RESEARCH ARTICLE

BD5: An open HDF5-based data format to represent quantitative biological dynamics data

Koji Kyoda[1,2], Kenneth H. L. Ho[1,2], Yukako Tohsato[1,2,3], Hiroya Itoga[1], Shuichi Onami[1,2]*

1 Laboratory for Developmental Dynamics, RIKEN Center for Biosystems Dynamics Research, Kobe, Japan, **2** Laboratory for Developmental Dynamics, RIKEN Quantitative Biology Center, Kobe, Japan, **3** Department of Information Science and Engineering, Ritsumeikan University, Shiga, Japan

* sonami@riken.jp, sonami@gsc.riken.jp

OPEN ACCESS

Citation: Kyoda K, Ho KHL, Tohsato Y, Itoga H, Onami S (2020) BD5: An open HDF5-based data format to represent quantitative biological dynamics data. PLoS ONE 15(8): e0237468. https://doi.org/10.1371/journal.pone.0237468

Editor: Jishnu Das, University of Pittsburgh, UNITED STATES

Received: April 27, 2020

Accepted: July 27, 2020

Published: August 12, 2020

Data Availability Statement: The data underlying the results presented in the study are available from http://ssbd.qbic.riken.jp and https://github.com/openssbd.

Funding: This work was supported in part by the National Bioscience Database Center (NBDC) of the Japan Science and Technology Agency (JST) (to S. O.); Core Research for Evolutionary Science and Technology (CREST) Grant Number JPMJCR1511, JST (to S.O.); JSPS KAKENHI Grant Number JP18H05412 (to S.O.); the Strategic Programs for R&D (President's Discretionary Fund) of RIKEN,

Abstract

BD5 is a new binary data format based on HDF5 (hierarchical data format version 5). It can be used for representing quantitative biological dynamics data obtained from bioimage informatics techniques and mechanobiological simulations. Biological Dynamics Markup Language (BDML) is an XML (Extensible Markup Language)-based open format that is also used to represent such data; however, it becomes difficult to access quantitative data in BDML files when the file size is large because parsing XML-based files requires large computational resources to first read the whole file sequentially into computer memory. BD5 enables fast random (i.e., direct) access to quantitative data on disk without parsing the entire file. Therefore, it allows practical reuse of data for understanding biological mechanisms underlying the dynamics.

Introduction

Recent advances in bioimage informatics and mechanobiological simulation techniques have led to the production of a large amount of quantitative data of spatiotemporal dynamics of biological objects ranging from molecules to organisms [1]. A wide variety of such data can be described in an open unified data format Biological Dynamics Markup Language (BDML), an Extensible Markup Language (XML)-based format [2]. BDML enables efficient development and evaluation of software tools for a wide range of applications.

The XML-based BDML format has the advantages of machine/human readability and extensibility. However, it is often problematic for accessing and retrieving data when the size of the BDML file becomes too large (e.g., our programs cannot load a BDML file over 20 GB on a standard workstation). This problem arises because parsing an XML-based file often requires large computational resources to first read the whole file sequentially into computer memory. In fact, many sets of quantitative data stored in the SSBD:database (Systems Science of Biological Dynamics database) [1] were divided into a series of BDML files for each time point to allow software to read them efficiently. One of the solutions to the above problem is to

Japan (to S.O.); and Open Life Science Platform, RIKEN, Japan (to S.O.). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

use another approach such as the eXtensible Data Model and Format [3] or FieldML [4]. In these formats, the data itself is described in HDF5 binary format and meta-information about the data is described in XML format. HDF5 is a hierarchical data format for storing large scientific datasets (http://www.hdfgroup.org/HDF5/). It is widely used for describing various kinds of large-scale biological data [4–9].

Here, we describe the development of the BD5 data format, based on HDF5, for representing quantitative biological dynamics data in a manner that enables quick access and retrieval.

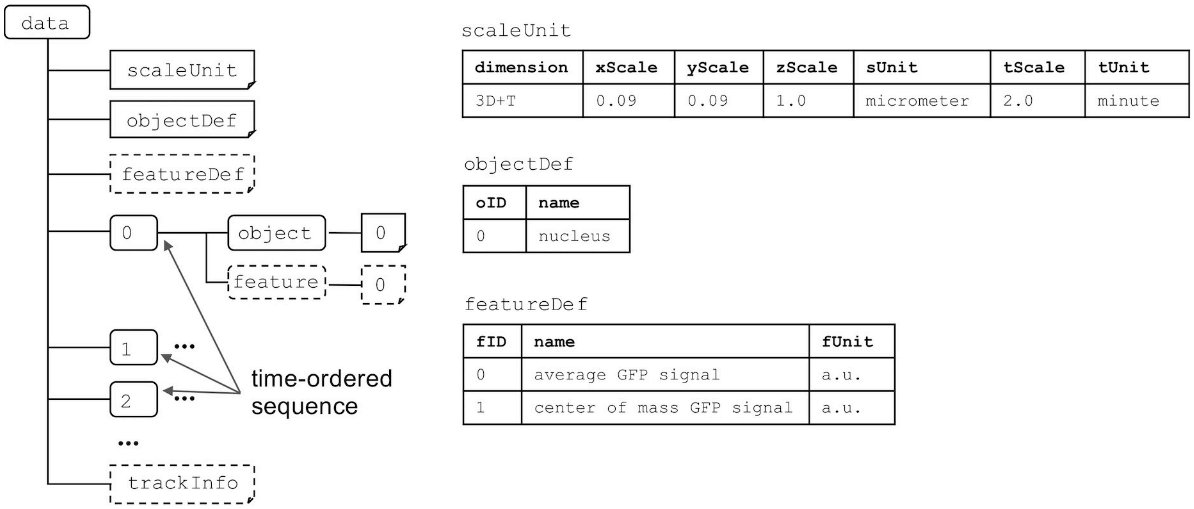
Materials and methods

Design and implementation

Here, we extended BDML to support HDF5-based storage of quantitative biological dynamics data. In contrast to XML documents, HDF5 format can allow random (i.e., direct) access to parts of the file without parsing the entire contents. Therefore, HDF5 is a more efficient file format for accessing and retrieving the contents of the file.

We developed the BD5 data format based on HDF5 for representing quantitative data. A BD5 file is organized into two primary structures, *datasets* and *groups*. Datasets are array-like objects that store numerical data, whereas groups are hierarchical containers that store datasets and other groups. Detailed information on BD5 is available at http://ssbd.qbic.riken.jp/bdml/. Here, we summarize the major BD5 datasets and groups. BD5 format has one container named *data* (Fig 1). It includes

- *scaleUnit* dataset for the definition of spatial and time scales and units,
- *objectDef* dataset for the definition of biological objects,
- *featureDef* dataset for features of interest,
- numbered groups (0, 1, . . ., n) corresponding to an index number of a time-ordered sequence,
- *trackInfo* dataset for the information of tracking of one object to another.



scaleUnit

dimension	xScale	yScale	zScale	sUnit	tScale	tUnit
3D+T	0.09	0.09	1.0	micrometer	2.0	minute

objectDef

oID	name
0	nucleus

featureDef

fID	name	fUnit
0	average GFP signal	a.u.
1	center of mass GFP signal	a.u.

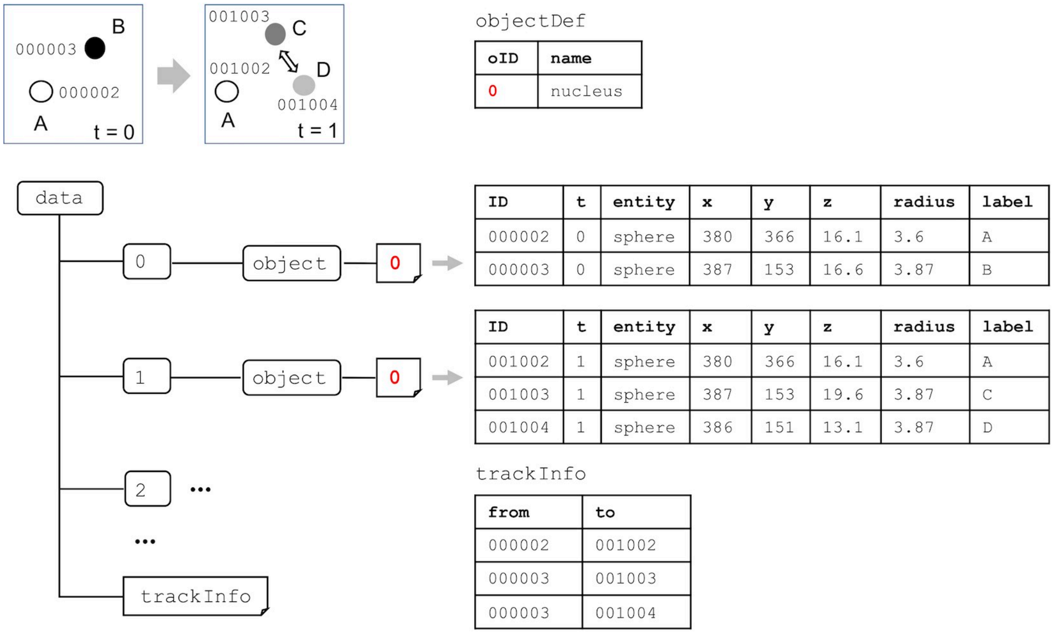
Fig 1. Outline of the BD5 data format. The *data* group includes *scaleUnit*, *objectDef*, *featureDef*, and *trackInfo* datasets; each *data* group is numbered to correspond to the index number of the time-ordered sequence. Each numbered group has spatial information about biological objects and numerical information about features related to the objects. Solid and dashed boxes represent the required and optional elements, respectively.

https://doi.org/10.1371/journal.pone.0237468.g001

Each of the numbered groups corresponds to an index of a time-ordered sequence that has *object* and *feature* groups. For a fixed time interval, the index will correspond to each sequential time point. For example, if the time interval is 2 minutes, group 0 will have t = 0 and group 1 will have t = 1, while tScale is 2 and tUnit is minute (Fig 1). For irregular time intervals, the index allows a time-ordered sequence to be saved and be read in the correct order. If the first time is 0 minutes, the second one is 2 minutes, and the third one is 7 minutes, then group 0 will have t = 0, group 1 will have t = 2 and group 2 will have t = 7. The tUnit is still minute, but the tScale in this case will be 1.

Each *object* group has numbered dataset(s) corresponding to the reference number of the biological object(s) predefined under the *objectDef* dataset. Each row of the numbered object includes an identifier of the object and its spatiotemporal information such as time point and *xyz*-coordinates (Fig 2). To represent biological objects such as line and face entities that have an arbitrary number of *xyz*-coordinates in BD5 format, a tabular dataset is used (Fig 3). The multiple *xyz*-coordinates are represented by using a sequential ID (sID) that allows us to connect the *xyz*-coordinates together to form a line or a face within a biological object. To represent multiple biological objects from different measurements, different numbered datasets are used (Fig 4). The datasets can include different entities to represent the corresponding biological objects.

Each *feature* group has numbered dataset(s) corresponding to the reference number of the object(s) predefined in the *objectDef* dataset. Each row of the numbered object includes an identifier of the object, an identifier of the feature (fID) predefined in *featureDef*, and the value of the feature (Fig 5). This format allows objects that do not possess all the features defined in *featureDef* to be recorded, because not all the features can necessarily be measured in practical biological experiments. For example, in the experiment in Fig 5, an object may have



objectDef

oID	name
0	nucleus

ID	t	entity	x	y	z	radius	label
000002	0	sphere	380	366	16.1	3.6	A
000003	0	sphere	387	153	16.6	3.87	B

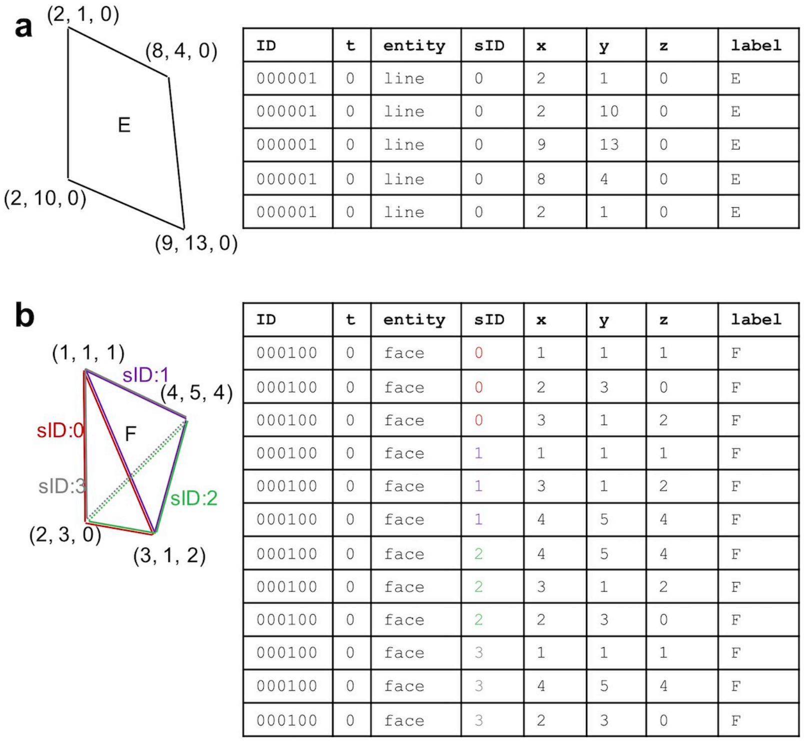
ID	t	entity	x	y	z	radius	label
001002	1	sphere	380	366	16.1	3.6	A
001003	1	sphere	387	153	19.6	3.87	C
001004	1	sphere	386	151	13.1	3.87	D

trackInfo

from	to
000002	001002
000003	001003
000003	001004

Fig 2. An example of the description of the spatiotemporal information of biological objects and their tracking information. The dataset name in *object* group corresponds to the identifier (ID) of the biological object (red). Each row in the dataset must have a unique ID and its spatiotemporal information. A label can optionally be attached for each object. The tracking information including object divisions and fusions can be stored in *trackInfo* dataset.

https://doi.org/10.1371/journal.pone.0237468.g002



ID	t	entity	sID	x	y	z	label
000001	0	line	0	2	1	0	E
000001	0	line	0	2	10	0	E
000001	0	line	0	9	13	0	E
000001	0	line	0	8	4	0	E
000001	0	line	0	2	1	0	E

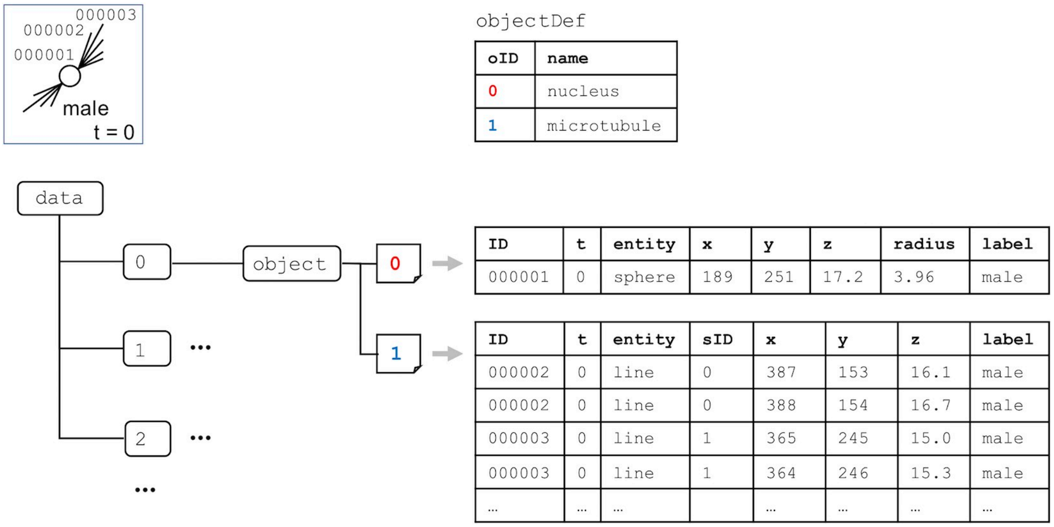
ID	t	entity	sID	x	y	z	label
000100	0	face	0	1	1	1	F
000100	0	face	0	2	3	0	F
000100	0	face	0	3	1	2	F
000100	0	face	1	1	1	1	F
000100	0	face	1	3	1	2	F
000100	0	face	1	4	5	4	F
000100	0	face	2	4	5	4	F
000100	0	face	2	3	1	2	F
000100	0	face	2	2	3	0	F
000100	0	face	3	1	1	1	F
000100	0	face	3	4	5	4	F
000100	0	face	3	2	3	0	F

Fig 3. An example of the description of the spatiotemporal information of 2D quadrilateral-shaped and 3D trigonal pyramidal objects by using line (a) and face entities (b), respectively. The sequential identifier (sID) represents a set of coordinates that can be connected from the top to describe an entity within one biological object.

https://doi.org/10.1371/journal.pone.0237468.g003

information for fID = 1 (name: center-of-mass GFP signal) but not fID = 0 (name: average GFP signal).

The *trackInfo* dataset enables information of the objects to be linked between different time points or time frames (Fig 2). For example, when a cell at t = 0 divides into two daughter cells



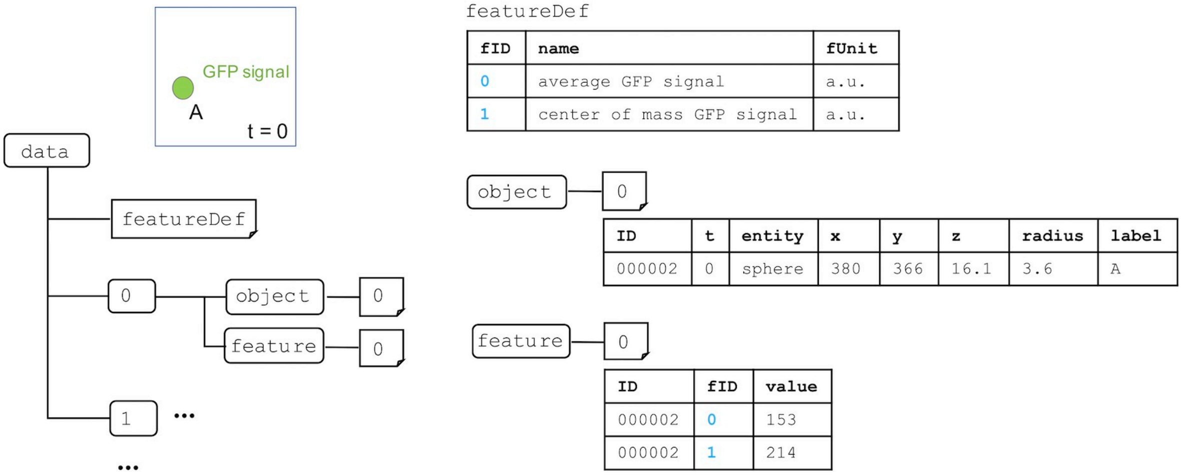
oID	name
0	nucleus
1	microtubule

ID	t	entity	x	y	z	radius	label
000001	0	sphere	189	251	17.2	3.96	male

ID	t	entity	sID	x	y	z	label
000002	0	line	0	387	153	16.1	male
000002	0	line	0	388	154	16.7	male
000003	0	line	1	365	245	15.0	male
000003	0	line	1	364	246	15.3	male
…	…	…		…	…	…	…

Fig 4. An example of the description of the spatiotemporal information of multiple biological objects (a nucleus and microtubules). The *object* group has two different datasets numbered 0 and 1 corresponding to the reference numbers of biological objects predefined in the *objectDef* dataset.

https://doi.org/10.1371/journal.pone.0237468.g004



fID	name	fUnit
0	average GFP signal	a.u.
1	center of mass GFP signal	a.u.

ID	t	entity	x	y	z	radius	label
000002	0	sphere	380	366	16.1	3.6	A

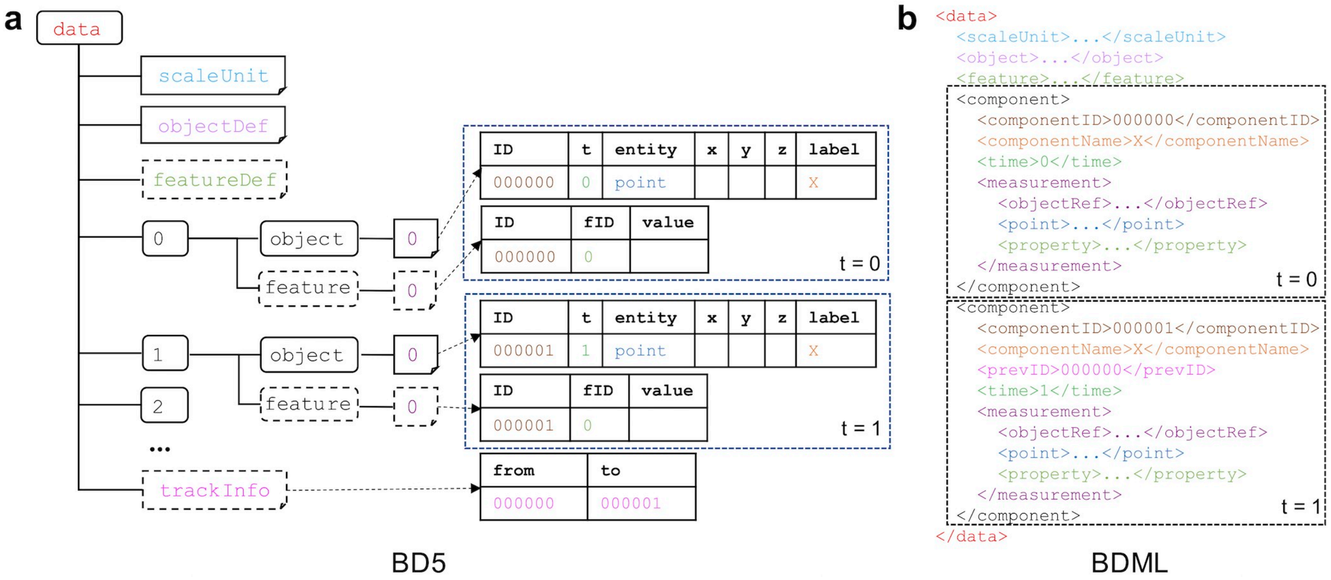
ID	fID	value
000002	0	153
000002	1	214

Fig 5. An example of the description of the feature information related to biological objects. The example object is a nucleus expressing green fluorescent protein (GFP) at t = 0 in a time series. The dataset name in *feature* group corresponds to the identifier (ID) of the biological object. Each row in the dataset has object ID, feature ID (fID, blue), and the feature value. In this example, fID is 0 for total GFP signal or 1 for average GFP signal. Object ID is 0 if the object is a nucleus. Feature value is the fluorescence intensity expressed in a.u. (arbitrary units).

https://doi.org/10.1371/journal.pone.0237468.g005

at t = 1, it has links from the parent cell to the daughter cells. The *trackInfo* dataset can be used to represent not only phenomena such as cell division but also those such as cell fusion.

To clarify the difference between BD5 and BDML, we compared the structures of the two formats. The structure of the BD5 format is based mainly on that of the BDML format (Fig 6). The *data* group in BD5 is an HDF5 data group under the root group. The name *data* originated from the top-level `data` element in BDML format. The *data* group in BD5 has the *scaleUnit*, *objectDef*, and *featureDef* datasets, each corresponding to the `scaleUnit`, `object`,



ID	t	entity	x	y	z	label
000000	0	point				X

ID	fID	value
000000	0	

ID	t	entity	x	y	z	label
000001	1	point				X

ID	fID	value
000001	0	

from	to
000000	000001

```
<data>
  <scaleUnit>...</scaleUnit>
  <object>...</object>
  <feature>...</feature>
  <component>
    <componentID>000000</componentID>
    <componentName>X</componentName>
    <time>0</time>
    <measurement>
      <objectRef>...</objectRef>
      <point>...</point>
      <property>...</property>
    </measurement>
  </component>
  <component>
    <componentID>000001</componentID>
    <componentName>X</componentName>
    <prevID>000000</prevID>
    <time>1</time>
    <measurement>
      <objectRef>...</objectRef>
      <point>...</point>
      <property>...</property>
    </measurement>
  </component>
</data>
```

Fig 6. Comparison of the structures between the BD5 (a) and BDML formats (b). Corresponding elements in the two formats have the same colors. Whereas BDML describes all the information in a linear structure, BD5 has an additional hierarchical time-ordered sequence (numbered groups) to separate spatial information about biological objects in time.

https://doi.org/10.1371/journal.pone.0237468.g006

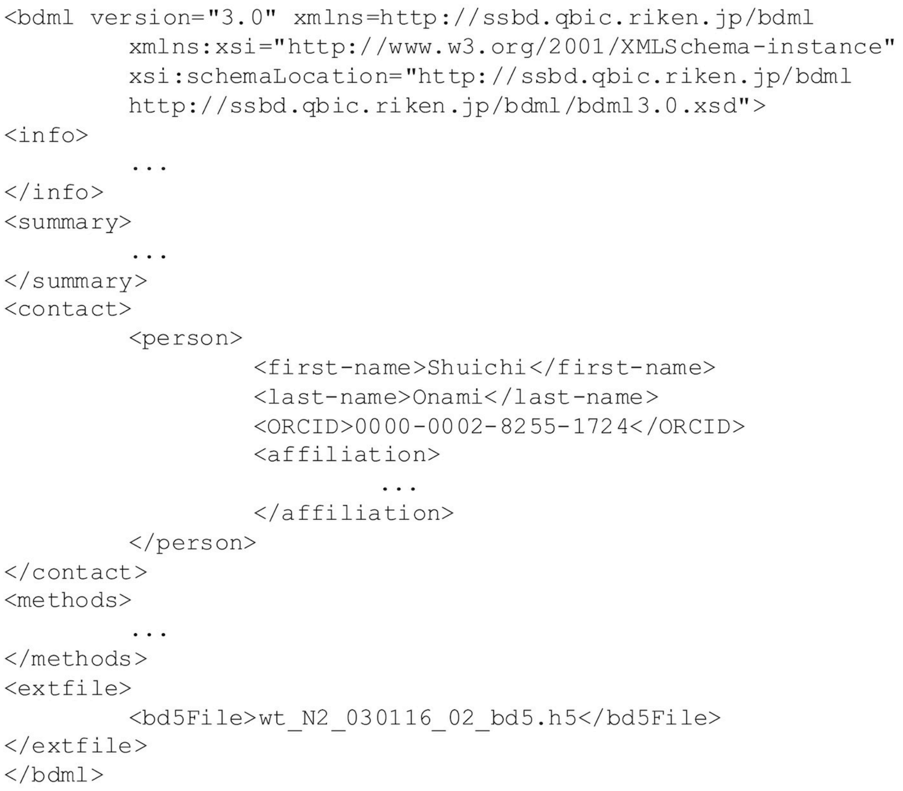
and `feature` element in BDML, respectively. The `component` element in BDML is no longer used in BD5. Instead, biological objects are stored separately in datasets that are ordered in time. Objects that have the same time are stored in the same numbered group in BD5. Each numbered dataset in the *object* and *feature* groups corresponds to each `measurement` including the *xyz*-coordinates and its `property`, respectively. In the numbered dataset in BD5, the ID corresponds to `componentID` and the label corresponds to the `componentName` element in BDML. Detected biological objects are often tracked over different time points; their tracking information can be stored separately in the *trackInfo* dataset, whereas prevID is used to describe tracking information within the `component` element in BDML.

To allow the use of BD5 to describe quantitative data, we needed to update the BDML format so that it could be used to describe the corresponding meta-information. The latest version of BDML (version 3.0) can handle an external file by using the `extFile` element (Fig 7). The `bd5File` element that we introduced within the `extFile` element can be used to point to an external BD5 file. In addition, this update allows the designation of multiple contact persons and the use of a unique persistent digital identifier, ORCID (https://orcid.org), in BDML format.

Results

Validation

To evaluate the performance of the BD5 format, we first compared the file access time between XML- and HDF5-based files (i.e., between pairs of BDML and BD5 files containing equivalent



```
<bdml version="3.0" xmlns=http://ssbd.qbic.riken.jp/bdml
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://ssbd.qbic.riken.jp/bdml
        http://ssbd.qbic.riken.jp/bdml/bdml3.0.xsd">
<info>
        ...
</info>
<summary>
        ...
</summary>
<contact>
        <person>
                <first-name>Shuichi</first-name>
                <last-name>Onami</last-name>
                <ORCID>0000-0002-8255-1724</ORCID>
                <affiliation>
                        ...
                </affiliation>
        </person>
</contact>
<methods>
        ...
</methods>
<extfile>
        <bd5File>wt_N2_030116_02_bd5.h5</bd5File>
</extfile>
</bdml>
```

Fig 7. A skeleton of a BDML version 3.0 file for describing meta-information. This version allows the use of an external file for describing the data itself, designation of multiple contact persons, and the use of ORCID, a unique persistent digital identifier of the research scientist.

https://doi.org/10.1371/journal.pone.0237468.g007

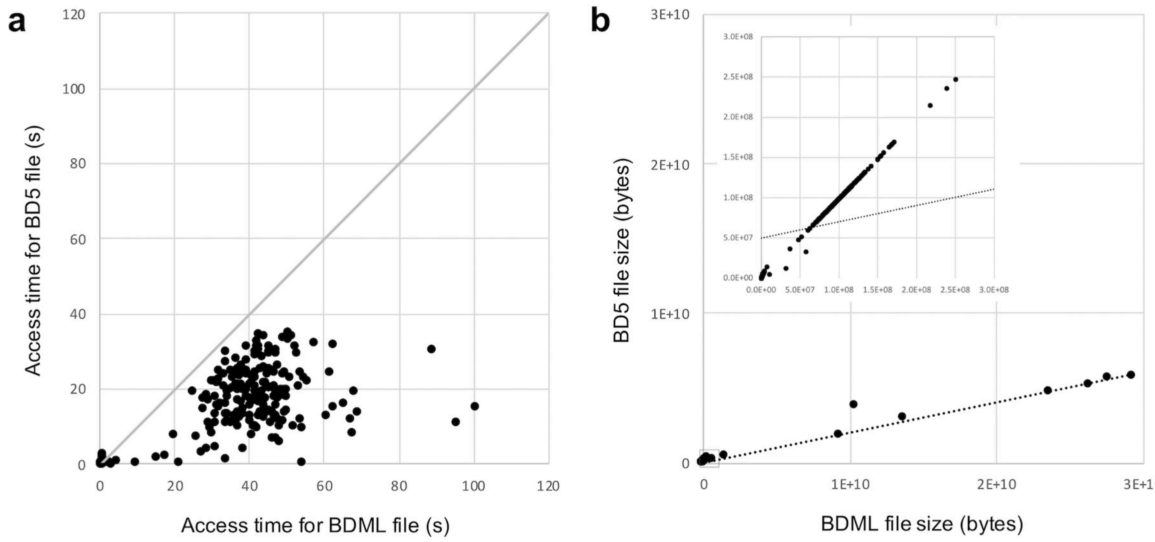


Fig 8. Comparison between the BD5 and BDML data formats. a) Access times of the BDML and BD5 files. Access time was measured as the time for accessing and displaying *xyz*-coordinate data at a randomly selected time point stored in the BDML and BD5 files. The time was measured on an Intel Xeon CPU 2.8 GHz processor with 32 GB of main memory. Each dot represents a biological quantitative dataset. We used 334 biological quantitative datasets, each of which has coordinate data and is stored in SSBD:database as a single BDML file. BD5 files were generated from the BDML files by using the BDML2BD5 program. b) Size of the BDML and BD5 files. Each dot represents a biological quantitative dataset. In this comparison, we used 347 biological quantitative datasets, each of which was produced by bioimage informatics techniques and is stored in SSBD:database as BDML file(s). As above, the BD5 files were generated from the BDML files by using the BDML2BD5 program. The dashed line represents the linear regression line for all dots. The data within the small rectangle near the origin in the large graph is plotted on expanded axes in the insert.

https://doi.org/10.1371/journal.pone.0237468.g008

data). We measured the time for accessing coordinate data at a randomly selected time point in the BDML and BD5 files (334 pairs of files) by using a Python-based program (Fig 8A). The access times of HDF5-based files were consistently faster than those of the corresponding XML-based files. Therefore, BD5, the new HDF5-based format, enables fast access to quantitative data for further analysis.

File size can be a critical benchmark for a data format because the transfer of large files often fails. Therefore, we next compared the disk space requirement between the XML- and HDF5-based files by comparing the size of BDML and BD5 files (347 pairs of files) (Fig 8B). We found that the BD5 format reduces the file size by ~80% in comparison with the BDML format when the data is large. When the data is small (< 300 MB), the size of a BD5 file is close to, but still less than, that of the corresponding BDML file. Because the size of HDF5-based files for large data is much less than that of the equivalent XML-based files, the BD5 format enables, in theory, fast transfer of large quantitative data to and from computers on a network and on the internet.

In addition, we determined the relationships between access time and file size for BDML and BD5 files (Fig 9). In BD5, we found fast access to the coordinate data even when the file size was large. This fast data access in BD5 originated from its random access to data. In BDML, the access time linearly increased with file size. This result suggests that parsing of XML was the main bottleneck of data access. Quantitative biological dynamics data tends to be large due to the advances in live-cell imaging techniques and imaging equipment. We anticipate that BD5 will play a key role in fast access to such large datasets.

Software tools and usage related to BD5

Sample programs and codes for writing and reading BD5 files using Python are provided as Jupyter notebooks at https://github.com/openssbd/BD5_samples. The codes include sample

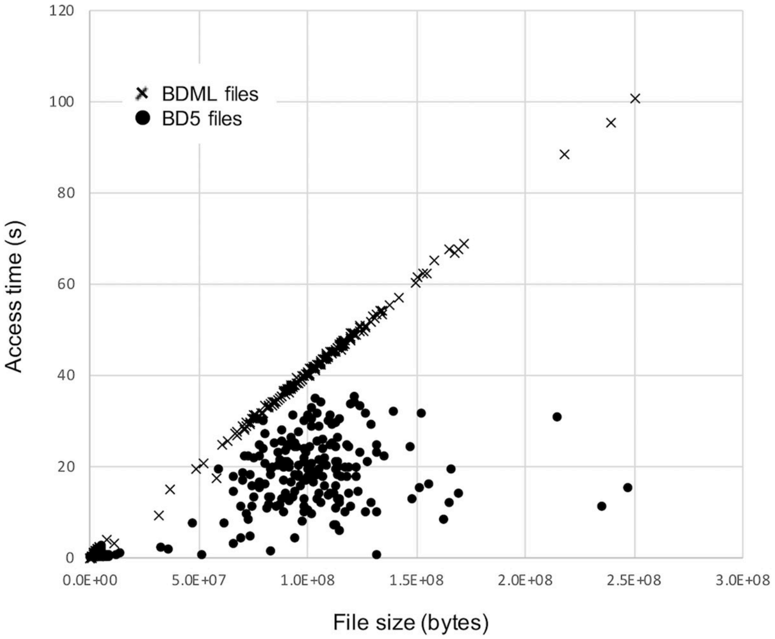


Fig 9. Relationship between access time and file size for BDML and BD5 files. The time for accessing and displaying coordinate data at a randomly selected time point is plotted against file size. Each cross represents a BDML file; each dot represents a BD5 file. In the comparison, we used BDML and BD5 files of the 334 biological quantitative datasets described in Fig 8A.

https://doi.org/10.1371/journal.pone.0237468.g009

codes that use NumPy arrays to describe five different entities (point, circle, line, sphere, and face) for representing biological objects and to write this information to a BD5 file. We also provided sample codes that read segmented regions of interest (ROIs) stored in a TIFF image file, and the ROIs are then saved in a BD5 file. Moreover, we provide a sample code to read data stored in CSV-formatted file and write the data to a BD5 file, and sample codes to read a BD5 file for further analysis using quantitative dataset. These sample codes will help developers of bioimaging software tools to understand and support BD5 format.

We also provide a program, bd5lint, for detecting bugs and inconsistencies in BD5 files. The program checks the structure of BD5 files and checks that the ordered numbered datasets in the *object* and *feature* groups correspond to the reference numbers of the objects and features predefined in the *objectDef* and *featureDef* datasets. It also checks the consistency of the dimensions declared and the actual dimensions used within the datasets. It provides type checking of the data and error warnings if the data do not conform to the BD5 specification. The Python source code is available at https://github.com/openssbd/bd5lint/.

To support data storage in older BDML files, we provide a C++-based software tool named BDML2BD5. By using this tool, BDML files can be converted into BD5 files. To compile the tool, the HDF5 library is required for HDF5 data writing, and CodeSynthesis XSD (http://www.codesynthesis.com/products/xsd/) is required for the BDML schema to C++ data-binding compiler. The source codes of the tool BDML2BD5 and a Docker container (bdml2bd5-docker) to build the executable are available at GitHub (https://github.com/openssbd/BDML2BD5/ and http://github.com/openssbd/bdml2bd5-docker, respectively).

Discussion

In this study, we developed a new BD5 data format based on HDF5 for representing quantitative biological dynamics data. Compared with BDML, which is based on XML, the BD5 format has two advantages: (a) fast access to and retrieval of quantitative data owing to random access to the HDF5-based file, and (b) fast transfer of files containing large quantitative data because the file size is dramatically reduced. A drawback of the BD5 is that human readability is low when compared with the BDML format. BD5 files cannot be opened by text editors because the file is binary formatted. However, the HDF group provides a software tool named HDFView that enables the user to open and read all HDF5-based files (https://www.hdfgroup.org/downloads/hdfview/). This tool can compensate for the lack of human readability.

Our proposed new approach uses a BD5 file to store only quantitative data, while a separate BDML file is used to store meta-information about the quantitative data. This separation of data and meta-information improves the efficiency of access to quantitative data, which tend to be large and are accessed mainly by using software tools. Reading from and writing to data storage using a binary file format are therefore more efficient than when using text-based BDML format. On the other hand, meta-information is mostly read and edited directly by users. Therefore, we keep using BDML to store meta-information to ensure that it can be easily edited by using any text editor and thereby to maintain human readability. The disadvantage of this approach is that it requires two files for data publication; such a multi-file approach often leads to the separation and fragmentation of data and meta-information. One way to solve this is to allow meta-information to be stored within a BD5 file as an option. This feature may be included in a future version of BD5 format.

Compared to the BDML format, BD5 file size is considerably smaller, but future advances in imaging techniques and devices will certainly produce larger datasets. One approach to tackle this is to use the HDF5 compression filter. We found that the filter (h5repack; https://support.hdfgroup.org/HDF5/doc/RM/Tools.html) dramatically reduced the file size, but slowed down access to the data because of the decompression cost. Therefore, we did not apply it to the BD5 files, but it can be a practical approach to reduce file size when larger datasets are produced. Another approach is to divide and store a dataset into multiple BD5 files followed by bundling these BD5 files together. To bundle multiple BD5 files, we can use external links supported by HDF5, or we can extend BDML format to handle multiple BD5 files. We will determine the specification or provide guidelines for dividing and bundling datasets in BD5 data format in the future.

BD5 format has already been used in the latest version of SSBD:database (http://ssbd.qbic.riken.jp), which is one of the major databases for sharing bioimage data and quantitative biological dynamics data [10]. At the time of writing, SSBD:database contained 687 files, which include a wide variety of quantitative biological dynamics data from molecules to cells to organisms, are available. This demonstrates that the BD5 format has high functionality and flexibility for representing quantitative biological dynamics data. SSBD:database also provides a RESTful application programming interface (API), which allows applications to access data and interact with external software tools through the use of the webservice h5serv (https://github.com/HDFGroup/h5serv). This enables SSBD:database to provide a web service for users to access quantitative data stored in BD5 files (http://ssbd.qbic.riken.jp/restfulapi/). Because HDF5 and XML are supported by many software platforms, BD5 is a promising data format for storing quantitative biological dynamics data.

Like BDML, BD5 can represent quantitative biological dynamics data that is associated with, but independent of, microscopy images. Such data has often been represented as ROIs on the corresponding microscopy images; for example, the ROIs in the OME data model (https://docs.openmicroscopy.org/ome-model/) and segmentation channels in Cell Feature Explorer (https://cfe.allencell.org). However, not all data can be represented as an ROI on a microscopy image. For example, in an automated cell lineage tracing study of *Caenorhabditis elegans*, each nucleus was represented as a sphere with center and radius, independently of the *z*-stack images [11]. Such flexible representation of BD5 (and also BDML) enables us to represent quantitative biological dynamics data obtained not only from bioimage informatics but also from mechanobiological simulation techniques.

To promote the support of BD5 format by software developers, we provide sample Python programs and Jupyter notebooks with explanations for writing and reading BD5 files. These sample programs will facilitate the understanding and use of BD5 format. For the standardization of this format, it is necessary to align our approach and our proposed format with those used by international projects and to support other open and widely used platforms. We have started to support the BD5 format on the Galaxy platform [12], e.g., https://github.com/openssbd/python-bdml. We have also been engaged in the Global BioImaging project (https://www.globalbioimaging.org/), and we are actively collaborating with the OME community (https://www.openmicroscopy.org/). We will continue to contribute to the bioimaging community by developing standard data formats for storing quantitative data of biological dynamics.

Acknowledgments

We are grateful to the members of the Onami laboratory, RIKEN Center for Biosystems Dynamics Research, Japan, for their feedback and discussions. We would also like to thank the reviewers for their technical advice and constructive suggestions.

Author Contributions

Conceptualization: Shuichi Onami.

Funding acquisition: Shuichi Onami.

Investigation: Koji Kyoda.

Software: Koji Kyoda, Kenneth H. L. Ho.

Supervision: Shuichi Onami.

Validation: Koji Kyoda, Kenneth H. L. Ho.

Writing – original draft: Koji Kyoda, Kenneth H. L. Ho.

Writing – review & editing: Yukako Tohsato, Hiroya Itoga, Shuichi Onami.

References

1. Tohsato Y, Ho KH, Kyoda K, Onami S. SSBD: a database of quantitative data of spatiotemporal dynamics of biological phenomena. Bioinformatics. 2016; 32:3471–2479. https://doi.org/10.1093/bioinformatics/btw417 PMID: 27412095
2. Kyoda K, Tohsato Y, Ho KH, Onami S. Biological Dynamics Markup Language (BDML): an open format for representing quantitative biological dynamics data. Bioinformatics. 2015; 31:1044–1052. https://doi.org/10.1093/bioinformatics/btu767 PMID: 25414366
3. Clarke JA, Mark ER. Enhancements to the eXtensible Data Model and Format (XDMF). Proceedings of the 2007 DoD High Performance Computing Modernization Program Users Group Conference; 2007 Jun; Washington DC, USA. IEEE Computer Society. pp. 322–327.
4. Britten RD, Christie GR, Little C, Miller AK, Bradley C, Wu A, et al. FieldML, a proposed open standard fort the Physiome project for mathematical model representation. Med Biol Eng Comput. 2013; 51:1191–1207. https://doi.org/10.1007/s11517-013-1097-7 PMID: 23900627
5. Baker M. Quantitative data: learning to share. Nature Meth. 2012; 9:39–41.
6. Dougherty MT, Folk MJ, Zadok E, Bernstein HJ, Bernstein FC, Eliceiri KW, et al. Unifying biological image formats with HDF5. Commun ACM. 2009; 52:42–47. https://doi.org/10.1145/1562764.1562781 PMID: 21218176

7. Hoffman MM, Buske OJ, Noble WS. The Genomedata format for storing large-scale functional genomics data. Bioinformatics. 2010; 26:1458–1459. https://doi.org/10.1093/bioinformatics/btq164 PMID: 20435580

8. Millard BL, Niepel M, Menden MP, Muhlich JL, Sorger PK. Adaptive informatics for multifactorial and high-content biological data. Nature Meth. 2011; 8:487–492.

9. Wilhelm M, Kirchner M, Steen JA, Steen H. mz5: space- and time-efficient storage of mass spectrometry data sets. Mol Cell Proteomics. 2012; 11:O111.011379. https://doi.org/10.1074/mcp.O111.011379 PMID: 21960719

10. Dance A. Find a home for every imaging data set. Nature. 2020; 579:162–163. https://doi.org/10.1038/d41586-020-00594-4 PMID: 32123362

11. Bao Z, Murray JI, Boyle T, Ooi SL, Sandel MJ, Waterston RH. Automated cell lineage tracing in *Caenorhabditis elegans*. Proc Natl Acad Sci USA. 2006; 103:2707–2712. https://doi.org/10.1073/pnas.0511111103 PMID: 16477039

12. Afgan E, Baker D, Batut B, van den Beek M, Bouvier D, Cech M, et al. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Res. 2018; 46 (Web Server issue):W537–W544. https://doi.org/10.1093/nar/gky379 PMID: 29790989