COMBINE Archive Specification

Frank T. Bergmann
fbergmann@caltech.edu
Computing and Mathematical Sciences
California Institute of Technology
Pasadena, CA, US

Nicolas Rodriguez
rodriguezn@babraham.ac.uk
Babraham Institute
Babraham Campus Cambridge
Cambridge, UK

Nicolas Le Novère
n.lenovere@gmail.com
Babraham Institute
Babraham Campus Cambridge
Cambridge, UK

Version 1, Release Candidate 2

July 24, 2014

This is a release candidate of the specification for the COMBINE archive and not a normative document. Please send feedback to the COMBINE mailing list at combine-archive@googlegroups.com.

This release of the specification is available at
http://co.mbine.org/documents/archive

combine

Contents

1 Document conventions

Following the precedent set by other COMBINE specification documents, we use UML 1.0 (Unified Modeling Language; Eriksson and Penker (1998); Oestereich (1999)) class diagram notation to define the constructs provided by this package.

We also use the following typographical conventions to distinguish the names of objects and data types from other entities:

Class: Names of ordinary (concrete) classes begin with a capital letter and are printed in an upright, bold, sans-serif typeface. In electronic document formats, the class names are also hyperlinked to their definitions in this specification document.

`SomeThing`, `otherThing`: Attributes of classes, data type names, literal XML, and generally all tokens *other* than UML class names, are printed in an upright typewriter typeface. Primitive types defined begin with a capital letter; the COMBINE archive also makes use of primitive types defined by XML Schema 1.0 (Biron and Malhotra, 2000; Fallside, 2000; Thompson et al., 2000), but unfortunately, XML Schema does not follow any capitalization convention and primitive types drawn from the XML Schema language may or may not start with a capital letter.

2 Background

2.1 Motivation

Computational modeling is an increasingly interdisciplinary field. Different aspects of the modeling activity come together, that need information to be stored in a cohesive unit. When sharing a model, not exchanging all relevant files might cause problems. Several different approaches have been tried to solve this issue, such as folder based project structures, or special versions of version control systems. Unfortunately these approaches are not as easy to support for tool authors as a single file based solution would.

This specification describes *COMBINE archive*, a file that contains all the information needed to describe a modeling and simulation experiment. The *COMBINE archive* is encoded with the *Open Modeling Exchange format* (OMEX).A *COMBINE archive* is a single file containing the various documents (and possibly in the future, references to documents), necessary for the description of a model and all associated datasets and procedures. This includes for instance, but not limited to, simulation experiment descriptions in SED-ML, all models needed to run the simulations in SBML or CellML and their graphical representations in SBGN-ML.

The COMBINE archive aims to augment these efforts by standardizing a manifest that describes all files in the archive, guidance for encoding metadata and a convention for bundling the information together. Such an approach also allow resources using a distributed version control system (such as PMR2, `http://models.cellml.org/`) to export COMBINE archives, by generating a manifest file during export. http://models.cellml.org/ Details of earlier independent proposals are provided below.

2.2 History

The COMBINE archive was engendered from the SED-ML archive, that was first introduced in the SED-ML mailing list in 2009, and briefly described in SED-ML Level 1 Version 1 (Waltemath et al., 2011). The SED-ML archive addressed the need of providing a way to include all relevant models used in a SED-ML simulation experiment along with the experiment specification. The SED-ML archive is basically the convention to provide, inside a ZIP file (with extension .sedx), a SED-ML document with the same base name as the archive and all models listed in the SED-ML file.

This specification has been created out of the many discussions that took place on the combine-discuss mailing list. The primary threads are located here:

- http://listserver.ebi.ac.uk/pipermail/combine-discuss/2011-October/thread.html
- http://listserver.ebi.ac.uk/pipermail/combine-discuss/2011-November/000016.html
- http://listserver.ebi.ac.uk/pipermail/combine-discuss/2012-January/thread.html

The discussion continued at the last Hackathons, HARMONY 2012 where for the first time a larger audience discussed a preliminary proposal put together by Richard Adams, Frank T. Bergmann and Nicolas Le Novère (Adams et al., 2012). Further discussions were held at HARMONY 2013.
(`https://docs.google.com/document/d/1ji2mOiWzXXl9ON6sU6Svd3E_IufowUCz24hklqci0iM/edit`).

Discussions about the development of OMEX and the COMBINE archive are now taking place on the combine-archive Google group (`https://groups.google.com/forum/?#!forum/combine-archive`).

3 Proposed syntax and semantics

In this section, we define the syntax and semantics of the COMBINE archive. We expound on the various data types and constructs defined, then in Section 4 on page 10, we provide complete examples of archives.

3.1 The archive format

The COMBINE archive is encoded using the "Open Modeling EXchange format" (OMEX). The archive itself is a "zip" file (Wikipedia, 2014). Zip is a file format used for data compression and archiving. A zip file contains one or more files that have been compressed, to reduce file size, or stored as is. The technical specification of the ZIP format is available from the PKWARE website (PKWARE Inc., 2012).

3.2 COMBINE archive extensions

The extension for the COMBINE archive is `.omex`

Additional extensions are available to indicate what is the main standard format used within the archive. This help users to choose between different archives, and select appropriate software tools and in particular parsers. As such, the purpose of the extensions is different from the master attribute of the manifest file, described in section Section 3.7, which is read by the processing software once the archive is loaded and parsed. Currently, the following extensions are specified:

- `.sedx` - SED-ML archive (Waltemath et al., 2011)
- `.sbex` - SBML archive (Hucka et al., 2003)
- `.cmex` - CellML archive (Cuellar et al., 2003)
- `.neux` - NeuroML archive (Gleeson et al., 2010)
- `.phex` - PharmML archive (Moodie et al., 2013)
- `.sbox` - SBOL archive (Galdzicki et al., 2014)

Note that a COMBINE archive may contain files in several different standard formats. Therefore the specific extension is only an indication. For instance, a database of models could distribute archives of the same models, encoded encoded in SBML with the extension `.sbex` and encoded in CellML with the extension `.cmex`. However, archives containing models in both formats could also be distributed and the extension `.omex` used. If the archives contain SED-ML files, the extension `.sedx` could be used and the selection between the model formats be devolved to the SED-ML file.

3.3 Content of the archive

The archive contains:

1. a mandatory manifest file, called `manifest.xml`, always located at the root of the archive, that describes the location and the type of each data file contained in the archive plus an entry describing the archive itself.

   The location of those files is defined by a relative path. In the current version of the COMBINE archive, all the files described must be included in the archive itself. It is envisioned that in the future the manifest might list files located elsewhere, using valid and resolvable http URIs.

2. metadata files containing clerical information about the various files contained in the archive, and the archive itself. A best practice is to include only one file for each file format metadata called `metadata.*` (where `*` means the suitable file extension)

3. all the remaining files necessary to the model and simulation project.

3.4 Namespace URI and other declarations necessary

In order to uniquely identify the manifest file and the whole archive, OMEX defines two namespaces.

The namespace URI for OMEX:

> "`http://identifiers.org/combine.specifications/omex`"

The namespace URI for the COMBINE archive manifest:

> "`http://identifiers.org/combine.specifications/omex-manifest`"

Versioned URIs will be generated as needed, following the practices of COMBINE standards.

3.5 Primitive data types

The COMBINE archive uses the XML Schema 1.0 data types (Biron and Malhotra, 2000). More specifically we make use of `string` and `boolean`.

3.6 The `manifest.xml` file and the OmexManifest class

One file must be present at the root of any COMBINE archive, named `manifest.xml`. This file contains an instantiation of the **OmexManifest** class.

It contains a number of **Content** children, one of which represents the COMBINE archive itself.

A valid manifest needs to have at least one entry, declaring the archive itself, but may contain as many entries as needed. All the files present in the archive but one must be listed in the manifest. The only optional entry describes the manifest.xml file. Indeed the presence of manifext.xml is mandatory and its declaration is not necessary for parsing (since the software would have already parsed it to access the declaration!).

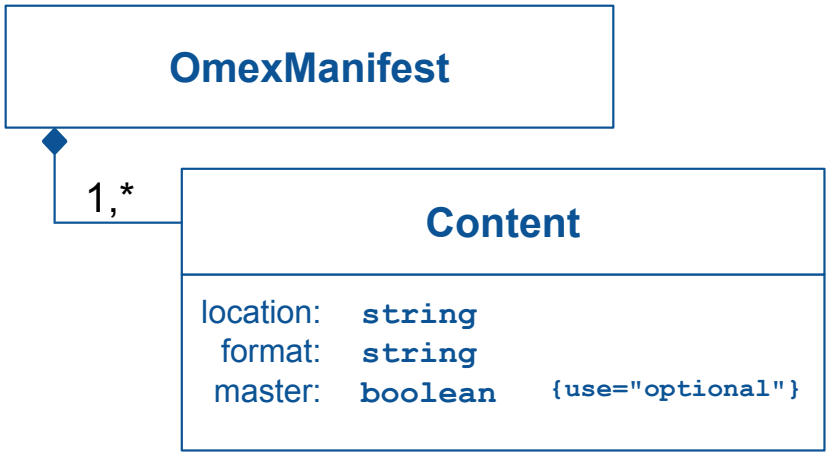

Figure 1: *A UML representation of the Manifest. Each manifest contains a number of Content elements.*

3.7 The Content class

An entry in the **OmexManifest** is represented by the **Content** class. It declares a file in the *COMBINE archive*. A content element possesses two required attributes, `location` and `format`, as well as an optional one, `master`.

The `location` *attribute*

The `location` attribute of type `string` is required. It represents the relative location of an entry within the archive. The archive is represented by a dot `'.'`.

The `format` *attribute*

The `format` attribute of type `string` is required. It indicates the file type of the **Content** element. The allowed values fall in two categories. Either the format denotes one of the COMBINE standards, in which case the `format` is the corresponding `Identifiers.org` URI. Or the `format` represents an Internet Media type (Freed and Borenstein, 1996) (previously known as "MIME type"), in which case the `format` indicates this Media type.

Using an `Identifiers.org` URI allows to unambiguously define a COMBINE format, including the precise version used. For example, the identifier: `http://identifiers.org/combine.specifications/sbml` would denote that the file declared by the **Content** element is encoded in the SBML format. That might be sufficient, as tools supporting one version of SBML often support others as well. However, if the software creating the COMBINE archive wanted to be more precise, it could specify that the document is encoded in SBML Level 2:

`http://identifiers.org/combine.specifications/sbml.level-2`

or even:

`http://identifiers.org/combine.specifications/sbml.level-2.version-4`.

The Internet Media type `purl.org` URI should be of the form `http://purl.org/NET/mediatypes/` followed by the Media type name. Here are a few examples:

```
for png (Portable Network Graphics) http://purl.org/NET/mediatypes/image/png
for pdf (Portable Document Format) http://purl.org/NET/mediatypes/application/pdf
for sbml http://purl.org/NET/mediatypes/application/sbml+xml
```

If both an `Identifiers.org` URI and a Media Type are available for one given format, the `Identifiers.org` URI must be used. Therefore the latter example should not be used. However, new Identifiers.org URI could be created for a format where an existing Media type was in used.

When creating a new COMBINE archive, the URI form should always be used for Media types. However, you may encounter some old COMBINE archives that use directly the Media type name for the format attribute, not in the URI form.

The `master` *attribute*

The `master` attribute of type `boolean` optional. When set to "true", it indicates that the file declared by the content element should be used first when processing the content of the archive. The file can be for instance the description of an upper model in a composed model, itself declaring the various submodels; a simulation description, declaring the different model descriptions and data sources used in the experiment. In most cases, one content element per archive will have its `master` attribute set to `true`.

For example the reading the snippet below, a software program should decide to present to the users first the file pointed by `location="simulation.xml"`.

```
<?xml version="1.0" encoding="utf-8"?>
<omexManifest xmlns="http://identifiers.org/combine.specifications/omex-manifest">
    <content location="." format="http://identifiers.org/combine.specifications/omex"/>
    <content location="model/model.xml"
        format="http://identifiers.org/combine.specifications/sbml"/>
    <content location="simulation.xml" master="true"
        format="http://identifiers.org/combine.specifications/sed-ml"/>
</omexManifest>
```

To simplify the identification, by a user or software, of the formats used by the documents inside an archive, file extensions (see also Section 3.2) can also be used. For instance, the archive containing the manifest file shown in the example above, could have the extension `.sbex`, to indicate that the software reading the archive need to support the `SBML` format to be able to interpret the content of the archive. If the software does support SED-ML as well as SBML, it should load the SED-ML file as the creator of the archive indicated it to be the '`master`' file. If the software does not support SED-ML, it can still open the SBML document, ignoring the master attribute.

Alternatively, if a software does not support either SBML or SED-ML it can choose to disregard the `master` attribute and the archive extension to present to the user the files present in the archive that it supports such as a PDF document or a diagram describing the model,

In some cases, the `master` attribute is more precise than the information given by the archive file extension.

```
<?xml version="1.0" encoding="utf-8"?>
<omexManifest xmlns="http://identifiers.org/combine.specifications/omex-manifest">
    <content location="."
        format="http://identifiers.org/combine.specifications/omex"/>
    <content location="main-model.xml" master="true"
        format="http://identifiers.org/combine.specifications/cellml.1.1"/>
    <content location="submodel1.xml"
        format="http://identifiers.org/combine.specifications/cellml.1.1"/>
    <content location="submodel2.xml"
        format="http://identifiers.org/combine.specifications/cellml.1.1"/>
</omexManifest>
```

In this case, the archive contains a modular CellML model. The file extension `cmex` is not sufficient to know which file should be processed first since there are several CellML files. The `master` attribute allows the software to know it should process `main-model.xml` first.

Ideally, COMBINE archives should contain only one model and should have only one `master` attribute set to `true`. However, in some cases, the creator of the archive might want to set several `master` attribute to `true`, for instance to indicate alternative renditions of a model in different formats, or to bundle together several models described in the same publication. In that case the reading software are free to open a specific file, all the files tagged as master, or display a dialog box to the user offering a choice. This might create some confusion as opening the same archive several times in the same software might result in a different model loaded. One option would be to not set any `master` attribute to `true`, being aware that proper interpretation of the archive migh be difficult, in particular for modular models.

3.8 Advised format for the archive metadata

Any type of file can be included in a COMBINE archive, and therefore any type of metadata format. However, in the interest of interoperability, and to ease the development of software support for metadata, a recommended format is provided as part of the specification of the archive.

The recommended format is based on several standards developed by other organisations:

- The Resource Description Format (Cyganiak et al., 2014) of the W3C, in particular its terms:
  - `RDF` (`http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/#coreSyntaxTerms`)
  - `Description` (`http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/#syntaxTerms`)
- vCard Ontology (vCard Ontology for describing People and Organizations, 2014), the RDF form of a file format standard for electronic business cards, in particular its terms:
  - `hasName` (`http://www.w3.org/2006/vcard/ns#hasName`),
  - `family-name`, (`http://www.w3.org/2006/vcard/ns#family-name`),
  - `given-name`, (`http://www.w3.org/2006/vcard/ns#given-name`),

- `hasEmail`, (http://www.w3.org/2006/vcard/ns#hasEmail),
- `organization-name` (http://www.w3.org/2006/vcard/ns#organization-name),

More information on how to use vCard in RDF can be found on the W3C website[1].

- Metadata terms[2] of the Dublin Core Metadata Initiative, in particular the terms:
  - `description` (http://purl.org/dc/terms/description),
  - `creator` (http://purl.org/dc/terms/creator),
  - `created` (http://purl.org/dc/terms/created),
  - `modified` (http://purl.org/dc/terms/modified),
  - `W3CDTF` (http://purl.org/dc/terms/W3CDTF).

  More information on the use of Dublin Core in RDF can be found on the Dublin Core website[3].

  More informationabout the definition of the date format used within `dcterms:created` and `dcterms:modified` elements is available on the W3C website [4].
- The BioModels qualifiers[5]

Users of the COMBINE standards may already be familiar with this approach, as it is also taken by SBML, SED-ML, SBGN-ML [6].

A *COMBINE archive* can include multiple metadata elements adding information about different content files. To identify the file a metadata element refers to, the `rdf:about` attribute of the relevant metadata structure should use the same value as used in the `location` attribute of the respective **Content** element, e.g.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dcterms="http://purl.org/dc/terms/"
   xmlns:vCard="http://www.w3.org/2006/vcard/ns#"
   xmlns:bqmodel="http://biomodels.net/models-qualifiers">
   <rdf:Description rdf:about="simulation.xml">
   ...
   </rdf:Description>
</rdf:RDF>
```

A complete example of the metadata related to a simulation description contained in a COMBINE archive is described in Section 4.

[1] http://www.w3.org/TR/vcard-rdf/
[2] http://dublincore.org/documents/dcmi-terms/
[3] http://dublincore.org/documents/dc-rdf/
[4] http://www.w3.org/TR/NOTE-datetime
[5] http://co.mbine.org/standards/qualifiers
[6] Note: These terms slightly differs from the terms used in SBML level 3 version 1 specification.

4 Illustrative examples of the syntax

This section contains a worked example showing the encoding of a model and associated data in a COMBINE archive. First the **OmexManifest** , that includes five entries. One of these entries represent the archive itself, it has a fixed location `location="."`. Additionally the archive includes an SBML model and a SED-ML description. It also includes a PDF file which format is specified through its Media type. Finally associated meta information for clerical data is included in `location="metadata.rdf"`.

```
<?xml version="1.0" encoding="utf-8"?>
<omexManifest xmlns="http://identifiers.org/combine.specifications/omex-manifest">
    <content location="."
             format="http://identifiers.org/combine.specifications/omex"/>
    <content location="model/model.xml"
             format="http://identifiers.org/combine.specifications/sbml"/>
    <content location="simulation.xml" master="true"
             format="http://identifiers.org/combine.specifications/sed-ml"/>
    <content location="doc/article.pdf"
             format="http://purl.org/NET/mediatypes/application/pdf"/>
    <content location="metadata.rdf"
             format="http://identifiers.org/combine.specifications/omex-metadata"/>
</omexManifest>
```

Here a complete example, on how the clerical data could be encoded in the metadata file:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dcterms="http://purl.org/dc/terms/"
         xmlns:vCard="http://www.w3.org/2006/vcard/ns#"
         xmlns:bqmodel="http://biomodels.net/model-qualifiers/">
    <rdf:Description rdf:about=".">
        <dcterms:description>
            Expanded version of the human metabolic reconstruction Recon 2.1
        </dcterms:description>
        <dcterms:creator rdf:parseType="Resource">
            <vCard:hasName rdf:parseType="Resource">
                <vCard:family-name>Le Novere</vCard:family-name>
                <vCard:given-name>Nicolas</vCard:given-name>
            </vCard:hasName>
            <vCard:hasEmail rdf:resource="lenov@babraham.ac.uk" />
            <vCard:organization-name>
                Babraham Institute
            </vCard:organization-name>
            <vCard:hasURL rdf:resource="http://orcid.org/0000-0002-6309-7327"/>
        </dcterms:creator>
        <dcterms:created rdf:parseType="Resource">
            <dcterms:W3CDTF>2014-06-26T10:29:00Z</dcterms:W3CDTF>
        </dcterms:created>
        <bqmodel:is
             rdf:resource="http://identifiers.org/biomodels.db/MODEL1311110001" />
        <bqmodel:isDescribedBy
             rdf:resource="http://identifiers.org/arxiv/1311.5696" />
    </rdf:Description>
</rdf:RDF>
```

5 Future development

In this section we highlight some open issues not addressed in this version of the COMBINE archive specification.

5.1 Linking to external documents

It was often discussed to also allow the `location` elements of a **Content** element point to an external document. However, in this first version we restrict them to local files, so as to make it easier to adopt in software tools. That way tools could focus on the primary use case of bundling up several local resources, rather than worry about retrieving information from online resources that may not always be available or may be more complex.

5.2 Cross References between entries

At HARMONY 2013, we spend some time discussing whether cross references between the individual entries in the archive ought to be in this first version of the specification. It was decided to leave the cross referencing to the individual standards for now, rather to impose them ad-hoc.

5.3 Alternative versions of the archive metadata

It was suggested to allow different versions of the archive metadata format. The manifest already provides a way for referencing alternate versions, all that would need to be changed, would be the format identifier to point to a different namespace rather than:

`http://identifiers.org/combine.specifications/omex-metadata`

However, as of the time of this writing no such format was proposed.

5.4 Convergence with other archive formats

Recently, the Workflow4Ever project (Wf4ever) funded by the European Union developed the Research Object Bundle (Soiland-Reyes, 2013), focusing on the preservation of scientific experiments in data-intensive science. The structure of the Research Object Bundle is close to the COMBINE archive. Based on Adobe Universal Container Format (UCF, 2013), it is also a Zip file containing a manifest and metadata. Discussions have started with the latter project. One way of starting convergence would be to share metadata vocabularies and format.

Further from scientific research, several industry standards are being largely used for building document archives based on Zip files. The main ones belong to two large families. The aforementioned Universal Container Format bears strong similarities with the Open Document Format (ODF, 2011) developed by the Organization for the Advancement of Structured Information Standards (OASIS), and the EPUB Open Container Format (OCF, 2011) developed by the International Digital Publication Forum (IDPF). As for the format described here, those archives contain a manifest file that lists and identifies the contents of the archive and a metadata file, either in xml or rdf. In parallel, Microsoft developed the Open Packaging Convention (OPC, 2008) used in its office suite and some other software such as MathWorks' Simulink. Those packages also list their contents in an xml file, and can carry metadata.

Those industry standards are relatively complex and rigid. In this initial iteration of the specification, the community opted for simplicity and flexibility, in order to get traction. The main differences between the COMBINE archive and the formats discussed above lie in the structure of the archive and the format of the file describing the content. Since the COMBINE archive is in general less specified, future convergences will be easy to implement once agreed upon.

Acknowledgments

We would like to thank all the people who contributed in various ways to the development of both the original proposal and this specification, especially Richard Adams and Stuart Moodie. We also would like to thank the attendees of dedicated sessions at COMBINE meetings and the members of the combine-discuss and combine-archive mailing lists and all others who contributed to discussions on various occasions.

References

Adams, R., Bergmann, F. T., and Le Novère, N. (2012). Combine archive format proposal.

Biron, P. V. and Malhotra, A. (2000). XML Schema part 2: Datatypes (W3C candidate recommendation 24 October 2000). Available via the World Wide Web at http://www.w3.org/TR/xmlschema-2/.

Cuellar, A., Lloyd, C., Nielsen, P., Bullivant, D., Nickerson, D., and Hunter, P. (2003). An overview of cellml 1.1, a biological model description language. *SIMULATION*, 79(12):740–747.

Cyganiak, R., Wood, D., Lanthaler, M., Klyne, G., Carroll, J. J., and McBride, B. (2014). RDF 1.1 concepts and abstract syntax. Available via the World Wide Web at http://www.w3.org/TR/rdf11-concepts/.

Eriksson, H.-E. and Penker, M. (1998). *UML Toolkit*. John Wiley & Sons, New York.

Fallside, D. C. (2000). XML Schema part 0: Primer (W3C candidate recommendation 24 October 2000). Available via the World Wide Web at http://www.w3.org/TR/xmlschema-0/.

Freed, N. and Borenstein, N. (1996). Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types. http://www.rfc-editor.org/rfc/rfc2046.txt http://www.ietf.org/rfc/rfc6350.txt http://www.iana.org/assignments/media-types/media-types.xhtml. Updated by RFC 6868.

Galdzicki, M., Clancy, K. P., Oberortner, E., Pocock, M., Quinn, J. Y., Rodriguez, C. A., Roehner, N., Wilson, M. L., Adam, L., Anderson, J. C., Bartley, B. A., Beal, J., Chandran, D., Chen, J., Densmore, D., Endy, D., GrÃijnberg, R., Hallinan, J., Hillson, N. J., Johnson, J. D., Kuchinsky, A., Lux, M., Misirli, G., Peccoud, J., Plahar, H. A., Sirin, E., Stan, G.-B., Villalobos, A., Wipat, A., Gennari, J. H., Myers, C. J., and Sauro, H. M. (2014). The synthetic biology open language (sbol) provides a community standard for communicating designs in synthetic biology. *Nat Biotechnol*, 32(6):545–550.

Gleeson, P., Crook, S., Cannon, R. C., Hines, M. L., Billings, G. O., Farinella, M., Morse, T. M., Davison, A. P., Ray, S., Bhalla, U. S., Barnes, S. R., Dimitrova, Y. D., and Silver, R. A. (2010). Neuroml: a language for describing data driven models of neurons and networks with a high degree of biological detail. *PLoS Comput Biol*, 6(6):e1000815.

Hucka, M., Finney, A., Sauro, H., Bolouri, H., Doyle, J., Kitano, H., Arkin, A., Bornstein, B., Bray, D., Cornish-Bowden, A., Cuellar, A., Dronov, S., Gilles, E., Ginkel, M., Gor, V., Goryanin, I., Hedley, W., Hodgman, T., Hofmeyr, J., Hunter, P., Juty, N., Kasberger, J., Kremling, A., Kummer, U., Novère, N. L., Loew, L., Lucio, D., Mendes, P., Minch, E., Mjolsness, E., Nakayama, Y., Nelson, M., Nielsen, P., Sakurada, T., Schaff, J., Shapiro, B., shimizu, T., Spence, H., Stelling, J., Takahashi, K., Tomita, M., Wagner, J., and Wang, J. (2003). The Systems Biology Markup Language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–31.

Moodie, S. L., Swat, M. J., Kristensen, N. R., and Novère, N. L. (2013). Pharmml: The pharmacometrics markup language. Available via the World Wide Web at http://ddmore.eu/pharmml.

OCF (2011). Epub open container format (ocf) 3.0. Available via the World Wide Web at http://www.idpf.org/epub/30/spec/epub30-ocf.html.

ODF (2011). Opendocument technical specification. Available via the World Wide Web at http://en.wikipedia.org/wiki/OpenDocument_technical_specification.

Oestereich, B. (1999). *Developing Software with UML: Object-Oriented Analysis and Design in Practice*. Addison-Wesley.

OPC (2008). Open packaging convention. Available via the World Wide Web at http://en.wikipedia.org/wiki/Open_Packaging_Convention.

PKWARE Inc. (2012). APPNOTE.TXT - .ZIP File Format Specification. http://www.pkware.com/documents/casestudies/APPNOTE.TXT. [Online; accessed 13-January-2014].

Soiland-Reyes, S. (2013). Wf4ever research object bundle. Available via the World Wide Web at http://wf4ever.github.io/ro/bundle/.

Thompson, H. S., Beech, D., Maloney, M., and Mendelsohn, N. (2000). XML Schema part 1: Structures (W3C candidate recommendation 24 October 2000). Available online via the World Wide Web at the address http://www.w3.org/TR/xmlschema-1/.

UCF (2013). Universal container format. Available via the World Wide Web at https://wikidocs.adobe.com/wiki/display/PDFNAV/Universal+Container+Format.

vCard Ontology for describing People and Organizations (2014). Ianella, renato and mckinney, james. Available via the World Wide Web at http://www.w3.org/TR/vcard-rdf/.

Waltemath, D., Adams, R., Bergmann, F. T., Hucka, M., Kolpakov, F., Miller, A. K., Moraru, I. I., Nickerson, D., Sahle, S., Snoep, J. L., and Le Novère, N. (2011). Reproducible computational biology experiments with sed-ml–the simulation experiment description markup language. *BMC Syst Biol*, 5:198.

Wikipedia (2014). Zip (file format) — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Zip%20(file%20format)&oldid=588974210. [Online; accessed 13-January-2014].