
BioExcel Building Blocks REST API (BioBB REST API), programmatic access to interoperable biomolecular simulation tools

Genís Bayarri¹, Pau Andrio², Adam Hospital^{1,*}, Modesto Orozco^{1,3,*}, Josep Lluís Gelpi^{2,3,*}

¹ Institute for Research in Biomedicine (IRB Barcelona), Barcelona Institute of Science and Technology, Barcelona 08028, Spain, ² Barcelona Supercomputing Center (BSC), ³ Department of Biochemistry and Molecular Biomedicine, University of Barcelona, Barcelona 08028, Spain

*To whom correspondence should be addressed.

Abstract

Motivation: The BioExcel Building Blocks (BioBB) library offers a broad collection of wrappers on top of common biomolecular simulation and bioinformatics tools. The possibility to access the library remotely and programmatically increases its usability, allowing individual and sporadic executions and enabling remote workflows.

Results: BioBB REST API extends and complements the BioBB library offering programmatic access to the collection of biomolecular simulation tools included in the BioExcel Building Blocks library. Molecular Dynamics setup, docking, structure modeling, free energy simulations, and flexibility analyses are examples of functionalities included in the endpoints collection. All functionalities are accessible through standard REST API calls, voiding the need for tool installation.

Availability: All the information related to the BioBB REST API endpoints is accessible from <https://mmb.irbbarcelona.org/biobb-api/>. Links to extended documentation, including OpenAPI endpoints specification and examples, Read-The-Docs documentation and a complete workflow tutorial can be found in the Suppl. Table 1.

Contact: adam.hospital@irbbarcelona.org, gelpi@ub.edu or modesto.orozco@irbbarcelona.org

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Published in Bioinformatics Applications Note: <https://doi.org/10.1093/bioinformatics/btac316>

1 Introduction

Application Programming Interfaces (APIs) are programming codes enabling data sharing through the Internet. APIs are massively used nowadays mainly because they provide a uniform way for accessing information or functionalities from different resources without having to know exactly how the queried platform or the programs behind them works. Representational State Transfer (REST API) is the most commonly used API architecture, mainly due to its compatibility with many data formats, including JavaScript Object Notation (JSON). APIs are nowadays routinely integrated in life science projects where transfer of data is required. Biomolecular data is not an exception, with a huge increase on the amounts of data generated over the last years, and central repositories such as the European Bioinformatics Institute (EMBL-EBI) search API doubling the number of requests every year (Madeira *et al.*, 2019). What is not so common is the use of REST API technology to offer remote execution services instead of just data (Software as a Service – SaaS).

Many bioinformatics approaches following the SaaS scheme have appeared in the last decades, but most of them based on web technologies with graphical interfaces (Web Servers), and not on remote programming access (Web Services). EMBL-EBI has traditionally provided access to analysis tools via Web Services since 2004 (Hamish McWilliam *et al.*, 2013), mainly focused on bioinformatics tools (e.g. BLAST, Clustal). In 2008, a set of bioinformatics Web Services were developed under the BioMoby project (Biomoby consortium, 2008), which defined an interoperable ecosystem to run bioinformatics tools, including biomolecular structure and simulation tools. More recently,

EMBL-EBI portal integrated a set of bioinformatics tools as a service through a collection of REST-API endpoints (Madeira *et al.*, 2019).

The biomolecular simulation field has been traditionally aside from these initiatives, mainly due to the heavy requirement of the calculations and the huge amount of data generated. To our knowledge, only our tool MDMoby (Hospital *et al.*, 2012) offers the possibility to remotely run a set of biomolecular simulation functionalities. MDMoby uses BioMoby Web Services to envelop Molecular Dynamics (MD) tools and offer them programmatically. Although the initiative evolved into a couple of web servers (MDWeb, Hospital *et al.*, 2012, and NAFlex, Hospital *et al.*, 2013), the Simple Object Access Protocol (SOAP) technology used by MDMoby Web Services revealed to be less suitable for handling large data coming from MD services.

With all these projects in mind, the European initiative BioExcel Center of Excellence started to work on the BioExcel Building Blocks (BioBB) project (Andrio *et al.*, 2019). The BioBB library is a collection of portable wrappers on top of common biomolecular simulation tools. The library is designed to i) increase the interoperability between the tools wrapped; ii) ease the implementation of biomolecular simulation workflows; and iii) increase the reusability and reproducibility of the generated workflows. BioBB library was designed to be easily deployable in a number of computation environments, through the use of several frameworks like Conda (<https://www.anaconda.com/>) or Biocontainers (da Veiga Leprevost *et al.* 2016, Bai *et al.* 2021). However, in any case it still requires some degree of installation by the final user. To further extend its usability, and using the expertise gained in our previous experiences, a REST API has been developed within the BioBB portfolio. Taking advantage of the evolution in computational power, network bandwidths, and REST technology, the BioBB REST API extends the BioBB library with the possibility to programmatically

access the available functionality, upgrading and extending the functionalities offered by the MDMoby Web Services. BioBB REST API final goal is to offer to the community the continuously growing collection of BioBB biomolecular functionalities with no software installation requirements.

2 Implementation and usage

BioBB REST API endpoints are implemented in PHP with the help of PHP slim framework (<https://www.slimframework.com>). The development follows best practices in building a usable REST API for life sciences (Tarkowska *et al.* 2018). API endpoints are generated automatically from the information contained in the Python docstring documentation at source code. This allows the continuous update of the API as the BioBB library evolves. Indeed, using GitHub Webhooks, updates in all BioBB packages automatically trigger an update on the BioBB REST API (Suppl. Figure 1). This ensures a consistency between the available BioBB building blocks and the REST API endpoints.

Tools are executed in asynchronous mode, as the time to completion is usually unknown and will depend on the tool and the available resources. Thus, endpoints should be called in three steps: *launch*, *poll*, and *data retrieval* (Suppl. Figure 2). Auxiliary functions and code snippets for polling and data retrieval are available in the documentation and tutorials (<https://mmb.irbbarcelona.org/biobb-api/tutorials/restapi>). Generic endpoints are available from the base BioBB REST API URL <https://mmb.irbbarcelona.org/biobb-api/rest/v1> (*baseURL*), whereas all endpoints are available from the BioBB REST API Tools Endpoints URL <https://mmb.irbbarcelona.org/biobb-api/tools-documentation?docExpansion=none>. A uniform syntax is used for all API endpoints. Tools are launched with the *baseURL/launch/{package}/{tool}* endpoints, with the particular *tool* name included and its corresponding *BioBB package* (e.g. *biobb_md/solvate*). Upon success, this launch service returns a *token* that can be used to query for the job status using the *baseURL/retrieve/status/{token}* endpoint. Once the execution is finished, generated data can be retrieved using the *baseURL/retrieve/data/{id}* endpoints, with *id* being a unique identifier for each of the produced files (included in the previous JSON response). The set of sample files included in the BioBB packages are available through the *baseURL/sample/{package}/{tool}* endpoints for testing purposes. All this information is included in the Swagger/OpenAPI specification available at <https://mmb.irbbarcelona.org/biobb-api/rest>.

BioBB REST API is powered by an on-demand private cloud infrastructure, deploying new Virtual Machines when needed (Suppl. Figure 3). There is no access restriction, with jobs being controlled by a queue manager and executed in a first come, first served regime, and a maximum execution time of 1h. Infrastructure readiness and correct execution of all endpoints are routinely checked with a continuous delivery process. Note that the available resources are not suitable for large (HPC based) simulation experiments, however the functionality provided allows to perform all required operations to prepare an HPC run and to analyze the results (i.e. the API is able to run a common MD setup process, but to run a production MD, which will take more than the 1h limit, user's resources, local or on available HPC clusters, would be required).

The BioBB REST API is complemented by an extended set of documentation, including tutorials and examples. The complete list of endpoints (150 as of January 2022) is presented in the website, with the possibility to test the process either manually (<https://mmb.irbbarcelona.org/biobb-api/tools-documentation>) or

automatically (<https://mmb.irbbarcelona.org/biobb-api/tools-execution>). In both cases users can provide their own data although sample data for each endpoint is available. Finally, the MD setup tutorial (<https://mmb.irbbarcelona.org/biobb-api/tutorials/mdsetup>) shows how a complete biomolecular simulation workflow can be executed using the BioBB REST API, and can be used as an example of how to use the API using a graphical interface such as Jupyter Notebooks. Links are summarized in the Suppl. Table 1.

3 Conclusion

The BioBB REST API presented here provides programmatic access to the available tools integrated in the BioBB library without the need for any local installation. To maximize the usability, syntax of the endpoints is simple and intuitive. Endpoints and sample files are automatically generated and extracted from the BioBB library source code, ensuring synchronization of the API with the growing collection of tools from the BioBB library, offering then simple remote and programmatic access to a huge variety of biosimulation tools.

BioBB REST-API information is gathered in a single landing webpage: <https://mmb.irbbarcelona.org/biobb-api/>

Funding

This work was funded by the Spanish Ministry of Science [Grant No. RTI2018-096704-B-100]; the Catalan SGR [Grant no. 2017SGR1110]; the European Union's Horizon 2020 research and innovation program [BioExcel-2 project, 823830]; the SIFECAT Ris3Cat Clúster Emergente del Cerebro Humano (CECH) [Grant No. 001-P-001682], cofounded by the Fondo Europeo de Desarrollo Regional (FEDER); the Instituto Nacional de Bioinformática, ISCIII, Biomolecular and Bioinformatics Resources Platform (Grants Nos. ISCIII PT17/0009/0007 and PT17/0008/0001), cofounded by the Fondo Europeo de Desarrollo Regional (FEDER); the MINECO Severo Ochoa Award of Excellence (Government of Spain) (awarded to IRB Barcelona).

Conflict of Interest: none declared.

References

- Andrio, P. *et al.* BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows (2019). *Sci Data* 6, 169.
- Bai J. *et al.* BioContainers Registry: Searching Bioinformatics and Proteomics Tools, Packages, and Containers (2021). *J. Proteome Res.*, 20, 4, 2056–2061.
- da Veiga Leprevost *et al.* BioContainers: an open-source and community-driven framework for software standardization (2017). *Bioinformatics*, 33, 16.
- Hospital, A. *et al.* MDWeb and MDMoby: an integrated web-based platform for molecular dynamics simulations (2012). *Bioinformatics*, Volume 28, Issue 9.
- Hospital A. *et al.* NAFlex: a web server for the study of nucleic acid flexibility (2013). *Nucleic Acids Research*, Volume 41, Issue W1.
- Madeira F. *et al.* The EMBL-EBI search and sequence analysis tools APIs in 2019. (2019). *Nucleic Acids Research*. Volume 47, Issue W1.
- McWilliam H. *et al.* Analysis Tool Web Services from the EMBL-EBI (2013). *Nucleic Acids Res.* 2013 Jul; 41(Web Server issue): W597–W600.
- Tarkowska A, *et al.* Eleven quick tips to build a usable REST API for life sciences (2018). *PLoS Comput Biol* 14(12): e1006542
- The BioMoby Consortium, Interoperability with Moby 1.0—It's better than sharing your toothbrush! (2008), *Briefings in Bioinformatics*, Volume 9, Issue 3.

BioExcel Building Blocks REST API (BioBB REST API), programmatic access to interoperable biomolecular simulation tools – Supplementary Material

Genís Bayarri¹, Pau Andrio², Adam Hospital^{1,*}, Modesto Orozco^{1,3,*}, Josep Lluís Gelpí^{2,3,*}

¹ Institute for Research in Biomedicine (IRB Barcelona), Barcelona Institute of Science and Technology, Barcelona 08028, Spain, ² Barcelona Supercomputing Center (BSC), ³ Department of Biochemistry and Molecular Biomedicine, University of Barcelona, Barcelona 08028, Spain

*To whom correspondence should be addressed.

Contact: adam.hospital@irbbarcelona.org, gelpi@ub.edu or modesto.orozco@irbbarcelona.org

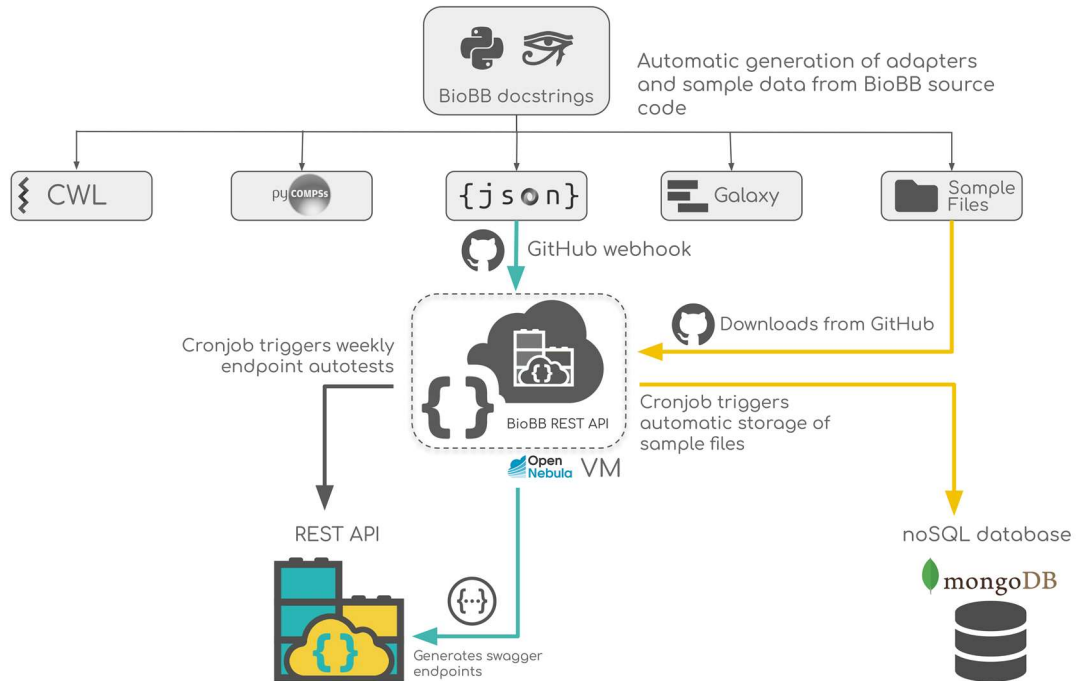
Supplementary information: Supplementary data are available at *Bioinformatics* online.

Published in Bioinformatics Applications Note: <https://doi.org/10.1093/bioinformatics/btac316>

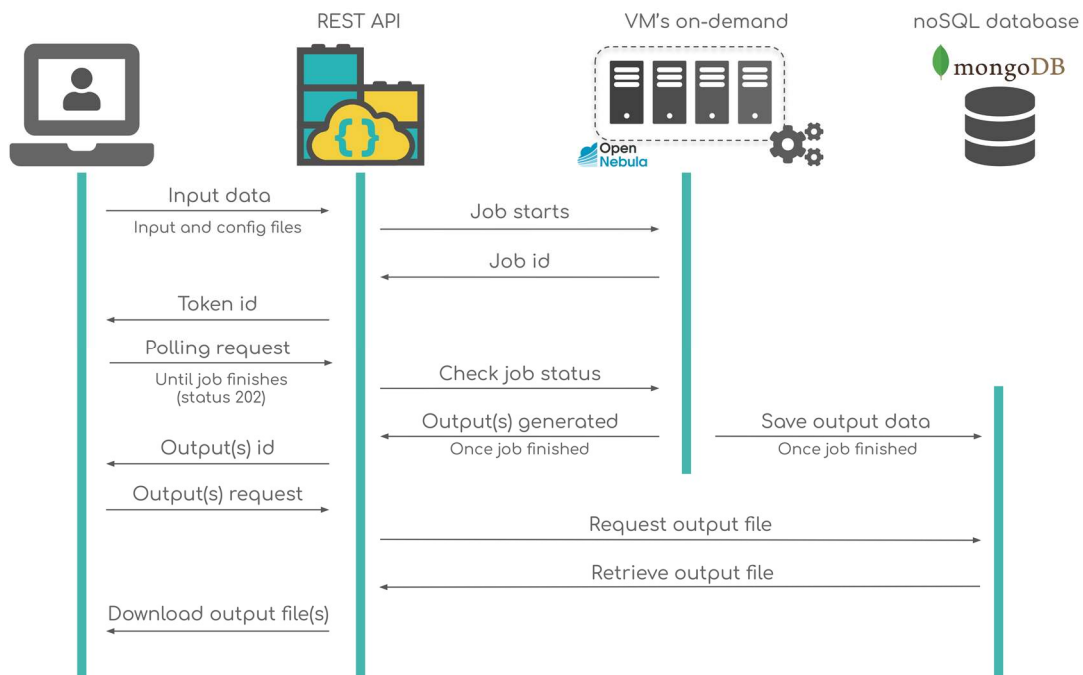
Suppl. Table 1: Documentation available from the **BioBB REST API** web site. Description of the information included in the web section and links to the respective section are shown in the table.

Description	Link
Landing Web Page	https://mmb.irbbarcelona.org/biobb-api/
OpenAPI Documentation	https://mmb.irbbarcelona.org/biobb-api/rest
Endpoints List	https://mmb.irbbarcelona.org/biobb-api/tools-documentation
Endpoints Examples	https://mmb.irbbarcelona.org/biobb-api/tools-execution
Endpoints Tutorial	https://mmb.irbbarcelona.org/biobb-api/tutorials/restapi
Protein MD Setup Workflow Tutorial	https://mmb.irbbarcelona.org/biobb-api/tutorials/mdsetup
Read-The-Docs Documentation	https://biobb-rest-api-documentation.readthedocs.io/
bio.tools entry	https://bio.tools/bioexcel_building_blocks_rest_api_biobb-api

Suppl. Figure 1: BioBB REST API automatic endpoints generation schema. Information contained in the Python docstring documentation for each of the BioBB building blocks is automatically parsed and used to generate the API endpoints. GitHub webhooks ensure consistency between the available BioBB building blocks and the REST API endpoints, triggering automatic updates with every push in the repositories. Sample files are also automatically downloaded from the corresponding repositories, stored in the API database, and used for the weekly auto tests.



Suppl. Figure 2: BioBB REST API user's interaction diagram. API endpoints executions comprise three different steps: launch, poll, and data retrieval. The launch process involves sending the input data files, with the endpoint returning a token id (asynchronous execution). The token is then used to query for the job status (Polling request). Once the execution is finished, a set of output id(s) (unique identifiers) for the generated data are sent, and can be used to retrieve the final results (Output request).



Suppl. Figure 3: BioBB REST API internal schema, including a private on-demand cloud infrastructure and a noSQL mongoDB.

