OPEN ACCESS Freely available online

PLoS one

Standard Biological Parts Knowledgebase

Michal Galdzicki[1], Cesar Rodriguez[2], Deepak Chandran[3], Herbert M. Sauro[3], John H. Gennari[1]*

1 Biomedical & Health Informatics, University of Washington, Seattle, Washington, United States of America, **2** BIOFAB, University of California, Berkeley, California, United States of America, **3** Bioengineering, University of Washington, Seattle, Washington, United States of America

Abstract

We have created the Knowledgebase of Standard Biological Parts (SBPkb) as a publically accessible Semantic Web resource for synthetic biology (sbolstandard.org). The SBPkb allows researchers to query and retrieve standard biological parts for research and use in synthetic biology. Its initial version includes all of the information about parts stored in the Registry of Standard Biological Parts (partsregistry.org). SBPkb transforms this information so that it is computable, using our semantic framework for synthetic biology parts. This framework, known as SBOL-semantic, was built as part of the Synthetic Biology Open Language (SBOL), a project of the Synthetic Biology Data Exchange Group. SBOL-semantic represents commonly used synthetic biology entities, and its purpose is to improve the distribution and exchange of descriptions of biological parts. In this paper, we describe the data, our methods for transformation to SBPkb, and finally, we demonstrate the value of our knowledgebase with a set of sample queries. We use RDF technology and SPARQL queries to retrieve candidate "promoter" parts that are known to be both negatively and positively regulated. This method provides new web based data access to perform searches for parts that are not currently possible.

Citation: Galdzicki M, Rodriguez C, Chandran D, Sauro HM, Gennari JH (2011) Standard Biological Parts Knowledgebase. PLoS ONE 6(2): e17005. doi:10.1371/journal.pone.0017005

Editor: Christian Schönbach, Kyushu Institute of Technology, Japan

Received August 31, 2010; **Accepted** January 19, 2011; **Published** February 24, 2011

Funding: This work was partially supported by grants from the National Library of Medicine (R41 LM010745, T15 LM007442), and the National Institute of Biomedical Imaging and Bioengineering (BE08407). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: gennari@uw.edu

Introduction

The engineering of new biological systems has begun to demonstrate the advantages of leveraging living cells as machines for the production of medicine [1], nutrients [2], biofuels [3,4], and as biosensors [5,6]. Driving the growth of these new technologies are advances in the approaches and tools used to control cellular processes [7,8] and to construct synthetic DNA [9,10]. Synthetic Biology offers the promise to address some of the world's most challenging problems [11].

To facilitate the process of development, synthetic biologists apply principles of engineering (i.e. standardization, abstraction, and decoupling) to specify the design, assembly, and validation of new biological systems [12]. In other engineering fields, such as mechanical, electrical, and computer engineering, these principles have led to the highly successful methods used today to build robust and complex products. The multiple scales, diversity, and dynamics inherent to biological systems and materials necessitate the use of computational methods to help manage this complexity. Synthetic biologists need software tools that support the engineering process of biological systems [13]. Several such software tools are currently in development and aim to aid the design of new systems by predicting their behavior, TinkerCell [14], BioNetCAD [15], SynBioSS [16,17], and BioJADE [18] planning the assembly process [19], and validating the design GenoCAD [20,21,22,23]. Such design tools require computational access to a library of parts, specifically the ability to query such a library.

The ability of synthetic biologists to manipulate the composition of DNA sequence should allow researchers to engineer cells with desired behavior. In particular, the modification of the basic elements of genetic regulatory networks, or "gene circuits" [24] is representative of a class of elementary behaviors [25] and can be thought of as modular [26]. Therefore, the abstraction of these segments of DNA as *biological parts* [27] for the purpose of engineering has been broadly adopted. The success of this approach is especially visible in the context of the International Genetically Engineered Machines (iGEM) competition (igem.org) [28], as evidenced by the growing number of biological parts in the Registry of Standard Biological Parts (partsregistry.org/cgi/partsdb/Statistics.cgi)[29]. This collection of parts, created by undergraduate students and independent synthetic biology laboratories is a ready source of components for engineering new biological systems.

Our research goal is to build a computationally accessible library of information about standard biological parts for synthetic biologists. We will design this library to support part re-use by leveraging the engineering principles of standardization, decoupling, and abstraction. If synthetic biologists had easy access to information about previously used parts, they could use this information to more efficiently design and plan for the assembly of new genetic devices. When already available components exist, and have been shown to work, their reuse would allow a synthetic biologist to focus on meeting design requirements, rather than re-creating prior work of others.

In this manuscript we present the Standard Biological Parts knowledge base (SBPkb), our initial version of a biological parts library that supports remote queries. This library builds on knowledge from the Registry of Standard Biological Parts (partsregistry.org), which we describe below. We adapted and transformed data from the registry into SBOL-semantic, that describes standard biological parts using RDF. Next, we demonstrate how the SBPkb can be queried using standard RDF technology (SPARQL queries) to retrieve parts

that may be relevant to a synthetic biologist. We take as our use case queries about promoter parts. In our results section, we show (a) that such queries cannot be pragmatically answered with current technologies, and (b) that our approach allows researchers to carry out query refinement. For the latter, we show that our promoter query can iteratively be made more specific, so that the query results in smaller lists of parts, and where these parts are more well-matched to specific design criteria.

Catalog of Parts: The Registry of Standard Biological Parts

The Registry of Standard Biological Parts (partsregistry.org) is a repository of biological parts for synthetic biology. The Registry is hosted at MIT and provides services to store and distribute plasmid DNA that conforms to certain specifications and descriptive information, i.e., a physical store and distribution point for biological parts. The Registry website is a publicly available source of information about those parts. The website is partially designed as a wiki, and therefore Registry users can edit its content directly. Registry staff also curate this information. From the point of view of a user it is organized from two main perspectives: one about individual part records and the second as a catalog or listings of various parts. The Registry also provides help and documentation sections, as well as user management features, such as groups and a user authentication system. Web pages describing individual part records provide detailed descriptive information about the DNA sequence, its design, and the availability of the part as physical DNA stored at the Registry. The second perspective is the Catalog which can be browsed to explore the contents and discover new parts. This section of the site is subdivided into categories ranging from listings of parts by their expected function (e.g. constitutive promoters) to listings of parts used in specific projects. For example, each iGEM team has a page of all parts created and used throughout the duration of the competition. While the Registry faces challenges maintaining integrity between the information and the DNA repository [29], it is a unique and rich resource for the synthetic biology community.

There are more than 13,444 part records within the Registry. This is the largest collection of publically available parts for synthetic biologists. In addition, like other fields within modern molecular biology, synthetic biology faces additional and rapid growth of this data. Efforts to standardize the characterization [30,31] and composition [27,32] of parts are gaining momentum in the synthetic biology community. There is now a need to standardize the electronic form of the knowledge about these parts. In addition to the Registry of Standard Biological Parts there are new notable software efforts addressing the need to manage information about biological parts. The Joint BioEnergy Institute Registry (JBEIr) provides a web based inventory platform as well as a graphical sequence annotator [33]. Clotho, a software framework for synthetic biology, offers a suite of tools for the design and management of new biological systems [34]. Furthermore, there are also efforts to store quantitative models that describe and predict functions of synthetic biology systems such as SynBIOSS [16,17] and the Repository of Standard Virtual Parts [35,36]. These systems, just like the design tools we mentioned earlier, would benefit greatly from computational access to the information contained in the Registry.

Transformation of Parts Data to SBOL-Semantic

To describe common concepts used in synthetic biology, we implemented SBOL-semantic, an information model for synthetic biology, using the Web Ontology Language (OWL). The Synthetic Biology Open Language (SBOL) (sbolstandard.org) is a collaborative effort of the Synthetic Biology Data Exchange Group to develop standards and technologies to facilitate information exchange for synthetic biologists. SBOL-semantic is based on the rough consensus of core synthetic biology concepts and their relationships and represents the semantics of synthetic biology theory and practice. We used an open process for the evolution and standardization of data models according to a framework for how data models in synthetic biology should be published [37]. This new work builds on the Provisional BioBrick Language (PoBoL) [38].

We have built SBOL-semantic using OWL so as to be compliant with Semantic Web information technology standards that allow SBOL data records to be read, manipulated, and interpreted using generic tools such as Protégé [39], RDFlib [40] and Sesame [41]. These tools were used for management of SBOL model structure, to create a scheme for unique identification of elements, and to reference the Sequence Ontology [42], a third party ontology. The choice of W3C recommended technology was made on the premise that modeling knowledge in a computable, standardized, and community supported format will provide long term benefit for the synthetic biology community. (See also our discussion and future work sections.)

The SBOL semantic structure is organized as a hierarchy of *classes* that refer to distinct categories of common information objects, such as Parts, Cells, Plasmids, and Sequence Features. The most general of the *classes* (Figure 1) constitute the core SBOL concepts. Instances of a *class* are *individual* data elements. Figure 2 shows the specific part known as BBa_B0015, a commonly used transcriptional terminator [29]. In this figure, the part has *annotations* that divide the part into segments such as BBa_B0010 that are themselves instances of the *Part* class. In our model, all such annotations are *properties* that capture relationship information between individuals. Data represented in this form can be conceptualized as a graph in which nodes are *individuals*, members of SBOL classes, and edges are the *properties* between them. Here we present results focused on *Parts* and the description of their nucleotide sequence, *Sequence Features*. The long term goal of SBOL is to represent information relevant to all levels of the engineering process in synthetic biology (Tissues, Cells, Plasmids, etc). Here, we demonstrate the open nature of the framework [37] by extending this class structure to support the needed concepts from the Registry.

To create the semantic knowledgebase for synthetic biology we used the information available from the Registry of Standard Biological Parts (partsregistry.org) to create an extension of the SBOL *class* structure. This extension uses SBOL-semantic in combination with the new terminology acquired from the Registry to describe biological parts. First, we extracted the Registry data and mapped its structure of tables, its relational schema, to SBOL-semantic. This mapping served as our translation table to transforming the Registry data of 13,444 part entries and the associated Sequence Features to OWL/RDF. Using a script, we converted 13,444 Registry part records with their associated Sequence Features from the Registry format to the SBOL semantic (OWL/RDF) form. Each Registry part record was also associated with the Registry's Sequence Feature table, a position based description of the nucleotide sequence (see Figure 2 for example sequence features such as a 'terminator'). We then mapped the Registry Sequence Feature table to the SBOL Sequence Annotation and Feature Class structures and performed the analogous translation into OWL/RDF.

As part of the transformation of Registry data we used the categories attribute of the Registry *parts* table to provide a richer

Sample	Aliquot of *Cells* or *DNA* material in a container
Cell	Basic functional unit of life
Physical DNA	Continuous DNA molecule
Plasmid	Extra chromosomal DNA capable of replicating independently from chromosomal DNA
Assembly Standard	Set of *Sequence Features* which designate a physical composition standard
Part	A standardized building block for synthetic biology
Vector Backbone	A special *Part* into which the construct of interest is inserted to be transfected into *Cells*
Sequence Annotation	Position and direction describing the region for a *Sequence Feature* of a *Part*
Sequence Feature	Description of primary *Annotations* of nucleic acid sequence
BioBrick Scar	Sequence between adjacent *Parts*, created as a byproduct of Assembly Standard 10
Terminator	Transcriptional terminator sequence, example of a type of Sequence Ontology term

Figure 1. Top level *Class* (bold) and example *sub-class* (regular face) SBOL semantic terminology with a simplified definition for clarity.
doi:10.1371/journal.pone.0017005.g001

description of parts. The Registry includes a total of 346 categories organized as a hierarchy of 28 top level categories (e.g. chassis, classic, dna, function, plasmid, plasmidbackbone, primer, promoter, proteindomain, proteintag, rbs, regulation, ribosome, rnap, terminator, etc. For full listing see Supporting Information Table S1, which contains the list of terms extracted from the Registry data, and File S1., which contains the generated OWL encoded semi-structured controlled vocabulary used throughout this work). These categories are a rich vocabulary used to describe parts and constitute a controlled vocabulary, created and maintained by the Registry staff, while its use is enforced by the Registry website software application. The categories form the basis of organization for the Registry Catalog website. Thus, to provide a good structure for querying the Registry information, we needed to augment our core SBOL-semantic ontology with this terminology. To do so, we auto-generated a class structure within SBOL-semantic that mimics the registry category structure. For an example, see Figure 3. Finally, we loaded the SBOL-semantic data into a framework for querying RDF data, creating the Standard Biological Parts knowledgebase resource (SBPkb) (see Implementation and Availability for details). As we show in our results section, we can use these categories to directly query the SBPkb for specific features of parts.

The semi-structured controlled vocabulary resulting from this process does not fulfill many of the criteria of formal ontology design [43]. The structure created reflects the organization found in the Registry, and is not a proper class hierarchy. Our effort, directed towards SPARQL query information retrieval, translates the existing Registry information to a Semantic Web technology standard to enhance its potential for re-use. This utilitarian approach provides immediate benefit of data access and lays out the scope of the knowledge engineering challenges which face the synthetic biology community. Challenges of formally structuring information for future use in multiple applications are especially evident in large collections such as the user-driven and community-supported data source for our work, the Registry of Standard Biological Parts. However, the main contribution of this work is to provide a pragmatic solution for synthetic biology users, and establish the need for improvement of information resources in the field.

Results

The Case of the Promoter

To illustrate the functionality of SBPkb we describe a hypothetical case for its use to research the availability of promoters for a new design. We asked the knowledge base to answer the following question, "Which promoters can I use for a design?" Because "promoter" is a class in our controlled vocabulary, this is a straightforward SPARQL query to ask of

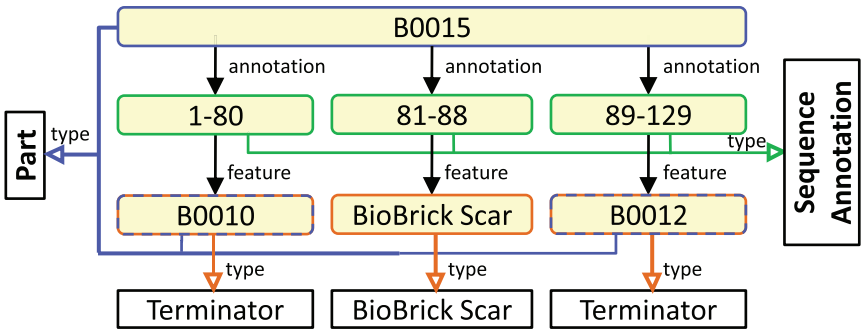

Figure 2. *Classes* (black rectangles) describe types (open faced arrows, colored by type) of *individual* data elements (yellow rounded rectangles) and the composition relationships between them (closed faced arrows).
doi:10.1371/journal.pone.0017005.g002

Registry Categories
//chassis/prokaryote/ecoli //promoter //regulation/positive

SBOL Category Classes
Class (chassis) → has_subclass → Class (ecoli_ prokaryote_ chassis)
Class (promoter)
Class (regulation) → has_subclass → Class (positive_regulation)

Figure 3. Example of Registry Categories to SBOL class structure conversion. These autogenerated classes are assigned to the partsregistry.org namespace to attribute them to the source and allow differentiation from SBOL-semantic classes, see the OWL implementation of SBOL-semantic File S1.
doi:10.1371/journal.pone.0017005.g003

our SBPkb (see query #1 in File S2), and it returns 538 parts that are annotated as promoters.

Although this query seems simple, we must compare the capabilities of SBPkb to current technology: How would one answer this question, with current technology, i.e., directly of the Parts Registry? Unfortunately, the only way to retrieve this set of parts is by manual browsing of web pages, and then manual compilation and analysis of the results listed on these web pages (also see the comparison section below). Additionally, SBPkb and SPARQL allow researchers to easily refine queries to both provide cleaner, more useful results, and to narrow the search to a more specific type of promoter. In this section, we describe how our initial query can be step-wise narrowed to a much more specific query that returns only six parts from our knowledgebase.

As a first step, we ask what information is associated with these parts—we carry out a SPARQL *describe* query (query #2 in File S2) that lists the complete set of properties associated with all promoters. This query would have a very long, large result, but we can sample only a few entries to explore the information space; Table 1 shows one sample entry from this query result. By looking at all available properties of a part, researchers may discover ways to narrow or improve their query. For example, an initial exploration may lead us to decide that the *status* property is important (we do not want any "deleted" parts), and that we only want parts that have DNA sequences listed. This refined query (query #3 in File S2 produces 529 parts (it eliminated seven "deleted" entries, and two without DNA sequences).

Trivially, we can also ask these sorts of "data cleaning" questions of the entire SPBkb. For example, we found that 12,152 of the 13,444 total part records have an associated DNA sequence and have not been marked for deletion (query #4 in File S2). Currently, many parts are larger in DNA sequence length than is financially prudent to directly synthesize, however not impossible using the latest methods [9]. Therefore, it is noteworthy that only 5,166 are marked as *Available* or as *Sent* to the Registry as clones (query #5 in File S2).

Comparison with current capabilities

To validate our (cleaner) result of 529 promoter parts found via our SPARQL query and the SBPkb, we also attempted to answer this question by exhaustively browsing the Parts Registry. First, we dismissed an information retrieval approach that might use heuristic algorithms based on text searches of the word "promoter" within the Registry's web pages (e.g., a Google search). Although careful construction of good heuristics might lead to accurate results, a simple text search will result in many entries that mention "promoter" but are not themselves promoter parts.

Thus, we used an exhaustive manual method, systematically exploring all web pages in the 'Promoter' category of the Parts Registry Catalog. When information appears about parts, the Registry Catalog typically displays the information in a table. Therefore, whenever we encountered a page with parts labeled as a category promoters, we copied the corresponding table into a spreadsheet application (MS Excel). This exploration results in 42 separate web pages (many with several tables) and a total of 833 promoter parts. (This data was collected by MG on Aug 3, 2010 from partsregistry.org/Promoters/Catalog). Because the same part can be found on multiple web pages, the same part identifier can be copied onto the spreadsheet multiple times. We removed these duplicate entries using the Remove Duplicates Data Tool in Excel^{TM} and obtained a unique list of 474 promoter entries. Finally, we noted that two of these lacked DNA sequence information, a requirement of our "cleaner" query.

The set of 472 entries that we found manually are all included in the set of 529 promoters returned by SBPkb. That is, there is no information "missed" by our knowledgebase. SBPkb also retrieved 57 additional entries that appear to be bona fide promoters, from a variety of subcategories. We attempted, but were unable to discover why these particular promoters were missing from our manual browsing of the web pages (see Table S2. for this list of 57 promoters).

It should be clear that exhaustive web page browsing is not a scalable approach to searching for a particular class of biological part. Indeed, the registry instead is a community-based, wiki-style collection of parts dedicated to capturing information about parts. Supporting such queries is a novel design consideration for a semantic web of data in synthetic biology. Query answering is a central design feature of the SPBkb, and as we demonstrate next, our initial query can be narrowed to return a much smaller set of parts, yet still maintain the ability to exhaustively search the knowledge base.

Design query refinement

The process of query refinement, or improvement of the query, as a specification of information needs, involves exploration in order to discover information about a topic [44]. We again look through the results of (query #2 in File S2) to find additional criteria by which to search SBPkb. The query driven exploration process helped us discover the rich source of structured information derived from the Registry categories. Among the results of this query (Table 1), we found that the example promoter part belongs to the type or category, 'sigma70_ecoli_prokaryote_ rnap'. The categories, represented as OWL *classes* in SBOL semantic, provide the capability to refine queries for promoters. For example, to narrow the selection to only those promoters which are expected to work with the *Escherichia coli* RNAP σ^{70} holoenzyme ($E\sigma^{70}$) and therefore to have an expected peak efficiency at the exponential growth phase [45]. This query (query #6 in File S2) results in 367 "$E\sigma^{70}$" promoters, a subset of the 529 promoters found in our initial query. This list of 367 are the most likely candidates to use for common synthetic biology experiments in *E.coli* for which measurements are taken at mid-exponential phase. The capability of retrieving specific parts from the thousands of entries within SBPkb by selection criteria such as the *class* structure of biological system contexts will allow synthetic biologists to find parts relevant to their design.

Not only were we able to retrieve promoter parts based on specific factors (σ), but available to us as selection criteria were also Registry categories which specify the expected mode of regulation. For example, during the design of a new genetic Barkai-Leibler oscillator [46,47] the synthetic biologist may want to find all pre-existing promoters that can be both 'positively regulated' AND 'negatively regulated', i.e., dual-regulated promoters (query #7 in

Table 1. Example result of a DESCRIBE SPARQL query for a selected single promoter part.

Subject	Predicate	Object
sbol:rQprqhqP5413	sbol:name	BBa_I746365
	rdf:type	sbol:ecoli_prokaryote_chassis
	rdf:type	sbol:sigma70_ecoli_prokaryote_rnap
	rdf:type	sbol:Part
	rdf:type	sbol:forward_direction
	rdf:type	sbol:promoter
	rdf:type	sbol:positive_regulation
	sbol:type	Regulatory
	sbol:shortDescription	PLL promoter from P4 phage
	sbol:longDescription	This is the PLL promoter taken from the P4 phage genome. It is an inducible promoter that is activated by a class of activators, including P2 ogr (I746350), PSP3 pag (I746351) and phiR73 delta (I746352). These different activators should cause different levels of activity of the PLL promoter.
	sbol:author	Stefan Milde
	sbol:status	Available
	sbol:id	9598
	sbol:owner_id	2122
	sbol:date	9/11/2007
	sbol:dnaSequence	cgctttattttgtgaatattttcagcagacgcaacaggggggatttgttcaggctgtcttacaatggctgtgtgtttttttgttcatctccac

doi:10.1371/journal.pone.0017005.t001

File S2). Our query returned just 36 unique promoter parts meeting these criteria (note that this query result is not necessarily a subset of the 367 "$E\sigma^{70}$" promoters). The Barkai-Leibler oscillator relies heavily on the availability of such dual-regulated promoters, therefore having knowledge of all dual-regulated promoters available in the Registry is highly advantageous to the synthetic biologist. Since a sufficient number of dual-regulated promoters are available, the search can be further limited to promoters for known specific inducers and repressors that are appropriate for the new design. The SBPkb includes information from the Registry Features table, therefore, for our final refinement, we further restricted our query to return promoters that have sequence annotations of known transcription factor binding sites, i.e., operator sites. This example query (Figure 4) returns just six parts and their known binding sites (Table 2). A selection of these six candidates provides a list small enough that each one can be examined in greater detail for relevance to a specific design.

During planning stages of a new synthetic biology research project investigation of prior work is an important phase of forming a new design. This process involves the exploration of available information resources for the purpose of discovery of candidate components to leverage in such a design. The SPARQL *describe* query in SBPkb can help identify information types or classes, such as Registry categories and data fields that hold information management, engineering, or biologically relevant information. These facts, or descriptions of parts, can then be used to search across the entire information collection to identify parts relevant to a particular design specification or criteria. This ability to quickly identify specific parts that match design criteria provides a method that enables fast and thorough exploration of prior work.

```
PREFIX sbol:<http://sbols.org/sbol.owl#>
PREFIX pr:<http://partsregistry.org/#>

SELECT DISTINCT ?name ?sdesc ?author ?fname
WHERE {?part a sbol:Part;
             a pr:promoter;
             sbol:name ?name;
             sbol:shortDescription ?sdesc;
             sbol:author ?author;
             a pr:positive_regulation;
             a pr:negative_regulation;
             sbol:annotation ?anot .
                             ?anot sbol:feature ?feat.
                                   ?feat sbol:name ?fname;
                                   a pr:binding.}
ORDER BY ?name
```

Figure 4. SPARQL query of SBPkb for dual-regulated promoter parts and their descriptions.
doi:10.1371/journal.pone.0017005.g004

Table 2. SBPkb promoter parts that can be both positively and negatively regulated with operator site sequence features.

Name	Short description	Author	Feature	Feature	Feature	Feature
BBa_I12036	Modified lamdba Prm promoter	Hans	OR1 lambda	OR2 lambda	OR1 434	OR2 434
BBa_I12006	Modified lamdba Prm promoter	mcnamara	OR1 lambda	OR2 lambda	OR1 434	
BBa_I12040	Modified lambda P(RM) promoter	ryhsiao	OR1 lambda	OR2 lambda	OR1 434	OR2 434
BBa_I14015	P(Las) TetO	Vijayan, V., Hsu, A., Fomundam, L.	TetR			
BBa_I14016	P(Las) CIO	Vijayan, V., Hsu, A., Fomundam, L.	CI lambda O1			
BBa_I1051	Lux cassette right promoter	Mahajan, V.S., Marinescu, V.D., Chow, B., Wissner-Gross, A.D., Carr, P.	cI (OR1)	LuxR/HSL		

doi:10.1371/journal.pone.0017005.t002

Implementation and Availability

To construct SBOL semantic we used Protégé 4.0.133 (protege.stanford.edu) and used a RDFlib (rdflib.net), a python library to perform programmatic additions of class terms and individuals during the data import process. We obtained the Standard Biological Parts Registry data from (partsregistry.org/Registry_API) on April 6, 2010. The downloaded information was provided in the form of two MySQL tables formatted as XML, a table of parts and a table of Sequence Features. These were converted into a text based delimited format to serve as input for SBPkb. We created python import scripts to parse the input tables from the Registry and libSBOL, a python library, to aid population of SBOL structures to generate the RDF/XML form of the data for SBPkb (synbiolib.sourceforge.net).

We have made the SBPkb data accessible via SPARQL a W3C recommended query language for RDF queries, with remote access (through a RESTful HTTP interface) provided using the Sesame 2.3.1 (openrdf.org) software. The SBPkb (sbpkb.sbolstandard.org) as a SPARQL accessible knowledge base is a publically available Semantic Web computational resource for the synthetic biology community.

Discussion

To effectively build new systems from prior work and best practices, synthetic biologists developed an initial framework and standards for the description of engineered biological devices [30,31]. The common approach of storing data about biological parts in a spreadsheet is convenient for a small laboratory. Our experience in synthetic biology research suggests that sharing such information between collaborating laboratories requires a significant coordination effort. Furthermore, *ad hoc* organization of part description information is too ambiguous to establish an efficient engineering pipeline for synthetic biology. The process of engineering synthetic biological systems relies on specialized software tools to: model systems, aid design, and plan assembly. For software to help researchers make appropriate design decisions, biological parts must be described using an unambiguous language, such as SBOL-semantic. To reconcile the need for engineering with base pair precision with the inherent complexity of biological system dynamics at multiple scales, there is a need for software tools to have the ability to exchange information about the entire spectrum of the domain of synthetic biology. Working towards the goal of defining an unambiguous computational language for synthetic biology, we have created Standard Biological Parts Knowledgebase (SBPkb). This public resource uses the Synthetic Biology Open Language semantics (SBOL-semantic) as its organizing structure and demonstrates its use for information retrieval.

Current methods for finding previously described biological parts are insufficient to realize new synthetic biology designs with increased sophistication. To create such integrated systems from parts and modules synthetic biologists must overcome significant challenges posed by the uncertainty and complexity of biology [48]. Synthetic biologists need to be able to draw on large numbers of examples of prior work to learn from the successes and failures of previous efforts. We have populated the SBPkb with the thirteen thousand parts from the Registry of Standard Biological Parts, and we have made it available for public use. Purnick & Weiss [48] reported that the most complex system built up to that time, as measured by the number of regulatory regions within a design, was six. Automatically searching the SBPkb, for existing candidate parts, will increase the number of part options to consider in designs. This ability, to quickly query part information from the large repository of knowledge provided by the Registry, removes one significant barrier in the exploration of prior work.

The ability to query SBPkb using a remote query protocol can serve to extend the capabilities of computational tools which support design work. Software designed to help synthetic biologists to plan designs can greatly benefit from a computationally accessible search interface. Information retrieved from SBPkb by SPARQL is returned as SBOL-semantic RDF/XML therefore can easily interpreted by the receiving application. For example, TinkerCell [14,49], a computer aided design application, could use SBPkb queries to fulfill designs based on combinations of specific requirements. We demonstrated one such hypothetical query for promoter parts controlled by dual modes of regulation. TinkerCell, and other design tools, could take advantage of query results to suggest these candidate parts to a user who is building a new Barkai-Leibler oscillator. The use of query refinement as a method for specifying design requirements would be an important methodological development towards automating the design to production pathway in synthetic biology.

SBOL-semantic is based on the robust principles and technology developed by the Semantic Web research program. The utility of the approach we described provides information retrieval services via a standard query language, SPARQL. However, we look forward to building on the foundation established by the SBOL-semantic framework to support additional capabilities, specifically to take advantage of reasoning services for ontologies formalized in OWL. Semantic Web inference engines, such as Pellet [50], Hermit [51], and Fact++ [52] perform consistency checking and classification/realization. These tools validate and generate new inferences about a set of axioms based on logical constraints and restrictions defined in OWL. Therefore, to develop significant improvements to SBOL-semantic, the terms from the controlled vocabulary provided by the Registry will have to conform with ontology design best

practices [43] and be defined using OWL-DL class restrictions. Therefore, to impart these capabilities we plan to formalize SBOL-semantic class definitions to make SBOL-semantic into an authoritative ontology for synthetic biology.

To aid in the design of transcriptional devices, we will extend SBOL-semantic in order to describe rules for how components can be combined together [22] and regulated. For example, to specify the interaction between transcriptional regulatory proteins and their cognate sequences, we will use simplified representation of functional relationships. Towards this goal we plan to leverage related work such as the BioPAX effort (biopax.org) [53,54] to specify the potential role of a promoter and factor pair, not the mechanism by which it occurs. A qualitative relationship between promoter parts and regulatory proteins will allow us to query and infer intended and unintended interactions. (The ability to carry out such inferences will require the use of a Semantic Web inference system such as Pellet.) For example, an instance of the promoter pLuxR (BBa_R0062) can be annotated as having an activating role on downstream expression in presence of LuxR protein and 3-oxo-hexanoyl-HSL ($3OC_6HSL$). Such a representation of gene regulation information is limited, but forms a framework for regulatory element information retrieval. In general, we aim to expand SBOL-semantic so that it can support consistency checking of designs as a way to do initial validation of a design and to help identify possible design problems early in the engineering process.

Summary and Future Directions

Due to the amount of detail inherent in any biological system and the distributed nature of scientific research, a semantic-web based solution for organizing synthetic biology data is the suitable choice. The SBOL-semantic framework described in this work can be used to unambiguously describe, remotely query, and therefore electronically retrieve information about biological parts. In the ideal scenario, researchers would use front-end software applications for submitting and retrieving parts from the SBPkb. SBOL-semantic plug-ins for TinkerCell and Clotho are already being planned to allow those software applications to export and import parts made available through SBPkb. Embedding SBPkb query utilities in the user friendly graphical interfaces of software will help us bring these capabilities into the workflow of active synthetic biologists.

Synthetic biology research is highly distributed. In the future we envision, not just a single library, but a network of libraries. Such part libraries may range from those that contain predominately parts described in peer reviewed publications, or be a collection of parts professionally fabricated by organizations such as the International Open Facility Advancing Biotechnology (BIOFAB). As long as all these libraries are compatible with SBOL-semantic, then researchers can retrieve parts from any selection of these libraries. The SBPkb is the first node in a framework for a semantic web of distributed knowledge in synthetic biology. This vision is a small scale synthetic biology application of the Semantic Web.

In the validation portion of this work we demonstrated that searching for part information using a manual process is not a scalable or pragmatic method. Searching the web pages requires manual compilation and curation for each information query; such methods are not scalable in the face of the continually growing number of available biological parts. Using SBOL-semantic to describe synthetic biology concepts not only allows electronic retrieval, but offers the ability to select specifically defined subsets of parts.

We plan to improve and extend SBOL-semantic in the near future. Our goal is to re-engineer SBOL-semantic into an ontology which supports the forward engineering practice of synthetic biologists. In particular, we aim to include enough information to support consistency checking and design coherence, as described in the discussion section. By automating reasoning, using the semantic definitions of biological components, we aim to provide improved design automation functionality for CAD software, such as TinkerCell. More broadly, we expect to leverage the ability of the OWL language to capture rich semantics, and to support 'intelligent' information retrieval and reasoning capabilities as envisioned by the Semantic Web. This further integration of SBOL-semantic with software will help encourage re-use of previously described components, a best practice of synthetic biology.

Additionally, we hope to work with the developers of other computational tools for synthetic biologists which could benefit from computational access to a large repository of knowledge about standard parts. SBOL is an open language. The success of the language, as well as that of the broader effort to standardize electronic information exchange in synthetic biology, depends on the active involvement of the interested community. We therefore extend an invitation to all interested readers to participate in the Synthetic Biology Data Exchange Group (sbolstandard.org and the discussion forum groups.google.com/group/synbiodex).

Reuse of components in synthetic biology research is one key way in which biologists can more easily engineer and construct new systems with increased complexity. The SBOL framework allows us to capture the semantics of richly-structured descriptions and to incorporate new information needed for design in synthetic biology. Automation of design promises to make building biological machines more efficient. Finding parts that meet the specifications of designs is a critical aspect of automation of the engineering process. Leveraging Semantic Web tools (such as SPARQL) to perform information retrieval can fulfill this need and offer additional benefits such as consistency checking and classification through automated inference. Adopting these capabilities to biological system design should allow engineers to use previously created solutions and apply them to solve novel problems.

Supporting Information

File S1 SBOL-semantic OWL file which contains the semi structured controlled vocabulary used to describe standard biological parts in the SBPkb, created August 24, 2010.
(TAR)

File S2 Text file containing SPARQL queries used to retrieve standard biological parts from SBPkb.
(DOC)

Table S1 List of Part Registry Categories, attributes obtained from the source database table.
(XLSX)

Table S2 Promoter parts discovered using SBPkb query, but not found during the manual browsing portion of our work. The descriptions of the 57 additional entries, such as the status and categories are shown in the table and do not reveal a pattern which would explain their exclusion from the Catalog portion of the Parts Registry website.
(XLSX)

Acknowledgments

We would like to express our thanks to Daniel L. Cook, University of Washington, Timothy Ham, Joint BioEnergy Institute, and members of

Herbert Sauro's laboratory Bryan Bartley, Sean Sleight, and Lucian Smith for their helpful discussions. We also acknowledge the Synthetic Biology Data Exchange Group for their input on early versions of this work.

Author Contributions

Conceived and designed the experiments: MG CR DC HMS JHG. Performed the experiments: MG. Analyzed the data: MG. Wrote the paper: MG JHG. Read and edited the manuscript: MG CR DC HMS JHG.

References

1. Ro D-K, Paradise EM, Ouellet M, Fisher KJ, Newman KL, et al. (2006) Production of the antimalarial drug precursor artemisinic acid in engineered yeast. Nature 440: 940–943.
2. Alper H, Miyaoku K, Stephanopoulos G (2005) Construction of lycopene-overproducing E. coli strains by combining systematic and combinatorial gene knockout targets. Nat Biotechnol 23: 612–616.
3. Bayer TS, Widmaier DM, Temme K, Mirsky EA, Santi DV, et al. (2009) Synthesis of Methyl Halides from Biomass Using Engineered Microbes. J Am Chem Soc 131: 6508–6515.
4. Steen EJ, Kang Y, Bokinsky G, Hu Z, Schirmer A, et al. (2010) Microbial production of fatty-acid-derived fuels and chemicals from plant biomass. Nature 463: 559–562.
5. Belkin S (2003) Microbial whole-cell sensing systems of environmental pollutants. Curr Opin Microbiol 6: 206–212.
6. Joshi N, Wang X, Montgomery L, Elfick A, French CE (2009) Novel approaches to biosensors for detection of arsenic in drinking water. Desalination 248: 517–523.
7. Tigges M, Marquez-Lago TT, Stelling J, Fussenegger M (2009) A tunable synthetic mammalian oscillator. Nature 457: 309–312.
8. Young E, Alper H (2010) Synthetic biology: tools to design, build, and optimize cellular processes. J Biomed Biotechnol. doi:10.1155/2010/130781.
9. Gibson DG, Young L, Chuang R-y, Venter JC, Iii CAH, et al. (2009) Enzymatic assembly of DNA molecules up to several hundred kilobases. Nat Methods 6: 12–16.
10. Gibson DG, Glass JI, Lartigue C, Noskov VN, Chuang RY, et al. (2010) Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. Science 329: 52–56.
11. Keasling JD (2008) Synthetic biology for synthetic chemistry. ACS Chem Biol 3: 64–76.
12. Endy D (2005) Foundations for engineering biology. Nature 438: 449–453.
13. Clancy K, Voigt CA (2010) Programming cells: towards an automated 'Genetic Compiler'. Curr Opin Biotechnol 21: 572–581.
14. Chandran D, Bergmann F, Sauro H (2009) TinkerCell: modular CAD tool for synthetic biology. J Biol Eng 3: 19.
15. Rialle S, Felicori L, Dias-Lopes C, Peres S, El Atia S, et al. (2010) BioNetCAD: design, simulation and experimental validation of synthetic biochemical networks. Bioinformatics 26: 2298–2304.
16. Hill AD, Tomshine JR, Weeding E, Sotiropoulos V, Kaznessis YN (2008) SynBioSS: the synthetic biology modeling suite. Bioinformatics 24: 2551–2553.
17. Weeding E, Houle J, Kaznessis YN (2010) SynBioSS designer: a web-based tool for the automated generation of kinetic models for synthetic biological constructs. Brief Bioinform 11: 394–402.
18. Goler JA (2004) BioJADE: A Design and Simulation Tool for Synthetic Biological Systems. MIT Computer Science and Artificial Intelligence Laboratory. doi: 1721.1/30475. <http://hdl.handle.net/1721.1/30475>.
19. Densmore D, Kittleson JT, Deloache W, Batten C, Anderson JC (2010) Algorithms for automated DNA assembly. Nucleic Acids Res. pp 1–10.
20. Cai Y, Hartnett B, Gustafsson C, Peccoud J (2007) A syntactic model to design and verify synthetic genetic constructs derived from standard biological parts. Bioinformatics 23: 2760–2767.
21. Goler JA, Bramlett BW, Peccoud J (2008) Genetic design: rising above the sequence. Trends in Biotechnology 26: 538–544.
22. Cai Y, Lux MW, Adam L, Peccoud J (2009) Modeling structure-function relationships in synthetic DNA sequences using attribute grammars. PLoS Comput Biol 5: e1000529–e1000529.
23. Cai Y, Wilson ML, Peccoud J (2010) GenoCAD for iGEM: a grammatical approach to the design of standard-compliant constructs. Nucleic Acids Res 38: 2637–2644.
24. Hasty J (2002) Engineered gene circuits. Nature 420: 224–230.
25. Savageau MA (2001) Design principles for elementary gene circuits: Elements, methods, and examples. Chaos 11: 142.
26. Kaern M, Blake WJ, Collins JJ (2003) The engineering of gene regulatory networks. Annu Rev Biomed Eng 5: 179–206.
27. Shetty RP, Endy D, Knight TF, Jr. (2008) Engineering BioBrick vectors from BioBrick parts. J Biol Eng 2: 5–5.
28. Brown J (2005) The iGEM competition: building with biology. Synthetic Biology, IET 1: 3–6.
29. Peccoud J, Blauvelt MF, Cai Y, Cooper KL, Crasta O, et al. (2008) Targeted development of registries of biological parts. PLoS One 3: e2671.
30. Canton B, Labno A, Endy D (2008) Refinement and standardization of synthetic biological parts and devices. Nat Biotechnol 26: 787–793.
31. Kelly JR, Rubin AJ, Davis JH, Ajo-Franklin CM, Cumbers J, et al. (2009) Measuring the activity of BioBrick promoters using an in vivo reference standard. J Biol Eng 3: 4.
32. Knight TF Idempotent Vector Design for Standard Assembly of BioBricks. Synthetic Biology Working Group Technical Reports. doi: 1721.1/21168. <http://hdl.handle.net/1721.1/21168>.
33. Ham T, Dmytriv Z, Hillson N, Keasling J. Towards Distributed Web of Registries: Design, Implementation and Practice of the JBEI Registry. In: Riedel M. Densmore D, ed. Anaheim, CA: Proc of the Int Workshop on Bio-Design Automation (IWBDA 2010); 2010;).
34. Densmore D, Devender AV, Johnson M, Sritanyaratana N. A platform-based design environment for synthetic biological systems. In: Berry N, ed. Portland, Oregon: Proc of the 5th Richard Tapia Celebration of Diversity in Computing Conference; 2009. ACM 24–29.
35. Rouilly V, Canton B, Nielsen P, Kitney R (2007) Registry of BioBricks Models using CellML. BMC Syst Biol 1: P79.
36. Cooling MT, Rouilly V, Misirli G, Lawson J, Yu T, et al. (2010) Standard virtual biological parts: a repository of modular modeling components for synthetic biology. Bioinformatics 26: 925–931.
37. Grunberg R (2009) Draft of an RDF-based framework for the exchange and integration of Synthetic Biology data. BBF RFC #30, doi: 1721.1/45143. <http://hdl.handle.net/1721.1/45143>.
38. Galdzicki M, Chandran D, Nielsen A, Morrison J, Cowell M, et al. (2009) Provisional BioBrick Language (PoBoL). BBF RFC #31, doi: 1721.1/45537. <http://hdl.handle.net/1721.1/45537>.
39. Protégé. Available: http://protege.stanford.edu/. Accessed 2010 Aug 20.
40. Krech D (2010) RDFLib. Available: http://www.rdflib.net. Accessed 2009 Apr 20.
41. Broekstra J, Kampman A, Van Harmelen F. Sesame: A generic architecture for storing and querying rdf and rdf schema. In: Horrocks I, Hendler J, eds. Sardinia, Italia: Proc of the 1st Int'l Semantic Web Conference (ISWC 2002); 2002. Springer 54–68.
42. Eilbeck K, Lewis SE, Mungall CJ, Yandell M, Stein L, et al. (2005) The Sequence Ontology: a tool for the unification of genome annotations. Genome Biol 6: R44–R44.
43. Smith B, Ashburner M, Rosse C, Bard J, Bug W, et al. (2007) The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nat Biotechnol 25: 1251–1255.
44. Hearst MA (2009) Search user interfaces. New York: Cambridge University Press. 404 p.
45. Gruber TM, Gross Ca (2003) Multiple sigma subunits and the partitioning of bacterial transcription space. Annu Rev Microbiol 57: 441–466.
46. Barkai N, Leibler S (2000) Biological rhythms: Circadian clocks limited by noise. Nature 403: 267–268.
47. Vilar JMG, Kueh HY, Barkai N, Leibler S (2002) Mechanisms of noise-resistance in genetic oscillators. Proc Natl Acad Sci U S A 99: 5988–5992.
48. Purnick P, Weiss R (2009) The second wave of synthetic biology: from modules to systems. Nat Rev Mol Cell Biol 10: 410–422.
49. Chandran D, Bergmann FT, Sauro HM (2010) Computer-aided design of biological circuits using TinkerCell. Bioeng Bugs 1: 276–283.
50. Sirin E, Parsia B, Grau BC, Kalyanpur A, Katz Y (2007) Pellet: A practical owl-dl reasoner. Web Semantics 5: 51–53.
51. Motik B, Shearer R, Horrocks I (2009) Hypertableau reasoning for description logics. J Artif Intell Res 173: 1275–1309.
52. Tsarkov D, Horrocks I. FaCT++ description logic reasoner: System description. In: Furbach U, Shankar N, eds. Proc of the 3rd International Joint Conference on Automated Reasoning (IJCAR 2006) 2006: 292–297.
53. Luciano J, Stevens R (2007) e-Science and biological pathway semantics. BMC Bioinformatics 8: S3.
54. Sahoo SS, Bodenreider O, Rutter JL, Skinner KJ, Sheth AP (2008) An ontology-driven semantic mashup of gene and biological pathway information: application to the domain of nicotine dependence. J Biomed Inform 41: 752–765.