BIOINFORMATICS APPLICATIONS NOTE Vol. 26 no. 21 2010, pages 2782–2783
doi:10.1093/bioinformatics/btq518

Sequence analysis

Advance access publication September 13, 2010

SynBioWave—a real-time communication platform for molecular and synthetic biology

Paul R. Staab[1,†], Jörg Walossek[2,†], David Nellessen[1,†], Raik Grünberg[3], Katja M. Arndt[2,4,5,6] and Kristian M. Müller[2,4,*]

[1]Department of Mathematics, [2]Department of Biology, University of Freiburg, [3]EMBL-CRG Systems Biology program, CRG, Barcelona, Spain, [4]Centre for Biological Signalling Studies (BIOSS), [5]Institute for Advanced Studies (FRIAS), University of Freiburg and [6]Institute for Biochemistry and Biology, University of Potsdam

Associate Editor: John Quackenbush

ABSTRACT

Summary: Synthetic Biology is advanced by many users and relies on the assembly of genetic elements to devices, systems and finally genomes. SynBioWave is a software suite that enables multiple distributed users to analyze and construct genetic parts in real-time collaboration. It builds on Google Wave and provides an extensible robot–robot–user communication framework, a menu driven user interface, biological data handling including DAS and an internal database communication. We demonstrate its use by implementing robots for gene–data retrieval, manipulation and display. The initial development of SynBioWave demonstrates the power of the underlying Google Wave protocol for Synthetic Biology and lays the foundation for continuous and user-friendly extensions. Specialized wave-robots with a manageable set of capabilities will divide and conquer the complex task of creating a genome *in silico*.

Availability: The robot is available at SynBioWave@appspot.com, the source code at http://synbiowave.sourceforge.net

Contact: kristian@biologie.uni-freiburg.de

Supplementary information: Supplementary data are available at *Bioinformatics* online and at http://www.synbiowave.org.

Received on April 5, 2010; revised on August 30, 2010; accepted on September 4, 2010

1 INTRODUCTION

Scientific research is a collaborative effort requiring continuous exchange of ideas and data interlinking researchers at every level of the work flow. Since geographic separation should pose no hindrance, such networking is best achieved using capabilities of the so called Web 2.0. In the life sciences, numerous databases provide information and web services solve specific tasks. Despite these efforts, a technology to share the research working process from the decision and design phase of an experiment to the data collection and the final evaluation remained elusive.

This gap is filled by an online platform named 'Wave', which was introduced in 2009 by Google (http://wave.google.com). Google Wave enables real-time collaboration between users and also integrates software tools. Various types of content are incorporated in one evolving communication document. Processes are handled by a wave server, which offers sophisticated capabilities for synchronization, annotation and versioning. The underlying Wave protocol as well as the server and client software are available as open source (http://www.waveprotocol.org).

There are currently two ways to extend Google Wave: Gadgets and Robots. While Gadgets are client side JavaScript applications that are synchronized by the Wave server, Robots are Java or Python applications running on a server and appearing as artificial communication participants in a Wave client. This technology has been praised (Neylon 2009; Noorden 2009), yet only a few single-task robots have been released for scientific applications.

2 SOFTWARE

2.1 Usage

As SynBioWave is based on Google Wave, usage differs from typical desktop or database-front ends. A user first opens a Wave client, which is a dynamic web site within a web browser, and creates or opens a conversation object called a Wave. The user may invite further people to this Wave and start to communicate. The SynBioWave environment is initialized by adding the SynBioWave robot (synbiowave@appspot.com) to the conversation. This robot adds a user interface (UI) to the Wave and provides core biological functionality. All users participating in the Wave can collaboratively use SynBioWave in real time.

For example, one user uploads a DNA sequence and another user wishes to translate the sequence and to search for reading frames. Since the latter functions are not included in the SynBioWave core, the user adds the participant/extension 'synbiowave-translate@appspot.com', which instantly inserts the translate option in the pull-down menu of the main toolbar. Finally, users can choose between various ways to display biological data (Supplementary Fig. S1). A more extensive user guide is available in the Supplementary Material.

2.2 Architecture

The SynBioWave core consists of one robot, the SynBioWave robot and an associated gadget. Together, these programs generate a UI, communicate with an internal database, and import or export annotated sequence formats. Specialized bioinformatics tasks are implemented in additional robots (Supplementary Fig. S2). Such further robots can be developed based on our extended

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first three authors should be regarded as joint First authors.

robot application programming interface (API), which provides an abstract class for creating robots. Hence, further robots can be built without detailed Wave know-how. By selecting task-relevant robots, end users can dynamically manage the complexity of the SynBioWave front end.

The concept of the SynBioWave core robot for organizing common tasks also overcomes a shortcoming of the original Wave platform, which does not offer coordination between third-party robots that may co-exist within a single Wave.

2.3 Graphical UI

Providing a consistent user experience and organization of robot services posed a major challenge. Therefore, we extended the Wave UI with a dynamically generated toolbar that integrates the features of the robots added to the Wave. This pull-down menu furnishes the web-based SynBioWave with a look and feel similar to a desktop application.

This was achieved by developing a new protocol, named qooxWave. The protocol defines a client-to-server and server-to-client communication for building UI-elements, event reporting and data transmission. Communication is realized via JSON strings that are stored inside a gadget's state object. Both server (robot) and client (gadget) can access the state object and both can react to changes of this state object (the latter is a feature of the Wave API). The client-side implementation of the qooxWave protocol is based on the rich-internet application (RIA) framework qooXdoo (http://qooxdoo.org). Wave gadgets import the qooXdoo Javascript application into the Wave-client. A robust store-model-view-controller architecture ensures a proper client-side implementation of the complex protocol. This technology can also be used for other scientific or non-scientific Wave extensions and provides a template for implementing new functions into SynBioWave.

2.4 Data handling

Biological information is provided in many different formats in databases, locally stored files and from user input. As Google Wave provides sharing capability and textual editing only, we improved the Google Java API to access, process and convert biological information. A very general internal Java representation of DNA, RNA and protein sequences was created allowing for easy conversion of the internal format to commonly used formats and vice versa (Supplementary Fig. S3). This representation merges the basic information of the GenBank format (http://www.ncbi.nlm.nih.gov/Sitemap/samplerecord.html) with the annotation system of the Google Wave API. Within SynBioWave sequence, information is stored in a separate workspace wave, which is automatically made accessible to all users added to the main wave. Import and export to databases is handled via the Java Data Objects (JDO) protocol. Use of JDO enables the user to choose compatible external databases (Supplementary Fig. S4). In addition, we implemented a manager class for the database protocol 'Distributed Annotation System' (DAS) (Dowell *et al.*, 2001). This protocol allows users to collect sequences and annotations from DAS-enabled databases, collate the information and display them in a single view within SynBioWave.

Java Wave robots *per se* represent Http Servlets originally restricted to serve robot events. Therefore, we extended their capabilities to enable file up- and downloads. File accession is handled based on the apache commons-upload project and information is then parsed via BioJavaX classes (Holland *et al.*, 2008). Due to the underlying BioJavaX API, the upload is easily extendable to sequence files in different formats. A simple function for converting directly typed or pasted sequences into the internal sequence format is also provided.

2.5 Extensibility

An important feature of the SynBioWave is extensibility by user contributed robots. The SynBioWave API provides easy access to the aforementioned features allowing a programmer to focus on specific tasks. The supplement gives an example of the API documentation (Supplementary Fig. S5) and a step-by-step developer guide.

3 DISCUSSION

SynBioWave builds upon Google Wave, a real-time online collaboration-software for molecular and synthetic biology. We added several features that are important for scientific tasks and provide a general-purpose extension framework. Wave is still undergoing significant development. A continuous co-development of SynBioWave is envisioned and we, for example, upgraded to Google's robot API v2. At present, performance of the Google Wave server varies and the long-term availability is unclear. Users can set up their own Wave servers and clients using open source software. (Supplementary Fig. S6). Due to its open architecture, SynBioWave can be adapted to various client–server implementations. To ensure wide access and compatibility with the latest updates, the SynBioWave robot is currently hosted on the Google Wave server. The code of SynBioWave is maintained over the sourceforge platform for open-source projects enabling subversion control and feature request. Primary contact point is the project web site (http://www.synbiowave.org), which provides documentation, a screencast, and a wordpress forum.

ACKNOWLEDGEMENTS

We thank the Qooxdoo development team for technical help.

Funding: Excellence Initiative of the German Federal and State Governments [EXC 294]; Genedata AG, Basel, Switzerland; Roche, Basel, Switzerland; Human Frontier Science Program (LT fellowship to R.G.).

Conflict of Interest: none declared.

REFERENCES

Dowell,R.D. *et al.* (2001) The distributed annotation system. *BMC Bioinformatics*, **2**, 7.

Holland,R.C. *et al.* (2008) BioJava: an open-source framework for bioinformatics. *Bioinformatics*, **24**, 2096–2097.

Neylon,C. (2009) Head in the clouds: Re-imagining the experimental laboratory record for the web-based networked world. *Autom Exp.*, **1**, 3.

Van Noorden,R. (2009) The science of Google Wave. *Nature.* Available at http://www.nature.com/news/2009/090824/full/news.2009.857.html (last accessed date July 7, 2010).