BioSystems 139 (2016) 12–16

Contents lists available at ScienceDirect

BioSystems

journal homepage: www.elsevier.com/locate/biosystems

Programmatic access to logical models in the Cell Collective modeling environment via a REST API

Bryan M. Kowal[a], Travis R. Schreier[a], Joseph T. Dauer[b], Tomáš Helikar[a,*]

[a] Department of Biochemistry, University of Nebraska-Lincoln, Lincoln, NE, USA
[b] School of Natural Resources, University of Nebraska-Lincoln, Lincoln, NE, USA

ARTICLE INFO

Article history:
Received 14 September 2015
Received in revised form 3 November 2015
Accepted 12 November 2015
Available online 14 November 2015

Keywords:
Logical models
Boolean models
REST API
Systems biology

ABSTRACT

Summary: Cell Collective (www.cellcollective.org) is a web-based interactive environment for constructing, simulating and analyzing logical models of biological systems. Herein, we present a Web service to access models, annotations, and simulation data in the Cell Collective platform through the Representational State Transfer (REST) Application Programming Interface (API). The REST API provides a convenient method for obtaining Cell Collective data through almost any programming language. To ensure easy processing of the retrieved data, the request output from the API is available in a standard JSON format.

Availability and implementation: The Cell Collective REST API is freely available at http://thecellcollective.org/tccapi. All public models in Cell Collective are available through the REST API. For users interested in creating and accessing their own models through the REST API first need to create an account in Cell Collective (http://thecellcollective.org).

Contact: thelikar2@unl.edu.

Supplementary information: Technical user documentation: https://goo.gl/U52GWo.

1. Introduction

Cell Collective is an interactive platform for the creation, simulations, and analyses of large-scale logical models (http://thecellcollective.org; Helikar et al., 2012a,b, 2015, 2013). The web-based software has been designed to make computational network modeling more accessible to the modeling, experimental, and life sciences education communities. The platform provides collaborative model creation, editing, and simulation that do not require manual editing of complex mathematical expressions, or writing code to execute these expressions. In addition, the Cell Collective Knowledge Base enables users to fully annotate every model interactions with supporting literature, providing an environment with fully transparent models that can be easily extended by others within a single platform. Currently, Cell Collective contains over 50 published logical models that are available to the community for simulations, analyses, and additional modifications. The platform also contains over 1500 private models created by Cell Collective users. Models can be created, edited, simulated, and analyzed directly in Cell Collective, or downloaded in multiple formats, including truth tables (.csv files), text files with logical expressions, and the recent SBML-qual format (Chaouiya et al., 2013). While other web-based network modeling platforms exist (e.g., Virtual Cell Slepchenko et al., 2003), they tend to utilize differential equations for the modeling technique to capture high levels of detail. The Cell Collective platform, on the other hand, utilizes logical models as they are more scalable, and still provide enough dynamical resolution.

Herein, we report a new REST API that provides users with programmatic access to Cell Collective models, simulations, and model annotations. This makes the access to logical models and the biological knowledge supporting them more convenient for analyses by other software tools, such as GINsim (Chaouiya et al., 2012), CellNOpt (Terfve et al., 2012), BoolNet (Müssel et al., 2010), and GNA (Batt et al., 2012).

2. Implementation

The REST API has been implemented as a Spring Boot Application. The API consists of a number of REST Controllers that have been registered via Spring Web to expose service endpoints that can be accessed over the Internet. The REST API Controllers utilize the existing Cell Collective pluggable services layer to retrieve information from the Cell Collective database. The REST API has been secured using Spring Security. Service endpoints

---

* Corresponding author.
E-mail address: thelikar2@unl.edu (T. Helikar).

http://dx.doi.org/10.1016/j.biosystems.2015.11.005
0303-2647/© 2015 Elsevier Ireland Ltd. All rights reserved.

Table 1
List of available API services. Note that information can be obtained only for models that are directly available to the user based on proper authentication and authorization.

Service	Description
getAvailableModelRef	Retrieve information {id, name, tags} for all model(s) that the requester has access to. Only published models will be returned for non-authenticated users. All private, shared, and published models will be returned for authenticated users.
getModelLogicalExpressions	Retrieve all logical (Boolean) expressions associated with a model. Expressions are in the form: X & !Y | Z with parentheses where required to ensure proper order of operations. "&" represents the AND relationship; "!" represents negation, and "|" corresponds to the OR relationship.
getModelShareInformation	Retrieve information about collaborators who have shared access to a model. The response consists of a list of users by username, as well as the date the model was shared with a given user, and whether or not the user has accepted the share request.
exportModel	Retrieve file bytes for an export of a model in the specified format(s) {Expression, Truth Table, SBML}. The file bytes can be written to a zip file. The expression export option will include a text file containing all logical expressions returned by the getModelLogicalExpressions method above. The truth table export option will return truth tables in separate text files for each species. Finally, the SBML export option will return the model converted to SBML Qualitative Models (Chaouiya et al., 2015, 2013).
getSpeciesReferences	Retrieve all model components {id, name}. The id of the components that are returned can be utilized within the getModules method to retrieve the regulatory Modules that are associated with the Component (if any). See Helikar et al. (2012a,b) for more information on how regulatory modules are defined in Cell Collective.
getModules	Retrieve all regulatory module(s) associated with a specified model component. See Helikar et al. (2012a,b) for more information on how regulatory modules are defined in Cell Collective. The information returned by this method includes the id of the module, the components that are part of the module, the type of module, and a description of the regulatory mechanism. The returned module id can be used to retrieve both dominance information (see below) via the getDominance API method (when applicable) and to retrieve conditions via the getConditions API method (when applicable).
getDominance	Retrieve all positive regulator module(s) over which a specified negative module is dominant. See Helikar et al. (2012a,b) for more details on dominance between positive and negative regulators.
getConditions	Retrieve condition(s) associated with a specified regulatory module. The information returned by this method includes the condition id, the type of condition, the name of the condition, the state of the condition, the species that participate in the condition, and a description of the condition. The condition id can be also used to retrieve sub-conditions via the getSubConditions API method (when applicable). See Helikar et al. (2012a,b) for more details on conditions and sub-conditions.
getSubConditions	Retrieve the sub-condition(s) associated with a specified condition of a regulatory module. The information returned by this method will include the sub-condition id, the type of sub-condition, the name of the sub-condition, the state of the sub-condition, the species that participate in the sub-condition, and an English description of the sub-condition. See Helikar et al. (2012a,b) for more details on conditions and sub-conditions.
getBiologic	Retrieve all regulatory biologic information (modules, conditions, sub-conditions, and dominance) for a specified model component. This API returns a combination of all of the information returned by the: getSpeciesReferences, getModules, getDominance, getConditions, and getSubConditions API methods.
getKBPage	Retrieve the contents of a Knowledge Base Page for a specified model component. The information returned by this method includes: the owner of the Knowledge Base Page, the date/time that the page was last updated, page content, and any bibliographic information included on the page.
getDynamicExperiments	Retrieves saved dynamical analysis experiment(s) for a specified model. The information returned by this method includes: the id of the experiment, the name of the experiment, and the settings that were used to run the experiment (number of simulations, number of time steps, transient time – see Helikar and Rogers (2009) for more details on these parameters). The experiment id can be used to retrieve the data generated by the experiment via the getDynamicExperimentData method.
getDynamicExperimentData	Retrieves the data generated during the specified experiment. The user can request either a byte stream of the data that can be used to write in a file on the REST client or the raw data arranged in a JSON structure.
searchModels	Search for model(s) by name or description contents. This method returns the id and name of the Model(s) that match the search criteria. Note that only Model(s) that are visible to the user who executed the request are returned.
searchSpecies	Search for model components by name or knowledge base content. This method returns the id and name of component(s) that match the search criteria. Note that only Species that belong to Models that are visible to the user that executed the request are returned.

are available to make the REST API accessible anonymously and with basic HTTP authentication (using Cell Collective credentials).

Every model in Cell Collective is uniquely identified. Associated with each model are simulations and analyses, as well as annotations of each model interaction as entered in the model Knowledge Base. The REST API provides read-only access to all of these components. All API responses are in JSON format. All API responses utilize a standard format to provide a uniform way to distinguish between requests that were successfully handled and requests that the API failed to handle. Detailed documentation for all available functions, and instructions on how to use them are available at: https://goo.gl/U52GWo.

3. Data privacy

Cell Collective models are either public or private. Public models are peer-reviewed and are available to anyone for simulations and analyses Users can make a private copy of any public model, that can be edited and/or extended in any way. Private models are available solely to the user who created them. These models can be shared selectively with others within the platform. Users can also create unique, read-only links to the models for anonymous viewings (e.g., to provide reviewers of manuscripts or grants access to the model for review). The API takes into consideration these restrictions through authentication, and provides users with read-only access to all public models. In addition, API users can only access their own private models and simulation results.

4. Model data

Models in Cell Collective are logical network models, consisting of components and directed interactions between them. Each component in the model can assume an active or inactive activity level, as governed by the biological regulatory mechanism described with a Boolean function (more details on Boolean networks can be found in various reviews, e.g., Albert and Thakar, 2014; Helikar et al., 2010). To make the models more accessible to the experimental