropic NMR Shielding",
    "value": {
        "number": 3.2,
        "unitstr": "ppm_Hz"
    }
}
```

Fig. 14 Adding all JSON-LD "term definition" declarations to the context

```
{
    "@context": [
        {"sci": "http://stuchalk.github.io/scidata/ontology/scidata.owl#"},
        {
            "@vocab": "http://www.w3.org/2001/XMLSchema#",
            "scope": {
                "@id": "sci:dataScope",
                "@type": "@id"
            },
            "quantity": {
                "@id": "sci:quantity",
                "@type": "string"
            },
            "property": {
                "@id": "sci:property",
                "@type": "string"
            },
            "value": "sci:numericValue",
            "number": {
                "@id": "sci:number",
                "@type": "float"
            },
            "unitstr": {
                "@id": "sci:unitOfMeasure",
                "@type": "string"
            }
        }
    ],
    "scope": "molecule/1",
    "quantity": "magnetic field strength",
    "property": "Anisotropic NMR Shielding",
    "value": {
        "number": 3.2,
        "unitstr": "ppm_Hz"
    }
}
```

Fig. 15 Cleaning up the URI term definitions in the context

```
{
    "@context": [
        "http://stuchalk.github.io/scidata/contexts/scidata_parameter.jsonld",
        {"@base": "http://stuchalk.github.io/scidata/examples/value.jsonld/"}
    ],
    "@id": "",
    "scope": "molecule/1",
    "quantity": "magnetic field strength",
    "property": "Anisotropic NMR Shielding",
    "value": {
        "@id": "value/1",
        "number": 3.2,
        "unitstr": "ppm_Hz"
    }
}
```

Fig. 16 Using an external context file and adding document references to parameter data ("@base" and "@id")

```
Subject <http://stuchalk.github.io/value.jsonld/>
Predicate <http://stuchalk.github.io/scidata/ontology/scidata.owl#dataScope>
Object <http://stuchalk.github.io/value.jsonld/molecule/1>.

Subject <http://stuchalk.github.io/value.jsonld/>
Predicate <http://stuchalk.github.io/scidata/ontology/scidata.owl#numericValue>
Object <http://stuchalk.github.io/value.jsonld/value/1>.

Subject <http://stuchalk.github.io/value.jsonld/>
Predicate <http://stuchalk.github.io/scidata/ontology/scidata.owl#property>
Object (Literal) "Anisotropic NMR Shielding".

Subject <http://stuchalk.github.io/value.jsonld/>
Predicate <http://stuchalk.github.io/scidata/ontology/scidata.owl#quantity>
Object (Literal) "magnetic field strength".

Subject <http://stuchalk.github.io/value.jsonld/value/1>
Predicate <http://stuchalk.github.io/scidata/ontology/scidata.owl#number>
Object (Literal) "3.2E0"^^<http://www.w3.org/2001/XMLSchema#double>.

Subject <http://stuchalk.github.io/value.jsonld/value/1>
Predicate <http://stuchalk.github.io/scidata/ontology/scidata.owl#unitOfMeasure>
Object (Literal) "ppm_Hz".
```

Fig. 17 RDF triples corresponding to the linked data in the JSON-LD parameter file in Fig. 10

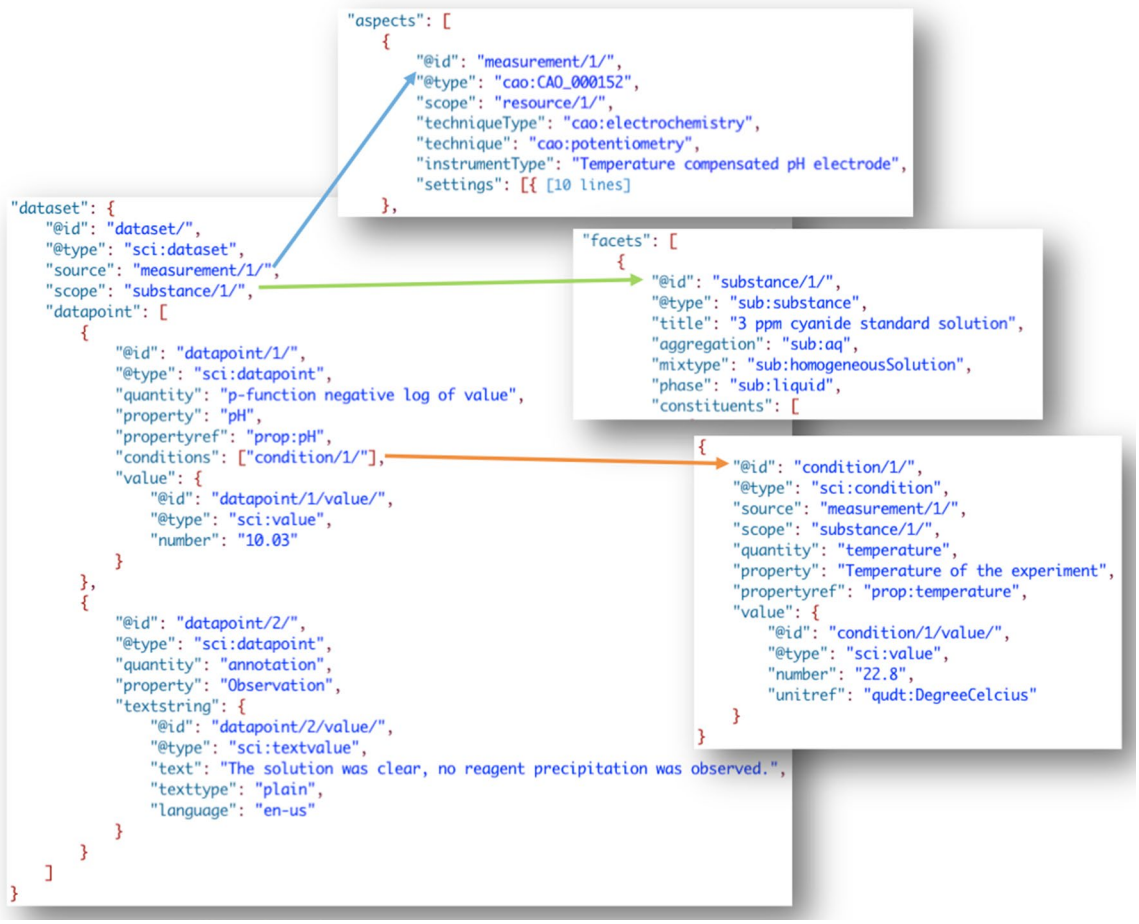

Fig. 18 SciData JSON-LD representation of numeric and textual data points

A scientific data model ontology

In order to give semantic meaning to the framework created by the SciData scientific data model an associated scientific data model ontology (SDMO) has been developed [54]. Each of the metadata elements that are specific to the framework is included in the ontology (over 60) along with reproduction of common metadata terminology (e.g. from Dublin Core [55]). Terms have been grouped into classes (see Fig. 10): metadata, context, dataset, methodology, scientific data, and unit of measure. The semantic annotation of the framework provides the structure that allows SPARQL [6] searches to be constructed that can mine data from multiple sources. Figure 11 shows an example SPARQL query to find all scientific data reports where a refractive index is reported using hydrochloric acid (via InChI Key) in the area of chemistry.

Encoding data in the data model

A full discussion of the use of JSON-LD to encode all of the metadata terms described above is beyond the scope of this paper, however, readers interested in viewing/using JSON-LD to explore this approach can go to the project website [1] where the full set of context files can be accessed along with example data documents. In addition, taking example files [56] and loading them into the JSON-LD playground [10] allows readers to see the RDF generated from JSON-LD encoded data.

To illustrate the use of JSON-LD to represent scientific data and what it means consider the JSON text below for a 'parameter' (Fig. 12). The JSON object represented in the figure is a collection of metadata strings and an embedded JSON object that represents the value of the parameter. Although a human can relatively easily understand the meaning of information presented, a computer sees the structure as strings and a numeric value. In order to add the meaning to this information so that a computer can represent it a JSON-LD context [57] needs to be included to reference the meanings of each of the JSON name-value pairs.

JSON-LD contexts [57] are indicated in a JSON file by the addition of a "@context" JSON object (see Fig. 13). In this example, the "ontological term definition" for "quantity" is added as a shortcut called "quantity" using a Uniform Resource Identifier (URI) (indicated by "@id") where a definition of the term is reported, and the value "magnetic field strength" is indicated by "@type" as being of type "string". Adding

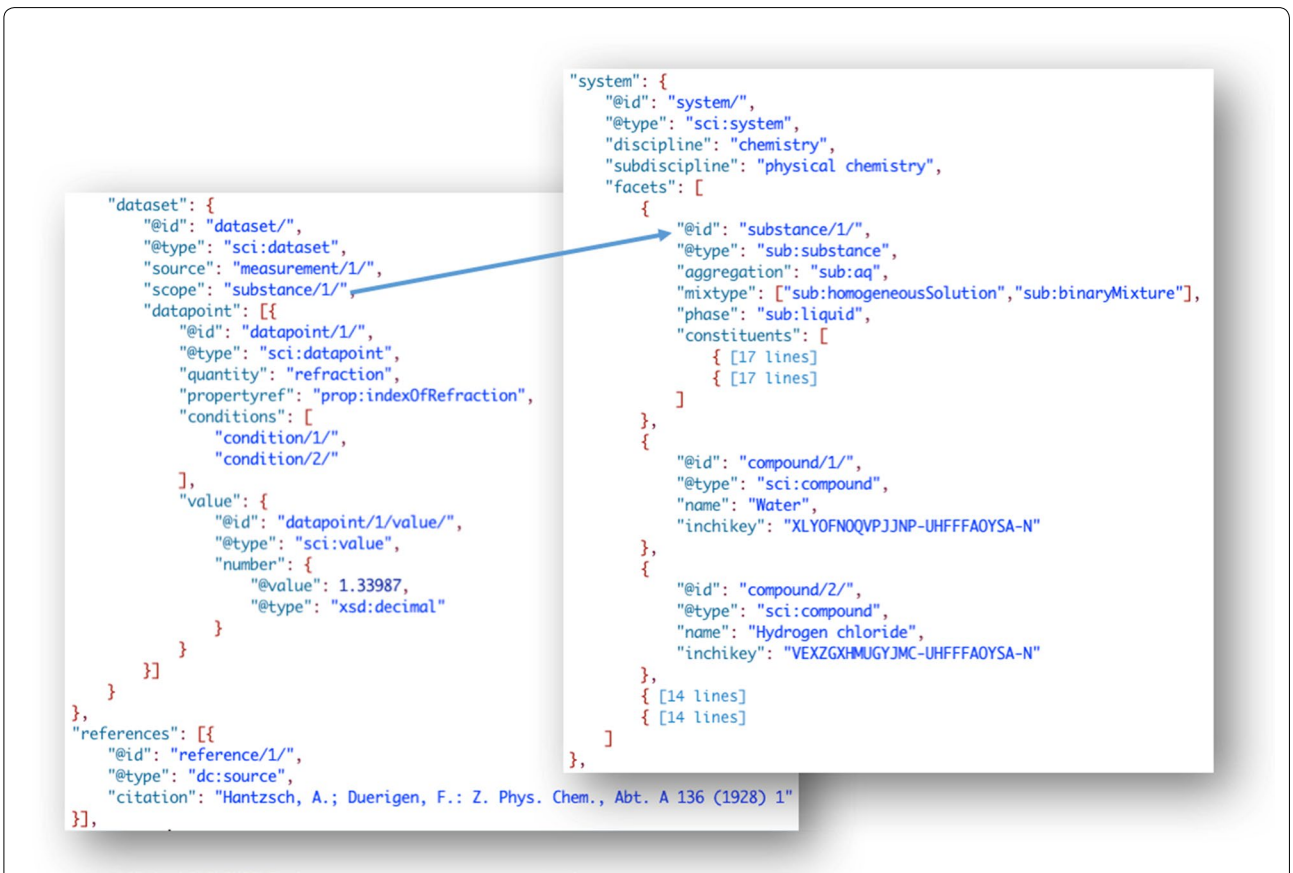

Fig. 19 SciData JSON-LD representation of the refractive index of hydrochloric acid from a research paper

term definitions for all name-value pairs gives the JSON-LD file in Fig. 14.

It can be seen that providing term definitions for all the elements and including all the full URIs makes the file much larger and complicated. Luckily, there are shortcuts that can be implemented to tidy things up quite a bit, i.e. the inclusion of a namespace abbreviation for the ontology URI ("sci") and the definition of a "@vocab" assignment to shorten the references to the data types. Figure 15 shows the cleaned up context array.

Finally, to ensure that this context specification can be used across many documents it can be extracted from the data file and saved as a stand-alone context file that is referenced in the parameter file (Fig. 16). Also note in Fig. 10 that an "@id" field is added to the root of each JSON object. This allows (along with the definition of the "@base" attribute) the generation of a unique URI for the parameter and separately its value. Copying and pasting this document into the JSON-LD playground [10] results in the triples shown in Fig. 17.

Application of the data model

The following portions of four examples show the application of the data model to different data needs. Each of these examples can be found in full on the example page of the project website 1. Additional examples showing the conversion of example data from PubChem, the Dortmund Data Bank, and a CIF file are also included on this page along with XML Stylesheet Language Transformation (XSLT) [58] files used to convert them.

The pH of a solution

This is an example of the most generic type of data—that of individual data points. In of itself a data point is the reporting of a numeric (or textual) value with or without a unit. In this case pH is measured along with an observation of a solution (Fig. 18). Included with the data are references to other parts of the file that contain the data about the measurement, substance, and condition under which the measurement was made (see [56] for complete file).

A measured property extracted from the literature

In order to make use of data reported previously in this linked-data age it is necessary to go to an original article and extract the reported value and its metadata into the data model. Below, in Fig. 19, is the data and original paper reporting the refractive index of a hydrochloric acid solution. Although not shown, the file also contains information about the measurement and conditions equivalent to that shown in Fig. 18 (see [56] for complete file).


```
"dataset": {
    "@id": "dataset/",
    "@type": "sci:dataset",
    "source": "measurement/1/",
    "scope": "substance/1/",
    "datagroup": [{
        "@id": "datagroup/1/",
        "@type": "sci:datagroup",
        "type": "spectrum",
        "attribute": [
            {
                "@id": "attribute/1/",
                "@type": "sci:attribute",
                "quantity": "count",
                "property": "Number of Data Points",
                "value": {
                    "@id": "attribute/1/value/",
                    "@type": "sci:value",
                    "number": "16384"
                }
            },
            ...
            { [11 lines]
        ],
        "dataserieses": [
            "dataseries/1/",
            "dataseries/2/"
        ]
    }],
    "dataseries": [
        {
            "@id": "dataseries/1/",
            "@type": "sci:independent",
            "label": "Excitation frequency (Hz)",
            "axis": "x-axis",
            "parameter": {
                "@id": "dataseries/1/parameter/",
                "@type": "sci:parameter",
                "quantity": "frequency",
                "property": "Radiofrequency",
                "valuearray": { [16435 lines]
            }
        },
        {
            "@id": "dataseries/2/",
            "@type": "sci:dependent",
            "label": "Signal (Arbitrary Units)",
            "axis": "y-axis",
            "parameter": {
                "@id": "dataseries/2/parameter/",
                "@type": "sci:parameter",
                "quantity": "Voltage",
                "property": "Free Induction Decay",
                "valuearray": { [16434 lines]
            }
        }
    ]
}
```


Fig. 20 SciData JSON-LD representation of an NMR spectrum of R-(+)-Limonene

NMR spectrum of a sample of R-(+)-Limonene

The majority of scientific data is measured using analytical instrumentation that produces data as spectra, chromatograms (2D and 3D data), kinetic traces, fiagrams, and many more. The collection and storage of this data can be readily done in 'dataseries' but some mechanism is needed to aggregate related 'dataseries'. This can be done

using a generic data group structure where its contents are two 'dataseries' plus additional metadata to describe what type of data group it is. Below (Fig. 20) is an example for storing a Nuclear Magnetic Resonance (NMR) spectrum (see [56] for complete file).

Computational chemistry calculation of electronic properties of glucose

The last example shows how results from computational chemistry calculations can be captured in the SciData format (Fig. 21). A large amount of data is generated from


```
"dataset": {
    "@id": "dataset/",
    "@type": "sci:dataset",
    "source": "calculation/1/",
    "datapoint": [
        {
            "@id": "datapoint/1/",
            "@type": "sci:datapoint",
            "scope": "system/1/",
            "quantity": "energy",
            "property": "SCF electronic energy",
            "value": {
                "@id": "datapoint/1/value/",
                "@type": "sci:value",
                "number": "-332111.1541",
                "unitref": "qudt:Kilocalories"
            }
        },
        ...
        { [11 lines]
    ],
    "datagroup": [
        {
            "@id": "datagroup/1/",
            "@type": "sci:datagroup",
            "scope": "system/1/",
            "group": "Wavefunction",
            "attributes": [{ [10 lines]
            "dataseries": [
                {
                    "@id": "datagroup/1/dataseries/1/",
                    "@type": "sci:dataseries",
                    "name": "Alpha orbital energies",
                    "parameter": { [72 lines]
                },
                { [19 lines]
                { [76 lines]
                { [77 lines]
                { [19 lines]
                { [76 lines]
            ],
            "datagroup": [
                {
                    "@id": "datagroup/1/datagroup/a/",
                    "@type": "sci:datagroup",
                    "group": "Alpha orbitals",
                    "dataseries": [
                        {
                            "@id": "datagroup/1/datagroup/a/dataseries/1/",
                            "@type": "sci:dataseries",
                            "uid": "ao1",
                            "name": "Molecular orbital coefficients",
                            "parameter": { [71 lines]
                        },
```


Fig. 21 SciData JSON-LD representation (partial) of the results from a glucose SCF calculation

the calculations (the JSON-LD file is over 17,000 lines long) yet the use of nested 'datagroup's allows straightforward organization of the spectrum data (see [56] for complete file).

Conclusion

With the current interest in big data and the movement toward open science there is a need for approaches to allow science data to be made available in an open and easily searchable format. This format needs to be flexible enough to accommodate data from scientific experiments of all kinds and the SciData data model and its implementation in JSON-LD fits that need. Assuming that this, or another framework, is accepted by the scientific community to collect, store, and disseminate semantically annotated scientific data, we can move to the next phase of tool development and data integration to move us toward the utopia of open, accessible and reliable semantically annotated scientific data.

Abbreviations

CCDC: Cambridge Crystallographic Data Centre; JSON: JavaScript Object Notation; JSON-LD: JavaScript Object Notation for Linked Data; NMR: nuclear magnetic resonance; NoSQL: not only SQL; OWL: Web Ontology Language; REST: Representational State Transfer; RDF: Resource Description Framework; SDM: scientific data model; SDMO: scientific data model ontology; SPARQL: SPARQL Protocol and RDF Query Language; SQL: Structured Query Language; URI: Uniform Resource Identifier; W3C: World Wide Web Consortium; XML: Extensible Markup Language; XSLT: XML Stylesheet Language Transformation.

Acknowledgements

Thanks to Neil Ostlund and Mirek Sopek for comments on the data model.

Competing interests

The author declares no competing interests

Availability of data and materials

All data associated with this project is available at https://github.com/stuchalk/scidata.

Funding

The author was not funded by any entity for this work

Received: 18 March 2016 Accepted: 4 October 2016
Published online: 14 October 2016

References

1. Chalk S (2016) SciData: a scientific data model. http://stuchalk.github.io/scidata/. Accessed 1 March 2016
2. Bird CL, Willoughby C, Coles SJ, Frey JG (2013) Data curation issues in the chemical sciences. Inf Stand Q 25(3):4–12. doi:10.3789/isqv25no3.2013.02
3. Murray-Rust P (2010) What is scientific data? http://blogs.ch.cam.ac.uk/pmr/2010/07/25/pp01-what-is-scientific-data/. Accessed 1 March 2016
4. Hey T, Tansley S, Tolle K (2009) The fourth paradigm: data-intensive scientific discovery. ISBN: 978-0982544204. http://research.microsoft.com/en-us/collaboration/fourthparadigm/
5. W3C (2016) Resource Description Framework (RDF) The World Wide Web Consortium. http://www.w3.org/RDF/
6. W3C (2016) SPARQL query language for RDF The World Wide Web Consortium. http://www.w3.org/TR/rdf-sparql-query/. Accessed 1 March 2016
7. W3C (2016) JSON-LD 1.0: a JSON-based serialization for linked data. The World Wide Web Consortium. http://www.w3.org/TR/json-ld/ Accessed March 1, 2016
8. Fredrich T (2016) What is REST? http://www.restapitutorial.com/lessons/whatisrest.html. Accessed 23 June 2016
9. Lanthaler M, Gütl C (2012) On using JSON-LD to create evolvable RESTful services. In: Third international workshop on RESTful design, ACM, pp 25–32. http://dx.doi.org/10.1145/2307819.2307827
10. W3C (2016) JSON-LD playground http://json-ld.org/playground/. Accessed 1 March 2016
11. Gower B (1997) Scientific method: a historical and philosophical introduction. Routledge. ISBN: 978-0415122825. https://www.amazon.com/dp/0415122821
12. RSC (2016) Development of the periodic table http://www.rsc.org/periodic-table/history/about. Accessed 23 June 2016
13. EOL (2016) What is biological classification? http://eol.org/info/461. Accessed 23 June 2016
14. UCB (2016) Phylogenetic systematics, a.k.a. evolutionary trees: reading trees—a quick review. http://evolution.berkeley.edu/evolibrary/article/phylogenetics_02. Accessed 23 June 2016
15. Langacker P (2012) Grand unification. Scholarpedia 7(10):11419
16. Lysakowski R, Gragg CE (eds) (1994) Computerized chemical data standards: databases, data interchange, and information systems, ASTM. ISBN: 978-0-8031-1876-8. http://www.astm.org/DIGITAL_LIBRARY/STP/SOURCE_PAGES/STP1214.htm
17. Perkins JA (2005) A history of molecular representation. Part one: 1800 to the 1960s. J Biocommun 31(1):1
18. Apodaca R (2007) Everything old is new again—Wiswesser Line Notation (WLN). http://depth-first.com/articles/2007/07/20/everything-old-is-new-again-wiswesser-line-notation-wln/. Accessed 23 June 2016
19. DCIS (2016) SMILES: a simplified chemical language daylight chemical information systems. http://www.daylight.com/dayhtml/doc/theory/theory.smiles.html. Accessed 23 June 2016
20. InChI Trust (2016) InChI and InChIKeys for chemical structures. http://www.inchi-trust.org/. Accessed 23 June 2016
21. OC (2016) OpenBabel: supported file formats and options OpenBabel community. http://openbabel.org/docs/2.3.0/FileFormats/Overview.html. Accessed 23 June 2016
22. Hanson R (2016) Jmol/JSmol file formats/coordinates http://wiki.jmol.org/index.php/File_formats/Coordinates-MOL_and_SD_.28Symyx_MDL.29. Accessed 23 June 2016
23. Murray-Rust P, Rzepa H (2012) Chemical markup language—CML. http://xml-cml.org/. Accessed 23 June 2016
24. W3C (2016) Extensible Markup Language (XML) https://www.w3.org/TR/xml/
25. Bernstein HJ, Bollinger JC, Brown ID, Grazulis S, Hester JR, McMahon B, Spadaccini N, Westbrook JD, Westrip SP (2016) Specification of the crystallographic information file format, version 2.0. J Appl Crystallogr 49(1). http://dx.doi.org/10.1107/S1600576715021871
26. Hall SR, Allen FH, Brown ID (1991) The crystallographic Information File (Cif): a new standard archive file for crystallography. Acta Cryst A 47:655–685
27. IUCr (2016) CIF international union of crystallography. http://www.iucr.org/resources/cif. Accessed 23 June 2016
28. Hall SR (1991) The Star file: a new format for electronic data transfer and archiving. J Chem Inf Comput Sci 31(2):326–333. doi:10.1021/ci00002a020
29. IUPAC (2016) IUPAC subcommittee on electronic data standards. http://jcamp-dx.org/. Accessed 1 March 2016
30. UCAR (2016) Network common data form (NetCDF) http://www.unidata.ucar.edu/software/netcdf/. Accessed 23 June 2016
31. ASTM (2016) Standard guide for analytical data interchange protocol for mass spectrometric data—E2078—00(2016) http://dx.doi.org/10.1520/E2078-00R16. Accessed 23 June 2016
32. ASTM (2016) Standard specification for analytical data interchange protocol for chromatographic data—E1947—98(2014) http://dx.doi.org/10.1520/E1947-98R14. Accessed 23 June 2016

33. ASTM (2016) The analytical information markup language (AnIML) AnIML working group. https://www.animl.org/
34. Oracle (2016) MySQL open-source database oracle corporation. http://www.mysql.com/. Accessed 1 March 2016
35. Microsoft (2014) SQL server 2014 microsoft corporation. http://www.microsoft.com/en-us/server-cloud/products/sql-server/. Accessed 1 March 2016
36. Oracle (2016) Oracle database oracle corporation. https://www.oracle.com/database/index.html. Accessed 1 March 2016
37. Reeve A (2012) Big Data and NoSQL: the problem with relational databases EMC corporation. https://infocus.emc.com/april_reeve/big-data-and-nosql-the-problem-with-relational-databases/. Accessed 1 March 2016
38. Sequeda J (2013) Introduction to: Triplestores Dataversity. http://www.dataversity.net/introduction-to-triplestores/. Accessed 1 March 2016
39. Dodds L, Davis I (2012) Linked data patterns. Data management patterns: named graph. http://patterns.dataincubator.org/book/named-graphs.html
40. NLM (2016) PubChem National Institutes of Health. http://pubchem.ncbi.nlm.nih.gov/ Accessed June 23, 2016
41. RSC (2016) ChemSpider: search and share chemistry. http://www.chemspider.com/. Accessed 23 June 2016
42. DDB (2016) Dortmund Data Bank DDBST GmbH. http://www.ddbst.com/. Accessed 23 June 2016
43. CCDC (2016) Cambridge Crystallographic Data Centre Cambridge, UK http://www.ccdc.cam.ac.uk/. Accessed 23 June 2016
44. Noy NF, McGuinness DL (2012) Ontology development 101: a guide to creating your first ontology Stanford University. http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html. Accessed 1 March 2016
45. W3C (2016) OWL 2 Web Ontology Language. The World Wide Web Consortium. http://www.w3.org/TR/owl-overview/. Accessed 1 March 2016
46. UNECE (2007) Globally harmonized system of classification and labelling of chemicals (GHS) (Rev.2). Geneva, Switzerland: United Nations Economic Commission for Europe. http://www.unece.org/trans/danger/publi/ghs/ghs_rev02/02files_e.html. Accessed 23 June 2016
47. IUPAC (1993) Nomenclature of organic chemistry. http://www.acdlabs.com/iupac/nomenclature/. Accessed 23 June 2016
48. Hartshorn RM, Hellwich KH, Yerin A, Damhus T, Hutton AT (2015) Brief guide to the nomenclature of inorganic chemistry. Pure Appl Chem 87(9–10):1039–1049. doi:10.1515/pac-2014-0718
49. FGED (2016) MIAME: minimum information about a microarray experiment. http://fged.org/projects/miame/. Accessed 23 June 2016
50. MIRAGE WG (2016) Minimum Information Required for a Glycomics Experiment Beilstein Institut. http://www.beilstein-institut.de/en/projects/mirage
51. STRENDA Commission (2016) STRENDA: Standards for Reporting Enzymology Data Frankfurt Beilstein Institut. http://www.beilstein-institut.de/en/projects/strenda. Accessed 23 June 2016
52. STRENDA Commission (2016) The STRENDA Guidelines Frankfurt Beilstein Institut. https://www.beilstein-strenda-db.org/strenda/public/guidelines.xhtml. Accessed 23 June 2016
53. Hodgson R, Keller PJ, Hodges J, Spivak J (2014) QUDT: quantities, units, dimensions and data types ontologies TopQuadrant, Inc. http://www.qudt.org/. Accessed 1 March 2016
54. Chalk S (2016) Scientific data model ontology (SDMO) http://stuchalk.github.io/scidata/ontology/scidata.owl. Accessed 23 June 2016
55. DCMI (2016) Dublin core metadata terms ASIS&T. http://dublincore.org/documents/dcmi-terms/. Accessed 1 March 2016
56. Chalk S (2016) SciData: example data files http://stuchalk.github.io/scidata/examples/. Accessed 23 June 2016
57. W3C (2016) JSON-LD 1.0: The Context The World Wide Web Consortium. http://www.w3.org/TR/json-ld/-the-context. Accessed 1 March 2016
58. W3C (2016) The extensible stylesheet language family (XSL). https://www.w3.org/Style/XSL/