COMBACH:
A case study to enhance the reproducibility of a dynamic JAK/STAT5 pathway model by creating a fully-featured COMBINE archive

Oliver Hölsken[1,4,5] ⓘ **, Felix Patricius Hans**[2,4] ⓘ **, Sami Habib**[3,4] ⓘ **, Abel Hodelin Hernandez**[3,4] ⓘ

[1]Charité – Universitätsmedizin Berlin, corporate member of Freie Universität Berlin and HumboldtUniversität zu Berlin, Laboratory of Innate Immunity, Department of Microbiology, Infectious Diseases and Immunology, Berlin, Germany
[2]Medical Center - University of Freiburg, University Emergency Department, Freiburg, Germany
[3]University Medical Center of the Johannes Gutenberg University Mainz, Mainz, Germany
[4]Master Program - Biomedical Informatics and Data Science (BIDS), Graduate School Rhein-Neckar in collaboration with the Medical Informatics for Research and Care in University Medicine (MIRACUM), Germany
[5]Heidelberg Bioscience International Graduate School, Heidelberg University, Heidelberg, Germany

Abstract

Introduction: The increase of size and complexity of simulation studies in systems biology and systems medicine proposes new challenges to sharing reproducible results. The Computational Modeling in Biology Network (COMBINE) improves the coordination of standard formats for several features of simulation studies [1]. On the other side, GitHub has been used as an essential common platform for managing software projects and supporting collaborative development [2]. In this case study we aimed to develop a fully-featured COMBINE archive of the Bachmann signaling model [3] using agile co-working technologies.

Methods: In this case study we implemented a GitHub environment to create a publicly traceable progress documentation of a COMBINE archive assembly. Systematic searches for available modeling files (Systems Biology Markup Language (SBML)) and software tools were conducted. The interaction between the tools and the modeling data was assessed and a fitting combination was chosen. Additionally we created the graphical notation files (Systems Biology Graphical Notation Markup Language (SBGN-ML) and Systems Biology Graphical Notation (SBGN)), that are needed to visualize the simulation results. Finally we reproduced the simulation data in a Tellurium [4] script and assembled a COMBINE archive that is publicly available.

Results: A powerful working environment could be created in the GitHub repository. Content versioning, code deposition and development as well as a wiki documentation delivered a prompt and sound scaffold for the development of the COMBINE archive. By systematic searches we identified the BACHMANN2011.XML (BIOMD0000000861) as the appropriate model file and were able to add the needed SBGN-ML and SBGN code. To fuse the above simulation components a Tellurium [4] script was developed to reproduce the numeric and graphical results of the original paper. The comprehensive information that was acquired by then was assembled within a COMBINE archive and is now able to reproduce merely all simulation results.

Conclusions: In this case study we could demonstrate the value of agile co-working and the possibility of a low-threshold approach for the development of a fully-featured combine archive of the Bachmann model. Although we we could not reproduce the full extend of the simulations given in the original work due to missing details in the publication we were able to create a fully functional archive that improves the reproducibility and availability of the original research results.

Introduction

Scientific background

Reproducibility of results is one of the most fundamental requirement for credibility in scientifc research [5]. However, in recent years it became evident that a substantial part of study results are not reproducible. A recent systematic review found that almost 50% of modeling studies could not be reproduced [5]. Especially in systems biology and systems medicine, the increase in size and complexity proposes new challenges to share reproducible results. Therefore, several initiatives such as the FAIDROM or COMBINE were formed aiming to develop and provide standardization efforts and tools to enhance reproducibility in systems biology [6]. One of these tools are so called COMBINE archives, which aim to improve the coordination of standard formats for several features of simulation studies, such as SBML, CellML, SBGN, and Systems Biology Result Markup Language (SBRML). These standards aim to encode, simulate and visualize biological models [1]. However, there is no comparable tool available, yet the number of modeling studies providing the data-set and meta-data in form of a COMBINE archives are limited.

To address this constraint COMBINE archive offer the unique opportunity to not only reproduce simulation results but also to access comprehensive metadata such as author information, publication IDs (e.g. Digital Object Identifier (DOI)) and simulation details in one single file. The vast majority of this information usually is stored in different data formats and locations that require a bundle of software tools to handle. Combine instead brings a single executable file, which is easy to access and comes with proper provenience information. It is obvious that this creates a much higher accessibility of complex data that derives from systems medicine and systems biology for researchers and provides a better reproducibility of scientific results.

Rationale for this study

Dyamic Pathway Simulations

Dynamic pathway modeling is needed to describe the complex regulatory system of feedback regulators, to answer this question, Bachmann *et al.* built a dual negative feedback model of JAK2/STAT5 signaling in primary erythroid progenitor cells isolated from mouse fetal livers [3].

Given the background of the only partially reproducible dynamic pathway model of SOCS family members in JAK2/STAT5 signaling from Bachmann *et al.* [3], we were asked to to create a fully featured archive and reproduce the simulation content both for educational purposes and for scientific completion of the original work. Furthermore, the created COMBINE Archive could serve as a template to enhance reproducibility of future modelling studies.

Agile working

Agile working has been a major drive for the evolution of working environment sespecially in information technologies. New definitions on how, where, with whom and when collaboration and the completion of tasks is done are enabled by digital cloud solutions and co-working platforms that integrate the allocation of tasks, versioning of content and the *ad-hoc*-formation of teams. GitHub as a provider of internet-hosted software

development and version control tools has been used as an essential common platform for managing software projects and supporting collaborative development. Lately some educational projects have begun to adopt it for hosting and managing course content to enhance transparency features in the creation, reuse, and remix of materials [2, 7]. In the development of this COMBINE archive we dedicated ourselves to the FAIR-principles and therefore built a completely publicly traceable working environment in github, that can be accessed via the link given in the appendices.

Objectives

The main objective of this case study was to create a fully featured COMBINE-archive including both scripts to reproduce all simulation figures and easy to access simulation data of the Bachmann model, a dynamic pathway model of JAK/STAT5 signaling [3]. Furthermore, we aimed to create an easy to use guideline on how to compile a COMBINE archive out of an existing simulation model. Besides these aims we were also interested in validating all generated scripts using already established tools. Finally, we aimed to evaluate the used tools in terms of usability and assessed the completeness of the provided data to reproduce the entire study.

Study design

This case study describes the implementation process of a COMBINE archive using agile co-working and publicly available documentation.

Research questions

The research question was whether there is a comprehensive way of developing a COMBINE archive out of publicly available data sources that allow the public to reproduce all simulation data including the resulting graphs. As secondary question we tested if this task could be performed, documented and archived in an agile working environment and according to the FAIR-principles.

Analysis procedures

To analyze the function of the COMBINE-archive we run several own simulations during the development process and compared the results with existing data from the original publication and with existing graphs ibidem.

Validity procedures

To validate the results of our COMBINE archive we conducted a systematic search for available modeling data, tested the content on a metadata level and created SBGN- and Simulation Experiment Description Markup Language (SED-ML)-files that were challenged against the original data in numeric and graphic analyses.

Materials and methods

The development of the COMBINE Archive was performed in accordance with a recently published guideline from Schwarm and Waltemath [1].

The tasks of this project were distributed on four teams and split into the sections

- Setup of an agile working environment

- Systematic review of existing materials and comparison of provided models

- Graphical representation

- Supply of a model script

Setup of an agile working environment

To provide the model from Bachmann *et al.* [3] as a fully featured COMBINE archive, we created a public repository using the open-source platform GitHub, with a Creative Commons (CC)0-1.0-license. We chose GitHub as a data management platform to supervise the course of the project as it provides an intuitive and easily customizable environment, along with features for documentation, and agile project management [2]. This repository contains the proposal directory's structure from Scharm & Waltemath [1] with the following directories:

- Documentation (files describing the model and its characteristics)

- Model (files describing and encoding the biological system, e.g. SBML format)

- Experiment (files describing and encoding the experimental setup, e.g. SED-ML)

- Result (files obtained from running *in silico* experiments, including plots and tables)

Systematic review of existing materials and comparison of provided models

In our systematic search, we found five SBML-Bachmann models in two different repositories, JWS Online and BioModels (see Baseline data). We chose the latest model dating from 14th November 2019 as it provided complementary files for the simulation.

das hier sind eigentlich results:

Software tools and Versions

Our search revealed five software tools for the simulation of biological systems:

1. **COmplex PAthway SImulator (COPASI)**
   A software application for the simulation and analysis of biochemical networks and their dynamics [8].

2. **JWS Online**
   Systems Biology tool for the construction, modification, and simulation of kinetic models and the storage of curated models [9]. On this repository was one of the found models.

3. **SED-ML_Web_Tools (SWT)**
   A suite of tools for creating, editing, simulating and validating SED-ML files [10].

4. **Tellurium**
   A tool to model, simulate and analyze biochemical systems [4].

5. **Webviewer Uni Rostock** (CombineArchiveWeb)
   A tool to visualize and manage COMBINE files [1]; this is the target application of our project.

Graphical representation

One of the objectives of our project was to provide a standardized graphical representation of the Bachmann model based on the SBGN. We performed research but could not find any SBGN of this model. Therefore, we decided to create an SBGN network *de novo* based on previous works of Le Novère [11] and Touré *et al.* [12]. In this step, we selected the SBGN language, and lastly created the Process Description (PD) map with the web tool Newt Editor (v3.0.3) [13].

   To validate the SBGN-ML we, imported it into several software and libraries including LibSBGN from Newt Editor, the open-source software Visualisation and Analysis of Networks containing Experimental Data (VANTED) [14], Krayon for SBGN [15], and SBGNViz [16]. Lastly, we cleaned up the map and colored the relevant features in the model to improve the developed map.

Supply of a model script

this section needs maintainance. soll das wirklich hier so stehen? oder ist das der abschnitt über die auswahl des richtigen files?

SED-ML Generation and Validation

For the development of new SED-ML files in order to reproduce the simulations performed by Bachmann *et al.*, we made use of two open source software tools:

1. default simulation in SED-ML WebTools as a basis for experiment-specific simulations in COPASI (as described by Scharm & Waltemath [1])

2. experiment-specific simulations in Tellurium, a Python platform for systems biology [4]

   The created SED-ML files were validated in SED-ML WebTools and integrated into the COMBINE archive. Subsequently, output files were created by

1. simulating all SED-ML files within the COMBINE archive using Tellurium and

2. loading individual SED-ML files in COPASI or SED-ML WebTools, and included in the COMBINE archive.

Assembly of the COMBINE archive

The final assembly of our fully-featured COMBINE archive of the Bachmann model was executed in the web platform *CombineArchiveWeb*. We used the same structure as described by Bachmann *et al.* (see 'setup of an agile working environment'), and extended the scaffold by additional sub-folders containing supplementary files. Thus we created

a new empty archive with our credentials on the *CombineArchiveWeb* with the name ::bachmann. The metadata section of our COMBINE archive includes a brief description of the study and the contact data of the members.

The folder 'documentation' includes the initial paper from Bachmann *et al.* with its additional materials, the paper from Scharm & Waltemath, and the file 'SystemsBiology.bib' with the used literature in our project. 'model' contains files with the latest version of the Bachmann model from 2019 (see 'Model selection') and the sub-folder sbgn, it contains pictures, SBGN-ML files, and other supplementary files to visualize the model. The folder 'experiment' provides Python-scripts, SED-ML files, and other additional files to support the simulation. The results of the different simulations and validations are in the folder 'result'. This folder also contains the sub-folders 'Default', 'Fig(3-5)', 'SuppFig9', and 'SuppFig9_COPASI' that incorporate comma-separated values (CSV) files with tables, PDF files with reports, and pictures of the performed simulations with different tools. 'result' contains a sub-folder 'validation' including the additional folders 'Test_ COPASI_2021-06-28' and 'Test_SWT_2021-06-28' with data files of the performed validations from COPASI and SWT.

The Archive can be accesses via the link in the 'appendices'-section.

Results

Systems Biology Markup Language (SBML) results

In our systematic search we found five available SBML-Bachmann models, one of them as support information of Bachmann *et al.* [3], this was the first delivered model. The others originated from two different repositories, JWS Online and BioModels.

We found three models on BioModels. The first model, 'BIOMD0000000347_url.xml', was submitted on 22nd July 2011 and modified on 31st January 2012. Together with this model were other files in different formats. Most of them were generated by tools to simulate, visualize, validate and document the model, one of them is another SBML model ('BIOMD0000000347_urn.xml').

The third and newest, 'Bachmann2011.xml', was posted on 14th November 2019. This file contained other complementary files for the simulation of this model. The models provided in JWS Online do not have any date of building or update, so it was not possible to know when these were built.

The first model in JWS online, 'bachmann.xml', is from *Mus musculus* and represents the STAT's pathway in a cell simulation *in silico*.

The second model, 'bachmann2.xml', was obtained from the BioModels database (BioModels ID: BIOMD0000000347).

Systems Biology Graphical Notation (SBGN) results

Figurative representation of data is a key factor to the rapid perception of information from it. The provision of the graphical representation of a model in a standardized, unambiguous form fosters the reusability and exchangeability of the model. The publication of the Bachmann model contains a process diagram of the model, which not uses a standard graphical notation.

Since no off-the-shelf standardised graphical representation could be found we decided to create our own SBGN map from scratch. Based on the authors´ diagram choice in the

publication and the suggestions of Le Novère *et al.* [11] and Touré *et al.* [12] we chose the 'Process Description' language for our COMBINE archive. Following the selection of the SBGN language, we identified several useful tools and ultimately created the PD map with 'Newt Editor'.

To confirm the validity of the developed SBGN we used the integrated LibSBGN-based validation feature of 'Newt Editor' as well as imported the SBGN-ML into VANTED/SBGN-ED, Krayon for SBGN and SBGNViz. **Figure 1** shows the visually improved map after cleaning it up and colouring the most important features of the model.

Simulation Experiment Description Markup Language (SED-ML) results

Following a review of available tools for creating SED-ML files [17] we decided on two separate approaches. Firstly we conducted default simulation in SED-ML WebTools as a basis for experiment-specific simulations in COPASI (as described by Scharm *et al.* [1]). Secondly we performed experiment-specific simulations in Tellurium, a Python platform for systems biology [4]. Tellurium includes a number of pre-installed Python libraries, plugins and tools for biological modelling and can be used via any Python frontend. We used Google Colaboratory (Colab), a cloud-based Jupyter notebook service for collaborative Python programming. The detailled documentation to the script can be found in the GitHub.

The created SED-ML files were validated in SED-ML WebTools and integrated into the COMBINE archive. Subsequently the output files were created by simulating all SED-ML files within the COMBINE archive using Tellurium and loading individual SED-ML files in COPASI or SED-ML WebTools.

Unexpected events and observations

Interestingly some required parameters to simulate specific experimental conditions were not included in the model. That way some model outputs are defined by observation functions that were not included as parameters in the model itself.

The plots for pJAK2, pEPOR and tSTAT5 (e.g., Fig3, Fig4) could not be generated since these parameters are missing from the model itself. Information on how to derive these observables from existing model parameters is included in the supplementary material (e.g., Supplementary Information, page 21).

In contrast to CIS and SOCS3, the overexpression of SHP1 could not be reproduced since the corresponding parameter (SHP1oe) was not included in the model (nor in any other Bachmann model files available online). We did not modify the model in order to simulate different level of Epo as described in the paper. While the corresponding figure legends and supplementary material contain some information on Epo concentrations used for these experiments, it was not clear ho to simulate this in the model.

Many figures (e.g. Figure 4) contain experimental data in addition to model simulations. Although this data is referenced as supplementary material in the online version of the paper, we did not include this in our plots. The ranges of the y axes do not correspond to the figures in the paper. Several experiments presented in the paper are based on testing a range of Erythropoietin (Epo) levels. We were not able to reproduce these experiments, because it was not clear how to simulate different Epo concentrations using the model.

Finally, we uploaded all data from our GitHub repository into the platform *CombineArchiveWeb*, except generated XML files report during the validations, because, the *CombineArchiveWeb* interface launches the error: Unknown Error: Cannot upload file, therefore we created a new ticket-issue in the GitHub repository from the *CombineArchiveWeb* project.

Reproducibility

As mentioned in the results section there have been some flaws in the reproducibility due to missing data given in the original work [3]. Nevertheless at the end of our project, we could build a fully-featured COMBINE archive and integrate the generated files, reports and pictures of the Bachmann model in the web platform CombineArchiveWeb.

Discussion

Systems biology studies generate many data files for the simulation of biological systems. These data files can be stored in COMBINE archives, standardized containers that facilitate the exchange of information from simulation studies [18], such as the fully-featured COMBINE archive described by Scharm *et al.* [1].

While we were able to re-create some (sub-) figures of the original paper using the model selected for this COMBINE archive, we encountered several problems. These will be addressed in separate GitHub issues for future reference (maybe for the authors of the original paper).

In reference to our study questions we could show that an agile working environment enables a comprehensive co-working structure in an educational setting. Here we could effectively develop a fully-featured COMBINE archive and reproduce a vast majority of the results given in the original work. The strength of our work beside the technical and documental tasks is to point our missing data or variables in the original work. Due to restricted course duration the main shortcoming of our study is that we were not able to resolve those issues within the project duration.

Given the vast extend of research and consecutive data in the field of system biology we could show that the assembly of a COMBINE archive is feasible, achievable and can be documented in a reproducible way with open source tools. This may be of utter importance for the future approach to systems medicine results.

Yet some questions remain unsolved at this time. We aimed to provide our SBGN model in a variety of formats to allow recurrence on the substitute files when the primary files are not applicable in a specific future application. Finally, we were able to provide different SBGN-ML 0.2 codes as well as a GraphML version of our model. The CellDesigner export did not work and also we provided no SBGN-ML 0.3 because we were not able to validate this with at least one other tool than Newt editor. Unfortunately, we were not able to further investigate the platform specific problems with exchange formats in detail as we experienced them, but provide a documentation for the community. During the time of our project, the CellDesigner export from Newt editor is already adressed in the discussion to an existing issue on GitHub (https://github.com/iVis-at-Bilkent/newt/issues/498). Also, we created a new issue adressing the manual modifications to the provided SBGN-ML version 0.2 main file to make it exchangeable to inform the developers of Newt editor (https://github.com/iVis-at-Bilkent/newt/issues/679).

An interesting field for future projects is definitely the integration of semantic annotations and functional aspects of the model into the SBGN map, with promising publications and reports from the community. As the approach in [2] to automatically create a first draft of SBGN was not suitable for the existing SBML files of the Bachmann model in our project, development of a guideline for modification of the SBML file to improve automated creation of SBGN would be very helpful.

Conclusions and Future work

In summary, in this case study we were able to reproduce a systems biology simulation model by Bachmann *et al.* [3] and add the necessary annotation to create a fully-featured COMBINE archive. This sprint collaboration was conducted in an agile environment using GuitHub that appeared to be a easy to use environment for team members of different experience to that tool. A fully publicly traceable documentation of the development and a reproducible simulation archive could be achieved in the context of a collaboratory educational sprint.

Limitations

Several parameters required to reproduce the full extend of individual plots were not included in the original data itself and therefor could not be reproduced. Given sthe sprint character of this collaboration we were not able to to added the missing data manually based on details provided in the supplementary material. Also some technical issues hindered a barrier-free assembly of the archive. The *CobineArchiveWeb* interface did not allow the XML reports from the validations to be be uploaded. Furthermore COPASI is limited in its functionality to support simulations working with more than one model. As this functionality is needed in our project to run simulations in the wildtype and overexpression conditions in parallel we changed our procedure and ultimately built and simulated SED-ML files using Tellurium.

Future work

The co-working environment in a GitHub appeared to be very easy to use and understand throughout the whole project team. We therefore conclude that a publicly available documentation of scientific collaboration and the development of repositories, code and archives can be facilitating Findable, Accessible, Interoperable, Reusable (FAIR) handling of research both in the process of making as well as in the handling of results itself. The authors suggest that, given the easy to access and freely available software tools, more researchers will take the effort to share their work in a traceable and reusable manner. Also in the context of corporate class work our results indicate a high potential to form a precedent for future development of FAIR research output.

Acronyms

BIDS   Biomedical Informatics and Data Science

CC   Creative Commons

COMBINE Computational Modeling in Biology Network

COPASI COmplex PAthway SImulator

CSV   comma-separated values

DOI   Digital Object Identifier

Epo   Erythropoietin

FAIR   Findable, Accessible, Interoperable, Reusable

MII   Medical Informatics Initiative

MIRACUM Medical Informatics for Research and Care in University Medicine

PD   Process Description

SED-ML Simulation Experiment Description Markup Language

SBGN   Systems Biology Graphical Notation

SBGN-ML Systems Biology Graphical Notation Markup Language

SBML   Systems Biology Markup Language

SBRML Systems Biology Result Markup Language

SOCS   suppressor of cytokine

SWT   SED-ML_Web_Tools

VANTED Visualisation and Analysis of Networks containing Experimental Data

References

[1] Martin Scharm and Dagmar Waltemath. "A fully featured COMBINE archive of a simulation study on syncytial mitotic cycles in Drosophila embryos". In: *Eurosurveillance* 5.2421 (2016). DOI: https://doi.org/10.12688/f1000research.9379.1.

[2] Joseph Feliciano, Margaret-Anne Storey, and Alexey Zagalsky. "Student Experiences Using GitHub in Software Engineering Courses: A Case Study". In: *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*. 2016, pp. 422–431. ISBN: 978-1-5090-2245-8.

[3] Julie Bachmann, Andreas Raue, Marcel Schilling, Martin E Böhm, Clemens Kreutz, Daniel Kaschek, Hauke Busch, Norbert Gretz, Wolf D Lehmann, Jens Timmer, and Ursula Klingmüller. "Division of labor by dual feedback regulators controls JAK2/STAT5 signaling over broad ligand range". In: *Molecular Systems Biology* 7.1 (2011), p. 516. DOI: https://doi.org/10.1038/msb.2011.50.

[4] J. Medley, Kiri Choi, Matthias König, Lucian Smith, Stanley Gu, Joseph Hellerstein, Stuart Sealfon, and Herbert Sauro. "Tellurium notebooks—An environment for reproducible dynamical modeling in systems biology". In: *PLOS Computational Biology* 14 (2018). DOI: 10.1371/journal.pcbi.1006220.

[5] Krishna Tiwari, Sarubini Kananathan, Matthew G Roberts, Johannes P Meyer, Mohammad Umer Sharif Shohan, Ashley Xavier, Matthieu Maire, Ahmad Zyoud, Jinghao Men, Szeyi Ng, et al. "Reproducibility in systems biology modelling". In: *Molecular Systems Biology* 17.2 (2021), e9982.

[6] Falk Schreiber, Gary D. Bader, Padraig Gleeson, Martin Golebiewski, Michael Hucka, Sarah M. Keating, Nicolas Le Novère, Chris Myers, David Nickerson, Björn Sommer, and Dagmar Waltemath. "Specifications of Standards in Systems and Synthetic Biology: Status and Developments in 2017". In: *Journal of Integrative Bioinformatics* 15.1 (Mar. 2018), p. 20180013. ISSN: 1613-4516. DOI: 10.1515/jib-2018-0013. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6167034/ (visited on 09/30/2021).

[7] Leonard Knegendorf, Felix Patricius Hans, and Carmen Haubold. *Github: Fuenfgeld/ATeamDatenmanagementUndArchivierung.* URL: https://github.com/Fuenfgeld/ATeamDatenmanagementUndArchivierung (visited on 09/30/2021).

[8] Stefan Hoops, Sven Sahle, Ralph Gauges, Christine Lee, Jürgen Pahle, Natalia Simus, Mudita Singhal, Liang Xu, Pedro Mendes, and Ursula Kummer. "COPASI—a COmplex PAthway SImulator". In: *Bioinformatics* 22.24 (Oct. 2006), pp. 3067–3074. DOI: 10.1093/bioinformatics/btl485. URL: https://doi.org/10.1093/bioinformatics/btl485.

[9] Brett G. Olivier and Jacky L. Snoep. "Web-based kinetic modelling using JWS Online". In: *Bioinformatics* 20.13 (Apr. 2004), pp. 2143–2144. DOI: 10.1093/bioinformatics/bth200.

[10] Dagmar Waltemath, Richard Adams, Frank T. Bergmann, Michael Hucka, Fedor Kolpakov, Andrew K. Miller, Ion I. Moraru, David Nickerson, Sven Sahle, Jacky L. Snoep, and Nicolas Le Novère. "Reproducible computational biology experiments with SED-ML - The Simulation Experiment Description Markup Language". In: *BMC Systems Biology* 5.198 (2011). DOI: 10.1186/1752-0509-5-198.

[11] Nicolas Le Novère, Michael Hucka, Huaiyu Mi, Stuart Moodie, Falk Schreiber, Anatoly Sorokin, Emek Demir, Katja Wegner, Mirit I. Aladjem, Sarala M. Wimalaratne, Frank T. Bergman, Ralph Gauges, Peter Ghazal, Hideya Kawaji, Lu Li, Yukiko Matsuoka, Alice Villéger, Sarah E. Boyd, Laurence Calzone, Melanie Courtot, Ugur Dogrusoz, Tom C. Freeman, Akira Funahashi, Samik Ghosh, Akiya Jouraku, Sohyoung Kim, Fedor Kolpakov, Augustin Luna, Sven Sahle, Esther Schmidt, Steven Watterson, Guanming Wu, Igor Goryanin, Douglas B. Kell, Chris Sander, Herbert Sauro, Jacky L. Snoep, Kurt Kohn, and Hiroaki Kitano. "The Systems Biology Graphical Notation". In: *Nature Biotechnology* 27.8 (2009), pp. 735–741. DOI: 10.1038/nbt.1558.

[12] Vasundra Touré, Nicolas Le Novère, Dagmar Waltemath, and Olaf Wolkenhauer. "Quick tips for creating effective and impactful biological pathways using the Systems Biology Graphical Notation". In: *PLOS Computational Biology* 14.2 (2018). DOI: 10.1371/journal.pcbi.1005740.

[13] Hasan Balci, Metin Can Siper, Nasim Saleh, Ilkin Safarli, Ludovic Roy, Merve Kilicarslan, Rumeysa Ozaydin, Alexander Mazein, Charles Auffray, Özgün Babur, Emek Demir, and Ugur Dogrusoz. "Newt: a comprehensive web-based tool for viewing, constructing and analyzing biological maps". In: *Bioinformatics* 37.10 (Nov. 2020), pp. 1475–1477. DOI: 10.1093/bioinformatics/btaa850.

[14] Hendrik Rohn, Astrid Junker, Anja Hartmann, Eva Grafahrend-Belau, Hendrik Treutler, Matthias Klapperstück, Tobias Czauderna, Christian Klukas, and Falk Schreiber. "VANTED v2: a framework for systems biology applications". In: *BMC Systems Biology* 6.1 (2012), p. 139. ISSN: 1752-0509. DOI: 10.1186/1752-0509-6-139.

[15] Frank T. Bergmann, Tobias Czauderna, Ugur Dogrusoz, Adrien Rougny, Andreas Dräger, Vasundra Touré, Alexander Mazein, Michael L. Blinov, and Augustin Luna. "Systems biology graphical notation markup language (SBGNML) version 0.3". In: *Journal of Integrative Bioinformatics* 17.2-3 (2020). DOI: doi:10.1515/jib-2020-0016.

[16] Mecit Sari, Istemi Bahceci, Ugur Dogrusoz, Selcuk Onur Sumer, Bülent Arman Aksoy, Özgün Babur, and Emek Demir. "SBGNViz: A Tool for Visualization and Complexity Management of SBGN Process Description Maps". In: *PLOS ONE* 10.6 (2015), pp. 1–14. DOI: 10.1371/journal.pone.0128985.

[17] sed-ml org sed-ml. *SED-ML — Software tools and libraries*. 2021. URL: https://sed-ml.org/showcase.html (visited on 09/30/2021).

[18] Frank T. Bergmann, Richard Adams, Stuart Moodie, Jonathan Cooper, Mihai Glont, Martin Golebiewski, Michael Hucka, Camille Laibe, Andrew K. Miller, David P. Nickerson, Brett G. Olivier, Nicolas Rodriguez, Herbert M. Sauro, Martin Scharm, Stian Soiland-Reyes, Dagmar Waltemath, Florent Yvon, and Nicolas Le Novère. "COMBINE archive and OMEX format: one file to share all information to reproduce a modeling project". In: *BMC Bioinformatics* 15.1 (2014). DOI: 10.1186/s12859-014-0369-z.

[19] Abel Hodelín Hernández, Leonard Knegendorf, Jessica Vasseur, Marisa Kaup, Oliver Hölsken, Felix Patricius Hans, Tania Lüty, Felix Schönecker, Andreas Bucher, and Dagmar Waltemath. *ahodelin/Bachmann_Archive: COMBINE_Bachmann*. 2021. DOI: 10.5281/zenodo.5543360.

Appendices

GitHub Repository

Our public GitHub repository is available here: `https://github.com/ahodelin/Bachmann_Archive` [19]. This repository is a collaborative effort of participants of the module "Bioinformatik und Systembiologie" in the Master Program BIDS, organized by the Graduate School Rhein-Neckar in cooperation with the MIRACUM, a consortium of the Medical Informatics Initiative (MII).

COMBINE Archive

The fully featured COMBINE Archive of our project is available here: `https://cat.bio.informatik.uni-rostock.de/rest/share/6b507398-c103-4b92-9c90-d6b105ff1372` under the vignette **::bachmann**. The original publication with the used model (Bachmann *et al.*) is available here: `https://doi.org/https://doi.org/10.1038/msb.2011.50`.