

# Lawrence Berkeley National Laboratory

LBL Publications

## Title

BioParts □ A Biological Parts Search Portal and Updates to the ICE Parts Registry Software Platform

## Permalink

<https://escholarship.org/uc/item/4fm7k354>

## Journal

ACS Synthetic Biology, 10(10)

## ISSN

2161-5063

## Authors

Plahar, Hector A

Rich, Thomas N

Lane, Stephen D

et al.

## Publication Date

2021-10-15

## DOI

10.1021/acssynbio.1c00263

Peer reviewed

# BioParts—A Biological Parts Search Portal and Updates to the ICE Parts Registry Software Platform

Hector A. Plahar, Thomas N. Rich, Stephen D. Lane, William C. Morrell, Leanne Springthorpe, Oge Nnadi, Elena Aravina, Tiffany Dai, Michael J. Fero, Nathan J. Hillson,\* and Christopher J. Petzold\*



Cite This: <https://doi.org/10.1021/acssynbio.1c00263>



Read Online

ACCESS |

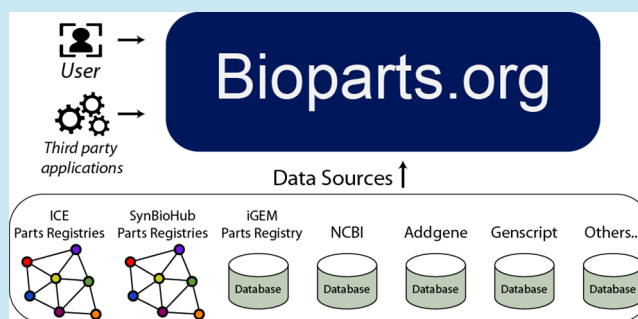
Metrics & More

Article Recommendations

**ABSTRACT:** Capturing, storing, and sharing biological DNA parts data are integral parts of synthetic biology research. Here, we detail updates to the ICE biological parts registry software platform that enable these processes, describe our implementation of the Web of Registries concept using ICE, and establish BioParts, a search portal for biological parts available in the public domain. The Web of Registries enables standalone ICE installations to securely connect and form a distributed parts database. This distributed database allows users from one registry to query and access plasmid, strain, (DNA) part, plant seed, and protein entry types in other connected registries. Users can also transfer entries from one ICE registry to another or make them publicly accessible.

BioParts, the new search portal, combines the ease and convenience of modern web search engines with the capabilities of bioinformatics search tools such as BLAST. This portal, available at [bioparts.org](http://bioparts.org), allows anyone to search for publicly accessible biological part information (e.g., NCBI, iGEM, SynBioHub, Addgene), including parts publicly accessible through ICE Registries. Additionally, the portal offers a REST API that enables third-party applications and tools to access the portal's functionality programmatically.

**KEYWORDS:** synthetic biology, repository, Web of Registries, Bioparts.org, OpenVectorEditor, sharing, biological parts search engine, multiplatform



Significant progress in synthetic biology and metabolic engineering is dependent on rapid, accurate data collection, analysis, and sharing. Central to these efforts are tools that standardize nomenclature, data types, and ontologies in a secure environment. For example, the Synthetic Biology Open Language (SBOL) data standard<sup>1–4</sup> simplifies nomenclature for biological parts, formalizing the data-exchange format used between software tools. These efforts are essential steps toward FAIR (findable, accessible, interoperable, and reusable) data principles.<sup>5</sup> The wide availability of software tools has resulted in a substantial resource of biological parts and associated data, much of which is publicly accessible. Notable examples are the Inventory of Composable Elements (ICE),<sup>6</sup> the iGEM Registry of Biological Parts ([igem.org/Registry](http://igem.org/Registry)), SynBioHub,<sup>7</sup> and several commercial tools such as those from TeselaGen Biotechnology and Benchling that store and manage information about biological parts. Physical samples for a subset of these parts are also readily available, thanks to repositories such as Addgene and GenScript, that store and distribute them worldwide.

Modern biological part registries need additional features to acquire the level of sophistication required by advances in

synthetic biology. ICE first introduced the concept of distributed and interconnected use in biological part registries to provide reusability and data sharing in synthetic biology. This technology, referred to as “web of registries”, has, over time, seen some implementations in other synthetic biology data repositories (e.g., SynBioHub<sup>7,8</sup>). While the web of registries concept presents many features and capabilities that leverage the capabilities of software platforms to push the field of synthetic biology forward, it does not bridge the gap between systems that do not have an existing, agreed-upon standard format. The use of REpresentational State Transfer (REST) Application Programming Interfaces (APIs),<sup>9</sup> and standards such as SBOL, addresses some of these issues. Nonetheless, the synthetic biology community will benefit from a tool providing easy accessibility to biological part data

Received: June 10, 2021

in a manner similar to that which search engines offer users wishing to find general data on the Internet. Owing to the nature and structure of biological DNA/RNA sequences, general-purpose search engines are unable to index biological registries via techniques used for regular Web sites. As such, the need exists for a specialized tool to account for the unique nature of biological sequences, index associated publicly available data, and make searching for them fast and convenient.

In this work, we describe updates to the ICE biological parts registry software platform, which we first introduced nearly a decade ago.<sup>6</sup> Since then, we have made several essential architectural improvements and introduced several new features. We detail here how we implement the web of registries concept to permit secure connections across different ICE parts registries and form a single distributed parts database using the ICE platform. This interconnectedness enables collaboration between users of various parts registry instances. It allows users from one ICE installation to query and access plasmid, strain, (DNA) part, plant seed, and protein entry types in other connected parts registries across distributed geographic locations. Users can also transfer entries from one registry instance to another and make them publicly accessible to everyone. Fine-grained access control privileges built into the ICE platform empower researchers to make their data private while it is part of ongoing research activity and then make it available to the general public at the appropriate time. We also describe a specialized tool in the form of a search portal available at [bioparts.org](http://bioparts.org). This tool indexes publicly available biological parts and enables searching using keywords, sequence fragments, or both. Enabling public-access privileges on ICE entries makes them available on the search portal.

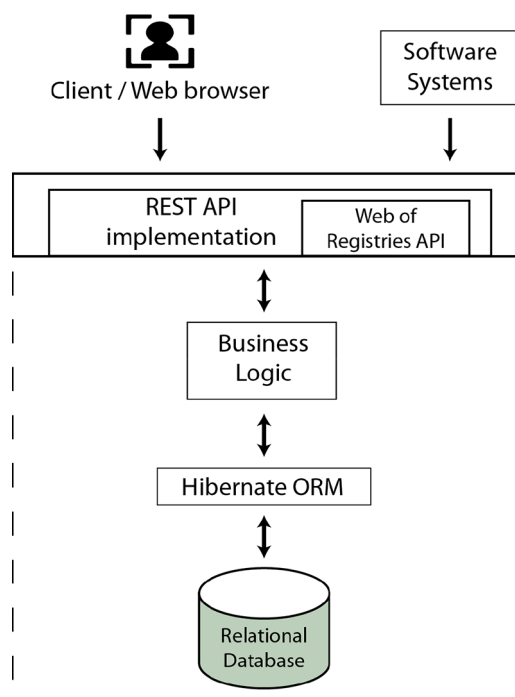
## MATERIALS AND METHODS

**Technical Details.** Since the initial ICE publication, there have been three major releases of the application. During this interval, ICE has undergone a significant architectural overhaul and user interface redesign, and has accrued several noteworthy new features (Figure 1). The current version, 5.9.0, incorporates the following major improvements:

1. The embedded graphical applications, formerly in Adobe Flex, have been reimplemented in JavaScript, resulting in broader device support, none of the associated security vulnerabilities, and improved performance.
2. The ICE API was reimplemented using a REST-style architecture to replace BlazeDS and Simple Object Accessible Protocol (SOAP).
3. A distributed platform to enable biological part sharing has been built into the application, enabling sharing and collaborative features across ICE instances.
4. A complete overhaul of the application's software architecture brings it up to modern standards and introduces several new features (detailed below).

The BioParts search portal application comprises four major components (Figure 2):

1. The Indexer, which is responsible for retrieving information on available biological parts from various data sources. These sources can include ICE, SynBio-Hub, and iGEM parts registries, as well as the NCBI, Addgene, GenScript, or other parts databases. It creates a locally stored and searchable index which is updated at



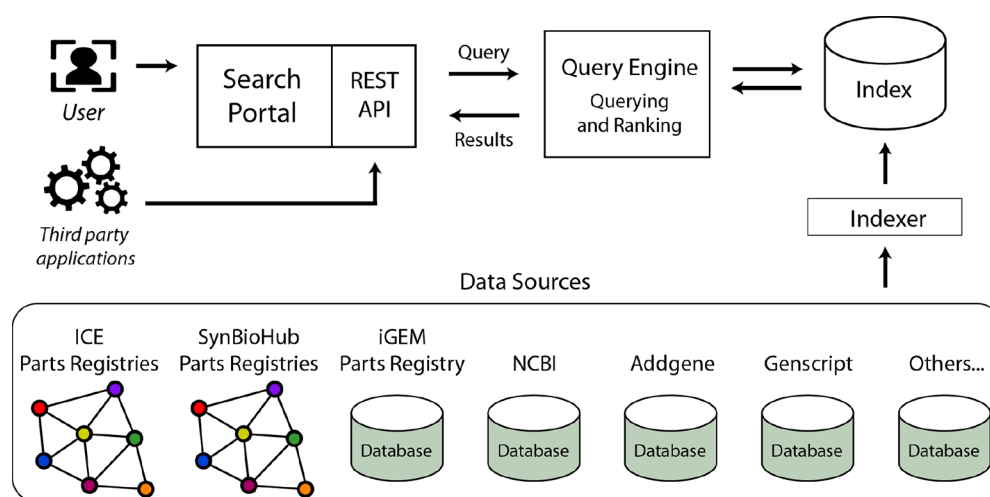
**Figure 1.** Schematic of the ICE system architecture, which utilizes a standard frontend-backend approach for web services. The frontend makes the REST API available to web browsers via the search portal web interface, and directly to nonbrowser-based software systems. The backend implements the business logic and communicates with the database persistent storage system via the Hibernate Object Relational Mapping (ORM).

least once a month. This searchable index is used by the web application to respond to user search requests.

2. The Query Engine, which is responsible for responding to query requests, processing them using the available index to find potential matches, and returning those to the user. The Indexer avoids duplicating the sequence information in the local data store, while the Query Engine pulls the additional information from the source of the data when a user requests details about a search result.
3. The Search Portal web application, which offers a user-friendly interface to perform queries. The portal runs in a web browser and does not require any programming expertise or technical know-how to use.
4. A REST API, which is made available for third-party tools and applications that wish to integrate with the portal's data discovery and transformation features.

**Software Availability.** ICE is an open-source application with a BSD license. The code is freely available on GitHub's online repository, at [github.com/JBEI/ice](https://github.com/JBEI/ice). There is a running instance available to the general public at [public-registry.jbei.org](http://public-registry.jbei.org). Anyone with an e-mail address can register for an account and use the application. Detailed user and API documentation is available at [ice.jbei.org](http://ice.jbei.org). For implementation details, we refer to the documentation and source code available at [github.com/JBEI/ice](https://github.com/JBEI/ice). BioParts is available as a web based application and hosted at [bioparts.org](http://bioparts.org). API documentation is also available at [bioparts.org/api](http://bioparts.org/api).

**Security and Sharing.** ICE supports two forms of user authentication: The Lightweight Directory Access Protocol (LDAP), and local authentication. LDAP support allows users



**Figure 2.** Diagram of the [bioparts.org](https://bioparts.org) search portal architecture. Queries from users or third-party applications are consumed by the search portal/REST API and sent to the query engine for processing. The engine consults the index and returns ranked results to the user or application via the portal or API interface. The indexer asynchronously and periodically updates the index based on data available from the various sources.

to log in using credentials managed by an external LDAP server; local authentication is available for situations where it is desirable to have the ICE instance manage user credentials. Security within the web of registries data exchange revolves around the use of secure tokens. When communicating with each other, instances can serve as both the client/user or the server, depending on the context. ICE instances never directly store the tokens; instead, hashed tokens are used to authenticate against whatever the client sends. The use of secure tokens is the primary means of maintaining security in the web of registry data exchange. Once a new repository requests access to the web of registries, each prior instance generates a new secure token specifically for that new instance. All data exchange between the repositories relies on the verification and validation of these secure tokens. The details of how the tokens are generated and used to secure communications between instances in the web of registries configuration are available on [ice.jbei.org](https://ice.jbei.org). ICE is implemented to align with the FAIR data framework and guiding principles.

## RESULTS AND DISCUSSION

**Organization of ICE Components.** Previously, we developed the ICE biological parts registry software platform to provide part storage functionality for synthetic biology research projects.<sup>6</sup> The initial ICE registry provided an open-source software library stack consisting of part storage, retrieval, manipulation, and machine accessibility via an API interface. At the time of the original ICE publication,<sup>6</sup> the release version was 3.1.0, which offered three primary methods of ICE access. The first version was via the web UI, which connected to the server instance through the Application Binary Interface (ABI). In the second and third versions, BlazeDS and SOAP services provided automated and programmatic access to the ICE API. The need for two separate mechanisms for programmatic access resulted from the embedded VectorEditor components, which were implemented as an Adobe Flex Rich Internet application. As a result, the ICE application required maintenance of these three separate access mechanisms.

Subsequently, the ICE registry application has been rearchitected using the client-server model (Figure 1). The

client side of the application is implemented with the Angular framework using HTML, CSS, and TypeScript and is intended to be run in a web browser. The server side of the application is developed in Java and offers an API implemented using the REST architectural style. ICE can support many popular relational databases (e.g., PostgreSQL, MySQL, Oracle) via the use of the Hibernate Object-Relational Mapping (ORM) framework. Communication between the client and server occurs over HTTP with JavaScript Object Notation (JSON) as the data exchange format. The client interacts with the server by connecting to the REST API and sends and receives data using the JSON format.

**JavaScript Implementation.** Earlier versions of ICE provided a number of graphical embedded applications written in Adobe Flex, based on the Adobe Flash platform, to enable sequence visualization and editing in real-time. These were referred to as the VectorEditor suite of components and consisted of the Sequence Checker, Vector Viewer, and Vector Editor. They formed a critical piece of the ICE platform and were also made available on GitHub as a standalone open-source DNA vector display and manipulation application. Issues with Flash—related to security risks, excessive computing resource requirements, and instabilities in the platform—resulted in significant web browser vendors' decision to drop support for the technology. Consequently, dependency on the Flash plugin was removed from ICE by replacing the VectorEditor suite of components with an open-source implementation developed by TeselaGen Biotechnology and made available at [github.com/TeselaGen/openVectorEditor](https://github.com/TeselaGen/openVectorEditor). Referred to as OpenVectorEditor, it was developed to include and expand the functionality and feature set offered by the Adobe Flex version of VectorEditor, but written in the modern React JavaScript framework. As such, third-party web plugins are no longer required by users of ICE to visualize, annotate, and manipulate their DNA sequences. Additionally, OpenVectorEditor introduced several new features and, more importantly, offered continued community support in the form of bug fixes and improvements (these aspects are discussed in greater detail below).

**REST API Implementation.** In version 3.1.0 of ICE,<sup>6</sup> the API was implemented via the SOAP data exchange format. Additionally, each ICE entry was assigned a Universally

Name*	Alias	Principal Investigator*	Creator*	Circular	Status*	BioSafety Level*	Origin of Replication	Promoters	Replicates In	Funding Source	Backbone	Keywords	Selection Mark
1 pak3-CopR1598		Ancilla Mukhopadhyay	Megan Garber	<input checked="" type="checkbox"/>	Complete	Level 1	ColE1	T7		ENIGMA	psbt3	Psst_1588	Kan
2 pGAPDH-Rt_AmFAR1-NAT(pGEN-291)		John Gladden	Gina Geiselman	<input checked="" type="checkbox"/>	Complete	Level 1					pG2		kanamycin, nourseotricin
3 pANT-GFKS_opt-NAT-pTEF1-RBTS1	pGEN-362	John Gladden	Xun Zhuang	<input checked="" type="checkbox"/>	Complete	Level 1	pVS1	ANT	E. coli, Agrobacterium tumefaciens, Rhodospiridium rubroideum	ABF	pANT-NAT-pTEF1_with_mod		Kanamycin
4 ANT-KanMX-with Mod (pGEN-292)		John Gladden	James Kirby	<input checked="" type="checkbox"/>	Complete	Level 1		ANT (was originally labeled ACPI1 but is actually ANE1, RTOK_12736)					
5 pmsf125	pmsf126	Dominique Loque	Patrick Ohn	<input type="checkbox"/>	Complete	Level 1	ColE1				pBcaB145	(P_AICesA7)	Carbenicillin

**Figure 3.** Bulk manipulation of biological parts within ICE via the built-in spreadsheet-like interface. In this example, plasmids are being uploaded; the system also supports bulk manipulation of the strain, (DNA) part, plant seed, and protein entry types. Detailed information about each entry includes its Name, the project PI, the entry Creator, its Status, Biosafety Level, and Selection Markers.

Unique Identifier (UUID) to enable distributed use. Currently, ICE incorporates the REST architectural style Web API, a simpler implementation and maintenance alternative to the more complex SOAP API, which also offers greater flexibility and scalability than SOAP. REST is currently considered the *de facto* standard for designing web services to facilitate communication across the internet. It offers support for all standard HTTP methods (e.g., GET, POST, PUT). Thus, instead of maintaining separate data-access end points for each access type (web UI and third-party applications), the REST API functions as the single end point for all data access. Unlike SOAP (which required XML for the data-interchange format), REST offers the flexibility of using JSON, XML, PDF, or other formats to represent resources, depending on the specific client requests or needs. Using a REST API also makes it easy to retrieve, share, and protect resources across ICE instances using HTTP methods. This makes implementing the Web of Registries concept an easier task than using SOAP or another similar protocol.

**New Features. Bulk Biological Part Manipulation.** As synthetic biology part design and build processes have become high-throughput, the original ICE part creation feature became a bottleneck. The current version of ICE supports bulk creation and editing of any number of parts. Users have a choice of using either the spreadsheet-like interface to enter information about the parts (Figure 3), or to upload an actual spreadsheet that contains this information. Additionally, to further the sharing and organizing of parts, users can create folders for any number of parts, move parts between them, or remove them from a folder. Users can also share folders with other users, and any parts contained within these folders inherit the permissions of that folder.

**Improved Real-Time Sequence Visualization and Editing.** The previous Flash-based sequence visualization and manipulation tools were replaced by OpenVectorEditor, an open-source tool suite implemented and maintained by TeselaGen Biotechnology (<https://github.com/TeselaGen/openVectorEditor>). It utilizes an industry-standard web framework that is fully supported by all modern browsers. Also, it is designed to be embeddable in different applications with customizable views (Figure 4).

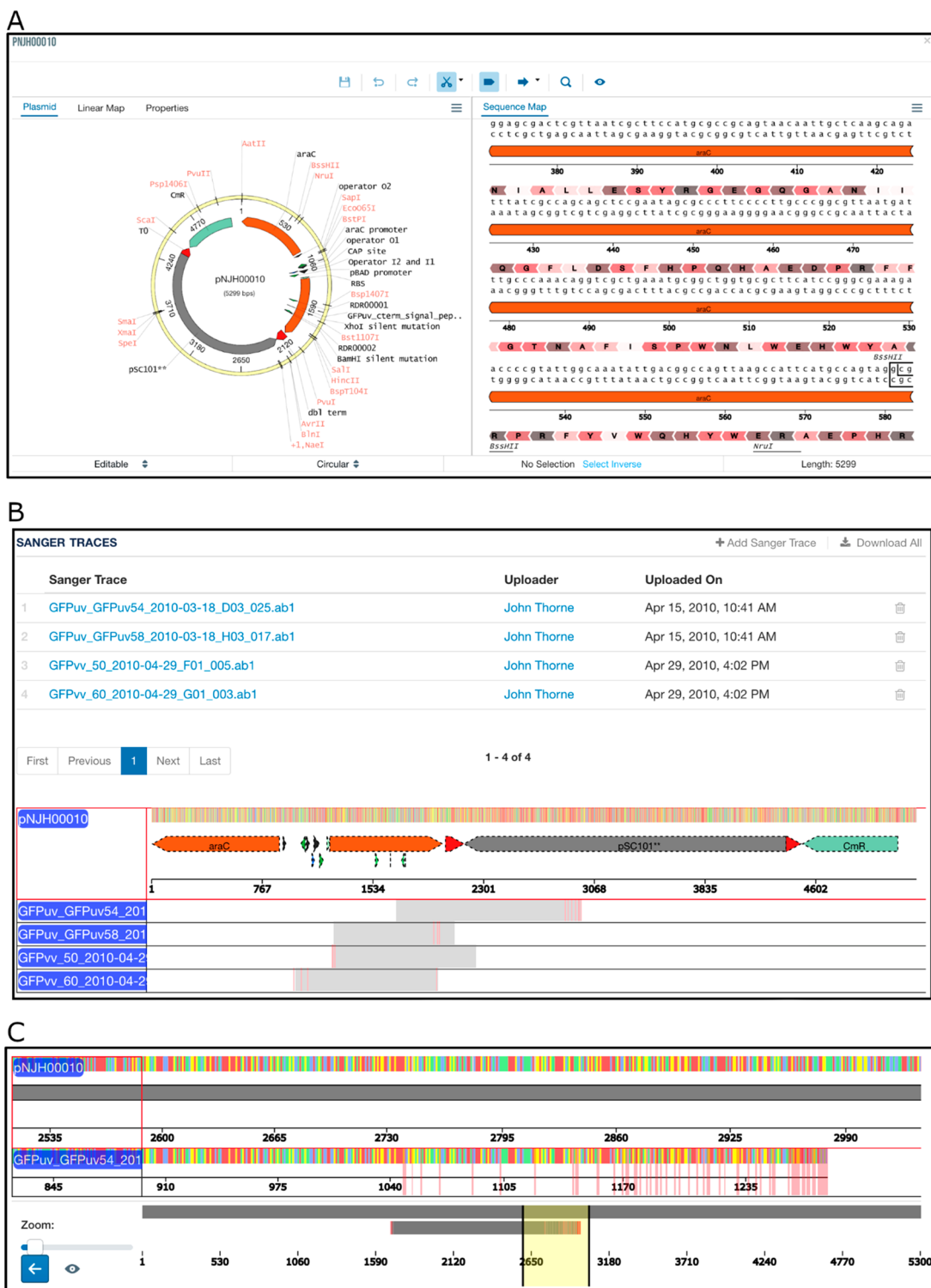
**Automatic Sequence Annotation.** To fully understand the functionality of a DNA sequence, it is important to identify the portions that are of interest. Annotating a DNA sequence is, therefore, an essential step in research activity, but manual sequence annotation is still a very tedious and time-consuming activity, subject to the potential errors of any “by hand” process

and creating a bottleneck for high-throughput workflows and pipelines. Recently, McGuffie et al. developed an open source web server to help annotate plasmids automatically.<sup>10</sup> BLAST has been integrated with ICE to allow users to automatically annotate a sequence of interest, based on publicly accessible annotated sequences that are available in the web of registries and other data sources, to increase the quantity, accuracy and quality of available annotations.

**Sample Tracking.** To aid institutional Laboratory Information Management System (LIMS) efforts, ICE provides built-in support for tracking the locations of physical samples within a lab or organization (Figure 5A,B). These locations depend on the type of samples. For example, the locations can be vials contained in 96-well plates contained in freezers in a predetermined location in a room or building. Users can request available samples from within the system using “Add to Cart” functionality identical to that found on e-commerce sites. Once users request an available sample, a notification is sent to a predetermined group of users available to respond to such requests. The real-time status of the request response is then visible to the user. Support for samples stored in community resources such as Addgene and GenScript is also available (Figure 5C,D). Users are thus able to indicate that their biological parts are available at these repositories and provide direct links to them from within the ICE application.

**Other Features.** Additional significant new features include the following:

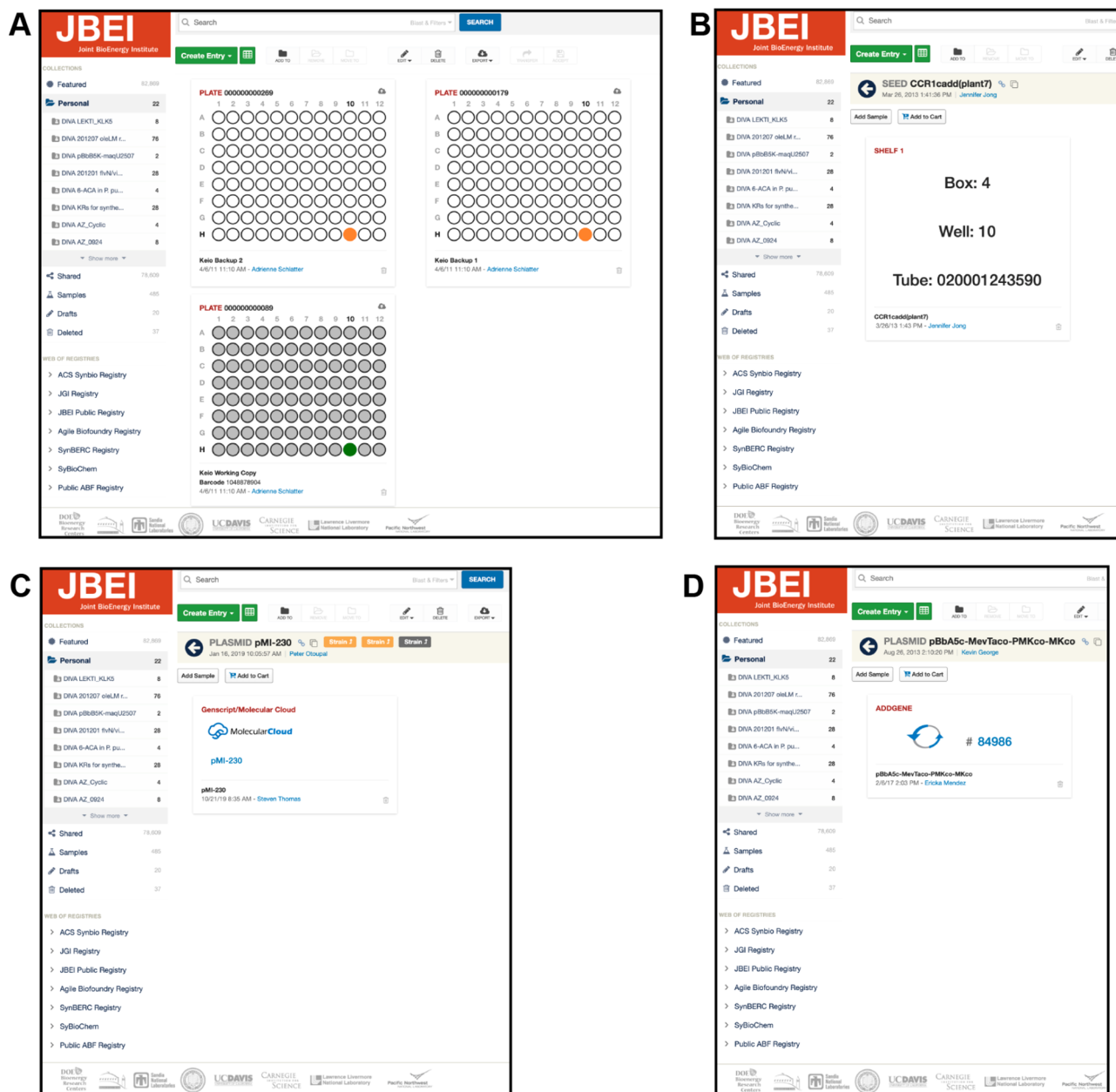
- support for hierarchically structuring the DNA components. As such, a generic part can be shown to be (physically) contained in one or more parts or plasmids, which can then be shown to be contained in one or more strains;
- a workflow to enable parts review in the context of manuscripts and publications, and make them publicly available once accepted for publication;
- an audit history showing users the operations (e.g., sequence modification or metadata edits and additions) performed on an entry since it was uploaded;
- storage of next-generation sequencing data (e.g., Illumina/PacBio) which can be downloaded and visualized using IGV software;
- integration with Experiment Data Depot<sup>11</sup> studies to track experiments using ICE parts;
- support for sequence import/export in GenBank, FASTA, SBOL, and GFF3 formats;



**Figure 4.** OpenVectorEditor tool suite in ICE. (A) The Vector Editor component offers a full-screen visualization of the sequence, enabling users to edit and manipulate it with real-time visualization of changes that can then be saved back to the embedding application. (B) The Sequence Alignment view enables Sanger traces to be aligned against the reference sequence with areas of mismatches highlighted to facilitate sequence analysis. (C) The pairwise alignment view enables a detailed view of the areas of mismatches between a single Sanger trace sequence and the reference sequence. ICE can also store next-generation sequencing data (e.g., Illumina/PacBio) which can be downloaded and visualized using IGV software.

- support for custom fields. With this feature, administrators of individual ICE instances can create and customize the input fields for the forms used to enter data into the registry without making code changes. As

such, institutions can specify which aspects of biological parts are essential to them and curate data entry to match those needs.



**Figure 5.** Sample visualizations of the ICE registry detailing sample storage support for (A) 96 well plates; (B) generic locations; (C) GenScript samples; (D) Addgene samples. Clicking on the Addgene and GenScript numbers will take you to the record on [addgene.org](https://www.addgene.org) and [genscript.org](https://www.genscript.com), respectively.

**Web of Registries: A Distributed Platform for Sharing Biological Parts.** One of the most significant improvements in ICE is the implementation of a distributed software platform to enable sharing biological parts across different ICE instances. This integrated platform enables ICE instances to connect to each other and form a global distributed parts database. Referred to as “web of registries”, the global distributed parts database forms an abstract layer on top of enjoined ICE instances to enable parts sharing in a public or private manner, publishing, and discovery using sequence fragments or search text. The web of registries implementation relies on point-to-point connections with other instances to retrieve data, in addition to a local search index for the faster presentation of results to the user. Through the web of registries, parts developed by any group in one part of the

world can be made available to the broader scientific community for use. Likewise, API access provides interactions with third-party tools and registries.

The web of registries infrastructure is built into the core ICE code base—no custom additions, add-on modules, or other additional code configurations are needed to access it. From the ICE installation administrator’s perspective, joining the web of registries is as simple as toggling an “Enable” button on the settings page (Figure 6A). This sets off the following series of actions by that ICE instance: the application contacts an administrator-configurable trusted registry instance (which defaults to [registry.jbei.org](https://registry.jbei.org)) and retrieves a curatable list of all other ICE instances that are in the web of registries configuration. For each ICE instance available, the new ICE instance connects to it and exchanges a secure and unique API

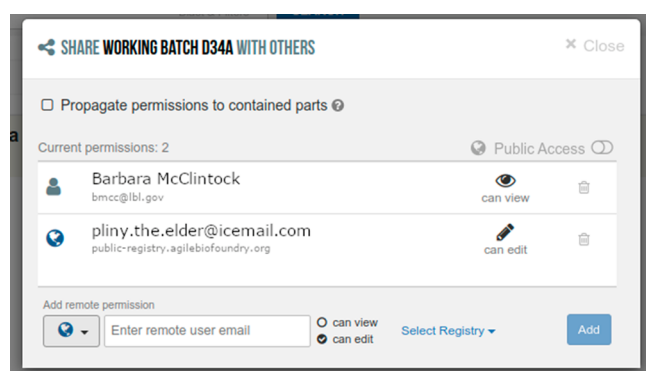




available under a “WEB OF REGISTRIES” section on the application’s menu page. This allows users to browse the publicly available entries on those instances from within their instance (Figure 6B). The Web of Registries platform also enables entries to be transferred from one instance to another. For example, the Joint BioEnergy Institute maintains two ICE instances: a private and a public instance. Access to the private instance is restricted to employees and researchers only and houses biological part data being actively researched. Once this data is ready to be made available to the public (usually as part of a research publication), it is transferred to the public instance, which is available to the general public. In keeping with ICE’s philosophy of allowing each instance to have complete and fine-grained control over its local data, entries have to be made “public” by their respective owners in order for them to be publicly accessible on the web. This ability by owners to make entries publicly accessible can be restricted even more by toggling an “Admins only” button on the Web of Registries settings page to only allow users with administrative roles to perform the transfer. The flexibility offered by these access controls allows institutions that prefer to maintain individual control of their own local instance the ability to do so, while offering the opportunity to join with other instances in a “web of registries” configuration and act like a single global instance. Other features of the web of registries, such as browsing entries on the web, are not affected. Even when entries are shared with other users, copies cannot be transferred from one ICE instance to another except by an administrator. An administrator on the receiving instance must then manually accept the transferred entries before they are visible to the general users on the receiving instance.

This decentralized method of connecting to other ICE registries is preferred to a centralized implementation where a single service has to approve each registry seeking to join the web, as the latter can be a single-point-of-failure. Also, different instances of web of registries can be formed and operated independently of each other.

**Private Remote Collaboration.** Private Remote Collaboration is a new feature made possible by the web of registries platform. It enables privately sharing entries on an ICE instance with specific users on another ICE instance. Assigning access privileges to users on a remote instance can be combined with access privileges assigned to users on the local instance (Figure 7).



**Figure 7.** Permissions dialogue box showing a folder being shared with a user on the local ICE instance and one on a remote ICE instance ([public-registry.jbei.org](http://public-registry.jbei.org)).

Users on remote ICE instances can retrieve these shared entries and interact with them in the same manner they interact with their entries on their local ICE instance (Figure 8). Only a single copy of the shared entries is stored (on the instance it is being shared from), which allows the owner to modify or revoke the entry’s access privileges at any time.

**BioParts Search Portal.** The Web of Registries concept aims to accelerate and broaden the distribution and discovery of biological parts data for biotechnology research applications. We have described an earlier implementation of this concept as part of the ICE parts repository software platform. However, this comes with some limitations. Here, we describe a broader implementation as a search portal, referred to as BioParts Search (BPS). It is available at [bioparts.org](http://bioparts.org), and it allows anyone to search for publicly accessible biological part information from over 10 sources such as NCBI, SynBioHub, Addgene, and parts publicly accessible through ICE Registries. The search portal combines the ease and convenience of modern web search engines with the capabilities of bioinformatics search tools such as BLAST. Users can search for available parts using keywords or sequence fragments. Clicking on a particular search result reveals more detailed information about the entry and supplements this detail with visualization and other tools that leverage features found within ICE. These tools include OpenVectorEditor sequence visualization (Figure 4) and the ability to convert sequences from one format (e.g., GenBank) to another (e.g., SBOL).

Users can visit [bioparts.org](http://bioparts.org) and use search terms (Figure 9A) or sequence fragments to identify available matching parts. We leveraged developments in ICE and the Web of Registries concept to include a number of capabilities in BPS that offer a uniform user experience when interacting with biological parts. An example of this includes the integration of the OpenVectorEditor plugin that enables users to interact with the sequences for discovered parts dynamically (Figure 9B). This plugin offers the ability to visualize the features, restriction, and cut sites and display protein translations and open reading frames all in real-time. This offers a consistent user experience even when the source of the data does not offer such tools. Additionally, the available BPS API can be used to enhance the feature-set of other synthetic biology tools. Search results from BPS can be integrated into BioCAD applications to locate and incorporate parts into designs. BPS offers an autoannotation feature that enables users to enter a sequence of base pairs, have it annotated based on the indexed data, and makes available for download a genbank file containing the user-selected annotations (Figure 10). There is also an API available for parts registries to make this autoannotation feature available.

## CONCLUSION

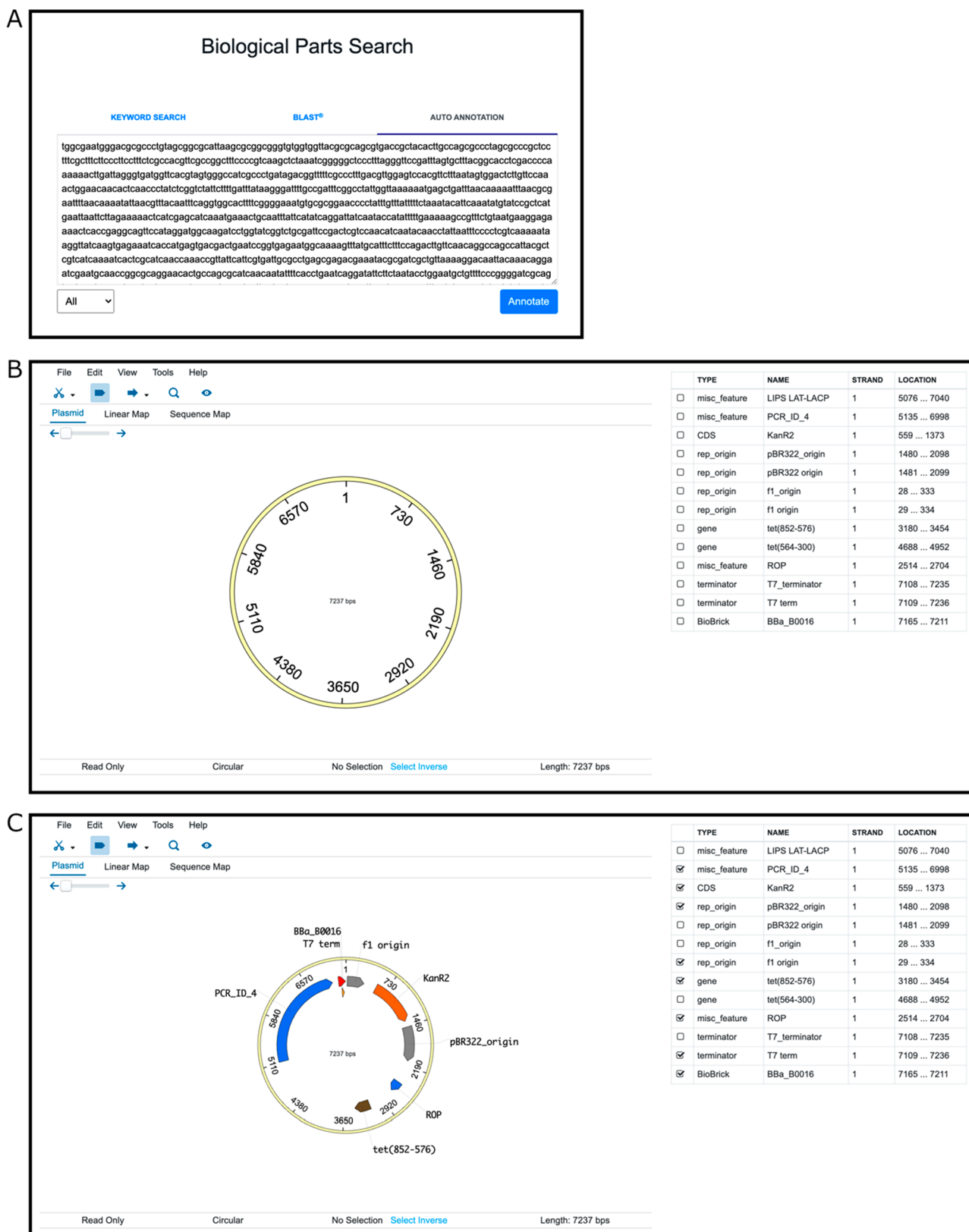
Well-organized, accessible biological parts information is a critical component of biotechnological research at scales far beyond the individual researcher.<sup>7,12</sup> As computational, automated “Design” and “Build” processes are developed (as part of the Design-Build-Test-Learn cycle), seamless interoperability between these tools becomes critical—to minimize bottlenecks, to guide rapid prototyping of engineered organisms, and to inform subsequent “Learn” processes.<sup>13–16</sup> The field needs efficient capturing, sharing, and reuse of genetic parts and designs between researchers and laboratories to enable this. In 2012, the original ICE parts repository provided important improvements in how parts for synthetic

Figure 8. View of the shared folder from the JBEI private registry on the JBEI public registry.

Figure 9. bioparts.org Web of Registries BPS portal. (A) The interface presented to the user is similar to that of other modern search engines. (B) Search results provide sequence, metadata, and annotation information, leveraging the OpenVectorEditor plugin.

biology research were described, communicated, tracked, and shared within an organization. This update to the ICE platform, and implementation of the web of registries concept,

provide significant improvements to collaborative synthetic biology research. The ICE platform has been updated both on the front end, with tools such as the implementation of



**Figure 10.** [bioparts.org](https://bioparts.org) autoannotation feature available on the web interface. (A) Users are able to enter a sequence they wish to annotate. (B) Matching annotations that can be applied to the sequence are presented to the user along with a visual representation of the sequence. (C) Live updates of the visual representation, as the user selects matching annotations.

OpenVectorEditor and shareable folders, and on the back end, with the modernization of the software implementation and the API. The BioParts Search application's introduction also

serves to meet the hitherto unmet need to make publicly available parts accessible conveniently, using modern search tools and techniques and visualization plugins such as

OpenVectorEditor. Future work will include expanding the information rendered in [Bioparts.org](https://bioparts.org) for a given entry and implementing the OAuth2.0 authorization framework, the industry standard for granting access to a resource or application, enabling authorization by third-party applications to the ICE REST API. Additional features and upgrades currently in development include upgrading the UI framework to the latest Angular, version control, provenance tracking of entries, and support for Synthetic Biology Open Language (SBOL) in BioParts. These features are slated for the next major ICE release.

## AUTHOR INFORMATION

### Corresponding Authors

**Nathan J. Hillson** – DOE Agile BioFoundry, Emeryville, California 94608, United States; DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Biological Systems and Engineering Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States; [orcid.org/0000-0002-9169-3978](https://orcid.org/0000-0002-9169-3978); Email: [njhillson@lbl.gov](mailto:njhillson@lbl.gov)

**Christopher J. Petzold** – DOE Agile BioFoundry, Emeryville, California 94608, United States; DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Biological Systems and Engineering Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States; [orcid.org/0000-0002-8270-5228](https://orcid.org/0000-0002-8270-5228); Email: [cjpetzold@lbl.gov](mailto:cjpetzold@lbl.gov)

### Authors

**Hector A. Plahar** – DOE Agile BioFoundry, Emeryville, California 94608, United States; DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Biological Systems and Engineering Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States

**Thomas N. Rich** – DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; TeselaGen Biotechnology Inc., San Francisco, California 94107, United States

**Stephen D. Lane** – DOE Agile BioFoundry, Emeryville, California 94608, United States; DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Sandia National Laboratories, Livermore, California 94550, United States

**William C. Morrell** – DOE Agile BioFoundry, Emeryville, California 94608, United States; DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Sandia National Laboratories, Livermore, California 94550, United States

**Leanne Springthorpe** – DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Biological Systems and Engineering Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States

**Oge Nnadi** – DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Biological Systems and Engineering Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States

**Elena Aravina** – DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; Biological Systems and Engineering Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States

**Tiffany Dai** – DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; TeselaGen Biotechnology Inc., San Francisco, California 94107, United States

**Michael J. Fero** – DOE Joint BioEnergy Institute, Emeryville, California 94608, United States; TeselaGen Biotechnology Inc., San Francisco, California 94107, United States

Complete contact information is available at:

<https://pubs.acs.org/10.1021/acssynbio.1c00263>

### Author Contributions

ICE development, H.A.P., W.C.M., S.D.L., L.S., O.N., E.A., and N.J.H.; Web-of-Registries development, H.A.P.; OpenVector-Editor development, T.N.R., and M.J.F.; supervision, C.J.P., N.J.H., and M.J.F.; writing original draft, H.A.P. and C.J.P.; writing-review and editing, all the authors.

### Funding

This work was part of the Agile BioFoundry ([agilebiofoundry.org](https://agilebiofoundry.org)) supported by the U.S. Department of Energy, Energy Efficiency and Renewable Energy, Bioenergy Technologies Office, and was part of the DOE Joint BioEnergy Institute ([jbei.org](https://jbei.org)) supported by the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research, through contract DE-AC02-05CH11231 between Lawrence Berkeley National Laboratory and the U.S. Department of Energy. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility or the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed or represents that its use would not infringe privately owned rights. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (<http://energy.gov/downloads/doe-public-access-plan>). Funding for open access charge: US Department of Energy.

### Notes

The authors declare the following competing financial interest(s): Thomas Rich, Tiffany Dai, Michael Fero, and Nathan Hillson have financial interest in TeselaGen Biotechnology Inc.

ICE is an open-source application with a BSD license. The code is made freely available on GitHub's online repository at [github.com/JBEI/ice](https://github.com/JBEI/ice). Detailed user and Application Programming Interface (API) documentation are available at [ice.jbei.org](https://ice.jbei.org). BioParts is a web based application hosted at [bioparts.org](https://bioparts.org). The API documentation is available at [bioparts.org/api](https://bioparts.org/api).

## ACKNOWLEDGMENTS

The authors thank Chris J. Myers and Sam Coradetti for helpful discussions, and Mark Kulawik for assistance with informatics.

## REFERENCES

- (1) Galdzicki, M.; Clancy, K. P.; Oberortner, E.; Pocock, M.; Quinn, J. Y.; Rodriguez, C. A.; Roehner, N.; Wilson, M. L.; Adam, L.; Anderson, J. C.; Bartley, B. A.; Beal, J.; Chandran, D.; Chen, J.; Densmore, D.; Endy, D.; Grünberg, R.; Hallinan, J.; Hillson, N. J.; Johnson, J. D.; Kuchinsky, A.; Lux, M.; Misirli, G.; Peccoud, J.; Plahar, H. A.; Sirin, E.; Stan, G.-B.; Villalobos, A.; Wipat, A.; Gennari, J. H.; Myers, C. J.; Sauro, H. M. The Synthetic Biology Open Language (SBOL) provides a community standard for communicating designs in synthetic biology. *Nat. Biotechnol.* **2014**, *32*, 545–550.
- (2) Quinn, J. Y.; Cox, R. S.; Adler, A.; Beal, J.; Bhatia, S.; Cai, Y.; Chen, J.; Clancy, K.; Galdzicki, M.; Hillson, N. J.; Le Novère, N.; Maheshwari, A. J.; McLaughlin, J. A.; Myers, C. J.; P, U.; Pocock, M.; Rodriguez, C.; Soldatova, L.; Stan, G.-B. V.; Swainston, N.; Wipat, A.; Sauro, H. M. SBOL visual: A graphical language for genetic designs. *PLoS Biol.* **2015**, *13*, No. e1002310.
- (3) Zhang, M.; McLaughlin, J. A.; Wipat, A.; Myers, C. J. Sboldesigner 2: an intuitive tool for structural genetic design. *ACS Synth. Biol.* **2017**, *6*, 1150–1160.
- (4) McLaughlin, J. A.; Beal, J.; Misirli, G.; Grünberg, R.; Bartley, B. A.; Scott-Brown, J.; Vaidyanathan, P.; Fontanarrosa, P.; Oberortner, E.; Wipat, A.; Gorochofski, T. E.; Myers, C. J. The synthetic biology open language (SBOL) version 3: simplified data exchange for bioengineering. *Front. Bioeng. Biotechnol.* **2020**, *8*, 1009.
- (5) Wilkinson, M. D.; Dumontier, M.; Aalbersberg, I. J. J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.-W.; da Silva Santos, L. B.; Bourne, P. E.; Bouwman, J.; Brookes, A. J.; Clark, T.; Crosas, M.; Dillo, I.; Dumon, O.; Edmunds, S.; Evelo, C. T.; Finkers, R.; Gonzalez-Beltran, A.; Gray, A. J. G.; Groth, P.; Goble, C.; Grethe, J. S.; Heringa, J.; 't Hoen, P. A. C.; Hooft, R.; Kuhn, T.; Kok, R.; Kok, J.; Lusher, S. J.; Martone, M. E.; Mons, A.; Packer, A. L.; Persson, B.; Rocca-Serra, P.; Roos, M.; van Schaik, R.; Sansone, S.-A.; Schultes, E.; Sengstag, T.; Slater, T.; Strawn, G.; Swertz, M. A.; Thompson, M.; van der Lei, J.; van Mulligen, E.; Velterop, J.; Waagmeester, A.; Wittenburg, P.; Wolstencroft, K.; Zhao, J.; Mons, B. The FAIR Guiding Principles for scientific data management and stewardship. *Sci. Data* **2016**, *3*, 160018.
- (6) Ham, T. S.; Dmytriv, Z.; Plahar, H.; Chen, J.; Hillson, N. J.; Keasling, J. D. Design, implementation and practice of JBEI-ICE: an open source biological part registry platform and tools. *Nucleic Acids Res.* **2012**, *40*, No. e141.
- (7) McLaughlin, J. A.; Myers, C. J.; Zundel, Z.; Misirli, G.; Zhang, M.; Ofiteru, I. D.; Goñi-Moreno, A.; Wipat, A. SynBioHub: A Standards-Enabled Design Repository for Synthetic Biology. *ACS Synth. Biol.* **2018**, *7*, 682–688.
- (8) Mante, J.; Zundel, Z.; Myers, C. Extending SynBioHub's Functionality with Plugins. *ACS Synth. Biol.* **2020**, *9*, 1216–1220.
- (9) Fielding, T. Architectural Styles and the Design of Network-based Software Architectures. Ph.D. Dissertation, University of California Irvine, 2000, pp 5–151.
- (10) McGuffie, M. J.; Barrick, J. E. pLannotate: engineered plasmid annotation. *Nucleic Acids Res.* **2021**, *49*, W516–W522.
- (11) Morrell, W. C.; Birkel, G. W.; Forrer, M.; Lopez, T.; Backman, T. W. H.; Dussault, M.; Petzold, C. J.; Baidoo, E. E. K.; Costello, Z.; Ando, D.; Alonso-Gutierrez, J.; George, K. W.; Mukhopadhyay, A.; Vaino, I.; Keasling, J. D.; Adams, P. D.; Hillson, N. J.; Garcia Martin, H. The Experiment Data Depot: A Web-Based Software Tool for Biological Experimental Data Storage, Sharing, and Visualization. *ACS Synth. Biol.* **2017**, *6*, 2248–2259.
- (12) Beal, J.; Nguyen, T.; Gorochofski, T. E.; Goñi-Moreno, A.; Scott-Brown, J.; McLaughlin, J. A.; Madsen, C.; Aleritsch, B.; Bartley, B.; Bhakta, S.; Bissell, M.; Castillo Hair, S.; Clancy, K.; Luna, A.; Le Novère, N.; Palchick, Z.; Pocock, M.; Sauro, H.; Sexton, J. T.; Tabor, J. J.; Voigt, C. A.; Zundel, Z.; Myers, C.; Wipat, A. Communicating structure and function in synthetic biology diagrams. *ACS Synth. Biol.* **2019**, *8*, 1818–1825.
- (13) Carbonell, P.; Currin, A.; Jervis, A. J.; Rattray, N. J. W.; Swainston, N.; Yan, C.; Takano, E.; Breitling, R. Bioinformatics for the synthetic biology of natural products: integrating across the Design-Build-Test cycle. *Nat. Prod. Rep.* **2016**, *33*, 925–932.
- (14) Carbonell, P.; Radivojevic, T.; García Martín, H. Opportunities at the intersection of synthetic biology, machine learning, and automation. *ACS Synth. Biol.* **2019**, *8*, 1474–1477.
- (15) Carbonell, P.; Jervis, A. J.; Robinson, C. J.; Yan, C.; Dunstan, M.; Swainston, N.; Vinaixa, M.; Hollywood, K. A.; Currin, A.; Rattray, N. J. W.; Taylor, S.; Spiess, R.; Sung, R.; Williams, A. R.; Fellows, D.; Stanford, N. J.; Mulherin, P.; Le Feuvre, R.; Barran, P.; Goodacre, R.; Turner, N. J.; Goble, C.; Chen, G. G.; Kell, D. B.; Micklefield, J.; Breitling, R.; Takano, E.; Faulon, J.-L.; Scrutton, N. S. An automated Design-Build-Test-Learn pipeline for enhanced microbial production of fine chemicals. *Commun. Biol.* **2018**, *1*, 66.
- (16) Chao, R.; Mishra, S.; Si, T.; Zhao, H. Engineering biological systems using automated biofoundries. *Metab. Eng.* **2017**, *42*, 98–108.