This document is no longer the current working document. Please do not edit it. Please make changes to the github HTML version https://github.com/joejimbo/HCLSDatasetDescriptions

Dataset Descriptions: HCLS Community Profile

W3C Interest Group Note

TO DO:

- run sparql queries over chembl to obtain real statistics (?)
- make identifiers.org vocab available (nick & camille)
- chris - chembl rdf subset example (michel)
- ~~ask eric for a URL for the note (michel)~~
- obtain use case from datahub.io (michel)
  - sent email. waiting for response
- develop descriptions
  - core metadata section - joachim, ~~alejandra~~
  - identifiers section - ~~michel~~, joachim
  - provenance section - scott, ~~alejandra~~
  - availability section - ~~alasdair~~, scott
  - statistics - ~~michel~~, alasdair
- Add diagram for provenance of a dataset

Done:

- 6-2-2014 (michel): substantial editing across the document. added boxes for examples. edited identifier section. divided statistics section into two parts (core/enhanced) and added sparql queries

Contributors:

- Peter Ansell, CSIRO, Australia
- Gary D. Bader, The Donnelly Centre, University of Toronto, Canada
- Asuka Bando, NBDC, Japan
- Joachim Baran, Ontario Institute for Cancer Research, Canada
- Alison Callahan, Carleton University, Canada
- José Cruz-Toledo, Carleton University, Canada
- Erich Gombocz, IO Informatics, USA

- Alejandra Gonzalez-Beltran, University of Oxford, UK.
- Paul Groth, VU University Amsterdam, Netherlands
- Melissa Haendel, Oregon Health and Science University, USA
- Harry Hochheiser, University of Pittsburgh, USA. harryh@pitt.edu
- Maori Ito, NIBIO, Japan
- Simon Jupp, EMBL-EBI, UK
- Toshiaki Katayama, Database Center for Life Sciences, Japan
- Kalpana Krishnaswami, Metaome, USA
- Simon Lin, Marshfield Clinic Research Foundation, USA
- Chris Mungall, Lawrence Berkeley National Laboratory, USA
- Nicolas Le Novère, Babraham Institute, UK
- Camille Laibe, EMBL-EBI, UK
- Nick Juty, EMBL-EBI, UK
- James Malone, EMBL-EBI, UK
- Laurens Rietveld, VU University Amsterdam, Netherlands

Editors

- Alasdair J G Gray, Heriot-Watt University, UK.
- Michel Dumontier, Stanford University, USA
- M. Scott Marshall, MAASTRO Clinic, The Netherlands

Abstract

Access to consistent, high-quality metadata is critical to finding, understanding, and reusing scientific data. This document describes a consensus among participating stakeholders in health care and the life sciences domain on the description of datasets using the Resource Description Framework (RDF). This specification meets key functional requirements, reuses existing vocabularies to that extent that it is possible, and addresses elements of data description, versioning, provenance, discovery, exchange, query, and retrieval.

Status of This Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This is an incomplete draft (November 24, 2013; md) of a specification for the description of datasets. This is a live document and is subject to change without notice.

The document was produced by the Semantic Web in Health Care and Life Sciences Interest Group (HCLS), part of the W3C Semantic Web Activity (see charter). Comments may be sent to the publicly archived public-semweb-lifesci@w3.org mailing list.

Publication as an Interest Group Note does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

Table of Contents

1. Introduction

Big Data presents an exciting opportunity to pursue large-scale analyses over collections of data in order to uncover valuable insights across a myriad of fields and disciplines. Yet, as more and more data is made available, researchers are finding it increasingly difficult to discover and reuse these data. One problem is that **data are insufficiently described** to understand what they are or how they were produced. A second issue is that **no single vocabulary provides all key metadata fields** required to support basic scientific use cases. A third issue is that **data catalogs and data repositories all use different metadata standards**, if they use any standard at all, and this prevents easy search and aggregation of data. Therefore, we need a guide to indicate what are the essential metadata, and the manner in which we can express it.

For the purposes of this document, we **define a dataset as** *“A collection of data, available for access or download in one or more formats”*[1] [DCAT]. For instance, a dataset may be generated as part of some scientific investigation, whether tabulated from observations, generated by an instrument, obtained via analysis, created through a mashup, or enhanced or changed in some manner. Research data are available in research publications and

[1] This generalization of the DCAT definition of a dataset removes the restriction that a dataset is “published or curated by a single agent” , thus allowing multiple authors or curators.

supplemental documents, in literature curated databases such as PharmGKB or the CTD, from research repositories such as BioMedCentral-BGI , GigaScience [GigaScience], Nature Publishing Group's Scientific Data [ScientificData], Dryad Digital Repository [Dryad], FigShare [FigShare], Harvard Dataverse [Dataverse]. Cross-repository access is possible through data catalogs such as Neuroscience Information Framework (NIF) [NIF], BioSharing [BioSharing], Identifiers.org Registry [Identifiers.org], Integbio Database Catalog [Integbio], Force11 [Force11], and CKAN's datahub [Datahub].

While several vocabularies are relevant in describing datasets, none are sufficient to completely provide the breadth of requirements identified in Health Care and the Life Sciences. The Dublin Core Metadata Initiative (DCMI) [DCMI] Metadata Terms offers a broad set of types and relations for capturing document metadata. The Data Catalog Vocabulary (DCAT) [DCAT] is used to describe datasets in catalogs, but does not deal with the issue of dataset evolution and versioning. [**add other metadata**] The Provenance Ontology (PROV) [PROV] can be used to capture information about entities, activities, and people involved in producing a piece of data or thing. The Vocabulary of Interlinked Datasets (VOID) [VOID] is an RDF Schema vocabulary for expressing metadata about RDF datasets. Schema.org has a limited proposal for dataset descriptions [SCHEMA]. Thus, there is need to combine these vocabularies in a comprehensive manner that meets the needs of data registries, data producers, and data consumers.

Here we describe the efforts of a multi-stakeholder effort under the auspices of the W3C Semantic Web for Health Care and Life Sciences [HCLS] Interest Group to produce a specification for the description of datasets that meets key functional requirements, uses existing vocabularies, and is expressed using the Resource Description Framework [RDF]. We discuss elements of data description including provenance and versioning, and describe how these can be used for data discovery, exchange, and query (with SPARQL). This then enables the retrieval and reuse of data to encourage reproducible science.

Specifically, we provide

1. A community specification for describing datasets (Section 5) in RDF.
2. Detailed requirements (Section 4) that have been drawn from a wide range of use cases (Section 8).

2. Scope

This document focuses on common data elements and their value sets for the description of data. Although use cases are drawn from Health Care and the Life Sciences, this document will focus on requirements that are broadly applicable.

3. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

The following namespace prefix bindings are assumed unless otherwise stated:

Prefix	Namespace	Description
cito	http://purl.org/spar/cito/	Citation Typing Ontology
dcat	http://www.w3.org/ns/dcat#	Data Catalog
dctypes	http://purl.org/dc/dcmitype/	Dublin Core Metadata Types
dct	http://purl.org/dc/terms/	Dublin Core Metadata Terms
foaf	http://xmlns.com/foaf/0.1/	Friend-of-a-Friend
freq	http://purl.org/cld/freq/	Collection Description Frequency Vocabulary
idot	http://identifiers.org/terms#	Identifiers.org vocabulary
lexvo	http://lexvo.org/ontology#	Lexical Vocabulary
pav	http://purl.org/pav/	Provenance Authoring and Versioning ontology (PAV)
prov	http://www.w3.org/ns/prov#	PROV Ontology (PROV-O)
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#	Resource Description Framework
rdfs	http://www.w3.org/2000/01/rdf-schema#	RDF Schema
sorg	http://schema.org/	Schema.org
sd	http://www.w3.org/ns/sparql-service-description#	SPARQL 1.1 Service Description
xsd	http://www.w3.org/2001/XMLSchema#	XML Schema
vann	http://purl.org/vocab/vann/	A vocabulary for annotating vocabulary descriptions (VANN)

4. Functional Requirements

In this section we describe a set of essential functionality that the dataset descriptions must provide.

4.1 Dataset Identification, Description, Licensing, and Provenance

High quality data descriptions are necessary to understand the nature and provenance of data including what the data is, the format the data is represented with, where the data can be retrieved from, what license is associated with the dataset, how it was generated, when it was generated, and who generated it. Importantly, such dataset descriptions should provide globally unique identifiers for specific versions and formats of datasets so that they may be used and referenced by others in downstream analyses.

4.2 Dataset Discovery (via Catalog)

Data consumers need an easy mechanism to find datasets of interest. Data catalogs, identifier indices, and data standard registries such as BioSharing [BioSharing], identifiers.org Registry [Identifiers.org], datahub.io [Datahub], NIF [NIF] are all important infrastructure that make it easier for users to find relevant data and even discuss their quality or utility. The availability of rich metadata will enable users to perform faceted searches of data items by placing restrictions the values of specific metadata fields.

4.3 Exchange of Dataset Descriptions

Dataset descriptions should be in a standardized format that enables facile exchange between data providers. This would allow data catalogs to synchronize and specialize their offerings. For example, BioDBCore [BioDBCore,Gaudet et al 2010] is a community-driven effort overseen by the International Society for Biocuration [ISB], which defines a checklist, or minimum information standard, including the core attributes for the description of biological databases. [BioDBCore]

4.4 Dataset Linking

The integration of data typically involves establishing a similarity between resources described in different datasets. Since datasets naturally evolve with time, it is important that information regarding the nature of the linking can be adequately captured e.g. the version and format of files and software used to generate the links.

A dataset may incorporate, or link to, data in other datasets, e.g. in the creation of a data mashup. Rather than repeating the dataset description of the source datasets, the derived dataset would link to the dataset description of the specific instance that they loaded.

4.5 Content Summary

A breakdown of the entities and their relationships in a dataset is useful for communicating the structure and content of the dataset. This information can be used to enable dataset navigation, facilitate query construction and compare different versions of a dataset.

4.6 Monitoring of Dataset Changes

The reproducibility of scientific investigations is often tied to the availability of the original data. However, as original dataset grow or change with time, it becomes important to understand what changes have occurred and how these may affect dependent analyses. A dataset description should provide the means by which to compare different dataset versions.

5. Dataset Description Levels

In this section, we describe the W3C HCLS recommendation for rich descriptions of datasets. There are three levels each covering a different type of resource describing the data:

- **Summary Level:** The summary level provides a description of a dataset that is independent of a specific version or format.

- **Version Level:** The version level captures version-specific characteristics of a dataset.
- **Distribution Level:** The distribution level captures metadata about a specific form and version of a dataset.

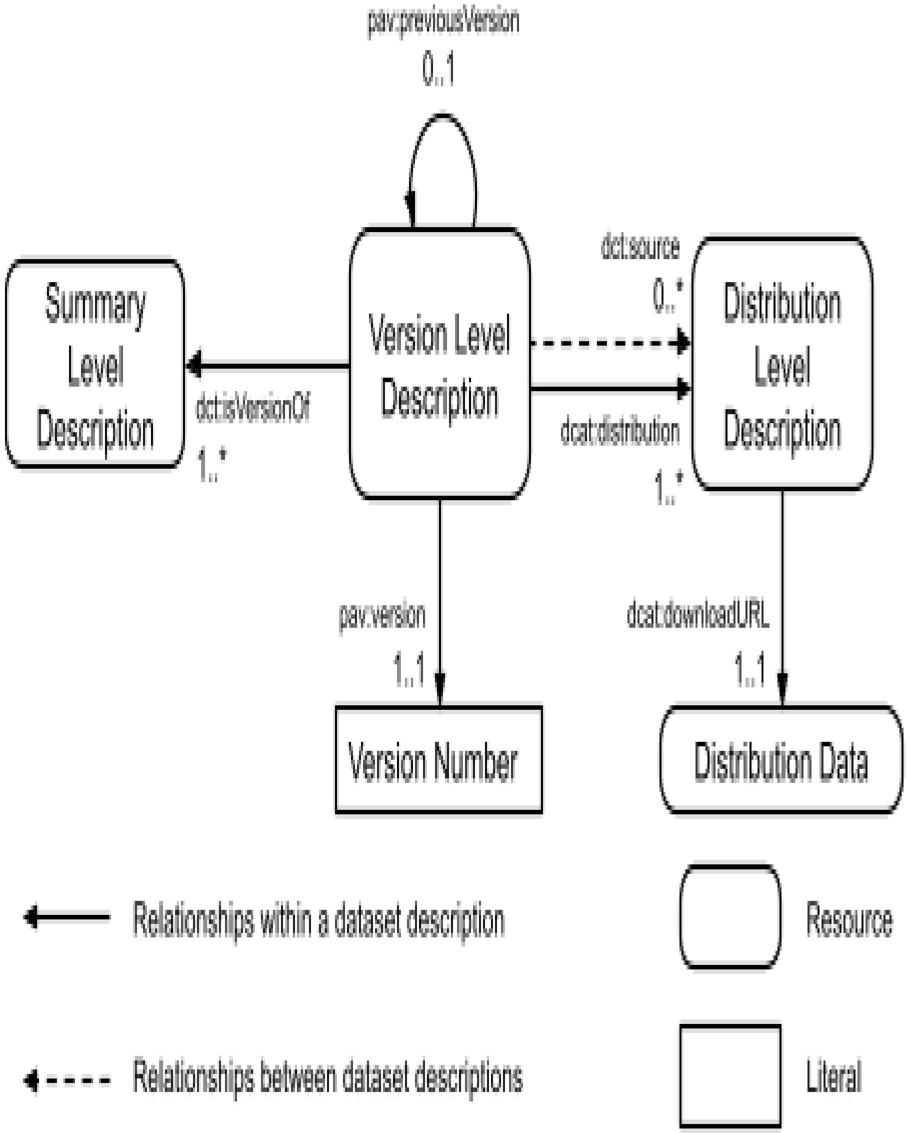

pav:previousVersion
0..1
Summary Level Description
Version Level Description
Distribution Level Description
dct:isVersionOf
1..*
dct:source
0..*
dcat:distribution
1..*
pav:version
1..1
dcat:downloadURL
1..1
Version Number
Distribution Data
Relationships within a dataset description
Relationships between dataset descriptions
Resource
Literal

Figure 1: An overview of the relationships between dataset description levels. A single summary level description for a dataset will be related to one or more version level descriptions using dct:isVersionOf. Incremental versions may be specified using pav:previousVersion. Each version level description will be linked to one or more distribution level descriptions using dcat:distribution. [comment].

Note that a distribution is the realisation of the data in a specific file format. The different distributions of a dataset may not be semantically equivalent due to differences between the data formats. For example, a chemical dataset may be released as a rich RDF dataset but also as an SD file. In this case, the SD file[2] will not contain the same data content as the RDF dataset.

For example, consider the ChEMBL dataset. It is manifested in 17 different versions, of which each version can be accessed in a variety of data formats. Thus, to be fully conformant with this specification, there would be one Summary Level description that is linked to 17 Version Level descriptions, and each of the Version Level descriptions would be linked to their corresponding Distribution Level descriptions. This is captured by the following example expressed in RDF/Turtle format:

```
# Summary Level Description
:chembl a dctypes:Dataset .

# Version Level Description
:chembl17 a dctypes:Dataset ;
    pav:isVersionOf :chembl;
    dcat:distribution :chembl17rdf,
        :chembl17relational, :chembl17csv .

# Distribution Level Descriptions
:chembl17rdf a dcat:Distribution, void:Dataset .
:chembl17relational a dcat:Distribution .
:chembl17csv a dcat:Distribution .
```

The properties expected for each metadata profile are given in the table below.
Table available in separate google spreadsheet for ease of editing![3]

6. Metadata Guidance Notes

Literals

Dates

Dates should be given as accurately as possible. It is recommended that xsd:dateTime [xsd:dateTime] is used (YYYY-MM-DDThh:mm:ss[Z], where [Z] is optionally specified as Z for UTC time or provided as a time zone offset e.g. -04:00 for EDT). In the case that the time is not precisely known, we recommend using the first valid value for the fields that are not known, i.e. the value ‘01’ for days and months, the value ‘00’ for hours, minutes and seconds.

Values as Strings

Values should be stated with a language tag, unless capturing an identifier or some structured value e.g. version identifier. Values may be captured in multiple languages.

[2] http://en.wikipedia.org/wiki/Chemical_table_file#SDF

[3] The complete table used in discussions: http://tinyurl.com/datasetdescription

Values as Strings or IRIs

For several predicates there is the choice of providing the value as an IRI or a string. We recommend that IRIs are used wherever possible.

6.1 Core Metadata

Dataset identification and declaration of type

It is a good general practice that data represented using Semantic Web technologies be typed as an instance of some class of entities. All datasets are typed as dctypes:Dataset with the exception of RDF formatted datasets which are typed as an instance of a void:Dataset.

```
:chembl
        rdf:type dctypes:Dataset .
```

Title, Alternative Names, and Short Code

A dataset can be known by several names. At least one of these must be provided for use in human interfaces for applications using the dataset. It is expected that the preferred – commonly used – name for a dataset will be provided using the dct:title term, while alternative forms of the name, or older names for the dataset, may be given using the dct:alternative property. We also recommend that data providers specify exactly one short-code for a dataset using the dct:identifier property and that it should be in lower case. For example, to provide a title, alternative name, and identier for the ChEMBL dataset:

```
:chembl
     dct:title “ChEMBL”@en ;
     dct:alternative “ChEMBLdb”@en ;
     dct:identifier “chembl”@en .
```

Description

It is expected that the description will be the same as the text that appears on the web page for the dataset (see below). It should be a few paragraphs that explain to a domain expert the contents of the dataset. Where available, the description should be provided in alternative languages, each with the appropriate language tag. For instance, to specify a description for the chembl dataset:

```
:chembl
        dct:description “ChEMBL is a database of bioactive compounds, their
        quantitative properties and bioactivities (binding constants, pharmacology
        and ADMET, etc). The data is abstracted and curated from the primary
        scientific literature.”@en .
```

Dates of Creation and Issuance

It is essential to know when a dataset came into existence. However for some datasets all that is known is when it was made public. Therefore, it is recommended that at least one of the creation or issued dates are provided. Both MAY be provided. For versioned or distribution dataset descriptions, state the date the dataset was generated using dct:created and/or the date the dataset was made public using dct:created. Both properties are restricted to a xsd:dateTime value. For instance, to specify that the chembl17 dataset (see section 6.4 for more details about :chembl17 declaration) was issued on 29 August 2013:

```
:chembl17
        dct:issued “2013-08-29T00:00:00”^^xsd:dateTime .
```

Note that since the time that the dataset was published is unknown, these are set to the beginning of the time period. If only the month and year had been known, the value would have been given as “2013-08-01T00:00:00”. Where the time is known, the timezone should be provided.

Other, more specific properties, e.g. the date the dataset was authored or curated, MAY be given using terms from the PAV ontology [PAV].

Authorship, Creation, Curation

Details of the individual or organisation responsible for creating a dataset MUST be provided using the dct:creator property. The value should be an IRI for the individual or organisation that can be resolved for more details. We recommend the use of ORCID ID [ORCID] for researchers. The value may be a string if a suitable IRI cannot be supplied.

```
:chembl17
        dct:creator <https://www.ebi.ac.uk/chembl/> .

:chembl_target_targetcmpt_linkset
        pav:authoredBy <http://orcid.org/0000-0002-8011-0300> ;
```

Fine-grained attribution of creation events such as authoring or curation MAY additionally be supplied using the terms from the PAV ontology

- Authorship
    - State the author(s) with pav:authoredBy and linking to URIs for the authors. The date of authorship should be given using pav:authoredOn with a xsd:dateTime.
- Creation
    - State the creator(s) with pav:createdBy and linking to URIs for the authors. The date of authorship should be given using pav:createdOn with a xsd:dateTime.
- Curation
    - State the author(s) with pav:curatedBy and linking to URIs for the authors. The date of authorship should be given using pav:curatedOn with a xsd:dateTime.

```
        :chembl_target_targetcmpt_linkset
                pav:authoredBy <http://orcid.org/0000-0002-8011-0300> ;
```

Declare the tool used to create the dataset using pav:createdWith linking to a URI representing the specific version of the tool.

Publisher

A link to the organisation responsible for publishing the dataset.

```
:chembl
              dct:publisher <http://www.ebi.ac.uk> .
```

Webpage

A link to the human-oriented web page for the dataset is provided using the foaf:page property [FOAF]. Due to the inverse-functional properties, foaf:homepage must not be used as it could result in the data about different versions of a dataset being combined. For instance, to specify that the chembl dataset is accessible at [ChEMBL].

```
        :chembl
                foaf:page <http://www.ebi.ac.uk/chembl/> .
```

Keywords

Keywords and topics of coverage of the dataset MAY be given. It is recommended that such terms are drawn from a suitable domain vocabulary and declared using the dcat:theme predicate. The dcat:keyword predicate is provided for the rare occasions where there is not a suitable term in a vocabulary

```
        @prefix ncit: <http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#>
        :chembl
                dcat:theme ncit:C48807. # chemical
                dcat:keyword “chemical”^^xsd:string, “assay”^^xsd:string .
```

Note that the ncit namespace refers to the National Cancer Institute (NCI) Thesaurus.

Licensing and Rights

The license under which the data is published MUST be stated to enable the reuse of the data. The details of the license may be summarised in a rights statement.

```
:chembl
                dct:license <http://creativecommons.org/licenses/by-sa/3.0/> .
```

Language

Declare the languages the data is published in using dct:language and values taken from the Lexvo.org Ontology [Lexvo].

```
        :chembl
                dct:language <http://lexvo.org/id/iso639-3/en> .
```

Literature Reference

Provide a literature reference as a CiTO entry [CiTO] using cito:citesAsAuthority property. For example, for the :chembl dataset the literature reference is [Gaulton et al 2011] .

```
        :chembl
                cito:citesAsAuthority <http://nar.oxfordjournals.org/content/40/D1/D1100>
```

Vocabularies

RDF datasets use one or more RDFS vocabularies or OWL ontologies to represent the data. It is recommended that the distribution level metadata, for RDF datasets, should state what terminologies were used in the dataset using void:vocabulary [VOID]. The value has to be a URI identifying the terminology used.

```
        :chembl
                void:vocabulary <> .
```

Conformance [comment]

Indicate that the dataset conforms to a particular format or standard using the dct:conformsTo property. For example,

```
        :dataset
                dct:conformsTo <> .
```

Subsets

The parts of a dataset MAY be described using dct:hasPart, or void:subset in the case of RDF datasets. For example, ChEMBL identifies several datasets, e.g. one containing data about molecules and another about targets. Each of these subsets can be assigned appropriate topics.

```
        :chembl
                void:subset :chembl_rdf_molecule_dataset, :chembl_rdf_target_dataset .
```

6.2 Identifiers

Preferred and Alternative Prefixes

A preferred prefix is a short label that is commonly used to refer to the dataset and may stand in place of the base IRI of the dataset (e.g. see http://prefix.cc). It is described by the idot:preferredPrefix property and MAY be used for **any** descriptions. It is recommended that that the idot:preferredPrefix is the same as the dct:identifier.
An alternate prefix is a short label that is used by some community to refer to the dataset, but is not the preferred prefix.

```
        :chembl
                idot:preferredPrefix “chembl” ;
                 idot:alternatePrefix “chembldb” .
```

Identifier, Resource, and Access Patterns

The idot:identifierPattern SHOULD be provided for X level descriptions, where it provides a regular expression for alphanumeric strings used to identify items or records in the dataset.

```
        :chembl17
                idot:identifierPattern “CHEMBL\d+”^^xsd:string .
```

The void:uriRegexPattern property SHOULD be provided for distribution level descriptions, where it provides a regular expression whose transitive closure denotes a superset of data item URIs in the dataset. The property must NEVER be used for summary level descriptions or version level descriptions.

```
        :chembl17rdf
                void:uriRegexPattern
 “http://rdf.ebi.ac.uk/resource/chembl/target/CHEMBL\d+” .
```

The idot:accessPattern MAY be used to specify how to access specific formats of the resources.

```
        :chembl
                idot:accessPattern [
                        rdf:type idot:AccessPattern;
                        rdf:type idot:PrimarySource;
                        dct:publisher <http://www.ebi.ac.uk>;
                        dct:format “text/html”;
                        rdf:value “http://www.ebi.ac.uk/chembl/compound/inspect/”;
                ]

                idot:accessPattern [
                        rdf:type idot:AccessPattern;
                        dct:format “text/html”;
                        rdf:value “http://identifiers.org/chembl.compound/”;
                ]
                idot:accessPattern [
                        rdf:type idot:AccessPattern;
                        dct:format “application/rdf+xml”;
                        rdf:value “http://linkedchemistry.info/chembl/chemblid/”;
                ]
                idot:accessPattern [
                        rdf:type idot:AccessPattern;
                        dct:format “application/rdf+xml”;
                        dct:publisher <http://bio2rdf.org>
                        rdf:value “http://bio2rdf.org/chembl:”;
                ]
```

Example Identifier and Resource

An example identifier MAY be provided using the idot:exampleIdentifier for any level descriptions.

```
        :chembl
                idot:exampleIdentifier “CHEMBL25”^^xsd:string .
```

An example resource MAY be provided using the void:exampleResource property for any level description.

```
        :chembl17rdf
                void:exampleResource
 <http://rdf.ebi.ac.uk/resource/chembl/compound/CHEMBL25> .
```

6.3 Provenance and Change

Versioning

A version level description MUST use the dct:isVersionOf property to relate to the summary level description. The versions of the summary level resource may then be inferred due to the declared inverse property pav:hasVersion.

```
        :chembl17
                 dct:isVersionOf :chembl .
```

For each version level description, the version identifier should be declared using pav:version and restrict its value to an xsd:string. For datasets that do not have a scheme for versioning numbers, the date of release could be used.
For instance, to specify that :chembl17 has a version number of “17”:

```
        :chembl17
                pav:version “17”^^xsd:string .
```

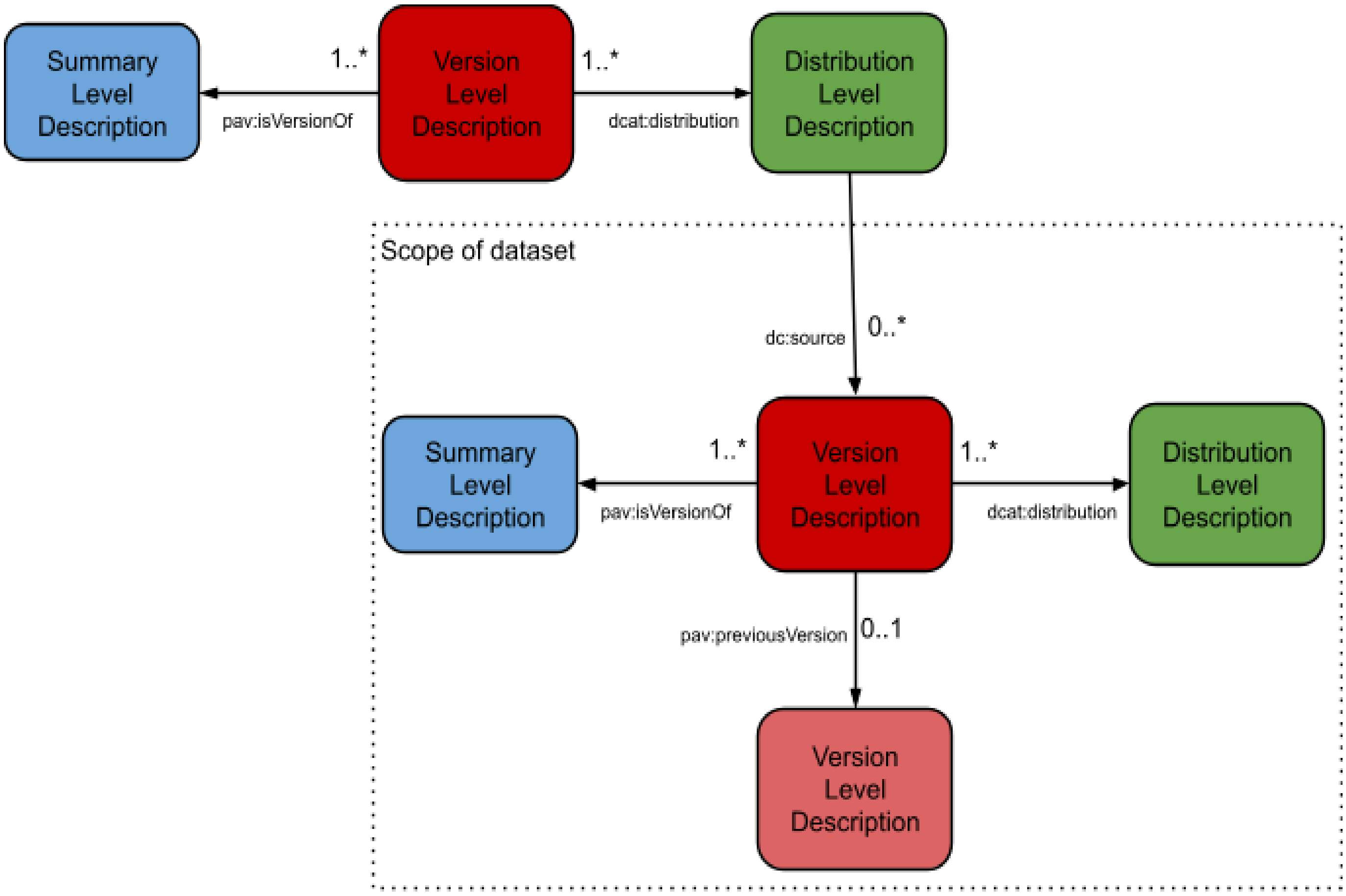
Links to the previous version of the dataset SHOULD be provided using pav:previousVersion. For instance, to state that :chemb16 is a prior version of :chemb17,

```
        :chembl17
                pav:previousVersion :chembl16 .
```

Data publishers MAY use pav:hasCurrentVersion to declare the current version of the dataset, **provided they are the authoritative source** of the data *and* the summary level description is always made current. Data aggregators are advised to maintain the most current summary level description so that only a single such reference appears.

```
        :chembl
                pav:hasCurrentVersion :chembl17 .
```

Dataset Provenance


Summary Level Description
1..*
pav:isVersionOf
Version Level Description
1..*
dcat:distribution
Distribution Level Description
Scope of dataset
dc:source
0..*
Summary Level Description
1..*
pav:isVersionOf
Version Level Description
1..*
dcat:distribution
Distribution Level Description
pav:previousVersion
0..1
Version Level Description

A version level description of a dataset SHOULD identify the datasets used to create the published data. Use **dct:source** when the source dataset was used in whole or in part. Use **pav:retrievedFrom** when the source dataset was used in whole and was not modified from its original distribution. Use **prov:wasDerivedFrom** when the source dataset was in whole or in part and was modified from its original distribution. [comment1]

```
:chembl
        dct:source :pubchem-bioassay-09-01-2014 .
```

A distribution level description of a dataset MAY describe a tool used to create the dataset (where appropriate). This information is useful for debugging automatic generation of the dataset. The property to be used for this purpose is:

- ○ pav:createdWith (need to capture the version of the tool)

```
:chembl17rdf
        pav:createdWith :chembl-sql2rdf-exporter .
```

Entity Provenance

Provenance for each data item SHOULD be provided using the void:inDataset property. It should be used with distribution level descriptions, and must NEVER be used for summary level descriptions or version level descriptions.

```
chembl.compound:chembl25
        void:inDataset :chembl17rdf .
```

Change Frequency

The update frequency of the dataset SHOULD be specified using dct:accrualPeriodicity with a value from the Dublin Core Frequency vocabulary [DCFreq]. This description appears at summary level.

```
:chembl
        dct:accrualPeriodicity freq:quarterly .
```

Modifications

It is recommended that modifications are NOT made to a dataset without changing its version information. If changes are required, a new version with its own description MUST be published, and the previous version information SHOULD be maintained with a link back to it from the version that superseded it.

```
:chembl17-1
        pav:previousVersion :chembl17 .
```

6.4 Availability and Distributions

Distributions and Formats

Each version level description SHOULD link to the distribution level descriptions that represent the files in different data formats. Each distribution level MUST state the file format in which the data is available. This SHOULD be stated as a IANA code [IANA-MT] otherwise dct:format should be used with different values (e.g. a URI from a controlled vocabulary).

In the case of RDF distributions, the distribution level description should be additionally typed as void:Dataset.

```
:chembl17
        dcat:distribution :chembl17rdf, :chembl17db .
:chembl17rdf
        a dctype:Dataset, void:Dataset ;
        dcat:format “text/turtle” ;
        dcat:format <http://www.w3.org/ns/formats/Turtle> .

:chembl17db
        a dctype:Dataset ;
        dct:format “application/sql” .
```

File Locations

It is recommended to use dcat:downloadURL to declare the distribution file in conjunction with dcat:byteSize to declare the size of the file. For RDF resources, the files should be declared using void:dataDump. Use dcat:accessURL to specify a directory containing the file(s) of interest.

```
        :chembl
                dcat:accessURL <ftp://ftp.ebi.ac.uk/pub/databases/chembl/ChEMBLdb>

        :chembl17
                dcat:accessURL
<ftp://ftp.ebi.ac.uk/pub/databases/chembl/ChEMBLdb/releases/chembl_17/>

        :chembl17sql
                dcat:downloadURL <ftp://ftp.ebi.ac.uk/pub/databases/chembl/ChEMBLdb/
                                        releases/chembl_17/chembl_17_mysql.tar.gz>
                dcat:byteSize “861443887”^^xsd:decimal ;

        :chembl17rdf
                void:dataDump <ftp://ftp.ebi.ac.uk/pub/databases/chembl/ChEMBL-RDF/
                                        17/chembl_17_molecule.ttl.gz> .
```

Query Endpoint

Publishers may also provide their data through a SPARQL endpoint. We recommend that such an endpoint should have a service description [SD] and this should link to the distribution level description that is currently loaded (see Figure 1).

Below is an example based on the one given in Section 6.5 of the VOID guidelines [VOID] to indicate how this may be achieved for the default graph of a triplestore.

```
<#service>
        a sd:Service;
        sd:defaultDatasetDescription [
                a sd:Dataset;
                sd:defaultGraph [
                        a sd:Graph;
                        dct:source :chembl17rdf ;
                ];
        ].
```

Note that once the file is loaded into the triplestore it becomes a different representation of the same dataset version. Hence the use dct:source rather than directly linking to the dataset description file.

We recommend that the void:sparqlEndpoint MUST NOT be used since it would either generate a maintenance issue for the generated descriptions be ambiguous as to which version of the data is loaded in the triplestore.

API and Dataset Documentation

Details of documentation associated with a dataset – either for the dataset itself or a web service through which it is made available – can be given using the `dcat:landingPage` predicate.

```
        :chembl17
                dcat:landingPage
        <ftp://ftp.ebi.ac.uk/pub/databases/chembl/ChEMBLdb/releases/chembl_17/chembl
        _17_release_notes.txt>
```

6.5 Statistics (Michel)

Dataset statistics offer one means by which to understand the contents of the dataset and their relation to other datasets. Below we provide recommendations for the capture of dataset statistics for RDF formatted files using the VoID vocabulary. These statistics can be computed on an RDF dataset using SPARQL queries [VOID-SPARQL]. The results of the SPARQL queries can be added to the description of the RDF-formatted dataset.

Core statistics

Core statistics provide basic information about datasets, such as the total number of triples in the dataset, number of unique entities (subject URIs), etc., as well as information about the number of unique classes, literals, and graphs in a dataset.

Basic statistics provide counts about RDF triples in the dataset, such as the total number of triples in the dataset, the number of unique subjects, properties, etc.

We recommend that the core statistics are provided for RDF distributions.

To specify the ***number of triples*** in the dataset:

```
        :rdfdataset
                void:triples “####”^^xsd:integer .
```

SPARQL : SELECT (COUNT(*) AS ?no) { ?s ?p ?o }

To specify the ***number of unique entities*** (an entity must have a URI) in the dataset:

```
        :rdfdataset
                void:entities “###”^^xsd:integer .
```

SPARQL : SELECT COUNT(distinct ?s) AS ?no { ?s a [] }

To specify the **number of unique subjects** in the dataset

```
        :rdfdataset
                void:distinctSubjects “###”^^xsd:integer .
```

SPARQL : SELECT (COUNT(DISTINCT ?s) AS ?no) { ?s ?p ?o   }

To specify the ***number of unique properties*** in the dataset

```
        :rdfdataset
                void:properties “##”^^xsd:integer .
```

SPARQL: SELECT count(distinct ?p) { ?s ?p ?o }

To specify the ***number of unique objects*** in the dataset

```
        :rdfdataset
                void:distinctObjects “###”^^xsd:integer .
```

SPARQL: SELECT (COUNT(DISTINCT ?o) AS ?no) { ?s ?p ?o filter(!isLiteral(?o)) }

Information about dataset contents quantifies the number of unique classes and literals within the RDF representation.

To specify the ***number of unique classes*** in the dataset

```
        :rdfdataset
                void:classPartition [
                        void:class rdfs:Class;
                        void:entities "###"^^xsd:integer;
                ] .
```

SPARQL: SELECT DISTINCT ?type { ?s a ?type }

To specify the ***number of unique literals*** in the dataset

```
    :rdfdataset
        void:classPartition [
            void:class rdfs:Literal;
            void:entities "###"^^xsd:integer;
        ] .
```

SPARQL: `SELECT (COUNT(DISTINCT ?o ) AS ?no) {  ?s ?p ?o  filter(isLiteral(?o)) }`

The number of RDF graphs denotes the number of sets of triples within the dataset.

To specify the ***number of graphs*** in the dataset

```
    :rdfdataset
        void:classPartition [
            void:class sd:Graph;
            void:entities "###"^^xsd:integer;
        ] .
```

SPARQL: `SELECT (COUNT(DISTINCT ?g ) AS ?no) { GRAPH ?g { ?s ?p ?o}}`

Enhanced Statistics

Enhanced statistics provide in-depth details that are not captured by basic statistics. These record the number of instances per class and multiple counts about properties. The enhanced statistics give an overview of the utilization of data within the dataset and to other datasets as linked data.

To specify the ***classes and the number of their instances*** in the dataset

```
    :rdfdataset
        void:classPartition[
            void:class <class-uri>;
            void:entities "###"^^xsd:integer;
        ] .
```

SPARQL: `SELECT  ?class (COUNT(?s) AS ?count ) { ?s a ?class } GROUP BY ?class`

To specify the ***properties and their frequency*** in the dataset

```
:rdfdataset
    void:propertyPartition [
        void:property <property-uri>;
        void:triples "###"^^xsd:integer;
    ]
```

SPARQL: `SELECT  ?p (COUNT(?p) AS ?count ) { ?s ?p ?o } GROUP BY ?p`

To specify the ***properties and the number of unique objects linked to the property*** in the dataset:

```
:rdfdataset void:subset
        [ a void:LinkSet;
                void:linkPredicate <property-uri>;
                void:objectsTarget [void:class rdfs:Class; void:entities "###"^^xsd:integer]
        ]
```

SPARQL: SELECT ?p (COUNT(DISTINCT ?o) AS ?count) { ?s ?p ?o } GROUP BY ?p

To specify the ***properties and the number of unique literals*** in the dataset

```
:rdfdataset void:subset
        [ a void:LinkSet;
                void:linkPredicate <property-uri>;
                void:objectsTarget [void:class rdfs:Literal; void:entities "###"^^xsd:integer]
        ]
```

SPARQL: SELECT ?p (COUNT(DISTINCT ?o) AS ?count) { ?s ?p ?o } GROUP BY ?p

To specify the number and list of ***unique subject types that are linked through a property to unique object types*** in the dataset:

```
:rdfdataset void:subset
        [ a void:LinkSet;
                void:linkPredicate <property-uri>;
                void:subjectsTarget[void:class <subject-type-uri>;void:entities
                "#"^^xsd:integer];
                void:objectsTarget [void:class <object-type-uri>; void:entities
                “#"^^xsd:integer];
        ]
```

SPARQL: SELECT (COUNT(DISTINCT ?s) AS ?scount) ?p (COUNT(DISTINCT ?o) AS ?ocount) { ?s ?p ?o } GROUP BY ?p

To specify the number and list of ***properties that link items from one dataset to another***

```
:rdfdataset void:subset
        [ a void:LinkSet;
                void:linkPredicate <property-uri>;
                void:subjectsTarget <subject-dataset-uri>;
                void:objectsTarget <object-dataset-uri>;
                void:triples “###”^^xsd:integer;
        ]
```

7. Tooling Support

To support the provision of dataset descriptions conforming with this specification, two tools have been provided.

7.1 Dataset Description Generation Tool

This tool helps with the creation of initial drafts of dataset descriptions. The dataset generator is available from http://voideditor.cs.man.ac.uk/.

Note that the current tool is for the creation of datasets conforming to the closely related Open PHACTS dataset description guidelines http://www.openphacts.org/specs/datadesc/. This will be updated.

It is anticipated that the dataset description will be created as part of the same process that generates the data, i.e. they are part the same (automated) production workflow.

7.2 Validation Tool

This tool allows for the validation of dataset descriptions against these specifications. The validator is available from http://www.macs.hw.ac.uk/~ajg33/HCLSValidator/.

Validation can be at different levels of conformance:

1. Minimal: all MUST properties have been provided. Datasets conforming to this minimal version of the specification can be branded with the following logo.
2. Strict: all MUST and some SHOULD properties have been provided but no additional properties have been given.
3. Recommended: all MUST properties have been supplied together with SHOULD and MAY properties. Datasets conforming to this specification can be branded with the following logo.

Currently, the validator requires the whole dataset description to be pasted into a web form. It is anticipated that an API will be provided.

8. Use Cases

brief intro. highlight elements of what the use case should express (michel)

- briefly describe organization
- describe workflow & make reference to dataset metadata
- express pain points
- express desired functionality
- examples of metadata fields needed

8.1 Marshfield Use Case

(Simon)

Marshfield Clinic is one of the largest physician group practices in the United States, who employs approximately 780 physicians representing 84 different specialties and 6,500 additional staff working on the main campus in Marshfield or one of 52 regional clinics serving the population of Wisconsin and the upper peninsula of Michigan. Integral to clinical practice, Marshfield Clinic Research Foundation (MCRF) conducts clinical and biomedical research projects.

Marshfield is a member of the Health Maintenance Organization Research Network (HMORN). To enable population-based health and health care research across 16 members in HMORN, Marshfield participates in a standardized federated data system called Virtual Data Warehouse (VDW). VDW enables the query to be distributed to different sites. The query result from each site is summarized at each site and then combined to be returned to the requester. In such a scenario, metadata like versioning would be critical for the scientific reproducibility of the query results.

Marshfield also participates in a clinical pharmacogenomics consortium called eMERGE-PGx. A goal of eMERGE-PGx is to discover novel associations between genotypes and pharmacogenomics responses. The more data accumulated from different participating sites, the higher the statistical power to detect novel associations. Standardized metadata description, especial descriptors of license conditions and rights, will ensure transparencies in genetic research and data sharing. With proper metadata available for each data set, pharmacogenomics discovery can be made by agent-based algorithms and proper reuse of data.

8.2 Metaome Transcriptomics Use Case

Gene expression data analysis has emerged as a powerful approach for understanding biology. RNA-seq, microarrays and qPCRs are some of the common approaches for analyzing gene expression. To understand gene function, it is often necessary to analyze gene expression over various experimental conditions. It is important that, from a large corpus of transcriptomics datasets, users are able to retrieve datasets that match specific experimental criteria. Further, based on the experiment metadata, the datasets can be enriched with relevant information such as related mutations, cell lines and phenotypes.

A well-structured and standardized dataset description is imperative for this operation. Such an approach of slicing large corpus of transcriptomics data, based on biological parameters and experimental conditions is recognized by our customers as a strong base for gene expression analytics. A structured approach to describing datasets combined with gene expression analytics could lead to applications such as better classification of tumor samples and drug-repurposing.

8.3 Radiotherapy Research Use Case

(Scott)

MAASTRO Clinic is a radiotherapy clinic with approximately 200 employees and 60 researchers. There are both internal and external projects that employ distributed data, and data discovery is an important feature to researchers.

The internal project at MAASTRO is a Research Data Archive (RDA) which creates a central access point for researchers to find many types of data for their research including: patient demographic data, treatment and planning data such as CT images, tumor volume, dose, and fractionation. The RDA is being built from a combination of a data warehouse and SPARQL federation and will contain a catalogue of all data sources, some of which will be data that has been analyzed in specific publications. A self-maintaining catalogue (i.e.

dynamically generated from dataset descriptions) would help to keep the RDA manageable as it grows to a larger scale, as well as lower the cost of adding data sources and maintenance of tools for data access.

In projects such as EuroCAT and EURECA, MAASTRO needs to facilitate data sharing with external partners. In EuroCAT, a system for machine learning has been built up based on a principle of a uniform data interface that enables machine learning algorithms to be sent to the data at the hospital and clinic. This circumvents the security and legal problems that arise when sending clinical data outside the walls of the hospital. In this case, each hospital could use dataset descriptions to make data discovery possible for other partners. The same principle can be applied to biobanks.

In the EURECA project, a legal and technical framework has been created that enables clinical care and clinical research data to be shared via a Center for Data Protection (CDP). However, as the data collection grows, it becomes steadily more difficult to find data based on attributes of interest. Therefore, a standardized dataset description will enable project partners to make better use of the data by enabling data discovery.

8.4 Computational Network Biology

Biological network data users

The Cytoscape software for network visualization and analysis [Cytoscape] and the GeneMANIA software for gene function prediction (http://genemania.org/) depend on loading biological network and related data from numerous and diverse sources to support various types of analysis. Cytoscape is a stand alone workbench application and data loading is driven by users and GeneMANIA primarily accesses data using an automated build process. Both systems will benefit from knowledge of the following aspects of a data set:

- license - generally needed to support commercial users who need to separate commercial from open data.
- data source short name (for GUI display purposes), full name (may go in a tooltip) and description
- homepage URL, PMIDs (find out more information about the data set/data source)
- Example data URL - useful to help browse contents of an online data source e.g. an example pathway database record
- production date - needed to communicate to users about how current the data is
- download date - needed to communicate to users about how current the data is
- version - needed to communicate to users about how current the data is
- update cycle - frequency of update, useful for build systems to figure out how often to check for data updates
- dataset statistics. Useful to gauge the overall size of the data. Ideally this would be the number of genes/protein/molecules and the number of interactions, broken down into type of interaction would be even better.

For GeneMANIA, to fully automate discovery of new data sets (a true intelligent agent), we would need to know the data type (e.g. protein-protein interaction, text mined co-citations, pathways, gene expression), the organism, the types of gene identifiers and the number of genes covered by a given number of interactions.

For Cytoscape, most users would also be interested to know how others have used the data, as otherwise, they would not know what to do with the list of data sets (at least if there was no further categorization).

Pathway Commons and Pathguide.org
The goal of Pathway Commons (http://www.pathwaycommons.org/about/) is to collect all publicly available biological pathway information and make it easily and widely available. Pathguide (http://www.pathguide.org/) tracks over 540 databases containing pathway related information. This tracking website is currently manually updated, but it would be great if it could be automatically updated by downloading metadata files from each database. In addition to the above metadata useful for GeneMANIA and Cytoscape, the nature of the data source in terms of originality is important for Pathway Commons, as originally curated information is desired, not redundant copies of data from meta-databases. Pathguide terms this ‘primary’ (originally curated or predicted) or ‘secondary’ (collected from other sources). An example Pathguide record for GeneMANIA is at http://pathguide.org/fullrecord.php?organisms=all&availability=all&standards=all&order=alphabetic&DBID=334

8.5 Safety Information Evaluation and Visual Exploration (“SIEVE”)

Suzanne Tracy, AstraZeneca; Stephen Furlong, AstraZeneca; Robert Stanley, IO Informatics; Peter Bogetti, AstraZeneca; Jason Eshleman, IO Informatics; Michael Goodman, AstraZeneca)

AstraZeneca (“AZ”) Patient Safety Science wanted to improve retrieval of clinical trial data and biometric assessments across studies. Traditionally, evaluation of clinical trials data across studies required manual intervention to deliver desired datasets. A proposal titled Safety Information Evaluation and Visual Exploration (“SIEVE”) was sponsored by Patient Safety Science. This took the form of collaboration between AZ and IO Informatics (“IO”). AZ provided the project environment, data resources, subject matter expertise (“SME”) and business expertise. IO provided semantic software, data modeling and integration services including solutions architecture, knowledge engineering and software engineering.

The project goal was to improve search and retrieve of clinical trials data. SIEVE was to provide a web-based environment suitable for cross-study analysis. The environment was to align across biomarkers, statistics and bioinformatics groups. Over-arching goals included decision-making for biomarker qualification, trial design, concomitant medication analysis and translational medicine.

The team analyzed approximately 42,000 trials records, identified by unique subjectIDs. IO’s Knowledge Explorer software was used by IO’s knowledge engineers in collaboration with AZ’s SMEs to explore the content of these records as linked RDF networks. Robust metadata descriptors were central to the integration. RowID was applied to match entries from a diverse source documents to unique rows in unique study documents. SubjectID and studyID were also important for combining data from separate rows into an integrated resource, for example to combine 2 or more rowIDs specific to a single study subject. Because almost all docs had both subjectID and studyID, concatenation of these two items as an individual identifier allowed connections that bridged multiple documents for data traversal. For data quality assessment, determining the error rate in making connections was possible by evaluating Gender and DOB associated with the concatenated individual identifier. About 6,000 patients could not be associated to both Gender and DOB, and were removed from corpus. Next, as Gender and DOB information were duplicated throughout the corpus, this allowed to test the consistency of data recording, which was reasonably high. Less than 40 individuals had problematic DOB and or gender information where one or more rowIDs did not agree for a subject. In summary, 36,000 records were found to contain valid data that could be usefully linked - each including a unique trial (StudyID), and unique and valid patient (SubjectID), and at least one row of valid laboratory data of interest.

IO created a semantic data model or “application ontology” to meet SIEVE requirements. The resulting data model and instances were harmonized by application of SPARQL-based rules and inference and were aligned with AZ standards. Data was integrated under this ontology, loaded into a semantic database and connected to IO’s “Web Query” software. The result is a web-based User Interface accessible to end users for cross-study searching, reporting, charting and sub-querying. Methods include “Quick Search” options, shared searches and query building containing nesting, inclusion / exclusion, ranges, etc. Advanced Queries are presented as filters for user entry to search subjects (“views” or “facets”) including Clinical Assays, Therapy Areas, Adverse Events and Subject Demographics. Reports include exporting, charting, hyperlink mapping and results-list based searches.

Results include reduced time to evaluate data from clinical trials and to facilitate forward looking decisions relevant to portfolios. Alternatives are less efficient. Trial data could previously be evaluated within a study; however there was no method to evaluate trials data across studies without manual intervention. Semantic technologies applied for data description, manipulation and linking provided mission-critical value. This was particularly apparent for integration and harmonization, in light of differences discovered across resources. IO’s Knowledge Explorer applied data visualization and manipulation, application of inference and SPARQL-based rules to RDF creation. This resulted in efficient data modeling, transformation, harmonization and integration and helped assure a successful project.

8.6 Sampling Large RDF graphs

SampLD is a tool for sampling large RDF graphs. We use the network topology of an RDF graph to detect which triples are 'interesting' enough to be included in a sample. SampLD uses common network analysis tools such as PageRank to detect 'interesting' triples. Because we cannot directly apply pagerank to RDF graphs (as an RDF graph has labelled edges), we first apply a rewrite step where we rewrite RDF to a graph with unlabelled edges (without losing too much information). We determine the quality of the sample by how good a sample is able to return answers to a set of queries (i.e. calculating recall). (This is why we use BioPortal, as we have access to the querylogs via the USEWOD challenge). However, the performance of each sample method (i.e. combination of rewrite method + network analysis algorithm) differs between datasets. The probable reasons for these differences are: (1) the structure of each dataset is different, (2) the queries for each dataset have a different structure. (2), is something which we are currently analyzing by extracting important features from queries. However, (1) is still an open case. What we need are features of an RDF graph. This is where the Bio2RDF dataset metrics are useful. They provide an easy way of generating information about the RDF structure itself.

8.7 Query Formulation Using Data Metrics

YASGUI (http://yasgui.org) is a query builder for SPARQL with a strong focus on usability. To assist the user in formulating queries, we provide autocompletion functionality as much as possible. Examples are prefix autocompletion using prefix.cc, and endpoint autocompletion using the CKAN-based datahub.io [Datahub] and Sparqles. However, what users need is mostly autocompletion for resources. The current version of YASGUI supports property and class autocompletion (more info on the taken approach is described here: http://laurensrietveld.nl/yasgui/help.html#autocomplete). However, our approach can be quite expensive. Additionally, the number of suggestions per autocompletion might be quite large. Where the Bio2RDF metrics are particularly useful are: fetching the number of distinct predicates, as this a lot cheaper than executing a "distinct ?p {?s ?p ?o}" query. Additionally, using these metrics we should be able to rank these predicates more meaningfully, based on the combination of its frequency and how this particular predicate is used in the query (e.g. is the object in this triple pattern a URI or a literal)

8.8 Data Providers

Open PHACTS

Open PHACTS [OpenPHACTS] is an open-source project to build a data integration platform for drug discovery data. The platform provides a domain specific API through which the integrated data can be retrieved. The platform is populated with open data sets including ChEMBL, UniProt, ChemSpider and Wikipathways. The data items within these datasets are related through VoID linksets [VOID]. The linksets are published as part of the delivery of the open platform.

The Open PHACTS platform is being further developed to rely on the VoID specification to drive the automated update of the integrated data from external providers. In particular, we would like to be able to rely on knowing when a dataset is updated and at what period it is updated. To know when to download new information. Furthermore, we need to do know the type of dataset we are dealing with whether it is a linkset or dataset. Importantly, knowing the correct predicate to find where to download or obtain dumps of provider data is crucial. Finally, it's important for us to be able to correctly link back to data providers (i.e. provenance) to give credit. To do this correctly, we need to know both human readable and machine readable provenance links.

EBI RDF Platform

The European Bioinformatics Institute (EBI) is the largest Bioinformatics resource provider in Europe. The recently released RDF platform [EBI-RDF] presents a coordinated effort to bring together RDF resources from multiple services and databases at the EBI. The EBI invests heavily in the curation and annotation of the source databases to ensure the most up-to-date and accurate information is readily available to the scientific community. Given that the generated RDF is typically the result of a conversions from the source database, it is important that our users understand the relationship between the RDF and the source. To address this we publish detailed provenance for each dataset that includes important information such as version number and release date. This data is available in RDF via content-negotiation from stable dataset URIs and is described using a variety of standard meta-data vocabularies. These dataset descriptions will conform to the recommendations outlined in this document.

Bio2RDF (Michel)

Bio2RDF [Bio2RDF] is an open-source project to provide Linked Data for the Life Sciences. Bio2RDF defines a set of simple conventions to create RDF(S) compatible Linked Data from a diverse set of heterogeneously formatted sources obtained from multiple data providers. Bio2RDF has developed a registry of over 2000 datasets that acts to provide basic information about bio-datasets and to normalize different identifiers to a common URI scheme. Bio2RDF generates a wide variety of statistics to describe the internal contents of each dataset so as to facilitate an understanding of their contents and to use in autocompletion services. Bio2RDF will implement the dataset description guidelines for the description of its datasets, the provenance of the datasets, and dataset statistics.

WormBase (Joachim)

WormBase [WormBase] is a curated genetic and genomic data resource of 19 nematode species including the widely used research model organism *Caenorhabditis elegans*. Its heavily cross-referenced datasets have continuously grown since the establishment of WormBase more than a decade ago and their growth has accelerated with the availability of cheap next-generation sequencing techniques. Data integration and query federation have become major driving factors in the further development of WormBase, which is currently being addressed by the translation of Genomic Feature Format Version 3 (GFF3) datasets into RDF and their subsequent exposure via a SPARQL endpoint. WormBase will implement the dataset description guidelines to advertise its data resources to the wider bioinformatics and computational biology communities.

8.9 Data Catalogs

BioSharing (Alejandra)

BioSharing [BioSharing] is a registry of:

i) community-developed data and metadata reporting standards (including minimum information checklists, ontologies and data formats),

ii) policies related to data preservation, managing and sharing; and
iii) databases
and the relationships between the three elements (standards, policies and databases) in the life sciences (broadly covering biological, natural and biomedical sciences).
Community-driven standardization efforts have worked in different domains to ensure that data and experimental details about how the data was generated are made available in an interoperable way. These are important requirements to enable science reproducibility.
The BioSharing catalogue works to serve those seeking information on the existing standards, to identify areas where duplications or gaps in coverage exist and to promote harmonization to stop wasteful reinvention, and developing criteria to be used in evaluating standards for adoption.
BioSharing also provides information about life science data, the tools that can be used to access them, the standards that have been used to generate the data or that can be used to access the data. It follows and extends the information required by the **BioDBCore** reporting guidelines. In particular, BioSharing identifies what reporting guidelines, terminological artifacts and/or exchange formats are considered within a particular dataset.
BioDBCore [BioDBCore,Gaudet *et al* 2010] is a checklist, or minimum information standard, including the core attributes for the description of biological databases. It is a community-driven effort overseen by the International Society for Biocuration [ISB], in collaboration with the BioSharing catalogue, which implements the BioDBCore guidelines [BioDBCore]. The BioDBCore checklist main goals involve to compile information on biological databases allowing to survey the current landscape and promote the interoperability of the resources by adoption of of syntactic and semantic standards. The proposed core attributes are listed at the BioDBCore website [BioDBCore].

Integbio Database Catalog

Integbio Database Catalog [Integbio] is a catalog which provides basic information on life science databases according to the uniformed description items such as URLs, database description and biological species to promote circulation of the databases created in Japan. Toward our goal of covering all databases scattered within Japan, we already merged the four ministries' existing database lists into Integbio Database Catalog and also continue a survey to collect information of the databases financially supported by research funds in Japan. The content of Integbio Database Catalog is already available to users as CSV format under the Creative Commons CC0 license. Also, the CSV dataset is translated into RDF and currently exposed through a SPARQL endpoint. We participate in the discussion for the dataset description guidelines to refine the RDF dataset.

identifiers.org

Identifiers.org [Identifiers.org] is a system which provides resolvable and persistent URIs to identify data of interest to the life sciences. It relies upon an underlying Registry which provides detailed information on numerous resources which are crucial to scientific research. It includes details on resolving locations where records for a particular resource may be accessed, defines identifier patterns for each data provider, and reflects accessibility of data by recording the up time for individual resource. The content of the Registry is available as RDF, which is defined using DCAT. The use of such terminologies has been agreed upon by numerous communities, and enables standardised descriptions of such repositories and facilitates data sharing and processing between them.

10 References

[Bio2RDF] http://bio2rdf.org

[BioDBCore] http://biodbcore.org/

[BioSharing] http://biosharing.org/

[ChEMBL] http://www.ebi.ac.uk/chembldb/

[CiTO] David Shotton and Silvio Peroni. CiTO, the Citation Typing Ontology. http://purl.org/spar/cito/

[Cytoscape] http://cytoscape.org/

[Datahub] http://datahub.io/

[Dataverse] Harvard Dataverse http://thedata.harvard.edu

[DCAT] http://www.w3.org/TR/2014/REC-vocab-dcat-20140116/

DCFreq Dublin Core Collection Description Task Group. Dublin Core Collection Description Frequency Vocabulary. http://dublincore.org/groups/collections/frequency/

[DCMI] Dublin Core Metadata Initiative http://dublincore.org/

[Dryad] http://datadryad.org/

[EBI-RDF] http://www.ebi.ac.uk/rdf

[FigShare] http://figshare.com/

[FOAF] http://xmlns.com/foaf/spec/

[Force11] http://www.force11.org/catalog

[Gaudet *et al* 2010] “Towards BioDBCore: a community-defined information specification for biological databases”. Nucleic Acids Research, Database Issue http://nar.oxfordjournals.org/content/early/2010/11/17/nar.gkq1173.abstract

[Gaulton et al 2011] ChEMBL: a large-scale bioactivity database for drug discovery"" Nucleic Acids Research. Volume 40. Issue D1. Pages D1100-D1107. 2011.

http://nar.oxfordjournals.org/content/40/D1/D1100

[GigaScience] GigaScience Journal http://www.gigasciencejournal.com/

[HCLS] http://www.w3.org/blog/hcls/

[IANA-MT] http://www.iana.org/assignments/media-types/media-types.xhtml

[Identifiers.org] Identifiers.org Registry http://identifiers.org

[Integbio] http://integbio.jp/dbcatalog/?lang=en

[ISB] http://www.biocurator.org/

Lexvo Gerard de Melo. Lexvo.org. http://www.lexvo.org/

Lexvo Paper Gerard de Melo. Lexvo.org: Language-Related Information for the Linguistic Linked Data Cloud. Submitted to Semantic Web Journal. http://www.semantic-web-journal.net/content/lexvoorg-language-related-information-linguistic-linked-data-cloud-0

[NIF] http://www.neuinfo.org/

[OpenPHACTS] http://www.openphacts.org

[ORCID] http://orcid.org/

PAV Paolo Ciccarese and Stian Soiland-Reyes. Provenance, authoring and versioning Ontology. http://purl.org/pav/.

PAV Paper Ciccarese, P.; Soiland-Reyes, S.; Belhajjame, K.; Gray, A. J. G.; Goble, C. and Clark, T. PAV ontology: Provenance, Authoring and Versioning. In Journal of Biomedical Semantics, 2013.

[PROV] http://www.w3.org/TR/prov-overview/

[RDF] http://www.w3.org/TR/rdf-primer/

[SCHEMA] http://schema.org/Dataset

[ScientificData] Nature Publishing Group’s Scientific Data http://nature.com/scientificdata/

[VOID] http://www.w3.org/TR/void/

[VOID-SPARQL] https://code.google.com/p/void-impl/wiki/SPARQLQueriesForStatistics

[WormBase] http://www.wormbase.org

[xsd:dateTime] http://www.w3.org/TR/xmlschema-2/#dateTime