SBOL: A community standard for communicating designs in synthetic biology

Michal Galdzicki, Kevin P. Clancy, Ernst Oberortner, Matthew Pocock, Jacqueline Quinn, Cesar A. Rodriguez, Nicholas Roehner, Mandy L. Wilson, Laura Adam, J. Christopher Anderson, Bryan A. Bartley, Jacob Beal, Deepak Chandran, Joanna Chen, Douglas Densmore, Drew Endy, Raik Grünberg, Jennifer Hallinan, Nathan J. Hillson, Jeffrey D. Johnson, Allan Kuchinsky, Matthew Lux, Goksel Misirli, Jean Peccoud, Hector A. Plahar, Evren Sirin, Guy-Bart Stan, Alan Villalobos, Anil Wipat, John H. Gennari, Chris J. Myers, Herbert M. Sauro

Abstract

The Synthetic Biology Open Language (SBOL) is a proposed data standard for exchanging designs within the synthetic biology community. SBOL represents synthetic biology designs in a community-driven, formalized format for exchange between software tools, research groups, and commercial service providers. The re-use of previously validated designs is critical to the evolution of synthetic biology from a research discipline to an engineering practice. As a community-driven standard, SBOL adapts as synthetic biology evolves, providing specific capabilities for different aspects of the synthetic biology workflow. The SBOL Developers Group has implemented SBOL as an XML/RDF serialization and provides software libraries and specification documentation to help developers implement SBOL in their own software. This paper also reports on early successes, including a demonstration of the utility of SBOL for information exchange between several different software tools and repositories from both academic and industrial partners.

Introduction

Synthetic biology treats biological organisms as a new technological medium with a unique set of characteristics, such as the ability to self-repair, evolve, and replicate. These characteristics create their own engineering challenges, but offer a rich and largely untapped source of potential applications across a broad range of sectors[1,2]**.** Applications such as bio-molecular computing[3], metabolic engineering[4], or reconstruction and exploration of natural cell biology[5,6] commonly require the design of new genetically encoded systems. As engineers, synthetic biologists most often base their designs on previously described *DNA segments* (terms in *italics* are defined in Supplementary Table 1) in order to meet their design requirements. Reuse of the DNA sequence for these segments involves their exchange between labs and their hierarchical composition to form devices and systems with higher level function.

Every engineering field relies on a set of *standards*[7] that practitioners follow to enable the exchange and reuse of designs for *systems*, *devices*, and *components*. Similarly, the representation of synthetic biology designs using computer-readable *data standards* has the potential to facilitate the forward engineering of novel biological systems from previously characterized devices and components. For example, such standards could enable synthetic biology companies to offer catalogs of devices and components via computer-readable data sheets, just as modern semiconductor companies do for electronics. Such standards could also enable a synthetic biologist to develop portions of a design using one software tool, refine the design using another tool, and finally transmit it electronically to a colleague or commercial fabrication company.

In order for synthetic biology designs to scale up in complexity, researchers will need to make greater use of specialized design tools and parts repositories. Seamless inter-tool communication would, for example, allow the separation of genetic network design from network simulation, and the separation of both from codon optimization and synthesis. The wide adoption of a design standard would allow the growing number of software tools to more directly support an integrated design workflow[8] involving synthetic biologists from both research and commercial institutions.

Furthermore, a *standard exchange format* for synthetic biology designs would dramatically improve the ability to reproduce published results[9]. Currently, it is extremely difficult to extract workable designs from literature because designs are usually described using imprecise and error-prone English prose. All too often, critical information is accidentally omitted or implicitly assumed, and critical data, such as the final, exact DNA sequences, are simply not available.

Although standards have been proposed for experimentally measuring some key characteristics of synthetic biological parts[10-12] and for constructing composite DNA[13], descriptions of the designs themselves have not been standardized. Furthermore, standard file formats for importing and exporting DNA sequences, such as FASTA[14], GenBank's flat file format[15] and GFF[16], cannot be easily adapted to accommodate the unique requirements of synthetic biology design. Synthetic biology is about the design of novel DNA to perform a desired function, rather than sequencing an extant molecule. (For a specific comparison of the differences between GenBank and SBOL file formats, see Supplementary Table 2.) These requirements include the ability to describe partial or incomplete designs, as well as the capacity to create hierarchical designs that organize *DNA components* to achieve a desired function. Ultimately, synthetic biology workflows require the ability to encode additional information beyond an annotated sequence including, among other things, environmental and experimental context information, computational models of behavior, measurements of performance characteristics, etc. Therefore, a new, extensible standard is required to achieve these goals.

Similar to the design of electronic circuits, synthetic biology designs are composed hierarchically from libraries of reusable components. Typical DNA components, such as *promoters*, *protein coding sequences* (CDS), and *transcriptional terminators,* are described in terms of the functions they perform in a defined context*.* Reusability requires that such functional descriptions are unambiguous. The supplier of a DNA component library and the designer who uses components from that library must both use the same term to describe, for example, a CDS. No ambiguity can exist as to whether the CDS includes a start codon; the meaning must be made explicit by the definition of the term, so that it is used consistently.

Another aspect of synthetic biology design is its iterative nature. At the early stages of a design, a synthetic biologist may not yet have a specific DNA sequence chosen. Therefore, the specific sequence of a DNA component should be optional, to be specified at a later stage of the engineering process. The hierarchical composition of synthetic biology designs allows for a mix of DNA components with specified and unspecified sequences, permitting the designer to assign the sequences as the design matures, and to exchange partial specifications with collaborators. Early stage design may, for example, be ignorant about the actual order of some DNA components. If a standard requires the introduction of such constraints prematurely, it is likely to lead to unexpected dependencies and design flaws.

To address these requirements, this paper describes the *Synthetic Biology Open Language* (SBOL), a proposed standard for the representation of synthetic biology designs. Our long-term goal is to increase productivity in the design, building, testing, and dissemination of synthetic biological organisms. The SBOL Developers Group is developing this standard to meet the specific needs of synthetic biologists. In addition to describing SBOL, this paper also presents preliminary work that demonstrates the potential benefit of SBOL and SBOL-compliant software tools to the community. In our illustrative example, SBOL allows synthetic biologists to create a partial design, send the design to other tools with different capabilities for further development, and then transmit the final design for archival in several repositories.

SBOL Developers Group

Since 2008, SBOL has been under development by the SBOL Developers Group, a diverse group of both experimental and computational synthetic biologists from academic, government, and commercial organizations. At this writing, the SBOL community has 76 delegates from 37 organizations (23 academic, 11 commercial, 2 government labs, and 1 independent), who work across organizational and international boundaries to set priorities and reach agreement on the standard. Any practitioner may join the SBOL Developers Group, and we are continuously reaching out to attract new members to broaden the representation of the synthetic biology community within the group. The outreach efforts of the SBOL Developers Group have helped to attract early adopters. Recently, 18 percent of self-identified synthetic biologists responding to a survey reported current use of SBOL and 10 percent past use[17], the highest use among standards and methods for measurement, functional composition, and data exchange in the survey. This base level of support forms a foundation for broader community adoption.

SBOL is an open standard in that participation in standardization activities is unrestricted to all affected interests[18], essential information is publicly accessible on the web, and the standard can be used without cost. Additionally, as the needs of the community evolve, SBOL is also open to change. Community engagement and a democratic decision-making process steer the standard so that no one person's or organization's interests dominate its development.

To facilitate the ongoing standardization process and the development of extensions, the SBOL community has developed a formal governance structure. The SBOL effort is coordinated by five elected editors under the guidance of an elected SBOL Chair. The editors represent the diverse backgrounds of the SBOL community, and serve two-year terms. They are responsible for documentation and community organization, while the SBOL Chair helps coordinate funding and the overall development process. The SBOL Editors monitor and incorporate amendments, proposals, and requests for revisions to the SBOL specifications coming from SBOL community members and from discussion within the SBOL

Developers Group. All decisions affecting the specification of the standard are voted on, with each member of the SBOL Developers Group having equal say.

SBOL's community engagement and outreach efforts have been inspired by the tremendous success of SBML[19]. The SBOL Developers took advantage of the "lessons learned" from the SBML community, including establishing open, democratic organization; early inclusion and engagement with young scientists; and regular meetings to build up and maintain excitement and consensus within the community. The community holds a minimum of two meetings per year to encourage familiarity with the field and to develop trust among the participants. The listing of these regular workshops can be found in Supplementary Figure 2.

The SBOL Standard

The SBOL standard's foundation is a *core data model* for the specification of DNA-level designs. This SBOL Core defines biological building blocks as DNA components, and enables their hierarchical composition, allowing specification of the substructure and lineage of each design component. SBOL Core also offers a *Collection* data structure to group DNA components into meaningful libraries and catalogs. Details of the core data model can be found here:

http://www.sbolstandard.org/sbolstandard/core-data-model

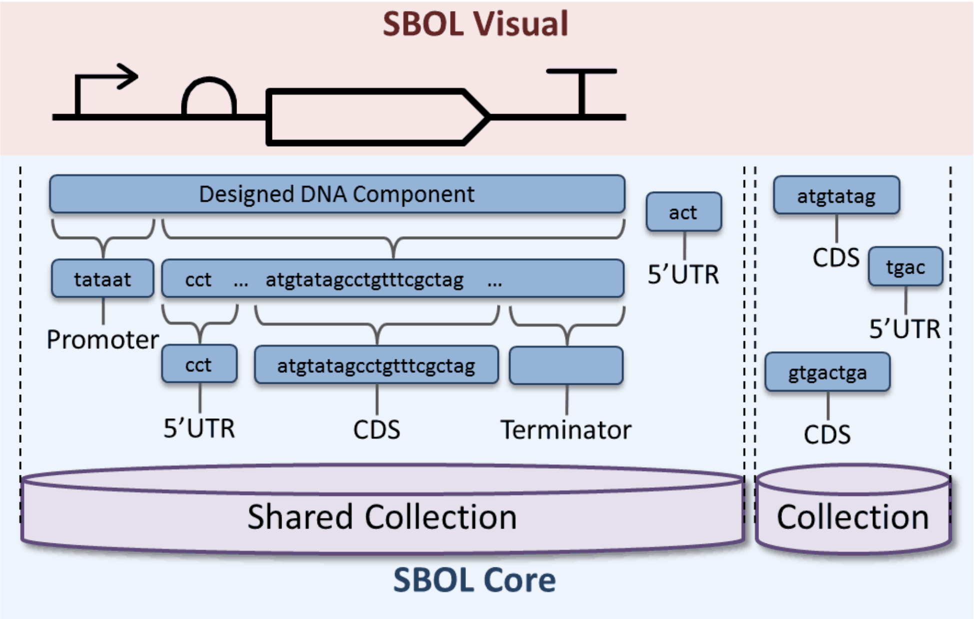

Figure 1. The SBOL core data model. A DNA component defines the design of a segment of DNA in terms of its required sub-components, their sequential arrangement (e.g., that one component must precede another) and, where known, its DNA sequence. This strategy allows us to specify: 1. designs in which the sequence is undefined, partially defined, or fully defined; 2. hierarchical compositions of components; 3. unambiguously defined component types using Sequence Ontology (SO) terms[20,21] e.g. Promoter, 5'UTR; and 4. collections of components for distribution to recipients. SBOL Visual[22], also being standardized by the SBOL Developers Group, enables the depiction of the structure of genetic designs in a standard graphical notation.

The SBOL Core leverages prior work in the development of the *Sequence Ontology* (SO)[20,21], a controlled vocabulary with a strictly defined set of concepts and relationships for DNA sequences involved in a biological process. SBOL uses SO terms to unambiguously label components in a design. Figure 1 shows an example of a hierarchical arrangement of components, each being labeled with an appropriate SO term.

The SBOL Core was first ratified and released by the SBOL Developers Group in November, 2011; Version 1.1.0 was released in October 2012 to address a couple issues requested by the community. The SBOL specification document[22] describes in detail the SBOL Core, the requirements of the standard, use cases, and software support. The use cases are derived from stakeholder requirements for exchanging synthetic biology designs. A description of the information exchange technology utilized can be found in the Supplementary Notes. Software support consists of libSBOLj, a Java library designed for developers to easily incorporate SBOL support into their tools. Table 1 presents a list of software tools that support SBOL. In order to provide feedback, report problems, and request features, SBOL users can contact the SBOL Developers as described here:

http://www.sbolstandard.org/contact-us

SBOL Demonstration

In general, the multidisciplinary nature of synthetic biology requires extensive collaboration between its practitioners, not only between academic groups, but also between public institutions and private companies. In the following demonstration, SBOL enabled six academic and commercial groups using five different computational tools and four repositories to collaborate on the design of a genetic toggle switch[38]. As illustrated in Figure 2, SBOL facilitated core principles of synthetic biology design, including collaboration between experts working on different levels of biological detail, and an iterative workflow that starts from the abstract design of a genetic circuit before moving towards the specification and refinement of actual DNA sequences (more details and all SBOL files involved in this demonstration are available in Supplementary Table 3).

In the first stage of the toggle switch design, researchers at the University of Washington designed four composite DNA components using SBOL Designer (http://clarkparsia.github.io/sbol/), a software tool for creating and visualizing basic genetic designs in SBOL. Each composite DNA component represents one possible cassette of the genetic toggle switch and is annotated with the following subcomponents: a repressible promoter, a repressor cistron, up to one reporter cistron, and a terminator. At this stage of the design, only DNA sequences for the repressible promoters and protein coding sequences (CDS) within each cistron are imported via the Standard Biological Parts knowledgebase (SBPkb)[36] from the iGEM Registry of Standard Biological Parts[35] using SBOL. The DNA sequences for the terminators and ribosome binding sites (RBS) within each cistron, on the other hand, are left unspecified, but their relative positions are indicated using SBOL *precedes* relationships.

Table 1. List of tools that support SBOL.

Application	Description	Affiliation	URL	Citation
Benchling	Web platform to edit, analyze, and collaborate on DNA sequences.	Benchling	https://benchling.com	23
Clotho (Hermes App)	A platform -based design tool for synthetic biology	BU	http://www.clothocad.org	24
DeviceEditor	A visual biological CAD canvas, front-end for j5.	JBEI/LBNL	http://j5.jbei.org	25
Eugene	A Language for solving combinatorial design problems in Synthetic Biology	BU	http://www.eugenecad.org	26
GenBank Converter	Interconverts SBOL and GenBank format files.	JBEI/LBNL	http://j5.jbei.org/bin/sbol_converter_entry_form.pl	
Gene Designer	DNA design tool.	DNA2.0	https://www.dna20.com/genedesigner2/	27
GeneGenie	Design and optimization of oligonucleotides	University of Manchester	http://oligomercedes.oligomercedes.appspot.com/	28
GSL	Internal language	Amyris		29
iBioSim	Automates modeling, analysis, and design of genetic circuits.	University of Utah	http://www.async.ece.utah.edu/iBioSim/	30
j5	Automates the design of DNA assembly protocols.	JBEI/LBNL	http://j5.jbei.org	31
JBEI-ICE	Repository for DNA sequences, microbial strains, and Arabidopsis seeds.	JBEI/LBNL	https://public-registry.jbei.org	32
MoSeC	Automates the derivation of DNA sequences from models.	Newcastle University	http://intbio.ncl.ac.uk/?projects=mosec	33
Proto BioCompiler	Automated design of genetic regulatory networks from high-level programs.	BBN	http://synbiotools.bbn.com/	34
Registry of Standard Biological Parts	Collection of standardized genetic parts, via SBOL Converter.	iGEM	http://parts.igem.org	35
SBPkb	Semantic information retrieval from Registry of Standard Biological Parts.	UW	http://www.sbolstandard.org/sbol-in-use/sbpkb	36
SBOL Designer	Create SBOL designs using SBOL visual icons and Geneious plugin	Clark & Parsia	http://clarkparsia.github.io/sbol	
TeselaGen	A visual biological CAD canvas. Automates the design of DNA assembly protocols.	TeselaGen	http://teselagen.com	
Tinker Cell (WikiDust plugin)	CAD tool for synthetic biology.	UW	www.tinkercell.org	37
VectorEditor	Viewing, annotating and *in silico* cloning of sequences	JBEI/LBNL	https://public-registry.jbei.org/static/vesa/VectorEditor.html	
Vector NTI®	Sequence analysis and design tools for molecular biology data	Life Tech	http://www.invitrogen.com/site/us/en/home/Products-and-Services/Applications/Cloning/vector-nti-software.html	
Virtual Parts	Catalog of parts and their composable models.	Newcastle University	http://www.virtualparts.org	

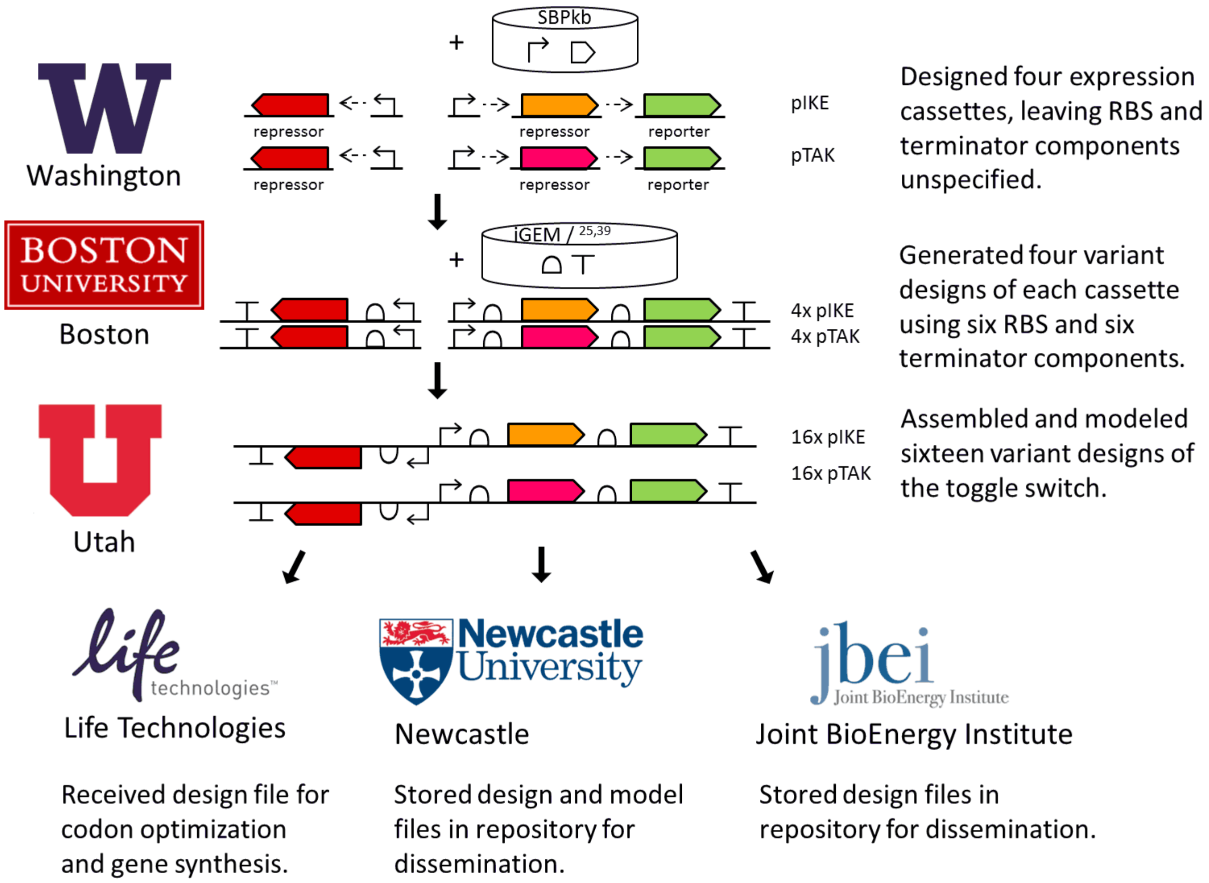
SBPkb
+
Washington
repressor
repressor
reporter
pIKE
repressor
repressor
reporter
pTAK
Designed four expression cassettes, leaving RBS and terminator components unspecified.
BOSTON UNIVERSITY
Boston
iGEM / 25,39
+
4x pIKE
4x pTAK
Generated four variant designs of each cassette using six RBS and six terminator components.
Utah
16x pIKE
16x pTAK
Assembled and modeled sixteen variant designs of the toggle switch.
life technologies
Life Technologies
Received design file for codon optimization and gene synthesis.
Newcastle University
Newcastle
Stored design and model files in repository for dissemination.
jbei Joint BioEnergy Institute
Joint BioEnergy Institute
Stored design files in repository for dissemination.

Figure 2. Demonstration of collaboration to design a genetic toggle switch. The Washington team specified abstract designs in SBOL for the two toggle switch classes based on the design by Gardner, et al.[38] for expression cassettes as DNA components *pIKE left*: TetR repressible promoter, LacI repressor CDS (red), and *pIKE right*: LacI regulated promoter, tetracycline repressor CDS (orange), and reporter GFP CDS(green); *pTAK left*: Lambda cI regulated promoter, LacI repressor (red); pTAK right: LacI regulated promoter, Lambda cI CDS (pink), and reporter GFP (green). The SBOL design file was sent to Boston University where researchers completed the DNA sequence of the cassettes with RBS and terminator sequences[25,39] generating four variants of each cassette. Upon receipt the University of Utah team combined the four variants into sixteen combinations and modeled the toggle switch functional design in SBML. They sent the output of the design to another three collaborators, Life Technologies, Newcastle University, and the Joint BioEnergy Institute demonstrating that the completed design can be read by and used in downstream applications.

During the second stage of the design, these partially abstract toggle switch cassettes are sent by email to Boston University, where researchers translated it into Eugene[26], a language to solve constrained combinatorial design problems in synthetic biology. In Eugene, these researchers imported the publicly available RBS and terminator sequences[25,39] from the iGEM Registry[35] via the Clotho platform[24], and specified non-publicly available terminator sequences manually. By using Eugene rules, the researchers pruned the number of possible toggle switch cassette variations that are fully annotated with DNA sequences.

In the final design stage, researchers at the University of Utah received the toggle switch cassettes and imported them into iBioSim[30], a software tool for the design and analysis of genetic circuits. Using iBioSim, these researchers built biochemical reaction models and composite DNA components for the toggle switches from all variants of the imported cassettes. The end result is a collection of hierarchically structured models written in SBML and composite toggle switch DNA components written in SBOL, describing the behavior and structure, respectively.

Next, researchers at Life Technologies imported the variants of the toggle switch design into VectorNTI® Express Designer, a software tool for sequence analysis and molecular biology design. Vector NTI® can, for example, import SBOL files, identify elements of a designed device, optimize codon usage in a coding sequence for a targeted organism, and request GeneArt® service to perform gene synthesis of the design.

Finally, the complete toggle switches are sent to the Joint BioEnergy Institute for storage in the public ICE[32] repository (https://public-registry.jbei.org), making them available to other researchers for future designs and construction. Additionally, and for these same reasons, the SBOL and SBML files containing the toggle switches are transmitted to researchers at Newcastle University for storage in their Virtual Parts repository (http://virtualparts.org).

The Future of SBOL

SBOL currently allows engineers to specify an unambiguous description of a DNA design in a hierarchical and fully annotated form; however, the complete specification of a design requires much more information than simply the DNA sequence. A complete description of a synthetic biology design also needs to represent other perspectives of the design, such as the dynamic behavior of the overall system, and the context of the host organism into which the design is introduced. For this reason, SBOL has been designed to be extensible, allowing additional information to be included as the synthetic biology field develops. Several extensions are under active development, including a context extension and a modeling extension.

The **SBOL Context Extension** describes the host organism used to realize the synthetic biology design, and the environment under which it must operate for its intended function to be guaranteed. The context extension provides information about the physical context, including the strain of the host, the medium in which the host resides, the container in which the medium is stored, the environmental conditions, and the measurement device used to study the context. Precise details about the experimental context are essential to the reproducibility of laboratory results. Details about this extension can be found here:

http://www.sbolstandard.org/community/sbol-working-groups/hostcontext

The **SBOL Modeling Extension** provides a mechanism for linking computational models to SBOL designs[40]. In this way, the modeling extension leverages the significant work done in the development of standards for modeling biological organisms, such as the *Systems Biology Markup Language* (SBML)[19], the *Biological Pathways Exchange* (BioPAX)[41], and the *Systems Biology Graphical Notation* (SBGN)[42]. The extension identifies the modeling language (SBML[43], CellML[44], MATLAB, BNGL[45], etc.) of the linked model, as well as its modeling framework (ODE, Stochastic, Boolean, etc.). Additionally, the extension can document interactions between components in a design, e.g. the interaction of a transcription factor with a promoter. Each interaction includes terms from the *Systems Biology Ontology* (SBO) to specify its type (repression, activation, etc.) and the roles (repressor, activator, etc.) played by its participating components. Details about this extension can be found here:

http://www.sbolstandard.org/community/sbol-working-groups/modelling

In order to connect these extensions with SBOL Core, the SBOL Developers Group has proposed extending the core with additional data structures for *devices* and *systems,* as well as, generalizing the notion of components to encompass protein and RNA components, in addition to DNA components. Devices gather components and sub-devices on the basis of shared function, while systems pair devices with their shared context. Models are associated with systems because the behavior of devices is closely tied to the context in which they are used. Figure 3 summarizes these proposed extensions and how they connect with SBOL Core. These extensions are being developed by small working groups within the SBOL Developers Group. Ultimately, extension specifications will be presented to and ratified by the entire group. As SBOL continues to mature, the SBOL Developers Group expects to add more extensions, handling an increasing range of the knowledge desired by practitioners to facilitate their interactions.

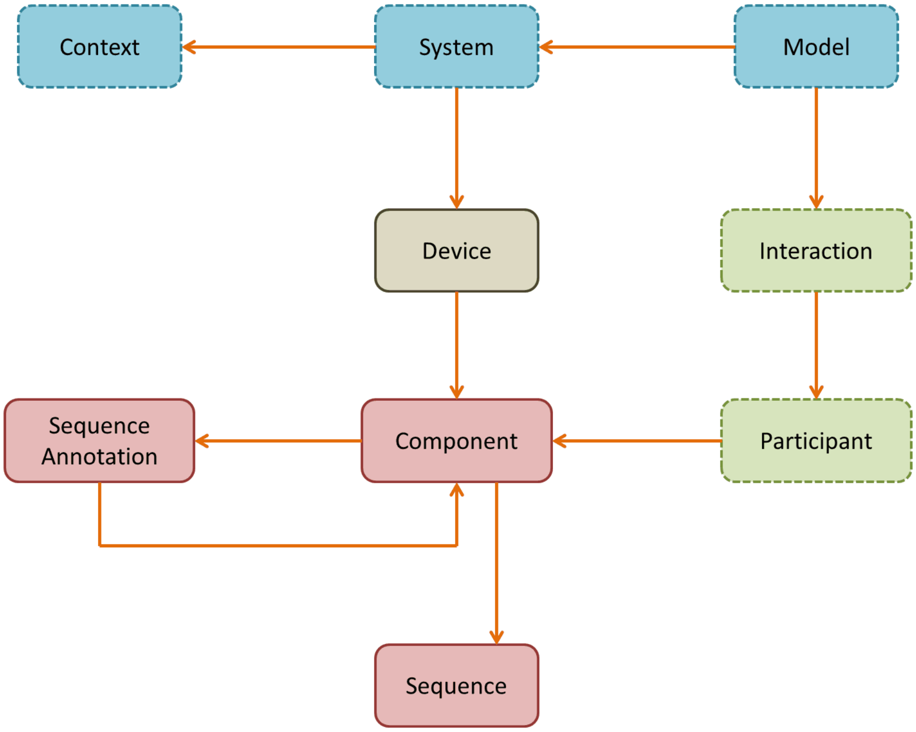

Figure 3. Extensions to the core data model and how they connect to each other. This diagram indicates the relationship between the different classes of information in a SBOL representation. At the head of the diagram is the 'System' class. Boxes in dotted outline represent proposed additions to the standard. For a more technical description of the proposed extensions see Supplementary Figure 2.

Conclusion

Since its inception in 2008, the SBOL community has grown to include academic, government, and commercial organizations, and it is on a path to become a widely adopted community standard. As of this writing, SBOL is supported by twenty-one software tools, including both commercial and academic efforts. To facilitate the adoption process, the SBOL Developers Group has developed a written specification document and associated software libraries to enable third-party developers to include SBOL in their workflow and software tools. As one way to improve productivity, SBOL encourages and facilitates the description and sharing of designs via libraries. By encouraging adoption of SBOL, we also hope to improve the reproducibility of results in the field, since SBOL files can be provided as supplementary material to journal articles, which would allow other researchers to more easily build on prior work.

More broadly, SBOL contributes to the implementation of principled engineering for biological organisms through standardization of the information exchange. However, SBOL faces several challenges, including a lack of dedicated funding for development, a need to better integrate efforts with other related standardization efforts, and the inherent challenges in coordinating efforts in an ever-growing developers group across many institutions, time zones and continents. Crucial to mitigating these challenges so far, and contributing to the success of this work, has been our open development process, organized around a diverse developers group that represents the broad activities in the synthetic biology field. Therefore, we hope that this paper serves both as an introduction and invitation to join this effort. We encourage synthetic biologists interested in joining to send an email to the SBOL editors (editors@sbolstandard.org). In establishing SBOL and its community, we strive to foster the translation of synthetic biology research into practice.

Acknowledgements

We acknowledge Haiyao Huang for her technical support on augmenting the Clotho platform for SBOL compliance. This work was initiated by an award from the Microsoft Computational Challenges in Synthetic Biology Initiative (2006). Subsequently the effort was supported by a variety of funding sources including AutoDesk, National Science Foundation (0527023, 1147158 and CCF-1218095), National Library of Medicine (R41 LM010745, T15 LM007442), National Human Genome Research Institute (R42 HG006737), Agilent Technologies' Applications and Core Technology University Research (ACT-UR) program, DARPA (HR0011-10-C- 0168), and the EPSRC-funded Flowers Consortium project (EP/J02175X/1). The portion of this work conducted by the Joint BioEnergy Institute is supported by the Office of Science, Office of Biological and Environmental Research, of the U.S. Department of Energy (Contract No.DE-AC02-05CH11231). The views and conclusions contained in this document are those of the authors and not the U.S. Government or any agency thereof. **Dedication:** We would like to dedicate this paper to the memory of Allan Kuchinsky who made significant contributions to SBOL through his support at our workshop meetings and critically to the development of libSBOLj.

Contributions

All authors helped develop the SBOL standard by contributing to the specification document and by participating in workshops or on the SBOL Developers mailing list and/or wrote software that supports SBOL. See Supplementary Table 4 for a full list of author contributions.

References

1. Khalil, A.S. & Collins, J.J. Synthetic biology: applications come of age. *Nature reviews. Genetics* **11**, 367-379 (2010).
2. Keasling, J. The Promise of Synthetic Biology. *The Bridge* **35**, 18-21 (2005).
3. Benenson, Y. Biomolecular computing systems: principles, progress and potential. *Nature reviews. Genetics* **13**, 455-468 (2012).
4. Woolston, B.M., Edgar, S. & Stephanopoulos, G. Metabolic engineering: past and future. *Annual review of chemical and biomolecular engineering* **4**, 259-288 (2013).
5. Nandagopal, N. & Elowitz, M.B. Synthetic biology: integrated gene circuits. *Science* **333**, 1244-1248 (2011).
6. Mukherji, S. & van Oudenaarden, A. Synthetic biology: understanding biological design from synthetic circuits. *Nature reviews. Genetics* **10**, 859-871 (2009).
7. Slattery, W.J. An index of U.S. voluntary engineering standards; covering those standards, specifications, test methods, and recommended practices issued by national standardization organizations in the United States. Vol. 329 (United States. National Bureau of Standards. Office of Engineering Standards Services, 1971).
8. Beal, J.*, et al.* An End-to-End Workflow for Engineering of Biological Networks from High-Level Specifications. *ACS Synthetic Biology* **1**, 317-331 (2012).
9. Peccoud, J.*, et al.* Essential information for synthetic DNA sequences. *Nature biotechnology* **29**, 22 (2011).
10. Beal, J., Weiss, R., Yaman, F., Davidsohn, N. & Adler, A. A Method for Fast, High-Precision Characterization of Synthetic Biology Devices. *MIT CSAIL Tech Report 2012-008* (2012).
11. Canton, B., Labno, A. & Endy, D. Refinement and standardization of synthetic biological parts and devices. *Nature biotechnology* **26**, 787-793 (2008).
12. Kelly, J.R.*, et al.* Measuring the activity of BioBrick promoters using an in vivo reference standard. *Journal of biological engineering* **3**, 1-13 (2009).
13. Müller, K.M. & Arndt, K.M. Chapter 2 Standardization in Synthetic Biology. **813**(2012).
14. Pearson, W.R. & Lipman, D.J. Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences of the United States of America* **85**, 2444-2448 (1988).
15. INSDC. The DDBJ/EMBL/GenBank Feature Table Definition Version 10.2. (2012).
16. Stein, L. Generic Feature Format Version 3 (GFF3) Version: 1.21. (2013).
17. Kahl, L.J. & Endy, D. A survey of enabling technologies in synthetic biology. *Journal of biological engineering* **7**, 13 (2013).
18. United States Standards Strategy Committee. The United States Standards Strategy. (American National Standards Institute, 2010).
19. Hucka, M.*, et al.* The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19**, 524-531 (2003).
20. Eilbeck, K.*, et al.* The Sequence Ontology: a tool for the unification of genome annotations. *Genome biology* **6**, R44 (2005).
21. Mungall, C.J., Batchelor, C. & Eilbeck, K. Evolution of the Sequence Ontology terms and relationships. *Journal of biomedical informatics* (2010).
22. Galdzicki, M.*, et al.* Synthetic Biology Open Language (SBOL) Version 1.1.0. in *BBF RFC #87* 1-26 (2012).

23. Li, S.C. Personal Communication: Re: New Software Tool Using SBOL (Benchling). (ed. sbol–dev@googlegroups.com) (2013).

24. Xia, B.*, et al.* Developer's and user's guide to Clotho v2.0 A software platform for the creation of synthetic biological systems. *Methods in enzymology* **498**, 97-135 (2011).

25. Chen, J., Densmore, D., Ham, T.S., Keasling, J.D. & Hillson, N.J. DeviceEditor visual biological CAD canvas. *Journal of biological engineering* **6**, 1 (2012).

26. Bilitchenko, L.*, et al.* Eugene - A domain specific language for specifying and constraining synthetic biological parts, devices, and systems. *PLoS one* **6**, e18882 (2011).

27. Villalobos, A., Ness, J., Gustafsson, C., Minshull, J. & Govindarajan, S. Gene Designer: a synthetic biology tool for constructing artificial DNA segments. *BMC Bioinformatics* **7**, 285 (2006).

28. Swainston, N. Re: SBOL for oligomer designing tool. (ed. <myers@ece.utah.edu>, C.J.M.) (2013).

29. Platt, D. Personal Communication: Thanks! [re: San Francisco SBOL meeting]. (sbol-dev@googlegroups.com, 2012).

30. Myers, C.J.*, et al.* iBioSim: a tool for the analysis and design of genetic circuits. *Bioinformatics* **25**, 2848-2849 (2009).

31. Hillson, N.J., Rosengarten, R.D. & Keasling, J.D. j5 DNA Assembly Design Automation Software. *ACS Synthetic Biology* **1**, 14-21 (2012).

32. Ham, T.S.*, et al.* Design, implementation and practice of JBEI-ICE: an open source biological part registry platform and tools. *Nucleic Acids Res* **40**, e141 (2012).

33. Misirli, G.*, et al.* Model annotation for synthetic biology: automating model to nucleotide sequence conversion. *Bioinformatics* **27**, 973-979 (2011).

34. Beal, J., Lu, T. & Weiss, R. Automatic compilation from high-level biologically-oriented programming language to genetic regulatory networks. *PLoS One* **6**, e22490 (2011).

35. Registry. Registry of Standard Biological Parts. (2012).

36. Galdzicki, M., Rodriguez, C., Chandran, D., Sauro, H.M. & Gennari, J.H. Standard Biological Parts Knowledgebase. *PLoS ONE* **6**, e17005 (2011).

37. Chandran, D., Bergmann, F.T. & Sauro, H.M. TinkerCell: modular CAD tool for synthetic biology. *Journal of biological engineering* **3**(2009).

38. Gardner, T.S., Cantor, C.R. & Collins, J.J. Construction of a genetic toggle switch in Escherichia coli. *Nature* **405**, 339-342 (2000).

39. Salis, H.M., Mirsky, E.A. & Voigt, C.A. Automated design of synthetic ribosome binding sites to control protein expression. *Nature biotechnology* **27**, 946-950 (2009).

40. Cai, Y., Lux, M.W., Adam, L. & Peccoud, J. Modeling Structure-Function Relationships in Synthetic DNA Sequences using Attribute Grammars. *PLoS Comput Biol* **5**, e1000529 (2009).

41. Demir, E.*, et al.* The BioPAX community standard for pathway data sharing. *Nature biotechnology* **28**, 935-942 (2010).

42. Le Novere, N.*, et al.* The Systems Biology Graphical Notation. *Nature biotechnology* **27**, 735-741 (2009).

43. Bornstein, B.*, et al.* BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Research* **34**, D689-691 (2006).

44. Lloyd, C.M., Halstead, M.D.B. & Nielsen, P.F. CellML: its future, present and past. *Progress in Biophysics and Molecular Biology* **85**, 433-450 (2004).

45. Faeder, J.R., Blinov, M.L. & Hlavacek, W.S. Rule-based modeling of biochemical systems with BioNetGen. *Methods in molecular biology* **500**, 113-167 (2009).