博士論文

A study on semantic integration of biological databases with the Semantic Web technologies.

（Semantic Web 技術を用いたバイオデータベースの
意味的統合に関する研究）

川島秀一

A Dissertation Presented

by

Shuichi Kawashima

Submitted to

Graduate School of Frontier Sciences

the University of Tokyo

In partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

February 2019

Acknowledgments

First, and foremost, I would like to express my sincere gratitude to Professor Toshihisa Takagi who gave me the opportunity to write this thesis. He has been devoting great efforts for many years to make life-science databases in Japan more useful and valuable. This research was conducted under his guidance as part of his grand vision. I could not complete this thesis without his patient support of my research with his warm personality.

I am deeply grateful to Professor Minoru Kanehisa who was my former supervisor. I had a lot of valuable experiences which have tremendously impacted my research career during the years I have spent in his laboratory as a graduate student and later as a research assistant.

My sincere gratitude goes to Prof. Kiyoshi Asai, Prof. Yutaka Suzuki, Prof. Kouji Kozaki, and Prof. Kiyoko F. Aoki-Kinoshita who constituted the thesis committee with Prof. Toshihisa Takagi. I would also like to acknowledge Professor Kenta Nakai and Kentaro Tomii for the initial developments of the AAindex database.

I gratefully acknowledge Katsuhiko Ohkubo, Takehiro Kato, Akio Nagano, Keita Urashima and Yoji Shidara for developing the software and maintaining the NBDC RDF portal. I would also like to thank all those who participated in the SPARQLthon and BioHackathon events for fruitful discussions. I thank all the colleagues and staff of the Bioinformatics Center of the Kyoto University, the Human Genome Center of the University of Tokyo, the National Bioscience Database Center of the Japan Science and Technology Agency, and the Database Center for Life Science of the Research Organization of Information and Systems.

Finally, I would like to express my gratitude to my parents, Haruyo and Keisuke Kawashima, my brother Takeshi Kawashima, my wife, Ranko Kawashima and my daughters Hanako and Haru Kawashima for giving warm encouragement continuously.

Contents

Chapter 1 General Introduction

Background

Biology has been a discipline focusing on observing, classifying and describing research subjects since the era of Aristotle in ancient Greece, who is considered to have initiated sciences related to the modern biology (Mager 2002). With the significant rise of molecular biology since the middle of the 20th century and the continued development of various kinds of high-performance scientific instruments, the amount of information in the biological field continues to this day to increase at an accelerating rate. Since the 1960s, biological information began to be compiled as databases. Among these, the Atlas of Protein Sequence and Structure, which is the pioneering work by Dr. Margaret Dayhoff, is known as the origin of biomolecular databases (Gauthier *et al*. 2018). This is a collection of known protein sequences as a book published in 1965. In 1971, Protein Data Bank (PDB), which is a database of protein 3D structures, was released as one of the earliest computerized biological databases (Berman 2008). Since the 1990s, studies producing a vast amount of data, such as various genome sequencing projects and many kinds of omics research, have become very common, and consequently the number of databases continues to increase until today. For example, the Nucleic Acids Research (NAR) journal started to publish an annual special issue dedicated to biological databases in 1993 (Imker 2018), and as of December 2018, 1,699 databases have been published as papers in NAR. As another example, Integbio Database Catalog, which is a catalog of biological databases developed in Japan, includes 1,694 databases. Since there are many databases that were either published in journals other than NAR or even unpublished, it is said that tens of thousands of databases have already been developed. In the 1980s, databases were generally distributed on magnetic tapes and on CD-ROMs, and gradually shifted to distribution via the Internet by using the

File Transfer Protocol from the end of the 1980s when commercial Internet services began. After the 1990s on which the Internet has become widespread, it has become general to provide databases on the World Wide Web (WWW).

Historically, biological databases have been developed as flat-files, usually a sequential collection of entries, which are stored in a set of text files. Database entries are structured texts, and they are designed for human readability. Even after the WWW became a major media to publish databases, it was still common to display database entries on the web browser directly with minimal formatting. Thereafter, in the case of databases developed using relational database management systems (RDBMS), the manner in which database contents are displayed have shifted to dynamic generation of the web page from information retrieved from multiple tables in the RDBMS.

Initially, databases were developed as a collection of information of specific targets such as DNA sequences and chemical compounds. As the next step, several database centers have started to provide database portal services such as Entrez developed by National Center of Biotechnology Information (NCBI) and GenomeNet by Kyoto University Bioinformatics Center. These services have provided data retrieval functionalities against multiple databases which are integrated in different ways by each service. Currently, we can regard most biological databases as integrated ones because they are constructed by incorporating the contents of various external databases and ontologies.

Although these integrated database services have been quite useful and become indispensable resources for life science research, users must make great efforts when collecting the necessary data from these databases and organizing them to carry out data science research. This is partly because they tend to be siloed: that is, databases are isolated from each other because of the lack of semantically explicit external links. In addition, databases utilize different information technologies, data formats, vocabularies, and ontologies, and metadata is often insufficient. These issues hinder the integrative use of databases.

In the WWW world, Sir Tim Berners-Lee proposed a new concept named the Semantic Web, as an extension of the existing WWW in 2001 (Berners-Lee *et al*. 2001). While the existing WWW consisting of hyperlinked documents are basically designed to be read by humans, the Semantic Web (SW) aims to be a web of data read by machines. It is supposed that the web of data makes it possible for machines to use it without human intervention. Since most biological databases have already been developed on the WWW, it can be expected that it makes them more machine-readable by applying SW technology. From that point of view, several pioneering projects such as UniProt (The Uniprot Consortium 2015) and Bio2RDF (Belleau et al. 2008) began to develop RDF versions of their databases. As a result, they have succeeded to provide machine-readable and reusable bio- databases. Today, various biological databases have been made available in RDF following these pioneering work.

Objectives

In this thesis, I will present a method for integrally using multiple biological RDF datasets. First, to introduce the classical flat-file format database, I will describe the AAindex database which I developed (Chapter 2).

As mentioned in the previous section, the semantic web is considered as a promising technology for addressing the issues described in the previous section. The semantic web consists of a set of specifications standardized by the World Wide Web Consortium (W3C) such as Resource Description Framework (RDF), SPARQL Protocol and RDF Query Language (SPARQL) and Web Ontology Language (OWL). In these specifications, RDF is used to describe the data. However, it has become clear that just exposing existing databases as RDF is not insufficient to realize the Web of Data, consisting of interlinked machine-readable data on the Web. This is because these specifications

provide no clues as to how to model particular knowledge or what type of ontology should be used to represent data in RDF. I have devised a set of guidelines which have been adopted by the National Bioscience Database Center (NBDC) to address these issues (Chapter 3). Then, I will describe the NBDC RDF portal which is an RDF-based life science dataset repository. All the datasets in this repository have been reviewed by the NBDC in terms of complying with the guidelines. I also show that these reviewed datasets enable us to efficiently query multiples datasets.

The contents of Chapter 2 and Chapter 3 in this thesis have been published as follows:

Shuichi Kawashima, Hiroyuki Ogata and Minoru Kanehisa, AAindex: Amino Acid Index Database. *Nucleic Acids Research* **27**, 368-369 (1999) published by Oxford University Press.

Shuichi Kawashima, Toshiaki Katayama, Hideki Hatanaka, Tatsuya Kushida and Toshihisa Takagi, NBDC RDF portal: a comprehensive repository for semantic data in life sciences. *Database*, doi: 10.1093/database/bay123 (2018) published by Oxford University Press.

Chapter 2 A development of a bio-database for collecting amino acid physicochemical properties

Introduction

The variety and specificity of protein three-dimensional structures and biological functions are due to the combination of the 20 different amino acids as specified by the genetic code. The amino acids are the building blocks of proteins each having different characteristics in terms of the shape, the volume, and chemical reactivity among others. A large body of experimental and theoretical research has been performed to characterize physicochemical and biochemical properties of individual amino acids; the derived property is often represented by a set of 20 numerical values that is called the amino acid index.

In addition to the properties of individual amino acids, the relations between amino acids are also represented by numerical values in the analysis of protein sequences and structures. In particular, the amino acid substitution matrix, also called the amino acid similarity matrix, is the basis for optimization in protein sequence alignments and similarity searches. The amino acid mutation matrix is generally a set of 20 x 20 numerical values, or symmetric. The AAindex database is a collection of published amino acid indices and mutation matrices.

Background

In 1988 Nakai *et al.* collected 222 amino acid indices from research papers and investigated their relationships by hierarchical cluster analysis (Nakai *et al.* 1988). They identified four major classes, α-helix and turn properties, β-strand propensity, hydrophobicity that can further be divided into subclasses, and other physicochemical properties such as bulkiness of amino acid residues. In 1996 Tomii and Kanehisa (Tomii and Kanehisa 1996) increased the size of the collection to include 402 indices and re-performed the clustering. The result was generally in good agreement with the previous work, but for the sake of convenience, the collection was divided into six major classes: α and turn properties, β propensity, amino acid composition, hydrophobicity, physicochemical properties, and other properties.

Tomii and Kanehisa also collected 42 amino acid mutation matrices from the literature and conducted extensive analysis on the correlations among them and with the amino acid indices. The AAindex database that was initiated by Nakai *et al.* was expanded by Tomii and Kanehisa and is still continuously updated (Kawashima *et al.* 1999; Kawashima and Kanehisa 2000; Kawashima *et al.* 2008).

In 2005, Pokarowski *et al*. compared 29 published matrices of protein pairwise contact potentials, i.e. energy functions that are obtained from the statistical analysis of protein structures (Pokarowski *et al*. 2005). These potentials have long been used to predict protein structures *in silico*. Pokarowski and coworkers elucidated that each of the contact potentials is similar to one of two popular matrices derived by Miyazawa and Jernigan (Miyazawa and Jernigan 1999). Recently, working on 29 mostly new amino acid substitution matrices and five contact potentials, the same team obtained segregation

of substitution matrices (Pokarowski et al. 2007) similar to Tomii and Kanehisa (Tomii and Kanehisa 1996). Moreover, they found intermediate links between substitution matrices, contact potentials matrices and potentials that exhibit mutual correlations of at least 0.8. In both works, Pokarowski and coworkers approximated matrices by simple functions of amino acid indices, which allow us to comprehend better the exchangeability of amino acids as well as the residue-residue interactions in proteins (Pokarowski et al. 2005, 2007). These relations between substitution matrices, contact potentials, and amino acid indices provide motivation to extend the AAindex database. I have compiled the data collected in the study on contact potentials (Pokarowski et al. 2007) as a new section of the AAindex database, named AAindex3.

The structure of the AAindex database

The AAindex database is a flat-file database that consists of three sections: AAindex 1 for the amino acid indices, AAindex2 for the amino acid mutation matrices and AAindex3 for the amino acid contact potentials. The contents and the format of the AAindex are as follows.

AAindex 1

The AAindex 1 section currently contains 434 amino acid indices. A sample entry of AAindex1 is shown in Figure 1. Each entry consists of an accession number, a short description on the index, the reference information, and the numerical values for the property of 20 amino acids. In addition, it contains neighbor information; namely, the cross-links to other entries with an absolute value for the Pearson correlation coefficient of 0.8 or larger. With the links, the user can identify a set of entries describing similar properties. In some instances, the values are not reported for all 20 amino acids.

When available I adopt the estimates by Kidera *et al.* (Kidera *et al.* 1985) who tried to fill missing values by statistical considerations. When the estimates were not available, the missing values were either replaced by the mean value of the rest or simply filled with zeros.

AAindex 2

The AAindex2 section currently contains 66 amino acid mutation matrices: 47 symmetric matrices and 19 non-symmetric matrices. A sample entry of AAindex2 is shown in Figure 2. The format of the entry is almost the same as that of AAindex 1 except that it contains 219 numerical values (20 diagonal and 20 x 19/2 off-diagonal elements) for a symmetric matrix and 400 or more numerical values for a non-symmetric matrix (some matrices include a gap or distinguish two states of cysteine).

AAindex 3

The AAindex3 section currently contains 47 amino acid contact potential matrices: 44 symmetric matrices and 3 non-symmetric matrices. The format of the entry is almost the same as that of AAindex2.

Availability

The AAindex database can be retrieved through the DBGET/LinkDB system of the Japanese GenomeNet service at http://www.genome.ad.jp/dbget/. The DBGET/LinkDB system (Fujibuchi *et al.* 1998) integrates various molecular biology databases and is especially suited for using hyperlinks to

related entries within the AAindex database as well as to other databases. Alternatively, the entire database may be copied and used locally. The URL for anonymous FTP is: ftp://ftp.genome.ad.jp/db/genomenet/aaindex/.

BioRuby (Goto *et al*. 2010), which is a bioinformatics library in the Ruby programming language, has provided useful functions to handle the AAindex database (http://bioruby.org/). Moreover, EMBOSS (Rice *et al*. 2000) has provided a program to extract the index data from the AAindex entry.

Discussion

AAindex has been used for various kinds of protein sequence analysis such as predicting protein subcellular localization (Sarda *et al*. 2005), immunogenicity of MHC class I binding peptides (Tung and Ho 2007), protein SUMO modification site (Liubc *et al*. 2007), and coordinated substitutions in multiple alignments of protein sequences (Afonnikov and Kolchanov 2004). As a more recent research example, Li *et al*. have developed a novel PTM prediction tool on the whole proteome scale (Li *et al*. 2018). They employed the AAindex to create descriptors of the physicochemical microenvironment of modified sites for their tool. Given the examples cited here, AAindex has acquired recognition as a useful resource in bioinformatics. However, as with other flat-file format databases, users must write a program in order to extract arbitrary elements from the database entries. In addition, the contents are not machine-readable because they were developed on the premise that the data would be interpreted by humans. In the next chapter, I will present an attempt to realize integrated databases with higher machine-readability.

```
H COWR900101
D Hydrophobicity index, 3.0 pH (Cowan-Whittaker, 1990)
R PMID:2134053
A Cowan, R. and Whittaker, R.G.
T Hydrophobicity indices for amino acid residues as determined by
  high-performance liquid chromatography
J Peptide Res. 3, 75-80 (1990)
C GUOD860101    0.920  BLAS910101    0.885  FAUJ830101    0.876
  EISD860103    0.868  EISD840101    0.863  WILM950101    0.860
  PLIV810101    0.857  JURD980101    0.855  MEEJ810102    0.849
  NADH010102    0.848  KYTJ820101    0.845  RADA880101    0.840
  NADH010103    0.825  MIYS850101    0.824  MEEJ810101    0.823
  CHOC760103    0.820  RADA880104    0.818  ROSM880105    0.817
  RADA880107    0.810  NADH010104    0.807  CIDH920104    0.803
  BULH740101   -0.804  MIYS990102   -0.825  MIYS990101   -0.826
  ROSM880101   -0.849  GRAR740102   -0.854  KIDA850101   -0.868
  WOLS870101   -0.883  ROSM880102   -0.897
I    A/L     R/K     N/M     D/F     C/P     Q/S     E/T     G/W     H/Y     I/V
    0.42   -1.56   -1.03   -0.51    0.84   -0.96   -0.37    0.00   -2.28    1.81
    1.80   -2.03    1.18    1.74    0.86   -0.64   -0.26    1.46    0.51    1.34
//
```

Figure 1. An example of a database entry in AAindex1. Each record of an entry is identified by one-letter codes: H, accession number; D, definition of the entry; R, reference database identifier; A, author(s); T, title of the journal article; J, journal citation information, C, accession numbers of similar entries having correlation coefficients of 0.8 (-0.8) or more (less); I, actual data in the specified order.

```
1H MIRL960101
D Statistical potential derived by the maximization of the harmonic mean of Z
  scores
R PMID:9000638
A Mirny, L.A. and Shakhnovich, E.I.
T How to derive a protein folding potential? A new approach
  to an old problem
J J. Mol. Biol. 264, 1164-1179 (1996)
M rows = ARNDCQEGHILKMFPSTWYV, cols = ARNDCQEGHILKMFPSTWYV
  -0.13
   0.43  0.11
   0.28 -0.14 -0.53
   0.12 -0.72 -0.30  0.04
   0.00  0.24  0.13  0.03 -1.06
   0.08 -0.52 -0.25 -0.17  0.05  0.29
   0.26 -0.74 -0.32 -0.15  0.69 -0.17 -0.03
  -0.07 -0.04 -0.14 -0.22 -0.08 -0.06  0.25 -0.38
   0.34 -0.12 -0.24 -0.39 -0.19 -0.02 -0.45  0.20 -0.29
  -0.22  0.42  0.53  0.59  0.16  0.36  0.35  0.25  0.49 -0.22
  -0.01  0.35  0.30  0.67 -0.08  0.26  0.43  0.23  0.16 -0.41 -0.27
   0.14  0.75 -0.33 -0.76  0.71 -0.38 -0.97  0.11  0.22  0.36  0.19  0.25
   0.25  0.31  0.08  0.65  0.19  0.46  0.44  0.19  0.99 -0.28 -0.20  0.00  0.04
   0.03  0.41  0.18  0.39 -0.23 -0.29  0.27 -0.38 -0.16 -0.19 -0.30  0.44 -0.42 -0.44
   0.10 -0.38 -0.18  0.04  0.00 -0.42 -0.10 -0.11 -0.21  0.25  0.42  0.11 -0.34  0.20  0.26
  -0.06  0.17 -0.14 -0.31 -0.02 -0.14 -0.26 -0.16 -0.05  0.21  0.25 -0.13  0.14  0.29  0.01 -0.20
  -0.09 -0.35 -0.11 -0.29  0.19 -0.14  0.00 -0.26 -0.19  0.14  0.20 -0.09  0.19  0.31 -0.07 -0.08  0.03
  -0.09 -0.16  0.06  0.24  0.08  0.08  0.29  0.18 -0.12  0.02 -0.09  0.22 -0.67 -0.16 -0.28  0.34  0.22 -0.12
   0.09 -0.25 -0.20  0.00  0.04 -0.20 -0.10  0.14 -0.34  0.11  0.24 -0.21 -0.13  0.00 -0.33  0.09  0.13 -0.04  0.06
  -0.10  0.30  0.50  0.58  0.06  0.24  0.34  0.16  0.19 -0.25 -0.29  0.44 -0.14 -0.22  0.09  0.18  0.25 -0.07  0.02 -0.29

//
```

Figure 2. An example of a database entry in AAindex2. The data format is the same as that described in Figure 1. The order of the matrix elements may be computed by the equation or examined in the database documentation file.

Chapter 3 NBDC RDF portal: a comprehensive repository for semantic data in the life sciences

Introduction

In the life sciences, enormous amounts of diverse data are continually being produced and numerous databases have been made available on the Internet (Rigden and Fernández 2018). It is becoming increasingly important to unify and integrate these databases in order to study complex biological phenomena (Stein 2003), but these independently-developed databases use a variety of different data formats, vocabularies, and identifiers, making it extremely difficult to use multiple databases in an integrated way (Slater *et al.* 2008). However, the semantic web (SW) is attracting attention as a promising approach to addressing these issues (Antezana *et al.* 2009; Chen *et al.* 2013).

The SW is a set of technologies that aims to create a Web of Data, consisting of interlinked machine-readable data on the Web. It includes the following core technologies: the Resource Description Framework (RDF) to describe the data, SPARQL Protocol and RDF Query Language (SPARQL) to query RDF datasets, RDF Schema (RDFS) to provide a vocabulary for modeling RDF data, and the Web Ontology Language (OWL) to describe the properties and classes needed to develop ontologies. RDF is a framework for representing information about resources on the Web in the form of subject–predicate–object triples. Subjects and predicates are described using Uniform Resource Identifiers (URIs) that act as global identifiers, while objects can be described using either URIs or literals. Objects represented by URIs can become the subject of another triple, thus connecting them and resulting in RDF datasets forming graph structures.

Life science data is currently being provided in a wide variety of formats, such as flat files and dump files from relational database management systems, as well as in JavaScript Object Notation (JSON), Extensible Markup Language (XML), and Comma-Separated Values (CSV) formats. It is often extremely time-consuming for users to extract the necessary data from these diverse sources and construct a dataset for use in their research. In fact, according to 'the first National Institutes of Health (NIH) Strategic Plan for Data Science' released on June 4, 2018 (NIH 2018), data scientists in a wide array of fields are reported to spend about 80% of their work time obtaining existing datasets and organizing data. In order to load the gathered data into a local relational database management system (RDBMS), it is also necessary to normalize the data and design a database schema. In contrast, with RDF, it is possible to load several different RDF datasets into an RDF store without any additional processing, avoiding the work that would otherwise be required. In addition, since RDF data is described using global URIs, there is no need to consider issues such as the same identifiers being assigned to different entities in different databases. Several attempts have been made to utilize such SW technology features, which enhance data interoperability in the life sciences (Belleau *et al.* 2008; Marshall *et al.* 2012; Katayama *et al.* 2013, 2014). In addition, fundamental databases, such as UniProt (The Uniprot Consortium 2015), PDB (Kinjo *et al*. 2017), PubChem (Fu *et al.* 2015), and Ensembl (Jupp *et al.* 2014), are already available in RDF.

The National Bioscience Database Center (NBDC) in Japan aims to promote the development of life science databases. Since its foundation, the NBDC has recognized the potential of SW technologies to integrate diverse databases. To achieve that goal, the NBDC and the Database Center for Life Science (DBCLS) have organized the BioHackathon series (The DBCLS BioHackathon Consortium 2010; Katayama *et al*. 2011, 2013, 2014), which is designed to encourage discussions about applying the SW to life science databases and to facilitate the development of RDF datasets and tools.

The NBDC has also funded the development of various life science databases and advised the groups involved to release them in RDF. This has led to a variety of databases becoming available in RDF, produced by both funded groups and other domestic research groups. Initially, each research group was left to decide how to publish their RDF datasets. However, it has proved difficult to provide SPARQL endpoints for all groups and it has become apparent that there is a need for a service that allows people to list, download, and query RDF datasets. Given this, I began developing the NBDC RDF portal to meet these needs.

The NBDC RDF portal has the following two features. First, it is an RDF dataset repository, hosting datasets developed by Japanese research groups in a wide variety of research fields. Second, each submitted dataset is reviewed by the NBDC and only those that ultimately pass this review are accepted. I have compiled a set of guidelines for converting databases into RDF and utilize these to review the quality of each dataset in terms of interoperability and queryability. This chapter describes the guidelines and the NBDC RDF portal in detail.

RDF portal guidelines and review policy

Background of creating the guidelines

All datasets provided by the RDF portal have been reviewed by the NBDC to assess their conformance to the guidelines below. In 2018, I also began using an automatic verification tool, which my colleagues and I developed and is described later in this chapter, prior to manual review. Before discussing the guidelines themselves, however, I first describe the background to creating them and the associated review policy.

The DBCLS hosts a monthly hackathon event, called SPARQLthon, that aims to promote SW applications in the life sciences and technical information sharing among developers. Based on experience and knowledge gathered from these events, I have compiled a set of useful practices known as the "DBCLS guidelines for RDFizing databases", which is available at https://github.com/dbcls/rdfizing-db-guidelines.

Several useful guidelines have already been published, such as a collection of patterns for modeling linked data (Dodds and Davis 2012), and instructions on how to represent data in RDF for exposure in Open PHACTS (Haupt *et al*. 2013) or select bio-ontologies (Malone *et al*. 2016). By combining these, our guidelines aim to answer some of the questions life science database developers with little SW experience may have when creating datasets in RDF.

From these guidelines, I then selected topics that could be used to objectively evaluate such datasets, compiling a guideline subset designed for the RDF portal (herein, called the RDF portal guidelines). Before being included in the RDF portal, all datasets are first reviewed according to these guidelines to ensure a sufficient level of interoperability.

RDF portal guidelines

Now, I summarize the RDF portal guidelines. The Qualified Name (QName) prefixes used in this article are shown in Appendix 1.

1. Primary resources should be instances of some ontology class

Life science databases usually cover either one or a few subjects, and their content is organized by subject. For example, UniProt (The Uniprot Consortium 2015) is a database of protein sequences, each represented as an instance of the up:Protein class in the UniProt RDF. As another example, ChEMBL is a database on the bioactivity of chemical compounds, and its entries are instances of classes such as cco:Assay, cco:Activity, or cco:Substance (Willighagen *et al*. 2013). URIs that represent such subjects (herein, called primary resources) should be defined as instances of an ontology class. This helps to reduce the search space of SPARQL queries. The following example represents the statement indicating that the resource refex: RFX0000000001 is an instance of the refexo: RefExEntry class.

```
(ex.) refex:RFX0000000001 rdfs:type refexo:RefExEntry .
```

2. Primary resources should have human-readable labels

Even though RDF is primarily intended to make data more machine-readable, providing natural-language labels for resources can be useful, especially when writing SPARQL queries or displaying application results. Linked Data Patterns, the previously-mentioned an online design pattern catalog for linked data development, advises to "Ensure that every resource in a dataset has an rdfs:label property." Our guidelines also recommend adding labels to as many URIs as possible, but at a minimum all primary URIs must be labeled using the rdfs:label property. When multiple labels are needed, I recommend using the skos:altLabel property.

Some of the datasets in the RDF portal contain labels written in Japanese, partly because they were developed in Japan. For resources with multiple labels in different languages, each label should have

a language tag so that labels in a specific language can be selected. On the other hand, language-independent literals, such as numerical values and database entry IDs, should not have language tags. The following example represents a statement where the resource ggdonto:CON00006 has both an English and Japanese label indicated by the rdfs:label property as its predicate.

```
(ex.) ggdonto:CON00006 rdfs:label "Fucosidosis"@en, "フコシドーシス"@ja .
```

3. Primary resources should provide their local database IDs.

The local database ID is generally placed after the last slash at the end of each primary URI. However, when printing search results and showing them in an application's user interface, users often find it easier to work with local database IDs rather than full URIs, and local IDs can also be convenient when writing SPARQL queries, for example. To enable this, the primary URI should have a dcterms:identifier property whose value is a literal containing the local ID. The following example represents a statement where the resource refex:RFX0000000001 has a local database identifier indicated by the dcterms:identifier property as its predicate.

```
(ex.) refex:RFX0000000001 dcterms:identifier "RFX0000000001" .
```

4. Links to external resources should be provided in a consistent format

With the SW, it is essential that both users and machines are able to explore the RDF-based Web of Data. Life science databases often provide abundant cross-links to external database entries, but there

are often several different URIs referring to the same database entry, and no general rules as to which URI to use when linking to external databases. Therefore, simply converting such databases into RDF may not enhance the Web of Data, because these different URIs, even if they are ultimately redirected to the same Internet URI, are regarded as different RDF resources.

To address this problem, I require all external resources to be referred to using the URIs provided by identifiers.org (Juty *et al*. 2012) and the rdfs:seeAlso property. This ensures that the same URI will always be used to refer to the same resource in different RDF datasets. One exception to this is that references to the primary resources within an RDF dataset officially released by the database provider must use the URIs defined in the dataset, because datasets do not usually use identifiers.org URIs to describe their own resources. In such cases, redundant links must, therefore, be included to both the canonical and identifers.org URIs. The canonical URIs used for the main RDF datasets are listed in Table 2.

There are two other exceptions to this rule for external resources. References to articles or books should use the relevant PubMed URI or Digital Object Identifier (DOI) with the dcterms:references property, and images should use the foaf:depiction property. The following example represents the statement where the resource refex:RFX0000000001 has a link to an entry of NCBI Gene as indicated by the rdfs:seeAlso property. The URI of this triple's object is provided by identifiers.org because NCBI Gene does not officially provide an RDF dataset.

```
(ex) refex:RFX0000000001 rdfs:seeAlso <http://identifiers.org/ncbigene/2> .
```

The next example represents the statement where the resource jpost:PRT201_1_B5MDL5 has a link to UniProt protein B5MDL5. The URI of this triple's object is the canonical URI used in the RDF dataset officially provided by UniProt.

```
(ex) jpost:PRT201_1_B5MDL5 rdfs:seeAlso
<http://purl.uniprot.org/uniprot/B5MDL5> .
```

This last example represents the statement where the resource pdb:2KVQ has a reference link to an article with PubMed identifier 20413501.

```
(ex) pdb:2KVQ dcterms:references <http://rdf.ncbi.nlm.nih.gov/pubmed/20413501> .
```

5. The minimum metadata should be provided

Dataset submitters should provide the following metadata: the dataset providers' and creators' names, version, date issued, license, and NBDC database classification tags. It is particularly important that license information is provided, so users can determine how the dataset can be used. This is also a condition for the dataset to be findable, accessible, interoperable, and reusable (FAIR) (Wilkinson 2016). The RDF portal only accepts datasets provided with some type of open license. Currently, most datasets are available under the Creative Commons License.

6. Existing ontologies should be used where possible

Using common ontologies for different datasets is one of the most important ways of enhancing the interoperability of RDF datasets. Although the semantics of individual RDF datasets are left to their developers, I encourage the use of existing ontologies where possible. The DBCLS guidelines for RDFizing databases, therefore, list the ontologies I recommend.

7. The domain and range of each user-defined property should be explicitly defined

When converting a database into RDF, it may be necessary to define new properties, particularly to express relationships between concepts. When doing so, each property's domain and range should be defined as explicitly as possible. This helps to make queries more efficient and create applications that build SPARQL queries automatically.

8. A schema diagram should be provided

A schema diagram greatly aids in writing SPARQL queries. Such a diagram should therefore be provided.

9. Sample queries should be provided

It is very helpful to see examples of typical queries when querying RDF datasets using SPARQL. At least one example query should, therefore, be provided.

10. DNA and protein sequence coordinates should be described using FALDO

Many life science databases provide structural and functional annotations to genome or protein sequences. The Feature Annotation Location Description Ontology (FALDO) (Bolleman *et al*. 2016) should be used to specify the point in a sequence to be annotated. This is already used in various RDF datasets, such as UniProt, Ensembl, and DDBJ (Mashima *et al*. 2017), and using common sequence coordinates will enable us to achieve highly interoperable annotations.

11. Structured values should be used for values with units

Structured values should be used to describe numerical values with units by using the Semanticscience Integrated Ontology (SIO) (Dumontier *et al*. 2014) and giving at least a sio:SIO_000300 property (i.e., sio:has-value) for each value and a sio:SIO_000221 property (*i.e.*, sio:has-unit) for each unit, as in the example below. Structured values should be typed using an appropriate ontology class, included as a sio:SIO_000216 property (i.e., sio:has-measurement-value). The Units of Measurement Ontology (UO) (http://bioportal.bioontology.org/ontologies/UO) should be used to express units where possible, but other ontologies can be used for units not included in the UO. The following example shows a resource (ex:m1) representing a measurement that the amount of fibrinogen (cmo:CMO_0000209) in a subject's blood was 2.15 milligrams per milliliter (uo:UO_0000273).

```
(ex.)
ex:m1 sio:SIO_000216 [
  rdf:type          cmo:CMO_0000209;
  sio:SIO_000300    2.15;
  sio:SIO_000221    uo:UO_0000273
] .
```

Review policy

With RDF, any type of information can be described explicitly on the Internet. However, current specifications provide no clues as to how to model particular knowledge or what type of ontology should be used to represent data or knowledge using an RDF. Different ontologies and models can be used to describe the same information, so just exposing databases in RDF will not necessarily improve interoperability from a semantic viewpoint without guidelines or agreement about the semantics. In order to achieve maximum interoperability, it is clearly essential for different communities to agree on common ontologies and models, but, at present, coming to such an agreement is extremely difficult. According to Splendiani *et al*., incentives like "the endorsement by granting agencies, by major journals as well as by main information providers" would help to promote SW in the life sciences (Splendiani *et al*. 2011).

With regard to semantics in the life sciences, my policy is essentially to respect the original description in each submitted RDF, because I assume that the developers working in each field fully understand these semantics. On the other hand, for general statements that appear in all research areas, such as linking to other database entries, labeling resources, mapping onto genome coordinates, and describing numerical values with units, I require the use of specific ontologies and models to increase interoperability among different RDF datasets. Developers can thus retain their original statements, except where they are required to use vocabularies defined in the RDF portal guidelines, due to RDF allowing redundant statements, an advantage that comes from the flexibility of its graph structure.

In the following simple example, resource ex:r1 cites document pubmed:12345 as providing an authoritative description:

(1) ex:r1 cito:citesAsAuthority pubmed:12345 .

However, the guidelines require the dcterms:references property to be used when referring to the literature:

(2) ex:r1 dcterms:references pubmed:12345 .

Although statement (1) has more detailed citation semantics than statement (2), using the same property in all datasets makes it easier to search across datasets. I would, therefore, instruct the submitter to add statement (2) to their dataset, leaving it to them to decide whether or not to include statement (1) as well. The SW also offers another solution that satisfies the need to both represent detailed meaning and to use common property for increased interoperability; namely defining a user-defined property that represents the detailed semantics as a sub-property of dcterms:references:

(3) ex2:newCitesAsAuthority rdfs:subClassOf dcterms:references

However, with regard to the RDF portal guidelines, I ask submitters to add statement (1), even if it is redundant. This is because doing otherwise would unnecessarily complicate writing queries and making inferences on extremely large life science datasets. With the current RDF store, it would also be generally impractical in terms of performance.

Implementation

The RDF portal currently uses OpenLink Virtuoso version 7.2.4 as its RDF store, running on a Unix server with 48 cores and 1.2 TB memory. The user interface of the site is implemented in Javascript using several libraries: CodeMirror 5.0, D3.js 4.13.0, JQuery v2.1.4, JQuery UI 1.11.4, jQuery Cookie Plugin 1.4.1, jQuery Easing 1.3 and webcomponents 0.5.5.

Although it would be desirable, from a usability standpoint, to store all the datasets in one RDF store instance, I have created separate Virtuoso instances for particularly large datasets because, in our experience, a single Virtuoso instance can handle at most twenty billion triples without problems in our environment. Currently, the DDBJ and DBKERO RDFs (Mashima *et al*. 2017; Suzuki *et al*. 2018) are each stored in their own instances. The metadata is always stored in the primary instance, for all datasets. Figure 3 shows an overview of the system architecture.

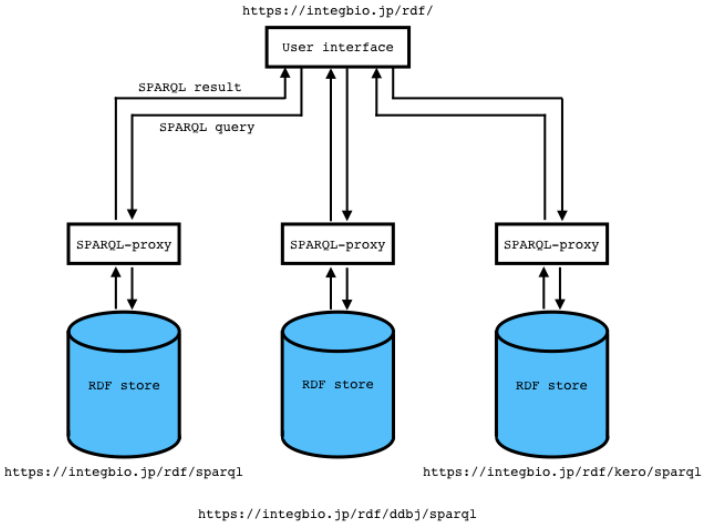

Figure 3. Overview of the system architecture. The RDF portal uses OpenLink Virtuoso as its RDF store. The SPARQL endpoint uses the SPARQL-proxy software for its front end. Currently, there are three Virtuoso instances for the primary instance, DDBJ RDF and DBKERO RDF.

SPARQL-proxy

Providing a SPARQL endpoint is one of the most effective ways that users can easily utilize RDF data. Various SPARQL endpoints are available for major RDF datasets in the life sciences such as UniProt and the EBI RDF platform. The RDF portal also provides SPARQL endpoints services. When providing a SPARQL endpoint, it is important to properly control the submitted queries so that the RDF data management system will not be burdened by heavy queries. Functionality to filter unsafe queries is also needed. In order to easily make use of such functionalities for any SPARQL endpoint running on various environments and variety of RDF stores, my colleagues and I have developed a portable web application named SPARQL-proxy. The SPARQL endpoint of RDF portal uses SPARQL-proxy for its front end.

SPARQL-proxy is implemented in Node.js. To start it, the user simply executes the following command from the directory where it is built.

```
$ PORT=3000 SPARQL_BACKEND=<url> npm start
```

It works as a proxy server for the SPARQL endpoint at the specified URL via the SPARQL_BACKEND environment variable. The provider of the SPARQL endpoint can expose the proxy URL instead of the original endpoint URL. In the above case, port 3000 is assigned but it can be 80 or the provider can configure an HTTP reverse proxy to point to that port. All other options such as a cache system of choice can also be set via the environment variables. SPARQL-proxy provides two web interfaces: one is the dashboard for administrators to monitor the execution of jobs (Figure 4) and the other is the query submission form for debugging use. Administrators can see the execution logs, cancel running/queued jobs and remove cached results. Submitted queries are validated to check for unsafe instructions, such as a SPARQL Update query, prior to passing them to the backend RDF

store. The job timeout and the number of concurrent requests can also be specified. In order to improve the response time of the requested query, SPARQL-proxy provides a function that caches each SPARQL result and returns a cached result when the same query is submitted. The provider of the service can select from one of the following caching mechanisms: a local file, memory, Redis, and Memcached. To reduce the size of the cache, cached results can be compressed using snappy.js which is a JavaScript implementation of Google's Snappy compression library. SPARQL-proxy is freely available, and the source code is provided on the GitHub repository at https://github.com/dbcls/sparql-proxy. The detailed usage is shown in Appendix 3.

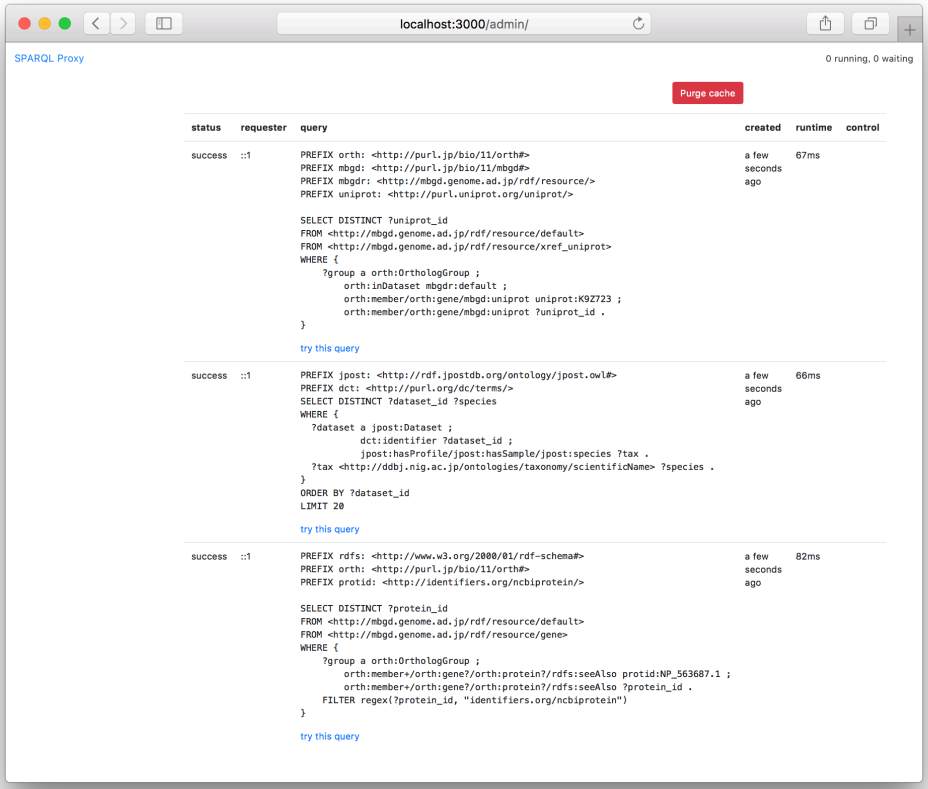

Figure 4. The dashboard page of SPARQL-proxy

Persistent URLs

Cool URI is a concept of ideal URIs to serve as fundamental blocks for the Semantic Web. The use of Cool (i.e., persistent) URIs is recommended for all SW URIs (https://www.w3.org/TR/cooluris) but designing them is not easy. In addition, it is sometimes necessary to use existing (non-Cool) URIs. For example, Cool URIs should not change, but if (for example) a research institute closes, its domain may also become unavailable. Persistent Uniform Resource Locators (PURLs) can address this problem to some extent by redirecting a fixed Uniform Resource Locator (URL) to the current actual Web address. To support RDF development, my colleagues and I have created the purl.jp PURL service, which can be used to create new URLs when converting datasets to RDF. It is intended as a general-purpose service, not limited to the life sciences, and issues new URLs for life science applications under http://purl.jp/bio/.

Monban; A RDF Lint Tool

To comprehensively verify the posted RDF dataset from the viewpoint of compliance with the guidelines, my colleagues and I have developed the RDF lint tool named Monban. Currently, Monban can verify whether primary resources comply with guidelines 1, 2 and 3. The Monban software is available on GitHub: https://github.com/dbcls/monban. The detailed usage is shown in Appendix 4.

Current status of the NBDC RDF portal

The NBDC RDF portal (https://integbio.jp/rdf/) was launched in November 2015. As of November 2018, it contains 21 RDF datasets submitted by Japanese research groups, comprising over 45.5 billion

triples. Table 1 shows the statistics of the RDF datasets. Fact sheets of datasets are shown in Appendix 1. An up-to-date list and other statistics are available at https://integbio.jp/rdf/?view=matrix. It includes datasets from a wide variety of research areas, such as protein orthology, cancer genomics, glycobiology, transcriptomes, and toxicogenomics. At present, most datasets are only accessible as SPARQL endpoints from this site. I rely on developers to provide dataset updates, but my colleagues and I regularly update the datasets as far as possible at their request. For example, we currently update wwPDB/RDF and BMRB/RDF every 3 months, and Integbio Database Catalog/RDF every week.

Each dataset has one or more database classification tags which are used in the Integbio Database Catalog developed by NBDC. In the datasets view (http://integbio.jp/rdf/?view=list), by clicking the icon in the lower left of the web browser, a pane to filter and sort datasets will appear (Figure 5). In this filter function, users can filter the displayed datasets by selecting the tags mentioned above. Users can also sort datasets in ascending or descending order by the date of last update, dataset name, the number of triples, or the name of the data providers.

Each dataset has its own page; the page for RefEx (Ono *et al*. 2017) is shown in Figure 6. These pages contain the dataset's metadata, the number of out-links and other statistics, RDF model schema diagrams, sample SPARQL queries (linked to the SPARQL endpoint), and links to download the submitted RDF files. The RDF model schema for RefEx RDF is shown in Figure 7.

When loading an RDF dataset, the number of triples representing out-links (complying with guideline 4), is counted and used to automatically generate a network view (Figure 8). This shows that the site's datasets complement the main existing RDF datasets and contribute to enriching linked open data in the life sciences. Recently, my colleagues and I developed an efficient command-line tool, named Aramashi, to count the number of links. The detailed usage of Aramashi is shown in Appendix 5.

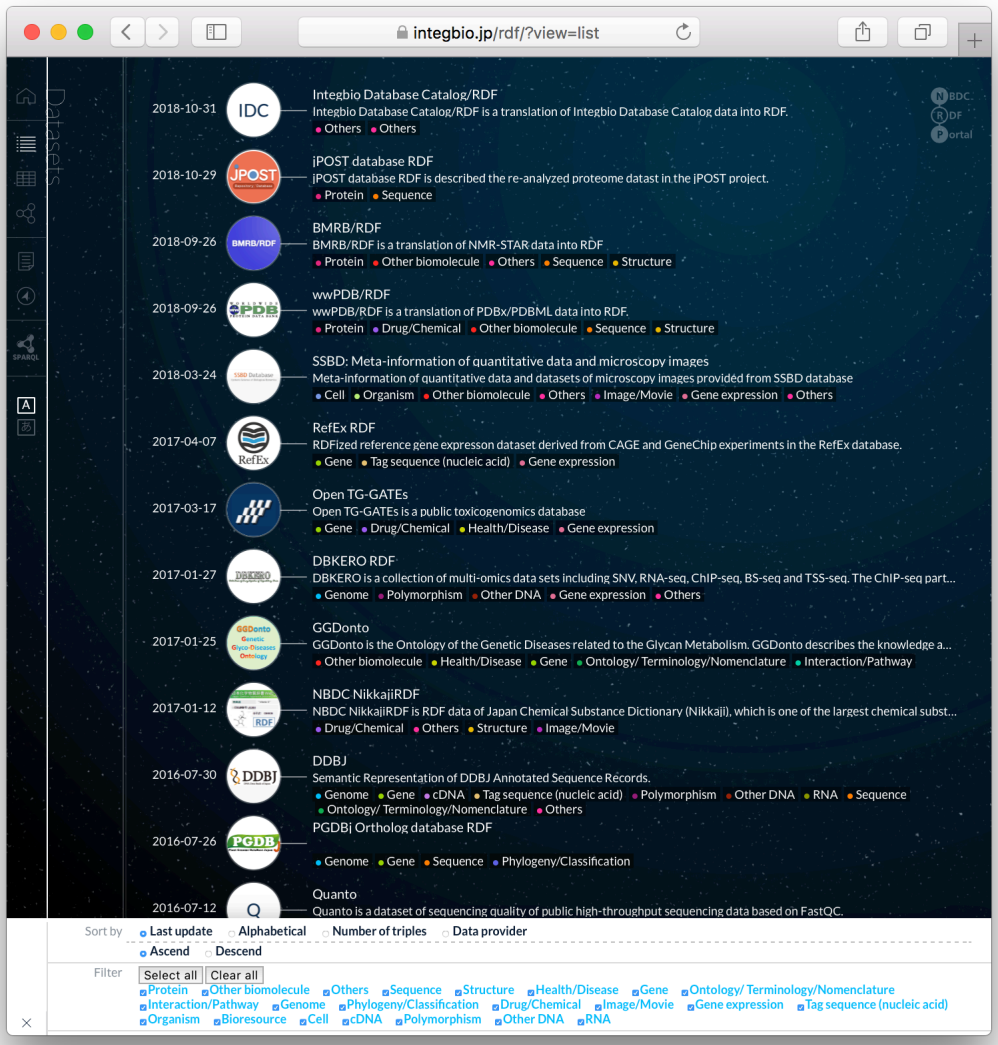

Figure 5. The dataset view of the NBDC RDF portal.

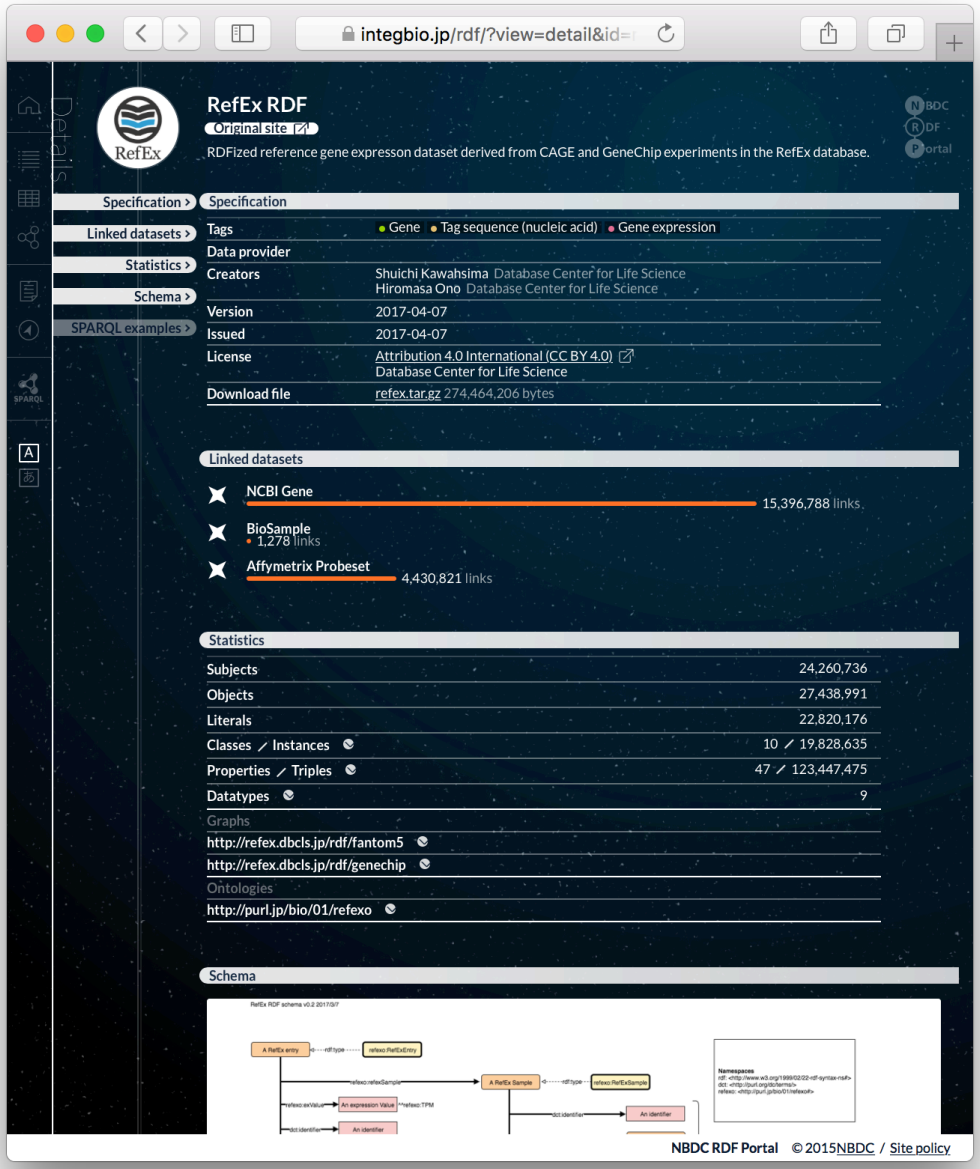

Figure 6. An example dataset page from the NBDC RDF portal. Each RDF dataset has its own page, which provides metadata, statistics, links to the RDF files, SPARQL query samples, and a link to the SPARQL endpoint

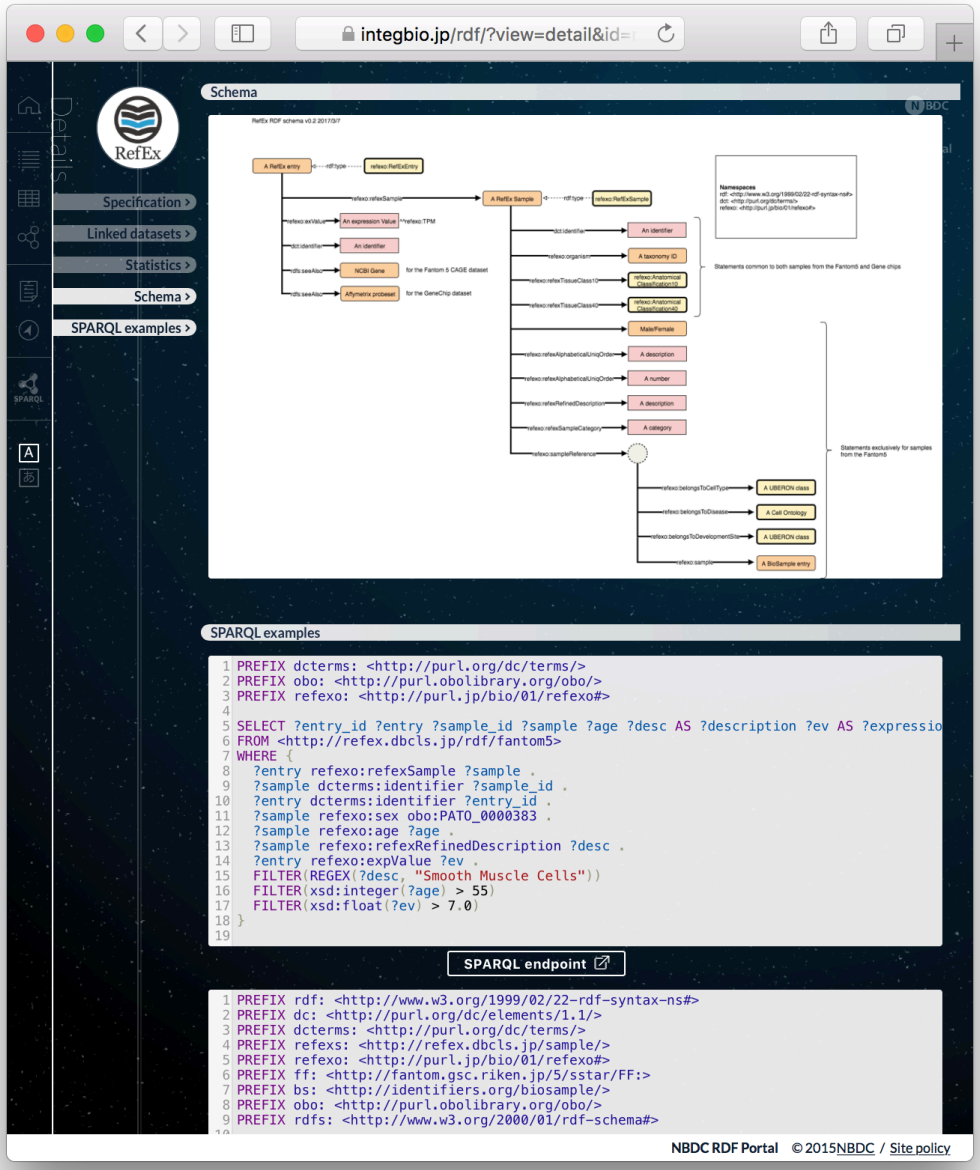

Figure 6. Cont.

Table 1. RDF datasets available via the NBDC RDF portal

RDF dataset	Number of triples
DDBJ	20,067,185,022
DBKERO RDF	11,017,998,412
Open TG-GATEs	6,800,384,609
wwPDB/RDF	4,481,680,698
MBGD RDF	1,609,018,143
Linked ICGC Dataset	577,082,774
NBDC KikkajiRDF	333,968,051
MBRB/RDF	281,996,472
RefEx RDF	123,447,370
Quanto	107,782,639
jPOST database RDF	99,128,038
FAMSBASE GPCR	21,297,786
PGDBj Ortholog database RDF	13,652,175
Dataset of WURCS-RDF	6,213,789
GlyTouCan	1,749,648
Integbio Database Catalog/RDF	92,875
PAConto	81,785
SSBD: Meta-information of quantitative data and microscopy images	40,300
GGDonto	39,439
GlycoEpitope	27,796
Metadata of JCM resources	8,896
Total number of triples	45,542,876,717

Table 2. Canonical URIs used in the major RDF datasets

RDF Dataset	A representative class of primary resources	Prefix of canonical URL
UniProt	core:Protein	http://purl.uniprot.org/uniprot/
Ensembl	obo:SO_0001217	http://rdf.ebi.ac.uk/resource/ensembl/
ChEMBL	cco:Substance	http://rdf.ebi.ac.uk/resource/chembl/molecule/
ExpressionAtlas	atlas:BaseLineExpressionValue	http://rdf.ebi.ac.uk/resource/expressionatlas/
	atlas:DifferentialExpressionRatio	http://rdf.ebi.ac.uk/resource/expressionatlas/
Reactome	biopax3:Pathway	http://identifiers.org/reactome/
BioModels	sbmlrdf:SBMLModel	http://identifiers.org/biomodels.vocabulary#
BioSamples	biosd-terms:Sample	http://rdf.ebi.ac.uk/resource/biosamples/sample
PubChem	compound	http://rdf.ncbi.nlm.nih.gov/pubchem/compound/
	substance	http://rdf.ncbi.nlm.nih.gov/pubchem/substance/
MESH	meshv:TopicalDescriptor	http://id.nlm.nih.gov/mesh/
wwPDB	PDBo:datablock	http://rdf.wwpdb.org/pdb/

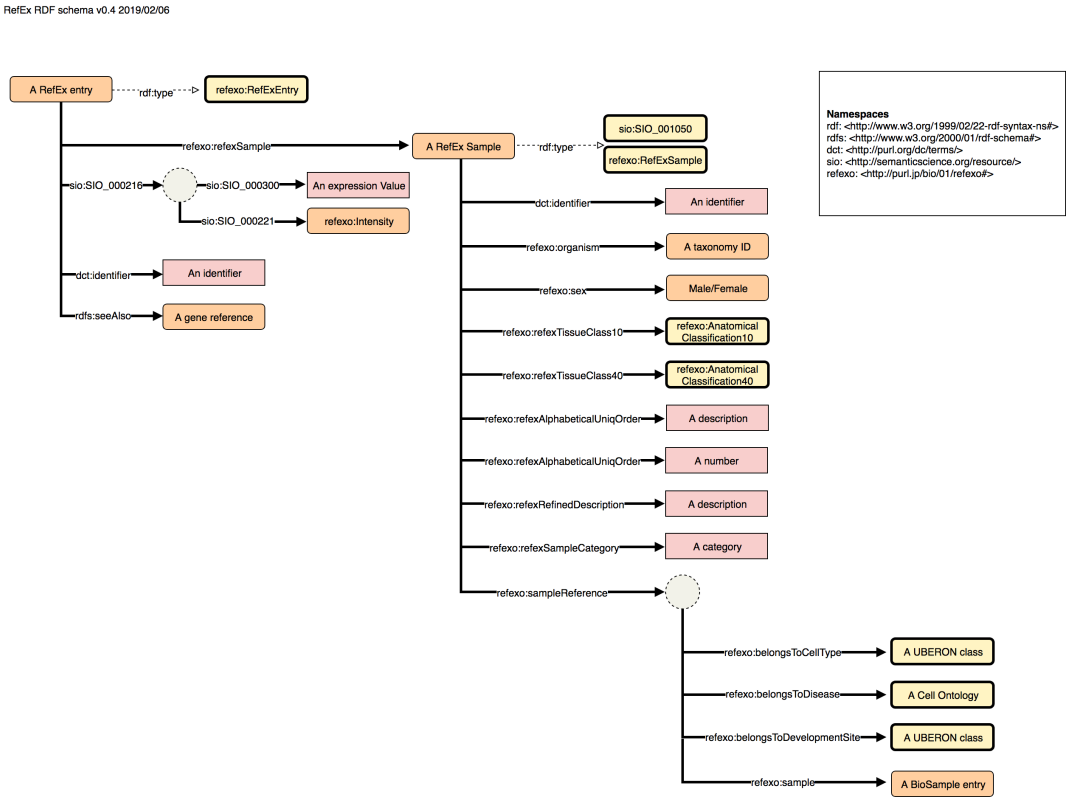

Figure 7. This example schema diagram is taken from the RefEx RDF. The orange, yellow, and pink rectangles represent instances, ontology classes, and literals, respectively, while the solid and dashed arrows represent properties and rdf:type relationships, and the dotted circles represent blank nodes.

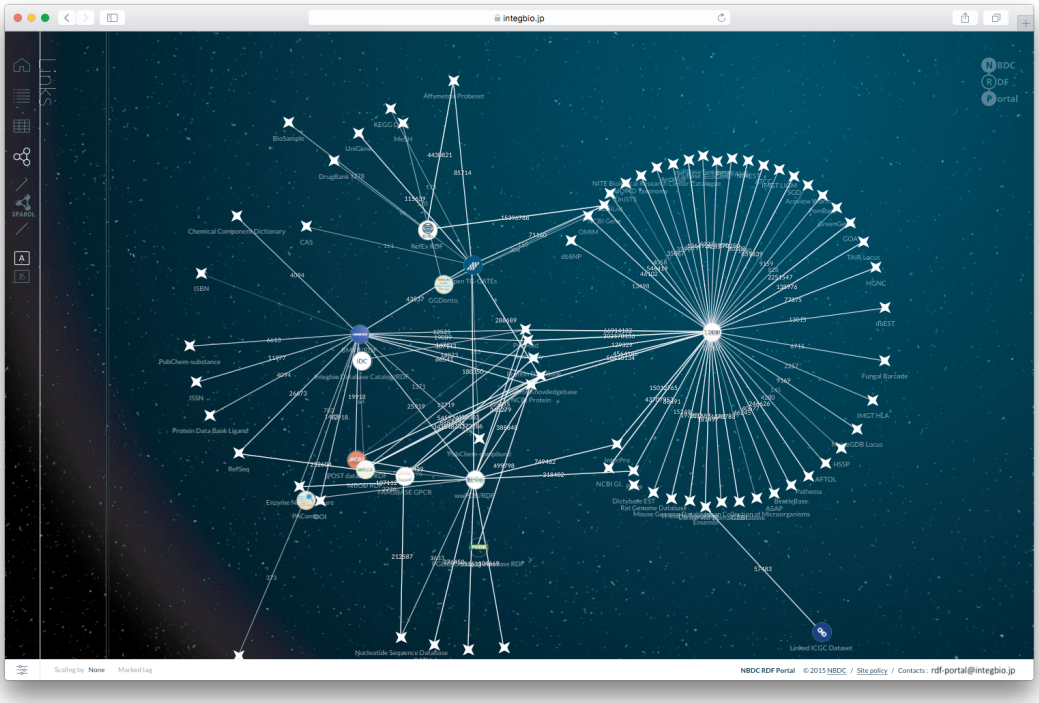

Figure 8. Network view of the NBDC RDF portal. This network view dynamically shows how the datasets are connected. The circles represent datasets registered with the RDF portal, while the stars represent external datasets. When two datasets are linked, they are connected by a solid line, and the number on the line represents the number of links.

An example of RDFizing a Biological Database

I have reviewed all RDF datasets included in the RDF portal and contributed to developing several of them, including the RDF datasets of RefEx and FAMSBASE. Here, I explain how to convert a biological database into an RDF dataset by taking RefEx RDF as an example.

RefEx is a web tool for browsing reference gene expression of human and mouse. It provides a faceted search allowing users to narrow down the results by specifying various filters such as gene names, body parts, gene ontologies, and protein families. Although several kinds of information from external databases such as Gene Ontology and InterPro are included in the RefEx dataset, these were taken from the original databases; when designing RDF it is not generally recommended to store redundant information that is available in external databases. This is because not only the file size increases but also when the information is updated in the original database, they will not be automatically synchronized. Therefore, I designed an RDF model of RefEx (Figure 7) by focusing on the unique information of the RefEx dataset: gene expression value and sample information. In addition, in order to represent the model, I created a small ontology, named RefExO. Figure 9 shows an example of the RDF of a gene expression value, and Figure 10 shows an example of the RDF of a sample in RefEx RDF. In the model, I defined a container resource for each gene expression value (Line 8 of Figure 9), and this resource was defined as an instance of an ontology class to comply with guideline 1 because it is regarded as a primary resource. In this case, this resource is defined as an instance of the refexo:RefExEntry class defined in RefExO (Line 9 of Figure 9). The the statement whose predicate is dcterms:identifier is necessary in order to comply with guideline 3 (Line 10 of Figure 9). The expression value calculated in Transcripts Per Million (TPM) is described as defined in guideline 11 (Line 12-15 of Figure 7). The statement linking to an NCBI gene usese the rdfs:seeAlso property for the predicate and an identifiers.org URI for the object, complying with guideline 4 (Line 16-17 of Figure 7). Each gene expression derived from a sample and the links to the sample is described with

the refexo:refexSample property (Line 11 of Figure 7). The resource of a sample is regarded as another primary resource and as such defined as an instance of both refexo:RefExSample and sio:SIO_001050 (sio:sample). This resource contains various meta-information of the sample, such as organism, sex, age, tissue, developmental stage, and some labels for display on the RefEx web site. As of December 2018, a part of the RefEx dataset which is derived from CAGE and GeneChip experiments were converted into RDF, resulting in 123,447,475 triples.

```
1  @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2  @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
3  @prefix dcterms: <http://purl.org/dc/terms/> .
4  @prefix refex: <http://refex.dbcls.jp/entry/> .
5  @prefix refexs: <http://refex.dbcls.jp/sample/> .
6  @prefix refexo: <http://purl.jp/bio/01/refexo#> .
7
8  refex:RFX0000002149
9    a refexo:RefExEntry;
10   dcterms:identifier "RFX0000002149";
11   refexo:refexSample refexs:RES00000481;
12   sio:SIO_000216 [
13     sio:SIO_000300 10.577177222478 ;
14     sio:SIO_000221 refexo:TPM
15   ] ;
16   rdfs:seeAlso <http://www.ncbi.nlm.nih.gov/gene/12>,
17                <http://identifiers.org/ncbigene/12> .
```

Figure 9. An example of RefEx RDF. This RDF shows gene expression values of human SERPINA3 (NCBI Gene ID: 12) from the sample RES00000481. The RDF of RES00000481 is shown in Figure 10

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix refexs: <http://refex.dbcls.jp/sample/> .
@prefix refexo: <http://purl.jp/bio/01/refexo#> .
@prefix ff: <http://fantom.gsc.riken.jp/5/sstar/FF:> .
@prefix bs: <http://identifiers.org/biosample/> .
@prefix obo: <http://purl.obolibrary.org/obo/> .

refexs:RES00000481
  a refexo:RefExSample, sio:SIO_001050 ;
  dcterms:identifier "RES00000481" ;
  refexo:sex obo:PATO_0001338 ;
  refexo:organism <http://identifiers.org/taxonomy/9606> ;
  refexo:originalDescription "TPM (tags per million) of liver, adult, pool1.CN
hs10624.10018-101C9" ;
  refexo:refexAlphabeticalUniqOrder "481" ;
  refexo:refexRefinedDescription "liver, adult" ;
  refexo:refexSampleCategory "02adult tissue" ;
  refexo:refexTissueClass10 refexo:v07_10 ;
  refexo:refexTissueClass40 refexo:v31_40 ;
  refexo:sampleReference [
    refexo:belongsToAnatomy obo:UBERON_0000061, obo:UBERON_0000062,
                            obo:UBERON_0000465, obo:UBERON_0000467,
                            obo:UBERON_0000468, obo:UBERON_0000475,
:
:
:
                            obo:UBERON_0005177, obo:UBERON_0006925,
                            obo:UBERON_0007023, obo:UBERON_0009569,
                            obo:UBERON_0010317 ;
   refexo:belongsToDevelopmentSite obo:UBERON_0001041, obo:UBERON_0002532,
                                   obo:UBERON_0003104, obo:UBERON_0004161,
                                   obo:UBERON_0006595, obo:UBERON_0009497,
                                   obo:UBERON_0010316 ;
        refexo:sample <http://fantom.gsc.riken.jp/5/sstar/FF:10018-101C9> ;
        rdfs:seeAlso bs:SAMD00005542
    ] .
```

Figure 10. An example of RefEx RDF. This RDF represents the sample referred to in Figure 9.

Querying multiple datasets

One consequence of the review process is that it enables us to efficiently query multiple datasets. For example, Figure 9 shows a SPARQL query that counts the number of PubMed document citations in each dataset; the results are shown in Table 4. Initially, I encountered cases where rdfs:seeAlso, dcterms:references, and other user-defined properties were used in literature citations. In addition, six different URIs were used to refer to the same PubMed resource (Table 3). Adding statements that used common vocabularies and specified URIs according to the guidelines, therefore, enabled us to increase the accuracy of queries across multiple datasets.

URIs of PubMed articles
http://identifiers.org/pubmed/
http://rdf.ncbi.nlm.nih.gov/pubmed/
http://identifiers.org/pubmed/
http://www.ncbi.nlm.nih.gov/pubmed/
http://rdf.ncbi.nlm.nih.gov/pubmed/
http://ncbi.nlm.nih.gov/pubmed/

Table 3. Six different URIs that refer to the same PubMed resource. In this way, the same resource may be referenced from different URIs, which is one of the reasons that interfere with RDF dataset interoperability.

```
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?graph COUNT(DISTINCT ?article) AS ?articles
WHERE {
  GRAPH ?graph {
    ?s dcterms:references ?article
    FILTER (REGEX(?article, "ncbi.nlm.nih.gov/pubmed"))
  }
} ORDER BY DESC(?articles)
```

Figure 11. SPARQL query that counts the references in each RDF graph. According to guideline 4, all datasets refer to the PubMed literature using the dcterms:references.

Next, Figure 12 shows an example SPARQL query against RefEx (Ono *et al*. 2017) and Open TG-GATEs (Igarashi *et al*. 2015), which store transcriptomic data. RefEx provides reference transcriptome datasets from 40 normal human, mouse, and rat tissues and cells, while Open TG-GATEs is a large-scale toxicogenomics database that includes transcriptome data for human samples exposed to various drugs. The query returns the expression values for probe 210049_at and the chemical compounds that human liver samples were exposed to from Open TG-GATEs, together with reference expression values for the same probe from RefEx; partial query results are shown in Table 5. Both databases include gene expression data measured using the same GeneChip technology, refer to organs in samples using the UBERON ontology (Mungall *et al*. 2012), and use a common RDF model to describe measured numerical data, enabling us to integrate them using a single SPARQL query. In addition to the two examples given here, I provide some examples of SPARQL queries that query multiple datasets in the documents section of the RDF portal.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX affy: <http://identifiers.org/affy.probeset/>
PREFIX tg-probe: <http://purl.jp/bio/101/opentggates/Probe/>
PREFIX tgo: <http://purl.jp/bio/101/opentggates/ontology/>
PREFIX pubchem: <http://identifiers.org/pubchem.compound/>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX refexo: <http://purl.jp/bio/01/refexo#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT DISTINCT ?refex_id ?refex_exp_value ?pubchem ?tggates_exp_value
WHERE {
  ?compound rdfs:seeAlso ?pubchem .
  ?condition tgo:exposedCompound ?compound .
  ?sample tgo:experimentalCondition ?condition .
  ?sample tgo:organ obo:UBERON_0002107 .
  ?sample tgo:chip ?chip .
  ?chip sio:SIO_000216 ?mv .
  ?mv sio:SIO_000300 ?tggates_exp_value .
  ?mv tgo:probe tg-probe:210049_at .
  FILTER(REGEX(?pubchem, "compound"))
  ?refex rdfs:seeAlso affy:210049_at .
  ?refex dcterms:identifier ?refex_id .
  ?refex sio:SIO_000216 ?refex_mv .
  ?refex_mv sio:SIO_000300 ?refex_exp_value .
  ?refex refexo:refexSample ?refex_sample .
  ?refex_sample refexo:refexTissueClass40 ?tissue .
  ?tissue rdfs:label "Liver/Hepato"@en .
  ?tissue skos:exactMatch obo:UBERON_0002107 .
  FILTER(REGEX(?pubchem, "identifiers.org"))
} ORDER BY DESC(?tggates_exp_value)
LIMIT 30
```

Figure 12. A SPARQL query that performs an integrated search of the RefEx and Open TG-GATEs RDFs. Both RefEx and Open TG-GATEs RDF include transcriptome data measured using the same GeneChip technology and use the RDF model defined in guideline 11 to describe measured numerical data.

Table 4. Results of the SPARQL query in Figure 9

RDF Dataset	Graph	Number of references
wwPDB	http://rdf.integbio.jp/dataset/pdbj	57546
BMRB	http://bmrbpub.protein.osaka-u.ac.jp/rdf/bmr	14679
MBGD	http://mbgd.genome.ad.jp/rdf/resource/organism	2690
GlycoEpitope	http://rdf.glycoinfo.org/glycoepitope	2354
IntegBio database catalog	http://rdf.integbio.jp/dataset/dbcatalog/main	1380
PACONTO	http://jcggdb.jp/rdf/diseases/paconto	214
SSBD	http://metadb.riken.jp/db/SSBD	46
GGDONTO	http://jcggdb.jp/rdf/diseases/ggdonto	15
INSDC ontology	http://integbio.jp/rdf/ontology/nucleotide	13
BMRB	http://bmrbpub.protein.osaka-u.ac.jp/rdf/bms	7
JPOST	http://jpost.org/graph/database	4

Table 5. Partial results of the SPARQL query in Figure 12.

From left to right, RefEx ID, expression value of the probe 210049_at in RefEx, URI of the compound exposed to the sample in Open TG-GATEs, expression value of the probe 210049_at in Open TG-GATEs.

ReFex ID	RefEx expression value	Exposed PubChem compound	Tggates expression value
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4449	319.3662702
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/31703	314.3898251
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/31703	310.6747304
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/31703	306.8218267
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4449	297.3405856
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/31703	264.2432302
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/186907	257.8708457
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/31703	253.6239994
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	238.6754244
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/5271566	234.1067549
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/186907	226.3806392
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/12699	223.208626
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/12699	217.2208698
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/10438	215.7555157
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/186907	210.6409975
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/8456	210.2461615
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/31703	210.0566659
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	209.0139089
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/5280965	208.8227747
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/186907	208.3912228
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/12699	207.4064151
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/7577	207.2949701
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	205.8646934
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/12699	205.6952544
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	205.4601065
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	205.3946991
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/12699	204.5959245
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	203.5228522
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/7577	203.3890369
RFX0016058250	12.3	http://identifiers.org/pubchem.compound/4725	203.3228314

Discussion

It is unrealistic to expect that independently-created RDF datasets will be highly interoperable. The EBI RDF platform succeeded in generating interoperable datasets by providing URI design guidelines and using common ontologies and RDF models as comprehensively as possible (Jupp *et al*. 2014). This was largely because they had the advantage that the groups developing the databases and the RDFs belonged to the same institute. Although we could not participate in developing each RDF, we were able to achieve reasonable interoperability by reviewing the RDFs when they were submitted.

With regard to the system's operational aspects, I faced the problem of being unable to include all the datasets in a single Virtuoso instance due to their enormous combined size. To deal with this, I have set up separate instances to host large datasets, such as DDBJ. However, this means I need to write federated SPARQL queries to query across instances, and these generally have performance issues, as well as not always returning answers to more complex queries. That said, I expect to improve the RDF store's performance in this area in the future.

Although I would like all datasets to comply with all the guidelines, I have been willing to accept non-compliance with some guidelines if there is sound reason. For example, wwPDB/RDF includes over 1000 classes and 5000 properties in its ontology, making it difficult to draw an appropriately-sized schema diagram, so it does not provide schema diagrams. Currently, the guidelines only require the use of certain limited property types. However, to further facilitate the semantic integration of life science data, I plan to ask developers to use more common properties and classes in the future. For example, I am asking developers to represent bio-sample resources as instances of sio:SIO_001050 (sio:sample). If I can introduce the use of common properties having biological meanings (herein,

called biological properties), it is expected that I can conduct more biologically meaningful queries against RDF datasets. The LinkDB database, which is a collection of links between databases entries, has only three link types: direct link, reverse link and equivalent link (Fujibuchi *et al*. 1998). Clearly, it would be useful for users if these can be extended to describe biological meanings, such as whether the relationship of proteins is binding or orthologous. However, so far there have been little effort to use common biological properties among RDF datasets. This may be caused by a tendency that many classes are provided by existing ontologies while properties are not. For example, BioPortal, the largest repository of biomedical ontologies, contains 686 OWL ontologies, whose statistics are available at the site, and include 7,926,030 classes. However, in contrast, only 42,064 properties are defined in these ontologies. This implies that the necessary properties may not be defined in any ontology. Recently, I have been working on a project, named med2rdf, aiming to develop RDFs of biomedical databases currently focusing on genomic variation (https://github.com/med2rdf). I have experimentally developed an ontology that includes properties to describe the relationships among primary subjects considered to be particularly important in the project such as genes, variations, diseases, and literature references. These biological properties will be used in the RDF datasets created by this project, which will enable users to retrieve the relationships between genes and its variations from multiple RDF datasets by a simple SPARQL query. In the future, I would like to introduce such biological properties to the RDF Portal to realize more biologically meaningful queries.

Conclusion

In this thesis, I presented a method for integrally using multiple RDF datasets using semantic web technology. First, as a concrete example of a classical flat-file format database, I described the AAindex database which I developed. AAindex is a collection of numerical indices and matrices representing various physicochemical and biochemical properties of amino acids and pairs of amino acids. Although AAindex has gained a certain evaluation as a useful resource itself, there is no mechanism to use it semantically combined with other databases. Existing biological databases generally have similar issues.

Next, from the viewpoint of integrating and utilizing multiple databases, I introduced the advantages of exposing biological databases as RDF to increase their interoperability. In RDF, a Uniform Resource Identifier (URI) is used to identify resources. Because URI is a globally unique identifier, I can refer to resources unambiguously by using URI. In addition, by using ontologies described in Web Ontology Language (OWL), it is possible to things consistent at the level of vocabulary among databases. However, in the Semantic Web, there is no rule on how to describe information as RDF. This is because of the following known problems: (1) In real databases, there are often several different URIs referring to the same resource on the Web, and there are no general rules as to which URI to use when linking to external resources. (2) There are often disparate ontology classes and properties representing same or similar concepts. (3) The same information can be modeled in different RDF schemas. These issues hinder integrated search across RDF datasets, which could potentially be possible. To address these, I have proposed a set of guidelines for developing RDF datasets with high interoperability. By complying with these guidelines when developing RDF, the RDF datasets become standardized even at the level of semantics.

With the cooperation of the NBDC, I have developed the NBDC RDF portal which is a repository service for RDF datasets. The portal provides a list of registered RDF datasets, a download service of RDF files and SPARQL endpoints for the RDF datasets. All datasets in this repository have been reviewed by the NBDC to ensure interoperability and queryability. In order to comprehensively carry out the reviewing processes, my colleagues and I also developed a verification tool for my guidelines. As a result, I have achieved higher interoperability among RDF datasets that were independently developed by different research groups.

As of November 2018, the NBDC RDF portal contains 21 RDF datasets of various research fields such as genes, protein 3D structures, epigenomes, cancer genomes, glycans, chemical compounds, and toxicogenomics. It has grown to become a considerable service, comprising over 45.5 billion triples. I hope that the portal will contribute to data science as a useful information infrastructure in the future.

Appendix

Appendix 1. The Qualified Name (QName) prefixes used in this thesis

Prefix	Vocabulary/Ontology name	URL
rdf	Resource Discription Framework	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	RDF Schema	http://www.w3.org/2000/01/rdf-schema#
owl	Web Ontology Language	http://www.w3.org/2002/07/owl#
dcterms	Dublin Core Metadata Initiative (DCMI) Metadata Terms	http://purl.org/dc/terms/
skos	Simple Knowledge Organization System	http://www.w3.org/2004/02/skos/core#
sio	Semanticscience Integrated Ontology	http://semanticscience.org/resource/
obo	The Open Biologicao and Biomedical Ontology	http://purl.obolibrary.org/obo/
bibo	The Bibliographic Ontology	http://purl.org/ontology/bibo/
cito	The Citation Typing Onotlogy	http://purl.org/spar/cito/
up	UniProt	http://purl.uniprot.org/core/
cco	ChEMBL Core Ontology	http://rdf.ebi.ac.uk/terms/chembl#
refex	RefEx	http://refex.dbcls.jp/entry/
refexo	RefEx ontology	http://purl.jp/bio/01/refexo#
ggdonto	GGDonto	http://jcggdb.jp/rdf/diseases/ggdonto#
jpost	jPOST	http://rdf.jpostdb.org/entry/
up	UniProt	http://purl.uniprot.org/uniprot/
pdb	wwPDB	https://rdf.wwpdb.org/pdb/

Appendix 2. The RDF datasets in the NBDC RDF portal

Here, the fact sheets of the RDF datasets in the NBDC RDF portal will be shown. Each sheet contains the dataset's name, description, tags, data provider, creators, version, date issued, license information, statistics, and linked datasets as of November, 2018.
In the linked datasets section, only the links complied with the RDF portal guidelines are counted. For DDBJ, the top 10 datasets are shown in descending order of the number of links from the dataset.

DDBJ					
Description	Semantic Representation of DDBJ Annotated Sequence Records.				
Tagas	Genome, Gene, cDNA, Tag sequence (nucleic acid), Polymorphism, Other DNA, RNA, Sequenec, Ontology/Terminology/Nomenclature, Others				
Data provider	National Institute of Genetics				
Creators	Takatomo Fujisawa (National Institute of Genetics) Toshiaki Katayama (Database Center for Life Science) Yasukazu Nakamura (National Institute of Genetics)				
Version	105.0				
Issued	2016-07-30				
License	Referred to in International Nucleotide Sequence Database Collaboration Policy (http://www.insdc.org/policy.html)				
Statistics					
Subject	2,913,415,115	Objects	3,907,105,857	Literals	950,890,006
Classes/Instances	109/1,878,437,947				
Properties/Triples	144/20,067,185,022				
Datatypes	4				
Linked datasets					
Taxonomy			202,178,136		
PubMed			66,914,132		
NCBI Protein			50,410,134		
NCBI GI			43,707,953		
InterPro			15,032,765		
UniProte Knowledgebase			4,564,185		
FlyBase			2,251,547		
GOA			1,264,928l		
SGD			659,839		
NCBI Gene			546,419		

DBKERO RDF					
Description	DBKERO is a collection of multi-omics data sets including SNV, RNA-seq, ChIP-seq, BS-seq and TSS-seq. The ChIP-seq part is big, so its lite version chip_seq_lite is included. The original big ChIP-seq data can also be downloaded at https://integbio.jp/rdf/download/kero/2017-01-27/all/chip_seq_all.tar.gz .				
Tags	Genome, Polymorphism, Other DNA, Gene expression, Others				
Data provider					
Creators	Shin Kawano (Database Center for Life Science) Hiroyuki Wakaguri (Graduate School of Frontier Sciences, The University of Tokyo) Yutaka Suzuki (Graduate School of Frontier Sciences, The University of Tokyo)				
Version	2017-01-27				
Issued	2017-01-27				
License	Creative Commons Attribution 4.0 International (CC BY 4.0)				
Statistics					
Subject	929,551,492	Objects	7,430,483,241	Literals	1,764,181,364
Classes/Instances	38 / 929,457,877				
Properties/Triples	60 / 11,017,998,412				
Datatypes	4				
Linked datasets					

Open TG-GATEs					
Description	Open TG-GATEs is a public toxicogenomics database.				
Tags	Gene, Drug/Chemical, Health/Disease, Gene expression				
Data provider	National Institutes of Biomedical Innovation, Health and Nutrition (NIBIOHN)				
Creators	Yoshinobu Igarashi (NIBIOHN) Shuichi Kawashima (Database Center for Life Science) Daisuke Satoh (Level Five Co., Ltd.) Chioko Nagao (NIBIOHN) Kenji Mizuguchi (NIBIOHN)				
Version	2017-03-17				
Issued	2017-03-17				
License	http://toxico.nibiohn.go.jp/english/agreement.html Toxicogenomics Project and Toxicogenomics Informatics Project				
Statistics					
Subject	1,497,955,718	Objects	2,765,961,664	Literals	1,267,395,887
Classes/Instances	65/1,497,955,718				
Properties/Triples	38/6,800,384,609				
Datatypes	5				
Linked datasets					
UniProt Knowledgebase			288,689		
UniGene			115,639		
wwwPDB/RDF			112,830		
Affymetrix Probeset			85,714		
NCBI Gene			71,160		
BMRB/RDF			26,223		
KEGG Drug			173		
PubChem-compoud			163		
CAS			161		
DrugBank			96		

wwPDB/RDF					
Description	wwPDB/RDF is a translation of PDBx/PDBML data into RDF				
Tags	Protein, Drug/Chemical, Other biomolecule, Sequence, Structure				
Data provider	Institute for Protein Research, Osaka University				
Creators	Akira Kinjo (Institute for Protein Research, Osaka University)				
Version	release_20180926				
Issued	2018-09-26				
License					
Statistics					
Subject	321,728,326	Objects	330,067,639	Literals	6,444,577
Classes/Instances	444/321,416,520				
Properties/Triples	2,688/4,725,251,340				
Datatypes	3				
Linked datasets					
InterPro	839,508				
Gene Ontology	595,071				
Taxonomy	580,464				
UniProt Knowledgebase	416,128				
CATH domain	388,479				
PubMed	248,142				
Pfam	238,923				
Enzyme Nomenclature	111,519				
SCOP	104,525				
Nucleotide Sequence Database	3827				

MBGD RDF					
Description	RDF dataset of Microbial Genome Database for Comparative Analysis (MBGD)				
Tags	Genome, Gene, Phylogeny/Classification				
Data provider	National Institute for Basic Biology				
Creators	Hirokazu Chiba (National Institute for Basic Biology) Hiroyo Nishide (National Institute for Basic Biology) Ikuo Uchiyama (National Institute for Basic Biology)				
Version	2015-01				
Issued	2015-06-20				
License	Creative Commons Attribution-ShareAlike 2.1 Japan (CC BY-SA 2.1 JP) MBGD RDF © MBGD development team, National Institute for Basic Biology licensed under Creative Commons Attribution-ShareAlike 2.1 Japan				
Statistics					
Subject	309,702,751	Objects	472,067,973	Literals	74,289,149
Classes/Instances	32/273,443,876				
Properties/Triples	79/1,609,018,143				
Datatypes	5				
Linked datasets					
NCBI Protein	34,354,877				
UniProt Knowledgebase	8,887,626				
Taxonomy	17719				
wwPDB/RDF	76				

Linked ICGC Dataset					
Description	Linked ICGC Dataset is a linked data version of the public ICGC (International Cancer Genome Consortium) data. This includes the information of the donors and the somatic mutations				
Tags	Genome, Health/Disease, Sequence				
Data provider	Research Center for Advanced Science and Technology, The University of Tokyo				
Creators	Ryota Yamanaka (Research Center for Advanced Science and Technology, The University of Tokyo)				
Version	release_20				
Issued	2016-01-04				
License	Creative Commons CC0 1.0 Universal (CC0 1.0) Public Domain Dedication				
Statistics					
Subject	51,410,016	Objects	30,906,909	Literals	20,735,685
Classes/Instances	9/51,410,015				
Properties/Triples	66/577,082,774				
Datatypes	1				
Linked datasets					
Ensembl	57,483				

BMRB/RDF					
Description	BMRB/RDF is a translation of NMR-STAR data into RDF				
Tags	Protein, Other biomolecule, Others, SequenceStructure				
Data provider	Institute for Protein Research, Osaka University				
Creators	Masashi Yokochi, Naohiro Kobayashi, Akira Kinjo, Takeshi Iwata, Haruki Nakamura, Chojiro Kojima, Toshimichi Fujiwara (Institute for Protein Research, Osaka University) Eldon L. Ulrich, John L. Markley (University of Wisconsin-Madison) Yannis E. Ioannidis (University of Athens) Miron Livny (University of Wisconsin-Madison)				
Version	2018-09-26				
Issued	2018-09-26				
License	Creative Commons Attribution 2.1 Japan (CC BY 2.1 JP) BMRB/RDF licensed under CC Attribution 2.1 Japan.				
Statistics					
Subject	30,773,713	Objects	32,554,901	Literals	1,710,303
Classes/Instances	403/30,534,687				
Properties/Triples	2,733/552,975,082				
Datatypes	4				
Linked datasets					
wwPDB/RDF	21,154,326				
Protein Data Bank	310,530				
NCBI Protein	66,518				
PubMed	35,032				
RefSeq	26,663				
Taxonomy	19,510				
DOI	14,833				
UniProt Knowledgebase	13,132				
ISSN	12,147				
PubChem-substance	6,613				

NBDC Nikkaji RDF					
Description	NBDC NikkajiRDF is RDF data of Japan Chemical Substance Dictionary (Nikkaji), which is one of the largest chemical substance databases in Japan.				
Tags	Drug/Chemical, Others, Structure, Image/Movie				
Data provider	Japan Science and Technology Agency (JST)				
Creators	Japan Science and Technology Agency (JST)				
Version	2017-01-12				
Issued	2017-01-12				
License	Creative Commons Attribution 2.1 Japan (CC BY 2.1 JP) NBDC NikkajiRDF © Japan Science and Technology Agency licensed under CC Attribution 2.1 Japan				
Statistics					
Subject	60,432,596	Objects	168,104,425	Literals	63,322,504
Classes/Instances	38,937/23,738,365				
Properties/Triples	41/333,968,051				
Datatypes	3				
Linked datasets					

jPOST database RDF					
Description	jPOST database RDF is described the re-analyzed proteome datast in the jPOST project.				
Tags	Protein, Sequence				
Data provider					
Creators	Yuki Moriya (Database Center for Life Science) Shin Kawano (Database Center for Life Science) Susumu Goto (Database Center for Life Science)				
Version	201807				
Issued	2018-07-31				
License	CC BY 4.0　j POST licensed under CC Attribution 4.0				
Statistics					
Subject	58,996,232	Objects	69,906,379	Literals	10,754,871
Classes/Instances	80 / 58,996,221				
Properties/Triples	83 / 209,474,019				
Datatypes	5				
Linked datasets					
UniProt Knowledgebase	596,005				
wwPDB/RDF	558,156				
BMRB/RDF	127,764				
PubMed	6				

RefEx RDF					
Description	RDFized reference gene expresson dataset derived from CAGE and GeneChip experiments in the RefEx database.				
Tags	Gene, Tag sequence (nucleic acid), Gene expression				
Data provider					
Creators	Shuichi Kawahsima (Database Center for Life Science) Hiromasa Ono (Database Center for Life Science)				
Version	2017-04-07				
Issued	2017-04-07				
License	Attribution 4.0 International (CC BY 4.0) Database Center for Life Science				
Statistics					
Subject	24,260,736	Objects	27,438,991	Literals	22,820,176
Classes/Instances	10 / 19,828,635				
Properties/Triples	47 / 123,447,475				
Datatypes	9				
Linked datasets					
NCBI Gene	15,396,788				
Affymetrix Probeset	4,430,821				
BioSample	1278				

Quanto					
Description	Quanto is a dataset of sequencing quality of public high-throughput sequencing data based on FastQC.				
Tags	Others				
Data provider	Database Center for Life Science				
Creators	Tazro Ohta (Database Center for Life Science)				
Version	0.1.2				
Issued	2016-07-12				
License	Creative Commons Attribution 4.0 International (CC BY 4.0) Quanto RDF dataset licensed under CC Attribution 4.0 International (CC BY 4.0)				
Statistics					
Subject	21,955,729	Objects	31,484,031	Literals	10,369,656
Classes/Instances	9 / 21,955,729				
Properties/Triples	37 / 107,782,639				
Datatypes	4				
Linked datasets					
Sequence Read Archive	1,995,973				

FAMSBASE GPCR					
Description	Predicted protein structures of GPCR				
Tags	Protein, Structure				
Data provider	Chuo University				
Creators	Mituo Iwadate Chuo University Shuichi Kawashima Database Center for Life Science				
Version	2016-03-24				
Issued	2016-03-24				
License	Creative Commons Attribution 4.0 International (CC BY 4.0) FAMSBASE GPCR RDF dataset licensed under CC Attribution 4.0 International (CC BY 4.0)				
Statistics					
Subject	5,858,909	Objects	6,378,250	Literals	488,759
Classes/Instances	16 / 5,858,908				
Properties/Triples	30 / 21,297,786				
Datatypes	3				
Linked datasets					
Protein Data Bank	490,303				
UniProt Knowledgebase	372,286				
RefSeq	252,604				
Nucleotide Sequence Database	212,587				

PGDBj Ortholog database RDF					
Description					
Tags	Genome, Gene, Sequence, Phylogeny/Classification				
Data provider	Kazusa DNA Research Institute				
Creators	Hisako Ichihara (Kazusa DNA Research Institute) Akihiro Nakaya (Osaka University) Hirokazu Chiba (National Institute for Basic Biology) Satoshi Tabata (Kazusa DNA Research Institute)				
Version	1.57.0				
Issued	2016-07-26				
License	Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0) PGBDj © Kazusa DNA Research Institute licensed under Creative Commons Attribution-ShareAlike 4.0 International				
Statistics					
Subject	1,963,741	Objects	5,728,073	Literals	1,858,372
Classes/Instances	11 / 1,963,733				
Properties/Triples	35 / 13,652,175				
Datatypes	2				
Linked datasets					
NCBI Protein	499,798				

Dataset of WURCS-RDF					
Description	Dataset of glycan structures described by WURCS				
Tags	Other biomolecule, Structure				
Data provider	The Noguchi Institute				
Creators	Issaku YAMADA (The Noguchi Institute) Masaaki MATSUBARA (The Noguchi Institute)				
Version	0.2				
Issued	2015-09-30				
License	Creative Commons Attribution 2.1 Japan (CC BY 2.1 JP) WURCS-RDF © GLIC licensed under CC Attribution 2.1 Japan				
Statistics					
Subject	1,365,653	Objects	1,138,140	Literals	40,435
Classes/Instances	14 / 817,535				
Properties/Triples	56 / 6,213,789				
Datatypes	4				
Linked datasets					

GlyTouCan					
Description	GlyTouCan is the international glycan structure repository.				
Tags	Other biomolecule, Structure				
Data provider	National Institute of Advanced Industrial Science and Technology (AIST)				
Creators	Hisashi Narimatsu (Glycoscience and Glycotechnology Research Group, Biotechnology Research Institute for Drug Discovery, National Institute of Advanced Industrial Science and Technology (AIST)) Kiyoko F. Aoki-Kinoshita (Soka University) Daisuke Shinmachi (Soka University)				
Version	Ver1.0				
Issued	2015-09-04				
License	Creative Commons Attribution 2.1 Japan (CC BY 2.1 JP) GlyTouCan licensed under CC Attribution 2.1 Japan				
Statistics					
Subject	375,657	Objects	502,463	Literals	126,879
Classes/Instances	20 / 375,657				
Properties/Triples	30 / 1,749,648				
Datatypes	5				
Linked datasets					

Integbio Database Catalog/RDF					
Description	Integbio Database Catalog/RDF is a translation of Integbio Database Catalog data into RDF.				
Tags	Others				
Data provider	National Bioscience Database Center (NBDC)				
Creators	Tomoe Nobusada (NBDC) Asuka Bando (NBDC)				
Version	release_20180919				
Issued	2018-10-16				
License	http://creativecommons.org/publicdomain/zero/1.0/ Integbio Database Catalog© National Bioscience Database Center licensed under CC Attribution 2.1 Japan				
Statistics					
Subject	11,332	Objects	29,210	Literals	15,033
Classes/Instances	9 / 8,320				
Properties/Triples	41 / 96,765				
Datatypes	3				
Linked datasets					
PubMed		1,506			

PAConto					
Description	PAConto is the RDF representation of PACDB (Pathogen Adherence to Carbohydrate Database) data and Ontology of Infectious Diseases known to be related to Glycan Binding. PACDB was developed by the Research Center for Medical Glycoscience (RCMG, AIST) and released in March 2010. At the present time PACDB provides information on about 370 strains of 120 microorganisms, and about 1,700 lectin-glycan interactions of two types: binding and not binding. Also, the PACDB provides information on about 100 infectious diseases in which the interaction between adherence molecules of pathogens and glycan ligands of the host cells plays an important role in the disease pathogenesis. All of the information for the creation of this database was obtained from scientific articles.				
Tags	Other biomolecule, Health/Disease, Interaction/Pathway, Structure				
Data provider	Glycoscience and Glycotechnology Research Group, Biotechnology Research Institute for Drug Discovery, National Institute of Advanced Industrial Science and Technology (AIST)				
Creators	Hisashi Narimatsu, Toshihide Shikanai, Elena Solovieva, Noriaki Fujita (Glycoscience and Glycotechnology Research Group, Biotechnology Research Institute for Drug Discovery, National Institute of Advanced Industrial Science and Technology (AIST))				
Version	v.1.0				
Issued	2016-06-01				
License	Creative Commons Attribution-NonCommercial-ShareAlike 2.1 Japan (CC BY-NC-SA 2.1 JP) PAConto © Glycoscience and Glycotechnology Research Group (AIST) licensed under CC Attribution-NonCommercial-ShareAlike 2.1 Japan				
Statistics					
Subject	9,329	Objects	16,396	Literals	8,586
Classes/Instances	63 / 9,296				
Properties/Triples	117 / 81,785				
Datatypes	3				
Linked datasets					
wwPDB/RDF			2,424		
BMRB/RDF			1,565		
MeSH			373		

SSBD: Meta-information of quantitative data and microscopy images					
Description	Meta-information of quantitative data and datasets of microscopy images provided from SSBD database				
Tags	Cell, Organism, Other biomolecule, Image/Movie, Gene expression, Others				
Data provider					
Creators	Yukako Tohsato (Osaka Electro-Communication University, Department of Engineering Informatics) Koji Kyoda (RIKEN Quantitative Biology Center, Laboratory for Developmental Dynamics) Kenneth H. L. Ho (RIKEN Quantitative Biology Center, Laboratory for Developmental Dynamics) Shuichi Onami (RIKEN Quantitative Biology Center, Laboratory for Developmental Dynamics)				
Version	SSBD31_20171218 release 20180324				
Issued	2018-03-24				
License	Creative Commons Attribution-ShareAlike 2.1 Japan (CC BY-SA 2.1 JP) Dataset © Shuichi Onami （RIKEN) licensed under CC Attribution-Share Alike 2.1 Japan				
Statistics					
Subject	6,644	Objects	9,329	Literals	3,807
Classes/Instances	18 / 6,644				
Properties/Triples	33 / 40,300				
Datatypes	4				
Linked datasets					

GGDonto					
Description	GGDonto is the Ontology of the Genetic Diseases related to the Glycan Metabolism. GGDonto describes the knowledge about Congenital Disorders of Glycosylation (CDG) and Lysosomal Storage Diseases (LSD). GGDonto provides the information on 120 genetic diseases of the glycan synthesis and the degradation and their causative genes.				
Tags	Other biomolecule, Health/Disease, GeneOntology/Terminology/Nomenclature, Interaction/Pathway				
Data provider	Glycoscience and Glycotechnology Research Group, Biotechnology Research Institute for Drug Discovery, National Institute of Advanced Industrial Science and Technology (AIST)				
Creators	Hisashi Narimatsu, Toshihide Shikanai, Elena Solovieva, Noriaki Fujita (Glycoscience and Glycotechnology Research Group, Biotechnology Research Institute for Drug Discovery, National Institute of Advanced Industrial Science and Technology (AIST))				
Version	v.1.0				
Issued	2017-01-25				
License	Creative Commons Attribution-NonCommercial-ShareAlike 2.1 Japan (CC BY-NC-SA 2.1 JP) GGDonto© Glycoscience and Glycotechnology Research Group (AIST) licensed under CC Attribution-NonCommercial-ShareAlike 2.1 Japan				
Statistics					
Subject	1,782	Objects	8,978	Literals	4,963
Classes/Instances	23 / 1,705				
Properties/Triples	943 / 39,439				
Datatypes	2				
Linked datasets					
MeSH	570				
OMIM	304				
NCBI Gene	150				

GlycoEpitope					
Description	GlycoEpitope is a database of useful information on carbohydrate antigens and antibodies.				
Tags	Other biomolecule, Structure				
Data provider	Ritsumeikan University				
Creators	Tshisuke Kawasaki (Ritsumeikan University) Shujiro Okuda (Niigata University)				
Version	version 3				
Issued	2015-11-18				
License	Creative Commons Attribution-ShareAlike 2.1 Japan (CC BY-SA 2.1 JP) GlycoEpitope licensed under CC Attribution-Share Alike 2.1 Japan				
Statistics					
Subject	8,678	Objects	9,769	Literals	5,453
Classes/Instances	24 / 5,726				
Properties/Triples	35 / 27,796				
Datatypes	2				
Linked datasets					

Metadata of JCM resources					
Description	A RDF-based meta-database of microbial strains used in various researches such as biology, environment and human health as bioresources. Microbial strains are available from Japan Collection of Microorganisms (JCM) in RIKEN BioResource Center. Please visit data browser at http://metadb.riken.jp/metadb/db/rikenbrc_jcm_microbe				
Tags	Organism, Phylogeny/Classification, Bioresource, Data provider				
Data provider	RIKEN				
Creators	Terue Takatsuki (Technology and development unit for knowledge base of mouse phenotype, RIKEN BioResource Center) Moriya Ohkuma (Micribe Division, RIKEN BioResource Center) Hiroshi Masuya (Technology and development unit for knowledge base of mouse phenotype, RIKEN BioResource Center)				
Version	beta				
Issued	2015-09-09				
License	Creative Commons Attribution-ShareAlike 2.1 Japan (CC BY-SA 2.1 JP) Metadata of JCM resources © RIKEN BRC licensed under CC Attribution-ShareAlike 2.1 Japan				
Statistics					
Subject	1,854	Objects	4,104	Literals	2,574
Classes/Instances	6 / 1,789				
Properties/Triples	25 / 8,896				
Datatypes	5				
Linked datasets					

Appendix 3. User manual for the SPARQL-proxy

SPARQL-proxy is a portable Web application that works as a proxy server for any SPARQL endpoint providing the following functionalities:

1. validation of the safety of query statements (omit SPARQL Update queries)
2. job scheduling for a large number of simultaneous SPARQL queries
3. providing a job management interface for time consuming SPARQL queries
4. (optional) cache mechanisms with compression for SPARQL results to improve response time
5. (optional) logging SPARQL queries and results
6. (experimental) splitting a SPARQL query into chunks by adding OFFSET & LIMIT

Docker

```
   $ docker run -p 8080:3000 -e SPARQL_BACKEND=http://example.com/sparql
dbcls/sparql-proxy
```

Prerequisites

Node.js (https://nodejs.org/)

Install

```
$ git clone git@github.com:dbcls/sparql-proxy.git
$ cd sparql-proxy
$ npm install
```

(Be patient, npm install may take a few minutes)

Run

```
PORT=3000 SPARQL_BACKEND=http://example.com/sparql ADMIN_USER=admin
ADMIN_PASSWORD=password npm start
```

Open http://localhost:3000/ on your browser.
Dashboard for administrators is at http://localhost:3000/admin .

Configuration

All configurations are set with the following environment variables.

PORT

(default: 3000)
Port to listen on.

SPARQL_BACKEND (required)

URL of the SPARQL backend.

ADMIN_USER

(default: admin)
User name for the sparql-proxy administrator.

ADMIN_PASSWORD

(default: password)
Password for the sparql-proxy administrator.

CACHE_STORE

(default: null)
Cache store. Specify one of the following:

- null: disable caching mechanism.
- file: cache in local files.
- memory: cache in the proxy process.
- redis: use redis.
- memcache: use memcached.

COMPRESSOR

(default: raw)
Cache compression algorithm. Specify one of the following:
raw: disable compression.
snappy: use snappy.

CACHE_STORE_PATH

(only applicable to CACHE_STORE=file case) (default: /tmp/sparql-proxy/cache)
Root directory of the cache store.

MEMORY_MAX_ENTRIES

(only applicable to CACHE_STORE=memory case)
Maximum number of the entries to keep in the cache.

REDIS_URL

(only applicable to CACHE_STORE=redis case)
(default: localhost:6379)
Specify URL to the redis server.

MEMCACHE_SERVERS

(only applicable to CACHE_STORE=memcache case)
(default: localhost:11211)
Specify server locations to the memcache servers (comma-separated).

JOB_TIMEOUT

(default: 300000)
Job timeout in millisecond.

DURATION_TO_KEEP_OLD_JOBS

(default: 300000)

Duration in millisecond to keep old jobs in the administrator dashboard.

MAX_CONCURRENCY

(default: 1)

Number of concurrent requests.

MAX_WAITING

(default: Infinity)
Number of jobs possible to be waiting.

TRUST_PROXY

(default: false)
Set true to trust proxies in front of the server.

MAX_LIMIT

(default: 10000)
Cap the LIMIT of queries.

ENABLE_QUERY_SPLITTING

THIS IS AN EXPERIMENTAL FEATURE.

(default: false)

Set true to enable query splitting. If enabled, content negotiation will be disabled; spaql-proxy will always use application/sparql-results+json. That is because merging results other than JSON is not supported.

MAX_CHUNK_LIMIT

(only applicable to ENABLE_QUERY_SPLITTING=true case)
(default: 1000)
Split queries into the chunk size specified.

QUERY_LOG_PATH

(default: null)
Log queries (and the corresponding responses) to the file, if specified.

Appendix 4. User manual of Monban

Monban: An RDF Lint Tool

Prerequisites

- Node.js(https://nodejs.org/) >= 8.10.0
- Yarn (https://yarnpkg.com) >= 1.5.1

Setup

```
$ git clone https://github.com/dbcls/monban
$ cd monban
$ yarn install
```

Usage of Monban

monban lints the file specified.

```
$ ./bin/monban [target file (.nt, .ttl)]
```

Options

--primal-classes <path.txt>

Path to primal classes definition. List classes one per line.

Example:

```
http://example.com/primaryClass1
http://example.com/primaryClass2
```

--uri-whitelist <path.tsv>

Path to white list definition for `rdfs:seeAlso` test. The file should be a Tab Separated Values (TSV) file.

1st column: label of the pattern
2nd column: RegExp of the pattern

Example:

```
Example1  ^http://example¥.com/1/
Example2  ^http://example¥.com/2/
```

--uri-blacklist <path.tsv>

Path to black list definition for `rdfs:seeAlso` test. The file should be a Tab Separated Values (TSV) file.

- 1st column: label of the pattern
- 2nd column: RegExp of the pattern

Example1 ^http://example¥.com/1/
Example2 ^http://example¥.com/2/

--ontology <path.ttl>

Path to ontology (in Turtle or N-Triples). This option can be specified multiple times.

Example1 ^http://example¥.com/1/
Example2 ^http://example¥.com/2/

--bib-patterns <path.tsv>

Path to bibliography resource patterns.
Example (this is the default):

PMC ^http://identifiers¥.org/pmc/
PubMed ^http://identifiers¥.org/pubmed/
DOI ^http://doi¥.org/

--report-limit <number>

Number of error instances to report per error. If a negative value specified, no limit.
Default: 10

--output-format <format>

Output format. json and markdown are available.
Default: markdown

Appendix 5. User manual of Aramashi

Aramashi

aramashi computes the statistics of an RDF file.

```
$ ./bin/aramashi [target file (.nt, .ttl)]
```

Option

--link-patterns <path.tsv>

Path to the link pattern definition. The file should be a Tab Separated Values (TSV) file.

- 1st column: label of the pattern
- 2nd column: RegExp of the pattern

Example:

DDBJ ^http://identifiers¥.org/insdc/
KERO ^http://kero¥.hgc¥.jp/rdf/

Aramashi-merge

aramashi-merge merges the outputs of aramashi.

```
$ ./bin/aramashi-merge [target file (.json)]
```

This can be used for a large graph consisting of many files; 1) use aramashi to compute the file-wise statistics, then 2) use aramashi-merge to merge the results. Example:

```
$ ./bin/aramashi file1.ttl > file1.json
$ ./bin/aramashi file2.ttl > file2.json
$ ./bin/aramashi-merge file1.json file2.json > merged.json
```

References

Afonnikov DA, Kolchanov NA. 2004. CRASP: A program for analysis of coordinated substitutions in multiple alignments of protein sequences. *Nucleic Acids Res* **32**: 64–68.

Antezana E, Kuiper M, Mironov V. 2009. Biological knowledge management: The emerging role of the Semantic Web technologies. *Brief Bioinform* **10**: 392–407.

Belleau F, Nolin M-A, Tourigny N, Rigault P, Morissette J. 2008. Bio2RDF: towards a mashup to build bioinformatics knowledge systems. *J Biomed Inform* **41**: 706–16.

Berman HM. 2008. The Protein Data Bank: A historical perspective. *Acta Crystallogr Sect A Found Crystallogr* **64**: 88–95.

Berners-Lee T, Hendler J, Lassila O. 2001. The Semantic Web. *Scientific American*.

Bolleman J, Mungall CJ, Strozzi F, Barran J, Dumontier M, Bonnal RJP, Buels R, Hoendorf R, Fujisawa T, Katayama T, et al. 2016. FALDO: A semantic standard for describing the location of nucleotide and protein feature annotation. *J Biomed Semantics* **7**: 39.

Chen H, Yu T, Chen JY. 2013. Semantic web meets integrative biology: A survey. *Brief Bioinform* **14**: 109–125.

Dodds L, Davis I. 2012. Linked Data Patterns A pattern catalogue for modelling , publishing , and consuming Linked Data and consuming Linked Data. http://patterns.dataincubator.org/book/.

Dumontier M, Baker CJO, Baran J, Callahan A, Chepelev L, Cruz-Toledo J, Del Rio NR, Duck G, Furlong LI, Keath N, et al. 2014. The semanticscience integrated ontology (SIO) for biomedical research and knowledge discovery. *J Biomed Semantics* **5**: 1–11.

Fu G, Batchelor C, Dumontier M, Hastings J, Willighagen E, Bolton E. 2015. PubChemRDF: Towards the semantic annotation of PubChem compound and substance databases. *J Cheminform* **7**: 1–15.

Fujibuchi W, Goto S, Migimatsu H, Uchiyama I, Ogiwara A, Akiyama Y, Kanehisa M. 1998. DBGET/LinkDB: an integrated database retrieval system. *Pac Symp Biocomput* 683–694.

Gauthier J, Vincent AT, Charette SJ, Derome N. 2018. A brief history of bioinformatics. *Brief Bioinform* 1–16.

Goto N, Prins P, Nakao M, Bonnal R, Aerts J, Katayama T. 2010. BioRuby: Bioinformatics software for the Ruby programming language. *Bioinformatics* **26**: 2617–2619.

Haupt C, Waagmeester A, Zimmermann M, Willighagen E. 2013. Guidelines for exposing data as RDF in Open PHACTS. http://www.openphacts.org/specs/2013/WD-rdfguide-20131007/.

Igarashi Y, Nakatsu N, Yamashita T, Ono A, Ohno Y, Urushidani T, Yamada H. 2015. Open TG-GATEs: A large-scale toxicogenomics database. *Nucleic Acids Res* **43**: D921–D927.

Imker HJ. 2018. 25 Years of Molecular Biology Databases: A Study of Proliferation, Impact, and Maintenance. *Front Res Metrics Anal* **3**: 1–20.

Jupp S, Malone J, Bolleman J, Brandizi M, Davies M, Garcia L, Gaulton A, Gehant S, Laibe C, Redaschi N, et al. 2014. The EBI RDF platform: Linked open data for the life sciences. *Bioinformatics* **30**: 1338–1339.

Juty N, Le Novère N, Laibe C. 2012. Identifiers.org and MIRIAM Registry: community resources to provide persistent identification. *Nucleic Acids Res* **40**: D580-6.

Katayama T, Wilkinson MD, Aoki-Kinoshita KF, Kawashima S, Yamamoto Y, Yamaguchi A, Okamoto S, Kawano S, Kim J-D, Wang Y, et al. 2014. BioHackathon series in 2011 and 2012: penetration of ontology and linked data in life science domains. *J Biomed Semantics* **5**: 1–13.

Katayama T, Wilkinson MD, Micklem G, Kawashima S, Yamaguchi A, Nakao M, Yamamoto Y, Okamoto S, Oouchida K, Chun H-W, et al. 2013. The 3rd DBCLS BioHackathon: improving life science data integration with semantic Web technologies. *J Biomed Semantics* **4**: 6.

Katayama T, Wilkinson MD, Vos R, Kawashima T, Kawashima S, Nakao M, Yamamoto Y, Chun H-W, Yamaguchi A, Kawano S, et al. 2011. The 2nd DBCLS BioHackathon: interoperable bioinformatics Web services for integrated applications. *J Biomed Semantics* **2**: 4.

Kawashima S, Kanehisa M. 2000. AAindex: Amino Acid index database. *Nucleic Acids Res* **28**: 374.

Kawashima S, Katayama T, Yamamoto Y. DBCLS RDFizing DB guidelines. https://github.com/dbcls/rdfizing-db-guidelines.

Kawashima S, Ogata H, Kanehisa M. 1999. AAindex: Amino acid index database. *Nucleic Acids Res* **27**: 368–369.

Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T, Kanehisa M. 2008. AAindex: Amino acid index database, progress report 2008. *Nucleic Acids Res* **36**: 202–205.

Kidera A, Konishi Y, Oka M, Ooi T, Scheraga HA. 1985. Statistical analysis of the physical properties of the 20 naturally occurring amino acids. *J Protein Chem* **4**: 23–55.

Kinjo AR, Bekker G-J, Suzuki H, Tsuchiya Y, Kawabata T, Ikegawa Y, Nakamura H. 2017. Protein Data Bank Japan (PDBj): updated user interfaces, resource description framework, analysis tools for large structures. *Nucleic Acids Res* **45**: D282–D288.

Li GXH, Vogel C, Choi H. 2018. PTMscape: an open source tool to predict generic post-translational modifications and map modification crosstalk in protein domains and biological processes. *Mol Omi* **14**: 197–209.

Liubc B, Lib S, Wangc Y, Lub L, Lib Y, Cai Y. 2007. Predicting the protein SUMO modification sites based on Properties Sequential Forward Selection (PSFS). *Biochem Biophys Res Commun* **358**: 136–139.

Mager LN. 2002. *A History of the Life Sciences, Revised and Expanded*. 3rd ed. CRC Press.

Malone J, Stevens R, Jupp S, Hancocks T, Parkinson H, Brooksbank C. 2016. Ten Simple Rules for Selecting a Bio-ontology. *PLoS Comput Biol* **12**: 1–6.

Marshall MS, Boyce R, Deus HF, Zhao J, Willighagen EL, Samwald M, Pichler E, Hajagos J, Prud'hommeaux E, Stephens S. 2012. Emerging practices for mapping and linking life sciences data using RDF—A case series. *Web Semant Sci Serv Agents World Wide Web*.

Mashima J, Kodama Y, Fujisawa T, Katayama T, Okuda Y, Kaminuma E, Ogasawara O, Okubo K, Nakamura Y, Takagi T. 2017. DNA Data Bank of Japan. *Nucleic Acids Res* **45**: D25–D31.

Miyazawa S, Jernigan RL. 1999. Self-consistent estimation of inter-residue protein contact energies based on an equilibrium mixture approximation of residues. *Proteins Struct Funct Genet* **34**: 49–68.

Mungall CJ, Torniai C, Gkoutos G V., Lewis SE, Haendel MA. 2012. Uberon, an integrative multi-species anatomy ontology. *Genome Biol* **13**: 1–20.

Nakai K, Kidera A, Kanehisa M. 1988. Cluster analysis of amino acid indices for prediction of protein structure and function. *Protein Eng* **2**: 93–100.

NIH. 2018. the first National Institutes of Health (NIH) Strategic Plan for Data Science. https://datascience.nih.gov/sites/default/files/NIH_Strategic_Plan_for_Data_Science_Final_508.pdf.

Ono H, Ogasawara O, Okubo K, Bono H. 2017. RefEx, a reference gene expression dataset as a web tool for the functional analysis of genes. *Sci Data* **4**.

Pokarowski P, Kloczkowski A, Jernigan RL, Kothari NS, Pokarowska M, Kolinski A. 2005. Inferring ideal amino acid interaction forms from statistical protein contact potentials. *Proteins Struct Funct Genet* **59**: 49–57.

Pokarowski P, Kloczkowski A, Nowakowski S, Pokarowska M, Jernigan RL, Kolinski A. 2007. Ideal amino acid exchange forms for approximating substitution matrices. *Proteins Struct Funct Genet* **69**: 379–393.

Rice P, Longden I, Bleasby A. 2000. EMBOSS: the European molecular biology open software suite. *Trends Genet* **16**: 276–77.

Rigden DJ, Fernández XM. 2018. The 2018 Nucleic Acids Research database issue and the online molecular biology database collection. *Nucleic Acids Res* **46**: D1–D7.

Sarda D, Chua GH, Li K Bin, Krishnan A. 2005. pSLIP: SVM based protein subcellular localization prediction using multiple physicochemical properties. *BMC Bioinformatics* **6**: 1–12.

Slater T, Bouton C, Huang ES. 2008. Beyond data integration. *Drug Discov Today* **13**: 584–589.

Splendiani A, Burger A, Paschke A, Romano P, Marshall M. 2011. Biomedical semantics in the Semantic Web. *J Biomed Semantics* **2**: S1.

Stein LD. 2003. Integrating Biological Databases. *Nat Rev Genet* **4**: 337–345.

Suzuki A, Kawano S, Mitsuyama T, Suyama M, Kanai Y, Shirahige K, Sasaki H, Tokunaga K, Tsuchihara K, Sugano S, et al. 2018. DBTSS/DBKERO for

integrated analysis of transcriptional regulation. *Nucleic Acids Res* **46**: D229–D238.

The DBCLS BioHackathon Consortium. 2010. The DBCLS BioHackathon: standardization and interoperability for bioinformatics web services and workflows. *J Biomed Semantics* 1–19.

The Uniprot Consortium. 2015. UniProt : a hub for protein information. **43**: 204–212.

Tomii K, Kanehisa M. 1996. Analysis of amino acid indices and mutation matrices for sequence comparison and structure prediction of proteins. *Protein Eng* **9**: 27–36.

Tung CW, Ho SY. 2007. POPI: Predicting immunogenicity of MHC class I binding peptides by mining informative physicochemical properties. *Bioinformatics* **23**: 942–949.

Wilkinson MD et al. 2016. Comment: The FAIR Guiding Principles for scienti fi c data management and stewardship. *Sci Data* **3**: 160018.

Willighagen EL, Waagmeester A, Spjuth O, Ansell P, Williams AJ, Tkachenko V, Hastings J, Chen B, Wild DJ. 2013. The ChEMBL database as linked open data. *J Cheminform* **5**: 1–12.