

The Synthetic Biology Open Language (SBOL) provides a community standard for communicating designs in synthetic biology

Michal Galdzicki¹, Kevin P Clancy², Ernst Oberortner³, Matthew Pocock⁴, Jacqueline Y Quinn⁵, Cesar A Rodriguez⁵, Nicholas Roehner⁶, Mandy L Wilson⁷, Laura Adam⁷, J Christopher Anderson⁸, Bryan A Bartley⁹, Jacob Beal¹⁰, Deepak Chandran⁵, Joanna Chen¹¹, Douglas Densmore³, Drew Endy¹², Raik Grünberg¹³, Jennifer Hallinan⁴, Nathan J Hillson¹¹, Jeffrey D Johnson¹⁴, Allan Kuchinsky¹⁵, Matthew Lux⁷, Goksel Misirli⁴, Jean Peccoud⁷, Hector A Plahar¹¹, Evren Sirin¹⁶, Guy-Bart Stan¹⁷, Alan Villalobos¹⁸, Anil Wipat⁴, John H Gennari¹, Chris J Myers¹⁹ & Herbert M Sauro⁹

The re-use of previously validated designs is critical to the evolution of synthetic biology from a research discipline to an engineering practice. Here we describe the Synthetic Biology Open Language (SBOL), 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 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. We describe 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. As a community-driven standard, SBOL will be updated as synthetic biology evolves to provide specific capabilities for different aspects of the synthetic biology workflow.

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 biomolecular computing³, metabolic engineering⁴, 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’ (see **Supplementary Table 1** for definitions of selected terms) to meet their design requirements. Reuse of the DNA sequence for these segments involves their exchange between laboratories 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 by means of 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 workflow⁸ involving synthetic biologists from both research and commercial institutions.

¹Biomedical and Health Informatics, University of Washington, Seattle, Washington, USA. ²Synthetic Biology Unit, Life Technologies, Carlsbad, California, USA. ³Department of Electrical and Computer Engineering, Boston University, Boston, Massachusetts, USA. ⁴School of Computing Science, Newcastle University, Newcastle upon Tyne, UK. ⁵Autodesk Research, Autodesk, San Francisco, California, USA. ⁶Bioengineering, University of Utah, Salt Lake City, Utah, USA. ⁷Virginia Bioinformatics Institute, Virginia Tech, Blacksburg, Virginia, USA. ⁸Department of Bioengineering, University of California, Berkeley, Berkeley, California, USA. ⁹Department of Bioengineering, University of Washington, Seattle, Washington, USA. ¹⁰Raytheon BBN Technologies, Cambridge, Massachusetts, USA. ¹¹Fuels Synthesis Division, Joint Bioenergy Institute (JBEI), Berkeley, California, USA. ¹²Department of Bioengineering, Stanford University, Stanford, California, USA. ¹³IRIC, University of Montreal, Montreal, Quebec, Canada. ¹⁴Plant and Microbial Biology, University of California, Berkeley, Berkeley, California, USA. ¹⁵Molecular Tools Laboratory, Agilent Technologies, Santa Clara, California, USA. ¹⁶Clark & Parsia, Arlington, Massachusetts, USA. ¹⁷Bioengineering, Imperial College London, London, UK. ¹⁸DNA2.0, Menlo Park, California, USA. ¹⁹Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, Utah, USA. Correspondence should be addressed to H.M.S. (hsauro@uw.edu).

Received 9 November 2013; accepted 20 December 2013; published online 6 June 2014; doi:10.1038/nbt.2891

Furthermore, a 'standard exchange format' for synthetic biology designs would dramatically improve the ability to reproduce published results⁹. 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 DNA¹³, descriptions of the designs themselves have not been standardized. Furthermore, standard file formats for importing and exporting DNA sequences, such as FASTA¹⁴, GenBank's flat file format (<http://www.insdc.org/documents/feature-table>) and GFF (<http://www.sequenceontology.org/gff3.shtml>), 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 and measurements of performance characteristics. 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 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 archiving in several repositories.

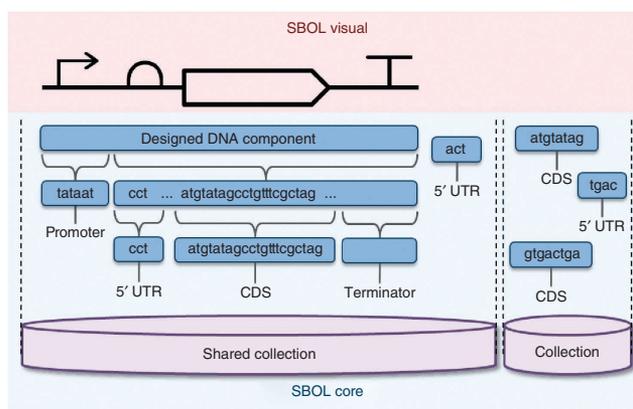
SBOL Developers Group

Since 2008, SBOL has been under development by the SBOL Developers Group, a diverse group of both experimental and computational synthetic biologists from academic, governmental and commercial organizations. At this writing, the SBOL community has 76 delegates from 37 organizations (23 academic, 11 commercial, 2 governmental 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 continually 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% of self-identified synthetic biologists responding to a survey reported current use of SBOL and 10% past use¹⁵, 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 interests¹⁶, 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

Figure 1 The SBOL core data model. A DNA component defines the design of a segment of DNA in terms of its required subcomponents, their sequential arrangement (for example, that one component must precede another) and, where known, its DNA sequence. This strategy allows us to specify (i) designs in which the sequence is undefined, partially defined or fully defined; (ii) hierarchical composition of components; (iii) unambiguously defined component types using Sequence Ontology (SO) terms^{18,19}, for example, Promoter, 5' UTR; and (iv) collections of components for distribution to recipients. SBOL visual²⁰, also being standardized by the SBOL Developers Group, enables the depiction of the structure of genetic designs in a standard graphical notation.



serve two-year terms. They are responsible for documentation and community organization, whereas 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 the 'Systems Biology Markup Language' (SBML)¹⁷. The SBOL Developers Group took advantage of the lessons learned from the SBML community, including establishing an open, democratic organization; early inclusion of 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 1**.

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 at <http://www.sbolstandard.org/sbolstandard/core-data-model>. The SBOL core leverages prior work in the development of the 'Sequence Ontology' (SO)^{18,19}, 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.

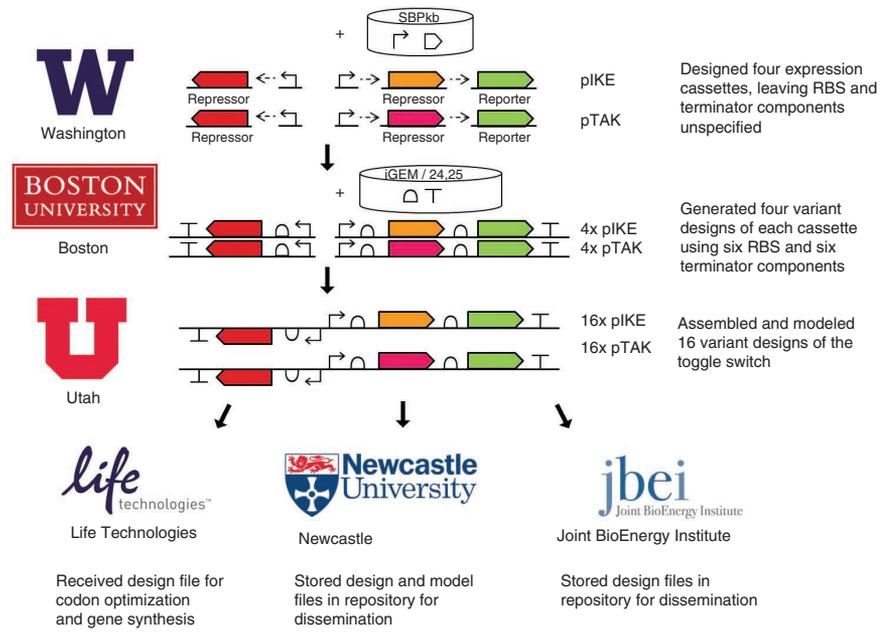
The SBOL core was first ratified and released by the SBOL Developers Group in November, 2011; version 1.1.0 was released in October 2012 to address a couple of issues requested by the community. The SBOL specification document²⁰ describes in detail the SBOL core, the requirements of the standard, use cases and software

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/	(S.C. Li, Benchling, Inc., personal communication)
Clotho (Hermes App)	A platform-based design tool for synthetic biology	BU	http://www.clothocad.org/	26
DeviceEditor	A visual biological CAD canvas, front-end for j5	JBEI/LBNL	http://j5.jbei.org/	24
Eugene	A language for solving combinatorial design problems in synthetic biology	BU	http://www.eugenecad.org/	23
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/	35
GeneGenie	Design and optimization of oligonucleotides	University of Manchester	http://oligomercedes.oligomercedes.appspot.com	(N. Swainston, University of Manchester, personal communication)
GSL	Internal language	Amyris	NA	(D. Platt, Amyris, Inc., personal communication)
iBioSim	Automates modeling, analysis and design of genetic circuits	University of Utah	http://www.async.ece.utah.edu/iBioSim/	27
j5	Automates the design of DNA assembly protocols	JBEI/LBNL	http://j5.jbei.org/	36
JBEI-ICE	Repository for DNA sequences, microbial strains and <i>Arabidopsis</i> seeds	JBEI/LBNL	https://public-registry.jbei.org/	28
MoSeC	Automates the derivation of DNA sequences from models	Newcastle University	http://intbio.ncl.ac.uk/?projects=mosec	37
Proto BioCompiler	Automated design of genetic regulatory networks from high-level programs	BBN	https://synbiotools.bbn.com/	38
Registry of Standard Biological Parts	Collection of standardized genetic parts, via SBOL Converter	iGEM	http://parts.igem.org/	
SBPkb	Semantic information retrieval from Registry of Standard Biological Parts	UW	http://www.sbolstandard.org/libsbol/sbpkb	22
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	TeselaGen	http://teselagen.com/	
Tinker Cell (WikiDust plugin)	Automates the design of DNA assembly protocols	UW	http://www.tinkercell.com/	39
VectorEditor	CAD tool for synthetic biology	JBEI/LBNL	https://public-registry.jbei.org/static/vesa/VectorEditor.html	
Vector NTI	Viewing, annotating and <i>in silico</i> cloning of sequences	JBEI/LBNL	https://public-registry.jbei.org/static/vesa/VectorEditor.html	
Virtual Parts	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	
	Catalog of parts and their composable models	Newcastle University	http://virtualparts.org/	

BU, Boston University; CAD, computer-aided design; NA, not available.

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.*²¹ 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 cl-regulated promoter, LacI repressor (red); pTAK right: LacI-regulated promoter, Lambda cl 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^{24,25} generating four variants of each cassette. Upon receipt the University of Utah team assembled the four variants to create 16 combinations and modeled these toggle switch designs 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.



support. The use cases are derived from stakeholder requirements for exchanging synthetic biology designs. A description of the information exchange technology used 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 (**Supplementary Software**). **Table 1** presents a list of software tools that support SBOL. To provide feedback, report problems and request features, SBOL users can contact the SBOL Developers as described at <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 switch²¹. SBOL facilitated core principles of synthetic biology design (**Fig. 2**), 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 toward the specification and refinement of actual DNA sequences (all SBOL files involved in this demonstration are available in **Supplementary Data Set 1** and described 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 represented one possible cassette of the genetic toggle switch and was annotated with the following sub-components: 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 CDS within each cistron were imported through the Standard Biological Parts knowledgebase (SBPkb)²² from the iGEM Registry of Standard Biological Parts (<http://parts.igem.org/>) using SBOL. The DNA sequences for

the terminators and ribosome binding sites (RBS) within each cistron, on the other hand, were left unspecified, but their relative positions were indicated using SBOL ‘precedes’ relationships.

During the second stage of the design, these partially abstract toggle switch cassettes were sent by email to Boston University, where researchers translated it into Eugene²³, a language to solve constrained combinatorial design problems in synthetic biology. In Eugene, these researchers imported the publicly available RBS and terminator sequences^{24,25} from the iGEM Registry through the Clotho platform²⁶, and specified nonpublicly 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 iBioSim²⁷, 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 Vector NTI 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 completed toggle switches were sent to the Joint BioEnergy Institute for storage in the public inventory of composable elements (ICE)²⁸ 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 were transmitted to researchers at Newcastle University for storage in their virtual parts repository (<http://virtualparts.org/>).



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 an SBOL representation. At the head of the diagram is the 'system' class. Boxes in dashed outline represent proposed additions to the standard. For a more technical description of the proposed extensions see **Supplementary Figure 2**.

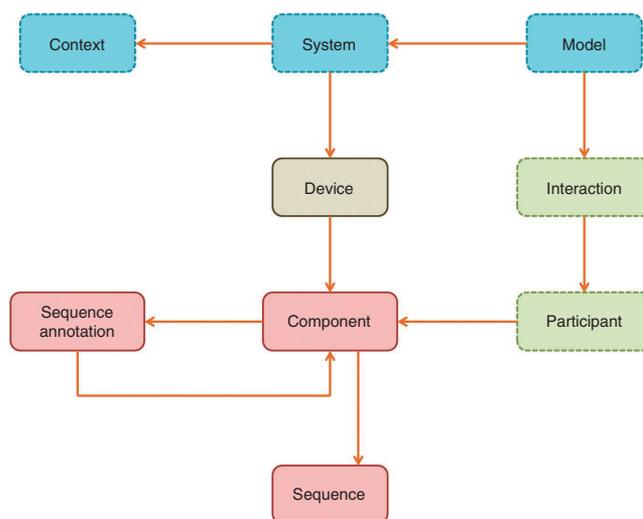
The future of SBOL

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

The SBOL modeling extension provides a mechanism for linking computational models to SBOL designs²⁹. In this way, the modeling extension leverages the significant work done in the development of standards for modeling biological organisms, such as SBML¹⁷, the 'Biological Pathways Exchange' (BioPAX)³⁰, and the 'Systems Biology Graphical Notation' (SBGN)³¹. The extension identifies the modeling language (for example, SBML³², CellML³³, MATLAB, BNGL³⁴) of the linked model, as well as its modeling framework (for example, ODE, Stochastic, Boolean). Additionally, the extension can document interactions between components in a design, for example, the interaction of a transcription factor with a promoter. Each interaction includes terms from the 'Systems Biology Ontology' (SBO) to specify its type (for example, repression, activation) and the roles (for example, repressor, activator) played by its participating components. Details about this extension can be found at <http://www.sbolstandard.org/community/sbol-working-groups/modelling>.

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 subdevices on the basis of shared function, whereas 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.



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 21 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 through libraries. By encouraging adoption of SBOL, we also hope to improve the reproducibility of results in the field; if SBOL files are provided as supplementary material to journal articles, other researchers can 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.

Note: Any Supplementary Information and Source Data files are available in the online version of the paper.

ACKNOWLEDGMENTS

We acknowledge H. 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, Inc., National Science Foundation (0527023, 1147158, EF-0850100 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, Defense Advanced Research Projects Agency (DARPA; HR0011-10-C-0168), the Engineering and Physical Sciences Research Council

(EPSRC)-funded Flowers Consortium project (EP/J02175X/1) and EPSRC-funded Centre for Synthetic Biology and Innovation at Imperial College (EP/G036004/1). The portion of this work conducted by the Joint BioEnergy Institute was supported by the Office of Science, Office of Biological and Environmental Research, the US Department of Energy (contract no. DE-AC02-05CH11231). The views and conclusions contained in this document are those of the authors and not the US government or any agency thereof. 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.

AUTHOR CONTRIBUTIONS

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

COMPETING FINANCIAL INTERESTS

The authors declare competing financial interests: details are available in the [online version of the paper](#).

Reprints and permissions information is available online at <http://www.nature.com/reprints/index.html>.

- Khalil, A.S. & Collins, J.J. Synthetic biology: applications come of age. *Nat. Rev. Genet.* **11**, 367–379 (2010).
- Keasling, J. The promise of synthetic biology. *The Bridge* **35**, 18–21 (2005).
- Benenson, Y. Biomolecular computing systems: principles, progress and potential. *Nat. Rev. Genet.* **13**, 455–468 (2012).
- Woolston, B.M., Edgar, S. & Stephanopoulos, G. Metabolic engineering: past and future. *Annu. Rev. Chem. Biomol. Eng.* **4**, 259–288 (2013).
- Nandagopal, N. & Elowitz, M.B. Synthetic biology: integrated gene circuits. *Science* **333**, 1244–1248 (2011).
- Mukherji, S. & van Oudenaarden, A. Synthetic biology: understanding biological design from synthetic circuits. *Nat. Rev. Genet.* **10**, 859–871 (2009).
- Slattery, W.J. An Index of US 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).
- Beal, J. *et al.* An end-to-end workflow for engineering of biological networks from high-level specifications. *ACS Synth. Biol.* **1**, 317–331 (2012).
- Peccoud, J. *et al.* Essential information for synthetic DNA sequences. *Nat. Biotechnol.* **29**, 22 (2011).
- 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).
- Canton, B., Labno, A. & Endy, D. Refinement and standardization of synthetic biological parts and devices. *Nat. Biotechnol.* **26**, 787–793 (2008).
- Kelly, J.R. *et al.* Measuring the activity of BioBrick promoters using an *in vivo* reference standard. *J. Biol. Eng.* **3**, 4 (2009).
- Müller, K.M. & Arndt, K.M. in *Synthetic Gene Networks: Methods and Protocols* (eds. Wilfried Weber, W. & Fussenegger, M.) Methods in Molecular Biology, vol. 813, 23–43 (2012).
- Pearson, W.R. & Lipman, D.J. Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA* **85**, 2444–2448 (1988).
- Kahl, L.J. & Endy, D. A survey of enabling technologies in synthetic biology. *J. Biol. Eng.* **7**, 13 (2013).
- United States Standards Strategy Committee. *United States Standards Strategy* (American National Standards Institute, 2010).
- 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).
- Eilbeck, K. *et al.* The Sequence Ontology: a tool for the unification of genome annotations. *Genome Biol.* **6**, R44 (2005).
- Mungall, C.J., Batchelor, C. & Eilbeck, K. Evolution of the Sequence Ontology terms and relationships. *J. Biomed. Inform.* **44**, 87–93 (2011).
- Galdzicki, M. *et al.* Synthetic Biology Open Language (SBOL) Version 1.1.0. in *BBF RFC #87* 1–26. <http://hdl.handle.net/1721.1/73909> (2012).
- Gardner, T.S., Cantor, C.R. & Collins, J.J. Construction of a genetic toggle switch in *Escherichia coli*. *Nature* **403**, 339–342 (2000).
- Galdzicki, M., Rodriguez, C., Chandran, D., Sauro, H.M. & Gennari, J.H. Standard Biological Parts Knowledgebase. *PLoS ONE* **6**, e17005 (2011).
- Billtchenko, L. *et al.* Eugene: a domain specific language for specifying and constraining synthetic biological parts, devices, and systems. *PLoS ONE* **6**, e18882 (2011).
- Chen, J., Densmore, D., Ham, T.S., Keasling, J.D. & Hillson, N.J. DeviceEditor visual biological CAD canvas. *J. Biol. Eng.* **6**, 1 (2012).
- Salis, H.M., Mirsky, E.A. & Voigt, C.A. Automated design of synthetic ribosome binding sites to control protein expression. *Nat. Biotechnol.* **27**, 946–950 (2009).
- 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 Enzymol.* **498**, 97–135 (2011).
- Myers, C.J. *et al.* iBioSim: a tool for the analysis and design of genetic circuits. *Bioinformatics* **25**, 2848–2849 (2009).
- 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).
- 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).
- Demir, E. *et al.* The BioPAX community standard for pathway data sharing. *Nat. Biotechnol.* **28**, 935–942 (2010).
- Le Novère, N. *et al.* The Systems Biology Graphical Notation. *Nat. Biotechnol.* **27**, 735–741 (2009).
- Le Novère, N. *et al.* BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Res.* **34**, D689–D691 (2006).
- Lloyd, C.M., Halstead, M.D.B. & Nielsen, P.F. CellML: its future, present and past. *Prog. Biophys. Mol. Biol.* **85**, 433–450 (2004).
- Faeder, J.R., Blinov, M.L. & Hlavacek, W.S. Rule-based modeling of biochemical systems with BioNetGen. *Methods Mol. Biol.* **500**, 113–167 (2009).
- 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).
- Hillson, N.J., Rosengarten, R.D. & Keasling, J.D. j5 DNA Assembly Design Automation Software. *ACS Synth. Biol.* **1**, 14–21 (2012).
- Misirli, G. *et al.* Model annotation for synthetic biology: automating model to nucleotide sequence conversion. *Bioinformatics* **27**, 973–979 (2011).
- Beal, J., Lu, T. & Weiss, R. Automatic compilation from high-level biologically-oriented programming language to genetic regulatory networks. *PLoS ONE* **6**, e22490 (2011).
- Chandran, D., Bergmann, F.T. & Sauro, H.M. TinkerCell: modular CAD tool for synthetic biology. *J. Biol. Eng.* **3**, 19 (2009).