

# Chapter 1

## **An Integrated Computer-Aided Design and Manufacturing Workflow for Synthetic Biology**

**Ernst Oberortner, Robert Evans, Xianwei Meng, Sangeeta Nath, Hector Plahar, Lisa Simirenko, Angela Tarver, Samuel Deutsch, Nathan J. Hillson, and Jan-Fang Cheng**

### **Abstract**

Biological computer-aided design and manufacturing (bioCAD/CAM) tools facilitate the design and build processes of engineering biological systems using iterative design–build–test–learn (DBTL) cycles. In this book chapter, we highlight some of the bioCAD/CAM tools developed and used at the US Department of Energy (DOE) Joint Genome Institute (JGI), Joint BioEnergy Institute (JBEI), and Agile BioFoundry (ABF). We demonstrate the use of these bioCAD/CAM tools on a common workflow for designing and building a multigene pathway in a hierarchical fashion. Each tool presented in this book chapter is specifically tailored to support one or more specific steps in a workflow, can be integrated with the others into design and build workflows, and can be deployed at academic, government, or commercial entities.

**Key words** DNA synthesis, DNA assembly, Computer-aided design, Computer-aided manufacturing, Software, Bioengineering, Synthetic biology

---

## **1 Introduction**

The JGI is a user facility providing a suite of genomics technologies capabilities including DNA and RNA sequencing, DNA synthesis, single cell technologies, and metabolomics in support of DOE mission-related scientific research. The JGI's DNA synthesis platform has established various dry- and wet-lab workflows for the design, build, and characterization tasks specific to each user project. These workflows include (a) phenotypic sequence data repository mining, (b) heterologous expression construct design, (c) screening of the sequences against biosecurity guidelines, (d) synthetic DNA requisition, (e) assembly and sequence verification of synthetic constructs or combinatorial libraries using Type-IIs/Golden Gate, Gibson chew-back, or Yeast-based assembly protocols, (f) transformation of the assembled constructs into

the target host organism, and (g) mass spectrometry secondary metabolite detection.

In this chapter we overview the biological computer-aided design and manufacturing (bioCAD/CAM) tools that the JGI has developed and/or uses, supporting the synthetic biologist throughout the design and build stages of an integrated, iterative design–build–test–learn (DBTL) engineering cycle. We demonstrate a common workflow for utilizing the highlighted bioCAD/CAM tools to design and build a biosynthetic pathway, using a hierarchical process. Specifically, we present a repertoire of bioCAD/CAM tools and demonstrate their applicability in a workflow of designing and building a pathway comprising multiple operons and genes. We follow a top-down design process that comprised of two levels, starting with the design of the pathway and the organization of its genes and regulatory elements into operons. Then, we define the assembly and cloning instructions for building the pathway following a Golden Gate or Yeast assembly protocol. Each part of the Golden Gate or Yeast assembly is synthesized and, in case it exceeds the maximum length synthesis, assembled using a Gibson-based assembly protocol.

At the JGI, staff use bioCAD/CAM tools to facilitate the design and build processes in service of the JGI’s DNA synthesis science program (<https://jgi.doe.gov/our-science/science-programs/synthetic-biology/>) enabling scientific users to gain, among others, access to large-scale DNA synthesis and assembly capabilities. The bioCAD/CAM tools described in this book chapter (a) provide web-based user interfaces (UIs), facilitating access and use, (b) can be deployed outside of the JGI, JBEI, or ABF, and (c) are available at no cost to noncommercial users and entities (*see Note 1*).

---

## 2 Materials

In the synthetic biology community, numerous bioCAD/CAM tools have been developed to computationally aid researchers [1, 2]. At the JGI, JBEI, and ABF, we use and develop bioCAD/CAM tools that support researchers throughout iterative DBTL engineering cycles. In this section, we provide an overview of the bioCAD/CAM tools that we have developed in order to facilitate the design and build processes at the JGI, JBEI, and ABF. We also provide a brief insight into our computational biosecurity screening pipeline.

## **2.1 Biological Computer-Aided Design (bioCAD) of Synthetic DNA**

### *2.1.1 Design, Implementation, and Validation Automation (DIVA)*

The Design, Implementation, and Validation Automation (DIVA) platform seeks to integrate bioCAD/CAM tools in order to provide seamless and useful workflows, liberating researchers from building DNA and enabling them to focus on designing and testing their experiments of interest. DIVA’s web-based UI allows researchers to graphically specify the design of synthetic DNA constructs using DeviceEditor [3] and (sub)sequences from ICE registry [4] plasmid or part entries. After finishing the design specification, researchers can submit their designs to a central construction queue, track DNA construction as it progresses, and then (in a few weeks) receive notice that their sequence-verified constructs have been completed. The DIVA platform integrates DeviceEditor, j5, ICE, BLiSS and BOOST, which we explain in more detail below. An instance of DIVA, specifically deployed for the specification of JGI’s designs, is publicly accessible at <https://diva.jgi.doe.gov>.

### *2.1.2 DeviceEditor [3]*

DeviceEditor mimics the intuitive visual whiteboard design process practiced in biological laboratories through a web-based graphical UI. DeviceEditor combines the visual design of combinatorial libraries, direct integration with scarless multipart DNA assembly design automation (j5 [5]), and a graphical user interface for the creation and modification of design specification rules (Eugene [6]). DeviceEditor follows a design paradigm to spatially organize abstractions of biological components. Therefore, DeviceEditor assists the aggregation and arrangement of the DNA sequences of genetic components (ribosomal-binding sites, promoters and terminators, genes, etc.) to be structurally assembled toward a desired functionality. To specify structurally composite designs in a graphical manner, DeviceEditor integrates the palette of glyphs from the SBOL Visual extension (SBOLv) [7], enabling the communication of designs in a standardized fashion. Once users have visually verified that the desired constructs are correctly designed, DeviceEditor can also direct j5 to design downstream automation processes. DeviceEditor is integrated into the DIVA platform and publicly accessible at <https://diva.jgi.doe.gov>.

### *2.1.3 j5 [5]*

j5 is a bioCAD tool to design scarless, multipart combinatorial DNA assemblies using established protocols such as Gibson and Golden Gate assembly. j5 generates cost-optimal designs for these assembly protocols. The j5 DNA assembly design algorithms are generally applicable across broad classes of DNA construction methodologies and complement other DNA assembly design tools, such as DIVA/DeviceEditor.

DeviceEditor (described above) can be used to specify the inputs for j5, including (a) the desired assembly methodology, (b) the biological “parts” to be assembled, (c) the sequential order and direction (forward or reverse) of each of the parts in

the final target construct(s), and (d) biological design specification rules, to constrain the combinations of parts that make up the final target constructs. j5 generates several output files, such as CSV format files describing the designed DNA assembly processes and Genbank (or FASTA or SBOL v1) format sequence files for the assembled constructs. The j5 bioCAD tool is accessible for non-commercial use through <https://j5.jbei.org>, as integrated functionality in DIVA (<https://diva.jgi.doe.gov>), and for commercial use j5 is exclusively accessible through TeselaGen (<https://teselagen.com>).

#### 2.1.4 *Open VectorEditor (OpenVE)*

OpenVE is an open-source, web-based sequence editor for the modification and visualization of circular as well as linear sequences of DNA including their feature annotations. The source code of OpenVE, which is developed by Teselagen Biotechnology Inc., is available at <https://github.com/TeselaGen/openVectorEditor>. We have deployed a stand-alone instance of OpenVE (<https://j5.jbei.org/VectorEditor/VectorEditor.html>), enabling researchers to utilize and experience its functionalities. In addition, OpenVE is integrated into the DIVA design platform for visualizing DNA construct designs and into the ICE registry for visualizing DNA plasmids and parts.

#### 2.1.5 *Build-Optimization Software Tools (BOOST) [8]*

Not every conceivable DNA sequence can be manufactured using today's state-of-the-art synthesizers. Extreme %GC regions and repeating k-mers complicate the synthesis and the quality assurance/control thereof using next-generation sequencing (NGS) technologies. To streamline the transition from in silico design to in vitro synthesis, synthetic DNA sequences need to be verified against such synthesis criteria.

Therefore, we have developed the Build-Optimization Software Tools (BOOST) that prepare designed DNA sequences for synthesis and assembly. BOOST comprises the following “apps”:

- The “Juggler” reverse-translates protein sequences into DNA sequences and codon-juggles DNA sequences.
- The “Polisher” verifies DNA sequences against DNA synthesis constraints and modifies the protein coding regions to resolve eventual violations of DNA synthesis constraints. Recently, we incorporated the APIs of commercial DNA synthesis vendors to ensure consistency among the complexity screening results of the vendors' portals and BOOST.
- The “Partitioner” fragments large DNA sequences into synthesizable building blocks with assembly-specific overlap sequences.
- The “Workflow Manager” combines the execution of the Juggler, Polisher, and Partitioner functionalities in one pass.

- The “Codon Usages Tables Merger” provides the functionality of combining two codon usage tables into one table when genes need to be expressed in multiple hosts.
- The “Builder” app allows for the specification of step-by-step instructions for manufacturing (synthetic) DNA constructs, such as the steps of purchasing (synthetic DNA fragments or primers), linearizing vectors (digestion or amplification), or PCR-amplification.

Researchers use BOOST to prepare designed constructs, such as genes, operons, or pathways, for synthesis by commercial DNA synthesis vendors and for the in-house assembly of the synthetic DNA fragments. Depending on the BOOST app and the inputs of the user, BOOST generates various output files. For example, the “Workflow Manager” can output ready-to-order synthetic DNA fragments, primers for PCR-amplification of either the synthetic DNA fragments or vectors, and step-by-step instructions for carrying out the tasks of DNA synthesis, assembly and transformation into target host organism. BOOST is publicly accessible at <https://boost.jgi.doe.gov>.

## **2.2 Biosecurity Screening of Synthetic DNA**

In 2010, US Department of Health and Human Services (HHS) issued the *Screening Framework Guidance for Providers of Synthetic Double-Stranded DNA*, outlining recommendations for synthetic DNA providers to ensure that existing regulations and best practices are followed in addressing biosecurity concerns. The HHS guidance defines a “*sequence of concern*” as sequences “derived from or encoding” entities on the HHS and Center for Disease Control’s Select Agents or Toxins list, which are regulated under the Select Agent Regulations, and/or agents on the Bureau of Industry and Security’s Commerce Control List (CCL) regulated under Export Administration Regulations. In accordance with the HHS guidance, the DNA Synthesis Science program at the US Department of Energy Joint Genome Institute (DOE JGI) has developed a computational pipeline—Black List Sequence Screening (BLiSS)—for screening all sequences that it manufactures either for internal research or for its scientific users.

As specified in the guidance, BLiSS detects “*sequences of concern*” of at least 200 nucleotides in length on either DNA strand. BLiSS follows a “Best Match” approach to determine whether a query sequence is unique to Select Agents or Toxins, or export-controlled agents, toxins or genetic elements, and to minimize false positives from closely related organisms or highly conserved “housekeeping genes” which do not pose a biosecurity threat.

At the JGI, and more recently at JBEI/ABF, researchers run BLiSS on all constructs that will be synthesized, after they are designed, but prior to building. If a construct is flagged as being a potential sequence of concern, follow-up screening is done to validate the end use of the construct (potentially resulting in contact with the US Federal Bureau of Investigation).

## **2.3 Biological Computer-Aided Manufacturing (bioCAM) of Synthetic DNA**

### *2.3.1 Tracking the DNA Synthesis and Assembly Workflow*

SynTrack is a web-based workflow-driven bioCAM platform that manages and tracks the DNA manufacturing processes. SynTrack imports the BOOST-specified build process information, which includes step-by-step instructions for staff (leveraging robotic platforms) to carry out all build operations.

Managing complex hierarchical, multistep DNA manufacturing processes requires (a) to define time- and cost-efficient synthesis and assembly strategies, (b) to manage the interplay between humans and automation technologies, such as liquid-handling robotics or microfluidic devices, (c) to assure the quality of the input/output components at well-defined stages in the process through adequate quality control (QA/QC), and (d) to track each step in the process including the provenance of each step's input components, such as DNA fragments/plasmids, master mixes, buffers, or enzymes.

SynTrack supports biologists in carrying out complex multistep processes that (a) begin with purchasing synthetic DNA fragments from commercial DNA synthesis vendors, (b) followed by preparing the received fragments for assembly using PCR amplification, (c) then setting up reactions based on the assembly protocol of the amplified constructs using manual labor and robotics platforms, and (d) lastly sequence verifying the assembled constructs using NGS platforms. With composed user-friendly web interfaces, SynTrack divides the routine build process into the “pre-assembly” and “assembly” stages.

The pre-assembly workflow comprises the tasks associated with purchasing DNA from commercial DNA synthesis vendors, including the tracking of the order and receiving date and the quantity of the synthetic DNA parts. As soon as all required synthetic DNA fragments are available, SynTrack can generate pipetting instruction for liquid handling robotics, such as the Biomek FX platform and Echo liquid handler to rearray received DNA fragments into the desired quadrant layout automatically. Prior to quantifying DNA fragments after cleanup, other tasks for the pre-assembly workflow include creating a set of instructions to handle liquid transfer on the condensed or uncondensed assay plates for PCR reactions.

SynTrack enables instantiation of one or more assembly workflows based on previously finished single or merged pre-assembly workflows. The assembly workflow comprises the tasks of (a) assembly and cloning using Gibson chew-back, Type-IIs/Golden Gate or Yeast-based assembly and cloning reactions, and (b) transforming the assembled and cloned plasmids into the target host and screening colonies for the presence of the transformed plasmids. To support biologists throughout the assembly process, SynTrack supports generating instructions for robotics platforms.

Throughout the entire manufacturing process, SynTrack effectively incorporates QA/QC results in order to maximize the

success rate of synthetic DNA constructs that are delivered to the scientific users. For example, SynTrack enables rework on constructs that fail on synthesis by, for example, purchasing the synthetic DNA fragment from a different vendor. Integrating QA/QC outcomes from colony pickers, plate readers and NGS sequencers, the tracking system performs related measurements on targeted master plates to monitor coordinated DNA construct changes subsequently. All DNA assembly data tracking has been condensed into secure, tabular-formatted web interfaces with the ability to observe and update the status of DNA constructs with time stamp. At the core of an effective quality control assembly processes, troubleshooting needs to be managed for altering the DNA assembly workflows. The workflow task supplies a flexible method based on the feedback of the QA/QC results to allow the workflow moving into backward direction when rework is an expected occurrence for troubleshooting.

### *2.3.2 Sequence Verification Using Next Generation Sequencing (NGS)*

For the verification of manufactured DNA constructs, we utilize DOE JGI's state-of-the-art next-generation sequencing (NGS) platforms. We developed a computational pipeline, SynBioQC, to process the raw reads of NGS sequencers in order to sequence verify the correctness of the manufactured DNA. That is, the SynBioQC pipeline ensures that the nucleotides of the in silico designed sequence matches the in vitro sequenced DNA.

To execute the SynBioQC pipeline, a user enters the sequences of the manufactured constructs and the library names of the sequenced construct pools into a web-based UI. Then, executable scripts are generated and sent to a job management system, which controls the execution of the scripts on a DOE NERSC super-computing cluster. The pipeline aligns the reference sequences with the raw reads and calls variants, such as single nucleotide polymorphism (SNP), insertions and deletions (indels). The results of the pipeline are then visualized in a web page, enabling the researchers to analyze the results in an efficient manner.

### **2.4 Repositories for Storing and Retrieving Synthetic DNA Constructs**

The Inventory of Composable Elements (ICE) [4] is an open-source, web-based registry platform for synthetic biological parts with support for plant seeds, microbial strains, DNA plasmids and parts, as well as protein sequences. ICE provides integrated tools for sequence visualization and editing (openVE, described above), as well as mechanisms for secure access and information sharing with other users and software tools using commonly used data exchange formats, such as FASTA, Genbank, and SBOL, as well as a RESTful API. The JGI instance of ICE is accessible at <https://registry.jgi.doe.gov>.

## **2.5 Data Exchange Using a Standardized Format**

We attempt to develop and use software tools that support commonly used data exchange formats and standards. Adopting community-driven standardization efforts, such as the Synthetic Biology Open Language (SBOL) [9], enables us to provide feedback to the community and to contribute to the development of standards accepted by the scientific community.

SBOL provides capabilities to support our approach to provenance tracking for synthetic DNA requisition and synthetic construct assembly. Provenance tracking is paramount for the purpose of managing scientific intellectual property, systematically reproducing the data process, and identifying defective designs that can be replaced with nondefective alternatives. SynTrack and BOOST communicate step-by-step assembly and cloning instructions, enabling SynTrack to determine each step in the manufacturing process and to track the start- and end-time of each step, who carried out the step, as well as the input and output components of each step. SynTrack underpins the transition from the *in silico* design phase to the *in vitro* build stage while aiming to form collaborations for build elements and activities through the SBOL-based community ontology, and to further incorporate provenance tracking standards into our integrated DBTL pipeline. Tracking the provenance of the design and build tasks provides the scientific user not only with the desired synthetic DNA or strains but also with a detailed, standardized report about what steps were performed to ultimately build the designed DNA constructs. Several of the bioCAD/CAM tools developed at the JGI, JBEI, and ABF support the input and output of data encoded in the SBOL standard by utilizing the SBOL libraries [10].

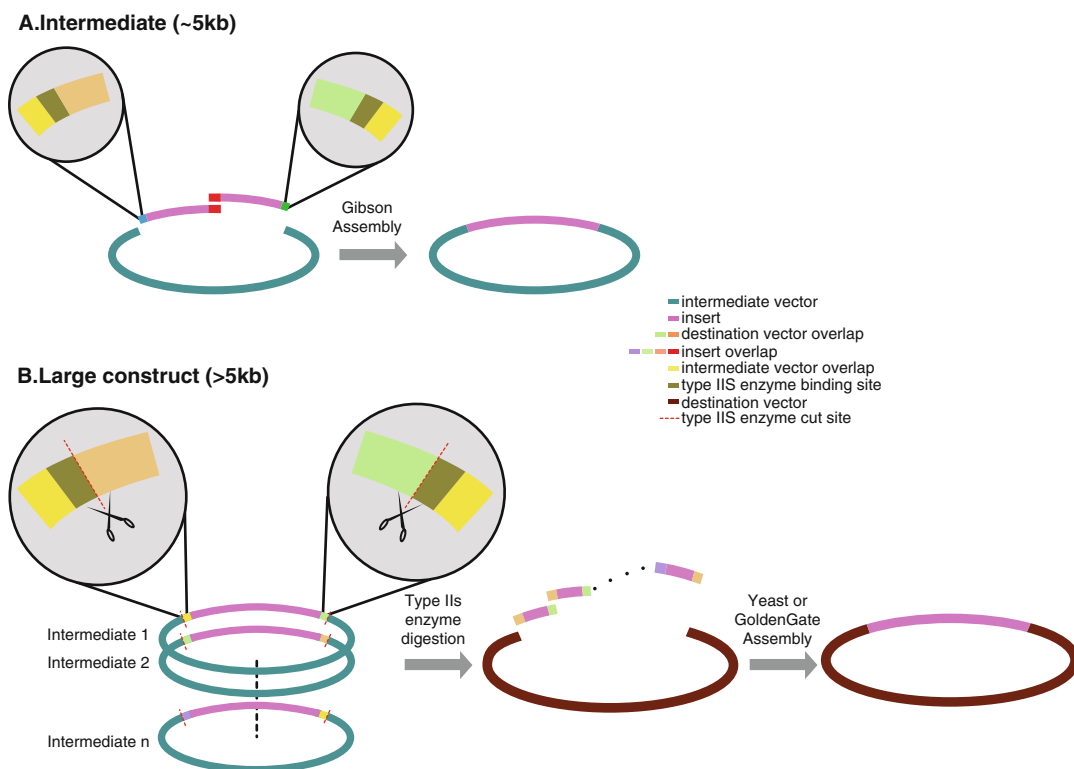
---

## **3 Methods**

In this section, we showcase the applicability of the bioCAD/CAM tools described above, demonstrating their use in a state-of-the-art workflow of the JGI's DNA synthesis platform. The illustrated workflow covers the stages of top-down design and bottom-up building of a synthetic biological system. The system under consideration is a 58 kb pathway consisting of five genes organized in five operons.

In Fig. 1, we outline the hierarchical bottom-up process of synthesis, assembly, and cloning. At the first level, Level-0, "intermediates" are made utilizing Gibson assembly [11] of synthetic DNA fragments with appropriate 5' and 3' overlap sequences for their assembly and insertion into the destination vector (Fig. 1a). In Level-1, sequence verified constructs built in the first level are digested with Type-IIs enzyme and subjected to Yeast [12, 13] or Golden Gate [14] assembly in the desired destination vector to generate the final construct (Fig. 1b).





**Fig. 1** Hierarchical DNA assembly of large constructs using a combination of the Gibson, Golden Gate, and Yeast assembly protocols. Bottom-up process of synthesis, assembly, and cloning of a large synthetic DNA construct (>5 kb). (a) Synthesis and Gibson assembly of intermediate constructs for the construction of large synthetic DNA constructs. (b) Assembly of intermediate constructs into large constructs

In the following sections we describe the top-down design process in a step-by-step manner and the use of bioCAD/CAM tools. The hierarchical design process focuses on the *in silico* design of the synthetic DNA constructs, their synthesis, assembly and cloning instructions (*see Note 2*). Subheadings 3.1, 3.2, and 3.3 focus on the design of the Level-1 construct, that is, the entire pathway, its design for Yeast/Golden Gate assembly, and the required destination vector. Subheadings 3.4, 3.5, and 3.6 focus on the design of the Level-0 constructs, that is, the intermediates, their sequence design for synthesis and chew-back assembly, as well as inserting the Level-0 constructs into the required destination vectors. As a result, the assembly of the intermediates (“Level-0”) leads to the manufactured pathway (“Level-1”). Lastly, we briefly mention tracking the assembly and cloning steps of synthetic DNA constructs.

### 3.1 Design Specification of Pathway Structure and Destination Vector Using DIVA

In the first step, the designer specifies the biological system using the DIVA platform. In Fig. 2 we show a screenshot of the DIVA platform and an excerpt of the design specification of a large pathway. The first column in DIVA’s design canvas represents the destination vector backbone and all other columns denote the



**Fig. 2** Graphical specification of the pathway design using DIVA

insert, that is, the pathway. The bins of the insert respectively specify the 5' vector linker sequences, the five operons of the pathway, each comprised of a promoter, two genes and a terminator, and the 3' vector linker sequence.

In this design scenario, however, we import the sequences of each part using the Genbank format. DIVA lets the user decide to either import the entire sequence of a Genbank file or to specify the start- and end-position of the (sub)sequence that should be imported into DIVA (*see Note 3*).

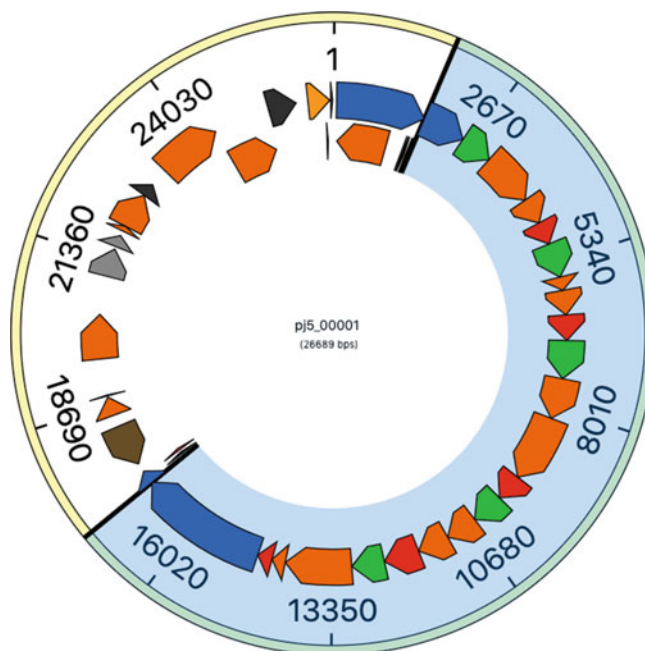
### **3.2 Using j5 to In Silico Assemble the Insert and Its Destination Vector**

In the next step, we utilize j5, taking as input the DIVA specification to assemble the destination vector with the insert (i.e., the pathway) into the final construct. To instruct j5 with the proper assembly strategies, we specify in DIVA that the destination vector backbone will be digested with a Type-IIs restriction enzyme and that the insert (5' vector linker, pathway, 3' vector linker) will be synthesized directly. In Fig. 3 we illustrate a screenshot of openVE, visualizing the final construct, that is, the pathway, which is highlighted, inserted into the destination vector.

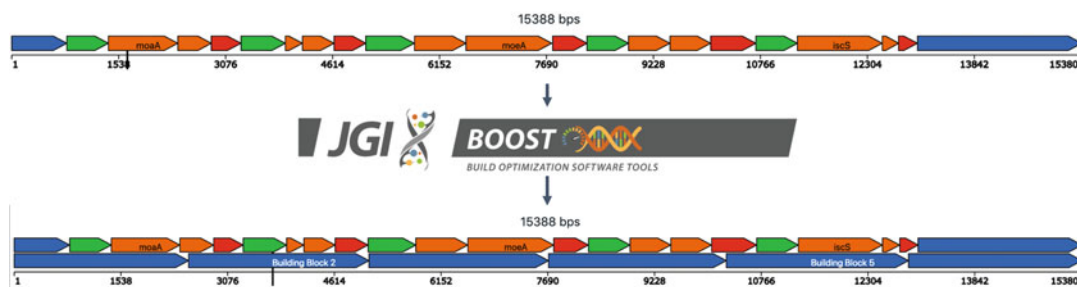
### **3.3 Partition the Insert for Yeast/ Golden Gate Assembly Using BOOST**

Input into BOOST is a GenBank file, generated by DIVA/j5. Depending on the designs' requirements, the user inputs partitioning parameters into BOOST in order to fragment the large pathway sequence into smaller fragments. The Yeast assembly protocol requires, for example, overlaps between 85 and 110 bp with a desired optimum overlap length of 100 bp. In case the build process is based on a Golden Gate assembly protocol, the user currently needs to instruct BOOST to look for proper partitioning areas in the large sequence but not extend the sequence fragments with overlapping sequences. In this case, the user can then utilize j5 for designing the flanking sequences required for subsequent Type-IIs restriction enzyme recognition.

In addition to the partitioning, the user can instruct BOOST to verify the pathway sequence against synthesis criteria by commercial DNA synthesis vendors. In case of violations, BOOST can modify the protein-coding sequences of the pathway to reduce the cost and time of synthesis as well as to increase the synthesis success rate.



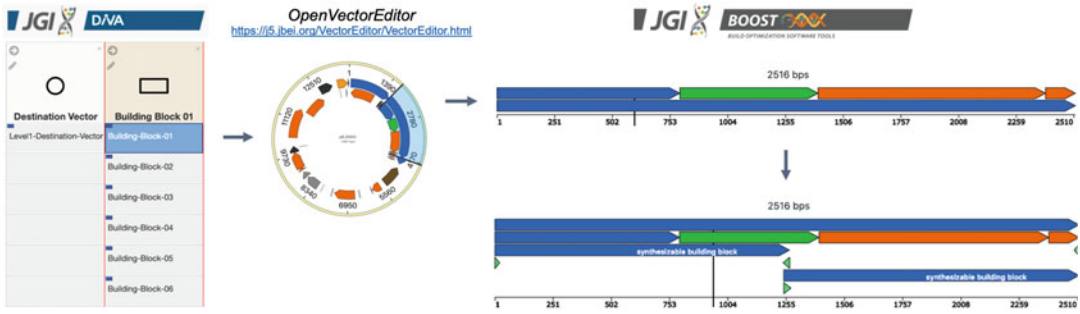
**Fig. 3** j5 generated plasmid map of the final construct (destination vector and inserted pathway)



**Fig. 4** Using BOOST to partition the insert into fragments for Yeast/Golden Gate assembly

BOOST outputs (a) instructions for assembly/cloning in XML format (not shown) and (b) a GenBank file of the partitioned pathway sequence. The Genbank output includes the original pathway sequence including feature annotations (genes, promoters, terminators) and annotations of the sequence partitions, including their overlap sequences.

In Fig. 4, we schematically illustrate the process of using BOOST for partitioning a large sequence into smaller fragments. The upper portion of Fig. 4 is the linear representation of the highlighted portion of the plasmid shown in Fig. 3. The user generates a Genbank file using openVE, provides the file as input to BOOST which partitions the input sequence into fragments that



**Fig. 5** Screenshots of DIVA, openVE, and the BOOST input/output sequences for the computer-aided design of the Level-0 constructs of the pathway

need to be synthesized and assembled using either Golden Gate or Yeast-based assembly protocol. In the lower portion of Fig. 4, we provide an openVE screenshot, visualizing the output of BOOST, namely, the original pathway sequence including annotations of the sequence partitions, which need to be designed for synthesis in the following steps.

In the following steps, we describe our workflow for designing the synthesis and assembly of the pathway’s sequence partitions—from now on called “Level-0 constructs”. In Fig. 5, we provide an overview of the design steps of the Level-0 constructs, following the same process as the design of the Level-1 construct and utilizing the same CAD tools for designing the Level-0 constructs, that is, DIVA, j5, openVE, and BOOST (*see Note 4*).

### 3.4 Using DIVA for the Specification and Design of Level-0 Constructs

The researcher starts with the specification of the Level-0 design in DIVA. Here, we build a combinatorial design of one destination (left column in DIVA design) and the six sequence partitions of the Level-1 construct, that is, the pathway (right column in DIVA design). To instruct j5, we specify “digestion” as assembly strategy for the destination vector and “direct synthesis” for the sequence partitions. The red vertical lines after each DIVA column denote the so-called DNA synthesis firewalls, instructing j5 to not further extend the DNA sequences of the column according to the selected assembly protocol.

### 3.5 Using j5 for Designing the Assembly of the Combinatorial Design

After the specification of the combinatorial design and the desired assembly strategies in DIVA, we use j5 for designing the assembly of the Level-0 intermediates and the destination vector. We show a screenshot of openVE, visualizing the assembled construct of the digested destination vector and one of the six synthesized building blocks, which is highlighted. The linear inserted partition into the destination vector is visualized to the right of the openVE screenshot.

### **3.6 Using BOOST for Synthesis and Gibson Chew-Back Assembly of the Level-0 Constructs**

In the last design step, we use BOOST to verify the Level-0 inserts against DNA synthesis criteria and to partition inserts, which exceed the maximum length of synthesis, into synthesizable fragments. Similarly, for Subheading 3.3, BOOST outputs the synthesizable partitions and, if desired, the PCR amplification primers. Also, BOOST generates assembly and cloning instructions about the chew-back assembly of the synthesized building blocks and their insertion into the Level-0 destination vectors.

After the Level-1 and the Level-0 constructs are designed and the required synthesis, assembly and cloning instructions are defined, the researcher can start the build process of the *in silico* designs.

### **3.7 Using SynTrack for Tracking the Build Process of Level-0 and Level-1 Constructs**

SynTrack takes as input the output of BOOST, namely, an XML file containing a description of synthesis, assembly, and cloning steps for building the desired construct. After the user uploads the XML file, then SynTrack can generate the order sheets of the synthetic fragments and PCR-amplification primers. SynTrack instructs the researchers through the assembly and cloning process, including QA/QC steps, generates plating instructions for robotics that automate certain steps along the process and also enables the user to attach the experiment data of each step.

After the assembly and cloning, the researchers perform a QC step, verifying that the build sequences matches the designed sequences. Therefore, we utilize a SynBioQC computational pipeline to perform sequence validations. Depending on the QA/QC results, the researcher either picks an error-free clone or must fix the errors, such as by reiterating the last cloning steps. The error-free clones are then stored in a master-plate, which is managed by SynTrack and shipped to the JGI user as soon as the user's constructs are ready.

---

## **4 Notes**

1. The following software tools are accessible via web-based UIs at no-cost to noncommercial users: BOOST, DIVA/DeviceEditor, ICE, openVE, and j5. The SynTrack tool as well as the BLiSS and SynBioQC pipelines are currently only available for internal use at the JGI, JBEI, and ABF. Commercial licenses (where required) for the tools are available through commercial distributors or the Innovations and Partnerships Office of Berkeley Lab (<https://ipo.lbl.gov>).
2. The bioCAD/CAM tools described in this book chapter are developed with the intention to support researchers throughout the design process. However, the tools themselves have certain limitations that currently preclude fully automated

design workflows. In the workflow described here, we often had to use a tool as a stand-alone, requiring to export the output of each tool, for example, into a Genbank file and to import the Genbank file into the next tool in our workflow, placing a burden on the user to organize and store several input and output files. Our future work will continue toward a more integrated solution, enabling to exchange data among the tools in a more sustainable and user-friendly manner. For example, DIVA/DeviceEditor, j5, ICE, openVE, and a subset of BOOST functionality are already well integrated.

3. Here, we do not demonstrate DIVA's capabilities of importing the sequences of the design elements (vector backbone, promoters, terminators, genes) from an ICE repository. In such a scenario, the user first needs to enter all parts into an ICE repository that is connected with DIVA. For example, the designer bulk-uploads all parts, their types, and their sequences into JGI's instance of ICE and specifies the design in JGI's instance of DIVA, which is configured to import parts from JGI's ICE instance.
4. The top-down design process, demonstrated in this book chapter, highlights that the design steps are repeated at each design level. For example, the steps for designing the Level-1 constructs are the same as the design steps of the Level-0 constructs.

---

## Acknowledgments

This work was part of the DOE Joint Genome Institute (<https://jgi.doe.gov>) and Joint BioEnergy Institute (<https://www.jbei.org>) supported by the US Department of Energy, Office of Science, Office of Biological and Environmental Research, and was part of the Agile BioFoundry (<https://agilebiofoundry.org>) supported by the US Department of Energy, Energy Efficiency and Renewable Energy, Bioenergy Technologies Office, through contract DE-AC02-05CH11231 between Lawrence Berkeley National Laboratory and the US Department of Energy. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the US Government or any agency thereof. Neither the US 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 US Government retains and the publisher, by accepting the article for publication, acknowledges that the US 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 US Government purposes.

We thank the DOE National Energy Research Scientific Computing Center (NERSC) (<https://www.nersc.gov/>) for hosting the JGI's instances of the bioCAD/CAM tools described in this chapter (e.g., DIVA/DeviceEditor/j5, BOOST, BLiSS, SynTrack, and ICE).

*Author Contributions:* R.E., S.N., A.T., N.J.H. and J.F.C. are end-users of the bioCAD/CAM tools. N.J.H. developed and maintains j5. X.M. developed and maintains SynTrack. E.O. developed and maintains BOOST and the SynBioQC pipelines. H.P. developed and maintains DIVA/DeviceEditor and ICE. L.S. developed and maintains BLiSS and manages the JGI's deployments of DIVA/DeviceEditor/j5, and ICE. S.N. designed and performed the pathway design workflow. All authors wrote, edited, and reviewed the manuscript.

*Conflicts of Interest Statement:* N.J.H. declares financial interests in the form of issued and pending patent applications related to the j5 software, and equity in TeselaGen Biotechnology, Inc.

## References

1. Nowogrodzki A (2018) The automatic-design tools that are changing synthetic biology. *Nature* 564(7735):291–292. <https://doi.org/10.1038/d41586-018-07662-w>
2. Appleton E, Madsen C, Roehner N, Densmore D (2017) Design automation in synthetic biology. *Cold Spring Harb Perspect Biol* 9: a023978
3. Chen J, Densmore D, Ham T, Keasling J, Hillson N (2012) DeviceEditor visual biological CAD canvas. *J Biol Eng* 6:1. <https://doi.org/10.1186/1754-1611-6-1>
4. Ham TS, Dmytriv Z, Plahar H, Chen J, Hillson NJ, Keasling JD (2012) Design, implementation and practice of JBEIICE: an open source biological part registry platform and tools. *Nucleic Acids Res* 40:e141
5. Hillson NJ, Rosengarten RD, Keasling JD (2012) j5 DNA assembly design automation software. *ACS Synth Biol* 1(1):14–21. <https://doi.org/10.1021/sb2000116>
6. Bilitchenko L, Liu A, Cheung S, Weeding E, Xia B et al (2011) Eugene – a domain specific language for specifying and constraining synthetic biological parts, devices, and systems. *PLoS One* 6(4):e18882. <https://doi.org/10.1371/journal.pone.0018882>
7. Quinn JY, Cox RS III, Adler A, Beal J, Bhatia S et al (2015) SBOL visual: a graphical language for genetic designs. *PLoS Biol* 13(12): e1002310. <https://doi.org/10.1371/journal.pbio.1002310>
8. Oberortner E, Cheng J-F, Hillson NJ, Deutsch S (2017) Streamlining the design-to-build transition with build-optimization software tools. *ACS Synth Biol* 6(3):485–496. <https://doi.org/10.1021/acssynbio.6b00200>
9. Galdzicki M, Clancy K, Oberortner E, Pocock MR, Quinn JY, Rodriguez CA, Roehner N, Wilson ML, Adam L, Anderson JC, Bartley B, Beal J, Chandran D, Chen JY, Densmore DM, Endy D, Grünberg R, Hallinan JS, Hillson NJ, Johnson JD, Kuchinsky A, Lux MW, Misirli G, Peccoud J, Plahar HA, Sirin E, Stan G, Villalobos A, Wipat A, Gennari JH, Myers CJ, Sauro HM (2014) The synthetic biology open language (SBOL) provides a community standard for communicating designs in synthetic biology. *Nat Biotechnol* 32:545–550
10. Zhang Z, Nguyen T, Roehner N, Misirli G, Pocock M, Oberortner E et al (2015) libSBOLj 2.0: a Java library to support SBOL 2.0. *IEEE Life Sci Lett* 1:34–37. <https://doi.org/10.1109/LLS.2016.2546546>

11. Gibson DG, Young L, Chuang RY, Venter JC, Hutchison CA 3rd, Smith HO (2009) Enzymatic assembly of DNA molecules up to several hundred kilobases. *Nat Methods* 6 (5):343–345. <https://doi.org/10.1038/nmeth.1318>
12. Shao Z, Zhao H, Zhao H (2009) DNA assembler, an in vivo genetic method for rapid construction of biochemical pathways. *Nucleic Acids Res* 37(2):e16. <https://doi.org/10.1093/nar/gkn991>
13. Kouprina N, Larionov V (2016) Transformation-associated recombination (TAR) cloning for genomics studies and synthetic biology. *Chromosoma* 125(4):621–632
14. Engler C, Kandzia R, Marillonnet S (2008) A one pot, one step, precision cloning method with high throughput capability. *PLoS One* 3 (11):e3647. <https://doi.org/10.1371/journal.pone.0003647>