
biotoolsSchema Documentation

Release latest

Apr 15, 2021

1	What is biotoolsSchema?	3
1.1	How to contribute to biotoolsSchema	3
1.2	Documentation	3
1.3	Download	4
1.4	Status	4
1.5	Motivation	4
1.6	Citing biotoolsSchema	4
2	Design considerations	5
3	biotoolsSchema elements	7
3.1	Element groups	7
3.2	Summary group	9
3.3	Function group	12
3.4	Labels group	15
3.5	Link group	18
3.6	Download group	18
3.7	Documentation group	20
3.8	Publication group	20
3.9	Relation group	22
3.10	Credit group	23
4	Controlled vocabularies	25
4.1	Tool type	26
4.2	Operating system	26
4.3	Programming language	27
4.4	License	27
4.5	Maturity	27
4.6	Cost	27
4.7	Accessibility	28
4.8	Link type	28
4.9	Download type	28
4.10	Documentation type	29
4.11	Publication type	30
4.12	Relation type	30
4.13	Entity type	31
4.14	Entity role	31

4.15	ID type	31
4.16	ELIXIR Platform	32
4.17	ELIXIR Node	32
4.18	ELIXIR Community	33
5	Contributors	35
5.1	Governance	35
5.2	Main authors & contributors	35
5.3	Contributors	36
6	License	37

This is the documentation for `biotoolsSchema`.

Contents:

What is biotoolsSchema?

biotoolsSchema is a formalised XML schema (XSD) which defines a description model for bioinformatics software. It can be used to describe bioinformatics tools - application software with well-defined data processing functions (inputs, outputs and operations). This includes simple tools with one or a few closely related functions, and complex, multimodal tools with many functions, tools available available for immediate use as online services, or in a form which which you can download, install, configure and run yourself.

biotoolsSchema defines over 50 important scientific, technical and administrative attributes that support cataloguing, discovery, use and interoperability of software. It concentrates upon the salient common features, with a minimal core of 3 attributes only (name, short description and homepage), to provide maximum flexibility for applications. To enable concise information, standard identifiers are used where possible, including [EDAM ontology](#) concept IDs for specialised scientific aspects. biotoolsSchema defines 18 controlled vocabularies for technical tool aspects. Verbose information is referred to by URL.

biotoolsSchema together with the [EDAM ontology](#) provide the foundation for an [information standard](#) for the description of tools. This standard is being adopted by bio.tools and defines the attributes that must be defined within a 5-tier scale of entry completeness and quality.

biotoolsSchema is used by the ELIXIR Tools & Data Services Registry ([bio.tools](#)).

1.1 How to contribute to biotoolsSchema

Contributions to biotoolsSchema are very welcome. Github being our primary communication channel, please do not hesitate to open issues, or comment on them. Pull requests are also welcome! The list of repositories hosting biotoolsSchema and its documentation are listed below. All materials are available under the [CC-BY-SA 4.0 license](#).

1.2 Documentation

These docs describe the latest stable version (currently 3.0.0) and are maintained in [GitHub](#). Additional documentation is available:

- [Technical docs](#) maintained in [GitHub](#) where you'll find example data files

- Usage guidelines including the [bio.tools Curators Guide](#) and [API Usage Guide](#)

Note: Version 3.0.0 is currently supported in [bio.tools](#).

1.3 Download

Latest stable version (3.0.0)

- <https://github.com/bio-tools/biotoolsSchema/blob/master/stable/biotools.xsd>

Current development version

- https://github.com/bio-tools/biotoolsSchema/blob/master/biotools_dev.xsd

Versioned releases

- <https://github.com/bio-tools/biotoolsSchema/tree/master/versions>

1.4 Status

biotoolsSchema is a mature schema having undergone a lot of community-driven revision (*e.g.* see [bio.tools events](#)). Development is use-case driven, primarily by [bio.tools](#). Future versions will not depart fundamentally from the current elements or structure. The development of biotoolsSchema can be followed at [GitHub](#). From version 2.0.0, version numbers follow the *MAJOR.MINOR.PATCH* pattern:

- *MAJOR* version for incompatible API changes
- *MINOR* version for addition of functionality in a backwards-compatible manner
- *PATCH* version for backwards-compatible bug fixes

Please contribute via [GitHub](#). See also the [bio.tools Contributors Guide](#).

1.5 Motivation

Bioinformaticians routinely use a large and diverse set of tools and data, and demand powerful and convenient means to organise, find, understand, compare, select, use and connect the available resources. These tasks rely on consistent, machine-understandable resource descriptions. The need - filled by biotoolsSchema - is for an information model that puts the description of a broad range of resources on a consistent syntactic basis.

1.6 Citing biotoolsSchema

If you use biotoolsSchema, please cite:

Ison J. et al. (2021) biotoolsSchema: a formalized schema for bioinformatics software description. *GigaScience*, **10** (1)

The article is [freely available](#).

doi: [10.1093/gigascience/giaa157](https://doi.org/10.1093/gigascience/giaa157)

Design considerations

The development of biotoolsSchema was guided by the following ten founding principles and design considerations:

- **Practical** - focus on salient attributes of practical value in everyday use; especially to support the discovery, use and practical interoperability of software; superfluous details are excluded.
- **General** - generally applicable, i.e. to all manner of bioinformatics resources (see Scope).
- **Consistent** - use ontologies and standardised enumerations of terms (see Controlled Vocabularies) where possible, to support precise searches over biotoolsSchema-formatted data and return of consistent and therefore comparable information.
- **Concise** - mandate URLs or standard identifiers where possible, helping to ensure the sustainable upkeep of biotoolsSchema-formatted data and support future integrations, applications and cross-linking with other resources.
- **Simple** - biotoolsSchema is as flat (unstructured) as is practicable, ensuring ease of use, whilst preserving essential structure, e.g. a meaningful model of tool function.
- **Compatible** - it is inevitable that tool providers, integrators, and cataloguers will continue to use a variety of models, methods and formats for tool descriptions; biotoolsSchema is broadly compatible (see Comparison to related efforts) to support future integration scenarios.
- **Extensible** - to cater for emerging requirements, and adaptable by others for their own purposes.
- **Stable** - the maintenance of software dependencies on mutating schema is expensive. Backwards incompatible changes are only made if absolutely required (see Development process and status).
- **Free and open source** - to encourage reuse and new applications.
- **Community-driven development** - to ensure end-user needs are satisfied.

biotoolsSchema elements

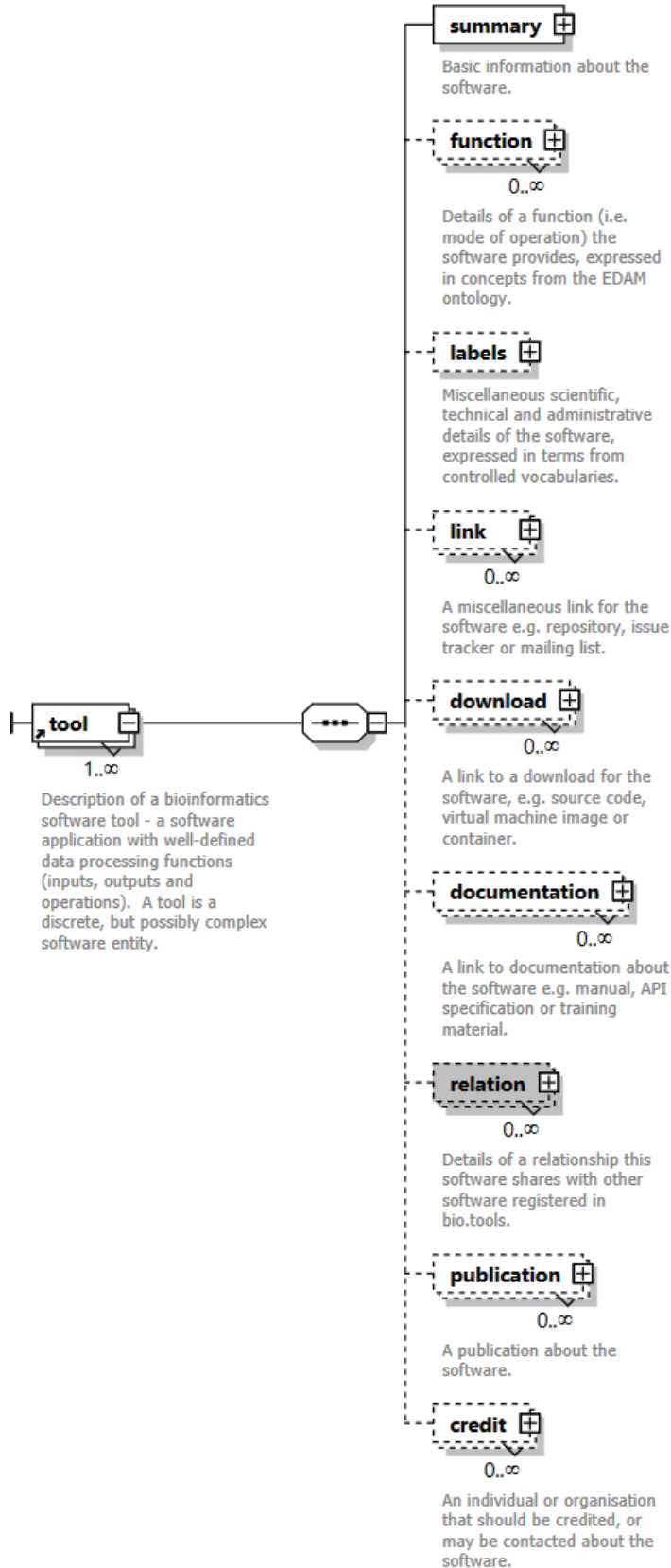
A summary of elements in biotoolsSchema is below. For more detailed information, please see the [technical documentation](#).

Important: When producing XML files compliant to the schema, it is essential to stick to the element order (including nested elements) as described below. See the sample [XML files](#).

3.1 Element groups

biotoolsSchema includes scientific, technical and administrative software attributes, organised for convenience into 8 logical groupings (see below).

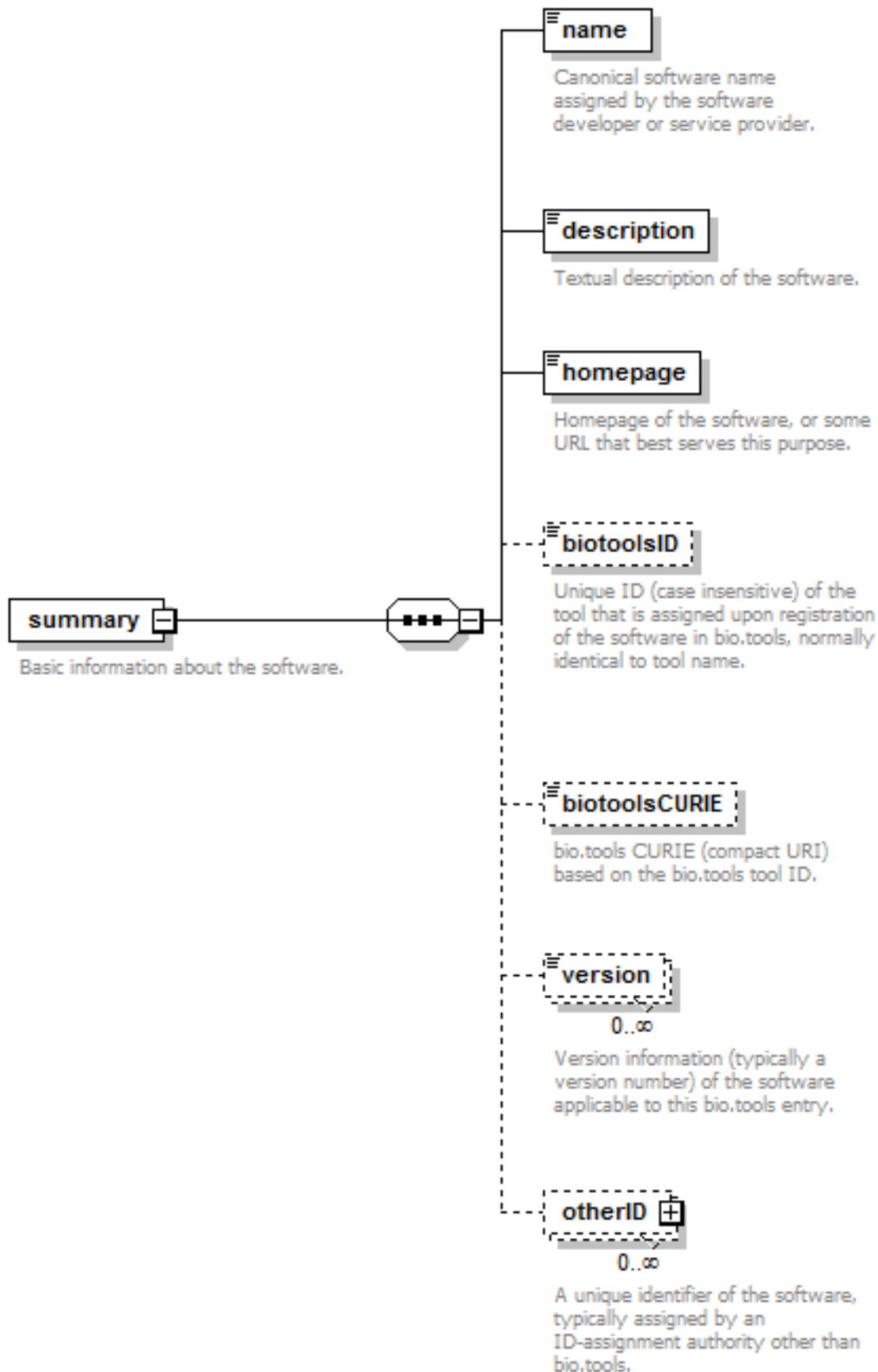
Note: As of biotoolsSchema 3.3.0, all of the element groups *other than* “Summary” and “Labels” are reflected as elements/objects in biotoolsSchema/biotoolsSchemaJ.

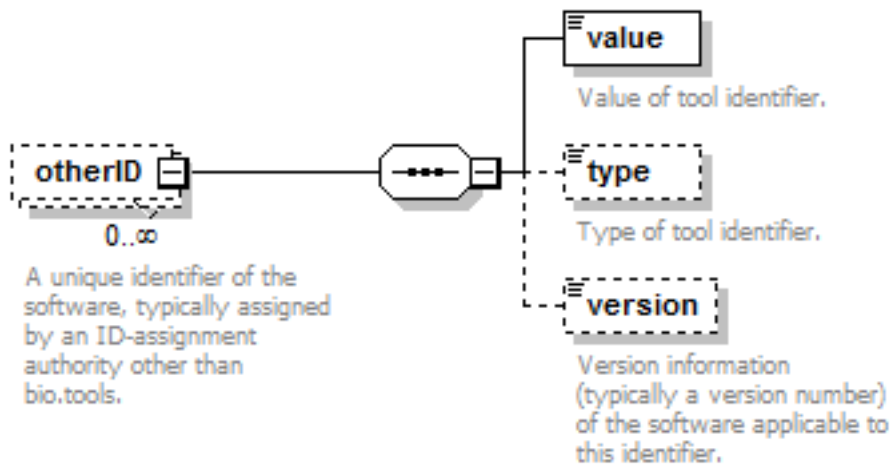


Group	XSD element	Description
Summary	summary	Basic information about the software.
Function	function	Details of the function(s) (i.e. modes of operation) the software provides, expressed in terms from the EDAM ontology.
Labels	label	Miscellaneous scientific, technical and administrative details of the software, expressed in terms from controlled vocabularies.
Links	link	Miscellaneous links for the software e.g. repository, issue tracker or mailing list.
Downloads	download	Links to downloads for the software, e.g. source code, virtual machine image or container.
Documentation	documentation	Links to documentation about the software e.g. user manual, API documentation or training material.
Publications	publication	Publications about the software.
Relations	relation	Details of a relationship this software shares with other software registered in bio.tools.
Credits	credit	Individuals or organisations that should be credited, or may be contacted about the software.

3.2 Summary group

Basic information about the software.





Element	Description	Type	Cardinality
name	Canonical software name assigned by the software developer or service provider.	xs:token (restriction)	1 only
description	Textual description of the software.	xs:token (restriction)	1 only
homepage	Homepage of the software, or some URL that best serves this purpose.	URL	1 only
biotoolsID	Unique ID (case insensitive) of the tool that is assigned upon registration of the software in bio.tools, normally identical to tool name.	URL (restriction)	0 or 1
biotoolsCURIE	bio.tools CURIE (compact URI) based on the bio.tools tool ID.	xs:token (restriction)	0 or 1
version	Version information (typically a version number) of the software applicable to this bio.tools entry.	xs:token (restriction)	0 or more
otherID	A unique identifier of the software, typically assigned by an ID-assignment authority other than bio.tools.	(see below)	0 or more
otherID->value	Value of tool identifier.	xs:token (restriction)	1 only
otherID->type	Type of tool identifier.	enum (see docs)	0 or 1
otherID->version	Version information (typically a version number) of the software applicable to this identifier.	xs:token (restriction)	0 or 1

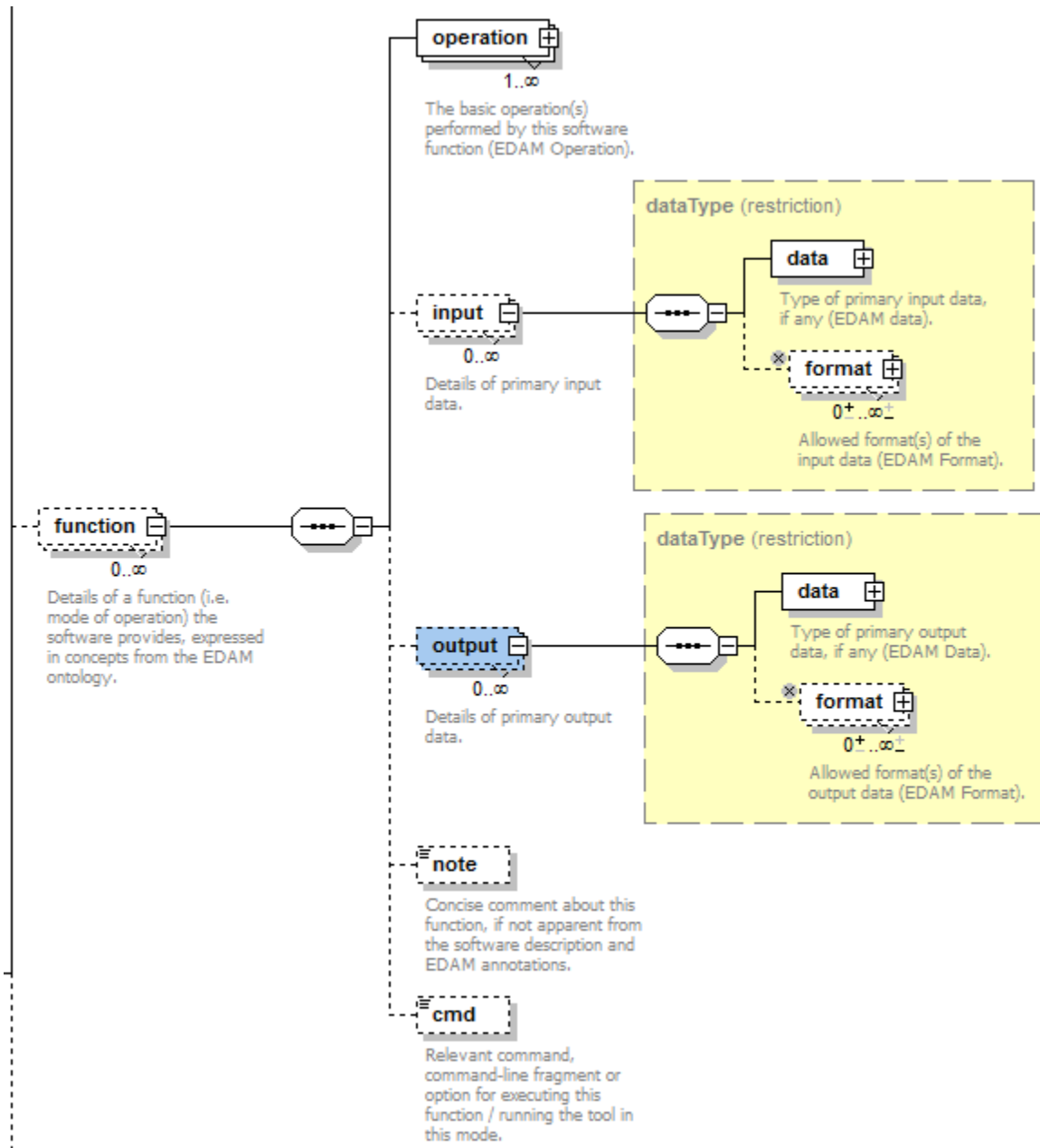
Note: See the [Curators Guide](#). As of biotoolsSchema 3.0.0, the Summary group does not have a corresponding element/object in biotoolsSchema/biotoolsSchemaJ (the schema was flattened).

3.3 Function group

Details of a function (i.e. mode of operation) the software provides, expressed in terms from the EDAM ontology.

Each software entity has one more functions, each corresponding to a mode of operation that the software provides. Each function performs one or more basic operations, and has zero or more primary input and/or output data, each of a

specified type and supported format(s). For each operation, data or format element, an EDAM ontology concept URL and/or term are specified.



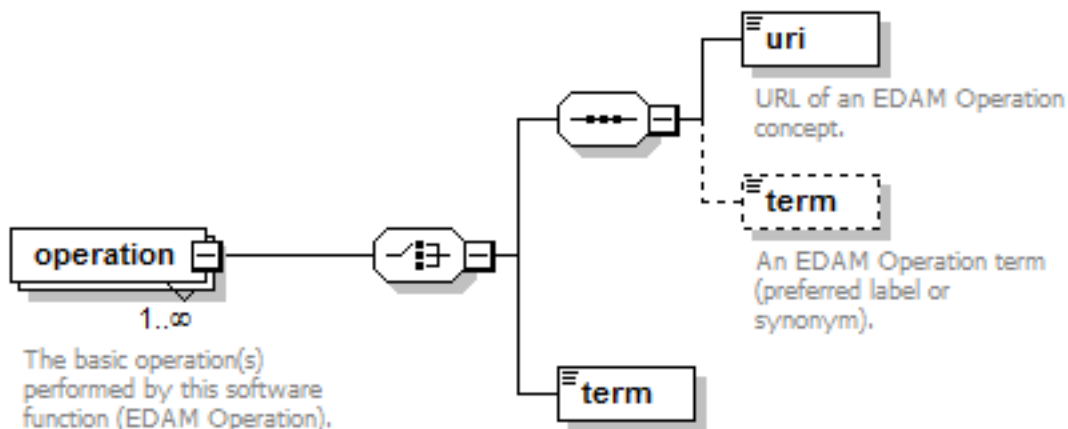


Fig. 1: Data and Format are modelled in the same way as Operation (shown above).

Element	Description	Type	Cardinality
operation	The basic operation(s) performed by this software function (EDAM Operation).	Ontology concept (restriction)	1 or more
input	<i>Details of primary input data.</i>		
input->data	Type of primary input data (EDAM data).	Ontology concept (restriction)	1 only
input->format	Allowed format(s) of the input data (EDAM Format).	Ontology concept (restriction)	0 or more
output	<i>Details of primary output data.</i>		
output->data	Type of primary output data (EDAM Data).	Ontology concept (restriction)	1 only
output->format	Allowed format(s) of the output data (EDAM Format).	Ontology concept (restriction)	0 or more
operation data format->url	URL of an EDAM Operation Data Format concept.	URL (restriction)	0 or 1
operation data format->term	An EDAM Operation Data Format term (preferred label or synonym).	xs:token	0 or 1
note	Concise comment about this function, if not apparent from the software description and EDAM annotations.	xs:token (restriction)	0 or 1
cmd	Relevant command, command-line fragment or option for executing this function	xs:token (restriction)	0 or 1
14	/ running the tool in this mode.		Chapter 3. biotoolsSchema elements

Important: The URL must be in the appropriate EDAM Operation | Data | Format namespace, *e.g.*

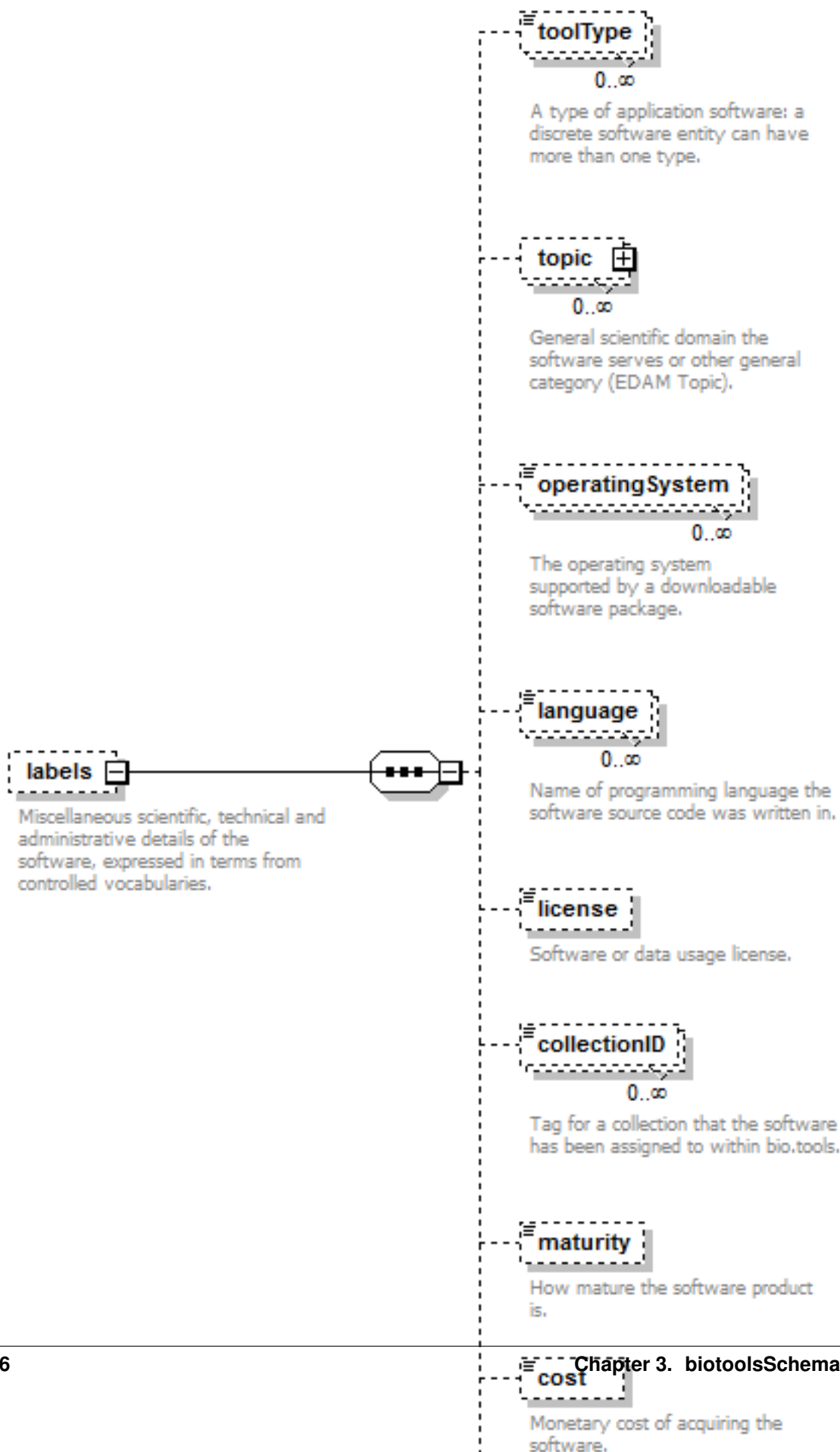
- http://edamontology.org/operation_0492
- http://edamontology.org/data_0863
- http://edamontology.org/format_1929

The term, *e.g.* “Multiple sequence alignment” must be either the preferred label of the concept or a synonym of this term, as defined in EDAM.

Note: See the [Curators Guide](#).

3.4 Labels group

Miscellaneous scientific, technical and administrative details of the software, expressed in terms from controlled vocabularies.

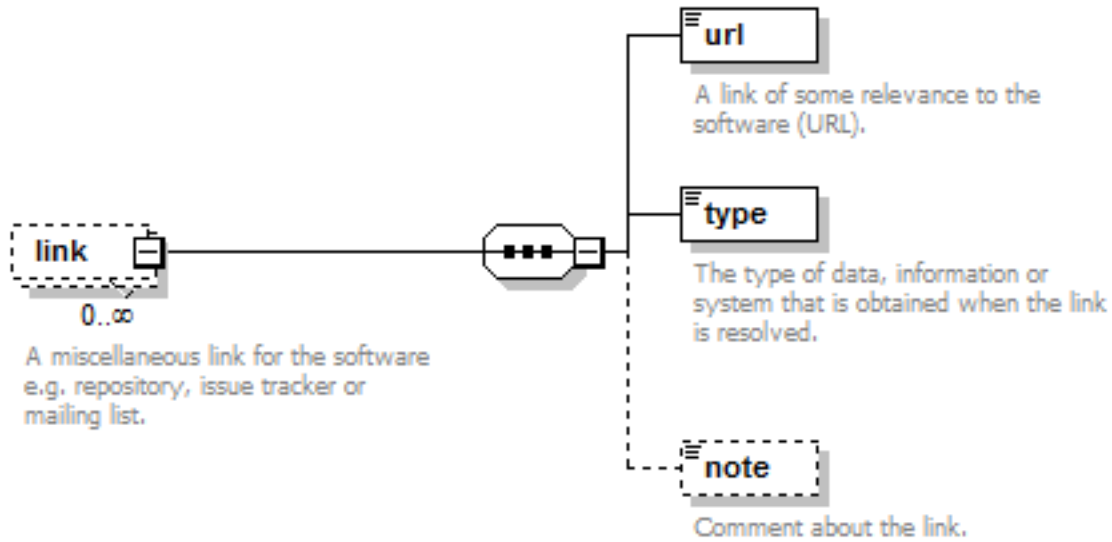


Element	Description	Type	Cardinality
toolType	A type of application software: a discrete software entity can have more than one type.	enum (see docs)	0 or more
topic	General scientific domain the software serves or other general category: one of EDAM Topic URL or term is specified.	Ontology concept (restriction)	0 or more
topic->url	URL of an EDAM Topic concept.	URL (restriction)	0 or 1
topic->term	An EDAM Topic term (preferred label or synonym).	xs:token	0 or 1
operatingSystem	The operating system supported by a downloadable software package.	enum (see docs)	0 or more
language	Name of programming language the software source code was written in.	enum (see docs)	0 or more
license	Software or data usage license.	enum (see docs)	0 or 1
collectionID	Tag for a collection that the software has been assigned to within bio.tools.	xs:token (restriction)	0 or more
maturity	How mature the software product is.	enum (see docs)	0 or 1
cost	Monetary cost of acquiring the software.	enum (see docs)	0 or 1
accessibility	Whether there are non-monetary restrictions on accessing an online service.	enum (see docs)	0 or more
elixirPlatform	Name of the ELIXIR Platform that is credited.	enum (see docs)	0 or more
elixirNode	Name of the ELIXIR Node that is credited.	enum (see docs)	0 or more
elixirCommunity	Name of relevant ELIXIR (or associated) community.	enum (see docs)	0 or more

Note: See the [Curators Guide](#). As of biotoolsSchema 3.0.0, the Labels group does not have a corresponding element/object in biotoolsSchema/biotoolsSchemaJ (the schema was flattened).

3.5 Link group

Miscellaneous links for the software e.g. repository, issue tracker or mailing list.

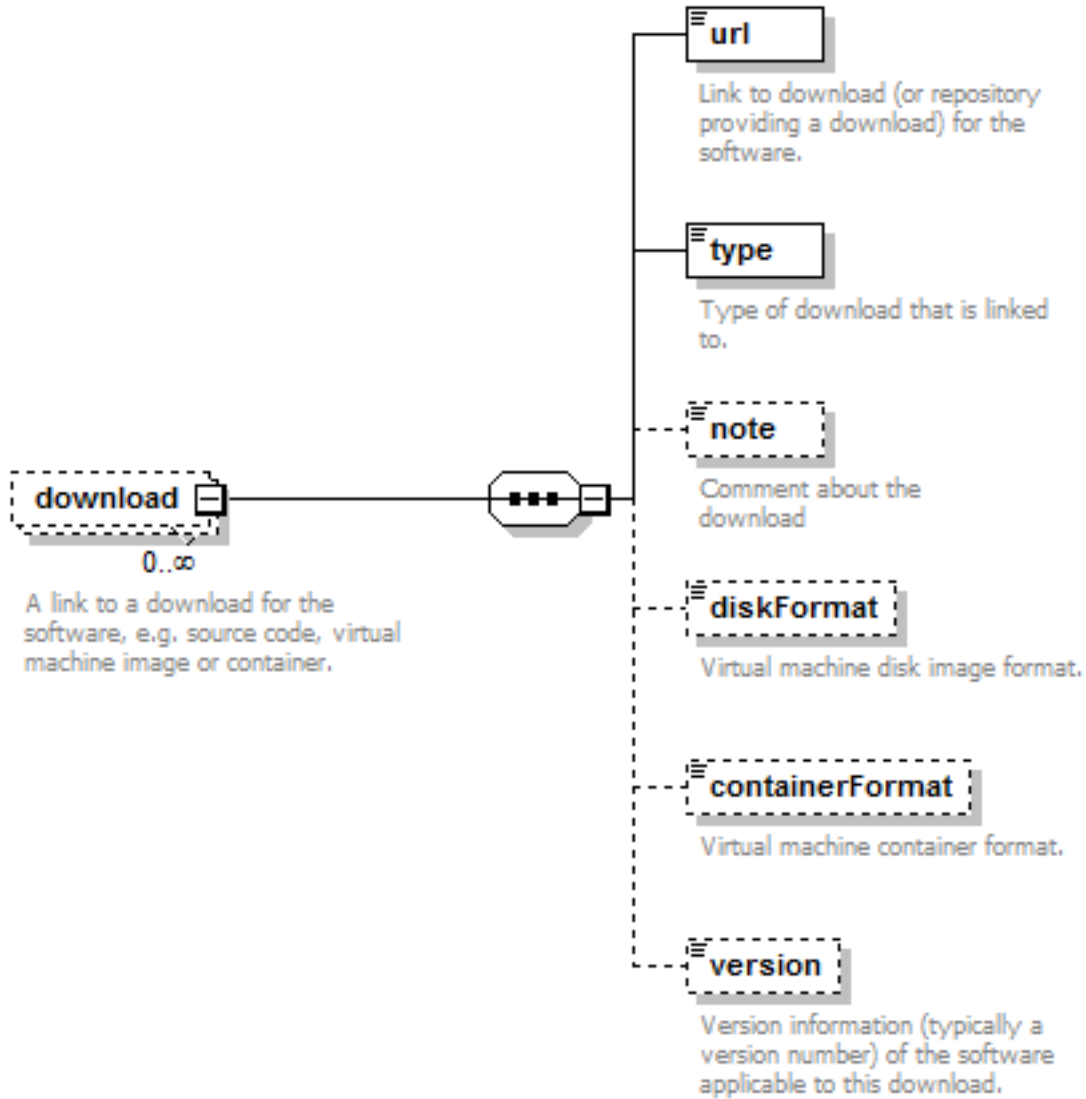


Element	Description	Type	Cardinality
url	A link of some relevance to the software (URL).	URL	1 only
type	The type of data, information or system that is obtained when the link is resolved.	enum (see docs)	1 or more
note	Comment about the link.	xs:token (restriction)	0 or 1

Note: See the [Curators Guide](#).

3.6 Download group

Links to downloads for the software, e.g. source code, virtual machine image or container.

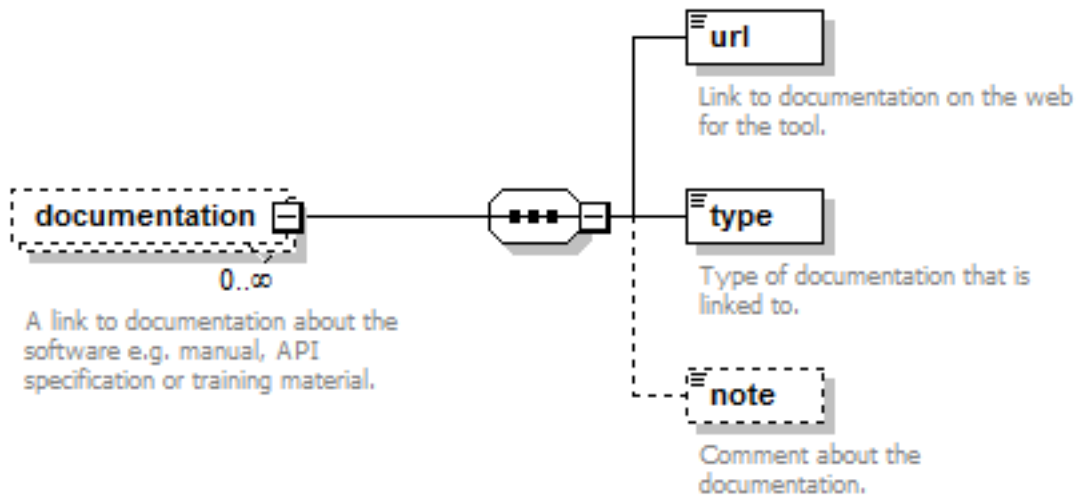


Element	Description	Type	Cardinality
url	Link to download (or repo providing a download) for the software.	URL	1 only
type	Type of download that is linked to.	enum (see docs)	1 only
note	Comment about the download.	xs:token (restriction)	0 or 1
version	Version information (typically a version number) of the software applicable to this download.	xs:token (restriction)	0 or 1

Note: See the [Curators Guide](#).

3.7 Documentation group

Links to documentation about the software e.g. manual, API specification or training material.

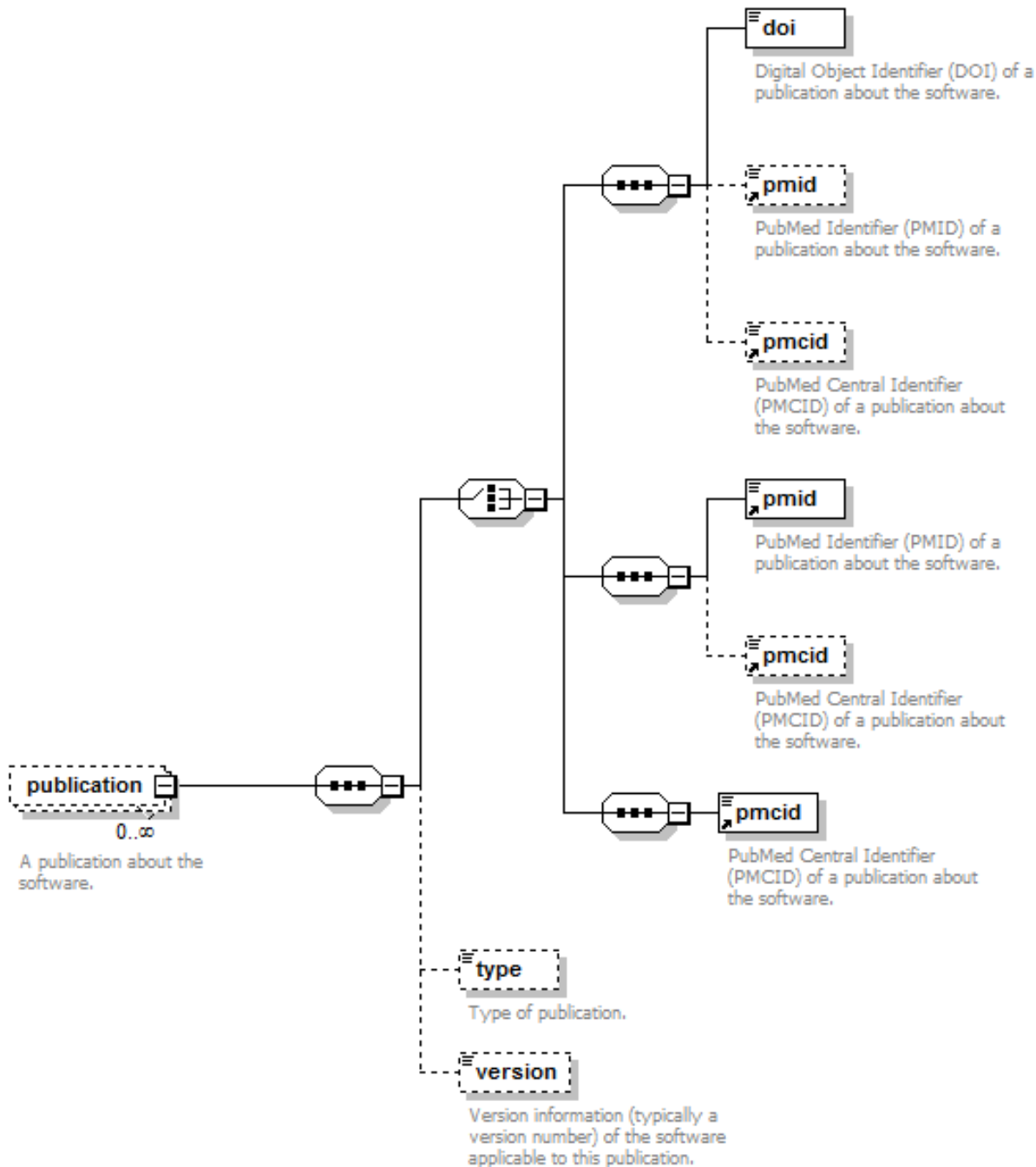


Element	Description	Type	Cardinality
url	Link to documentation on the web for the tool.	URL	1 only
type	Type of documentation that is linked to.	enum (see docs)	1 or more
note	Comment about the documentation.	xs:token (restriction)	0 or 1

Note: See the [Curators Guide](#).

3.8 Publication group

Publications about the software

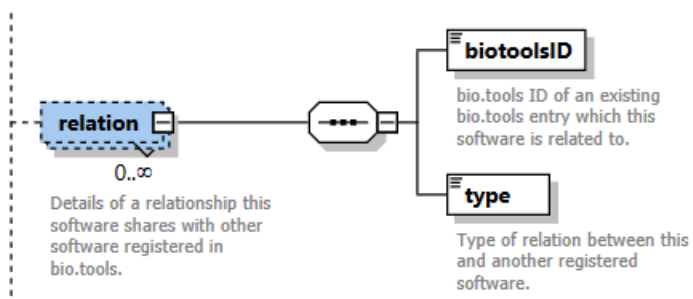


Element	Description	Type	Cardinality
pmcid	PubMed Central Identifier of a publication about the software.	xs:token (restriction)	0 or 1
pmid	PubMed Identifier.	xs:token (restriction)	0 or 1
doi	Digital Object Identifier.	xs:token (restriction)	0 or 1
type	Type of publication.	enum (see docs)	0 or more
note	Comment about the publication.	xs:token (restriction)	0 or 1
version	Version information (typically a version number) of the software applicable to this publication.	xs:token (restriction)	0 or 1

Note: See the [Curators Guide](#).

3.9 Relation group

Details of a relationship this software shares with other software registered in bio.tools.

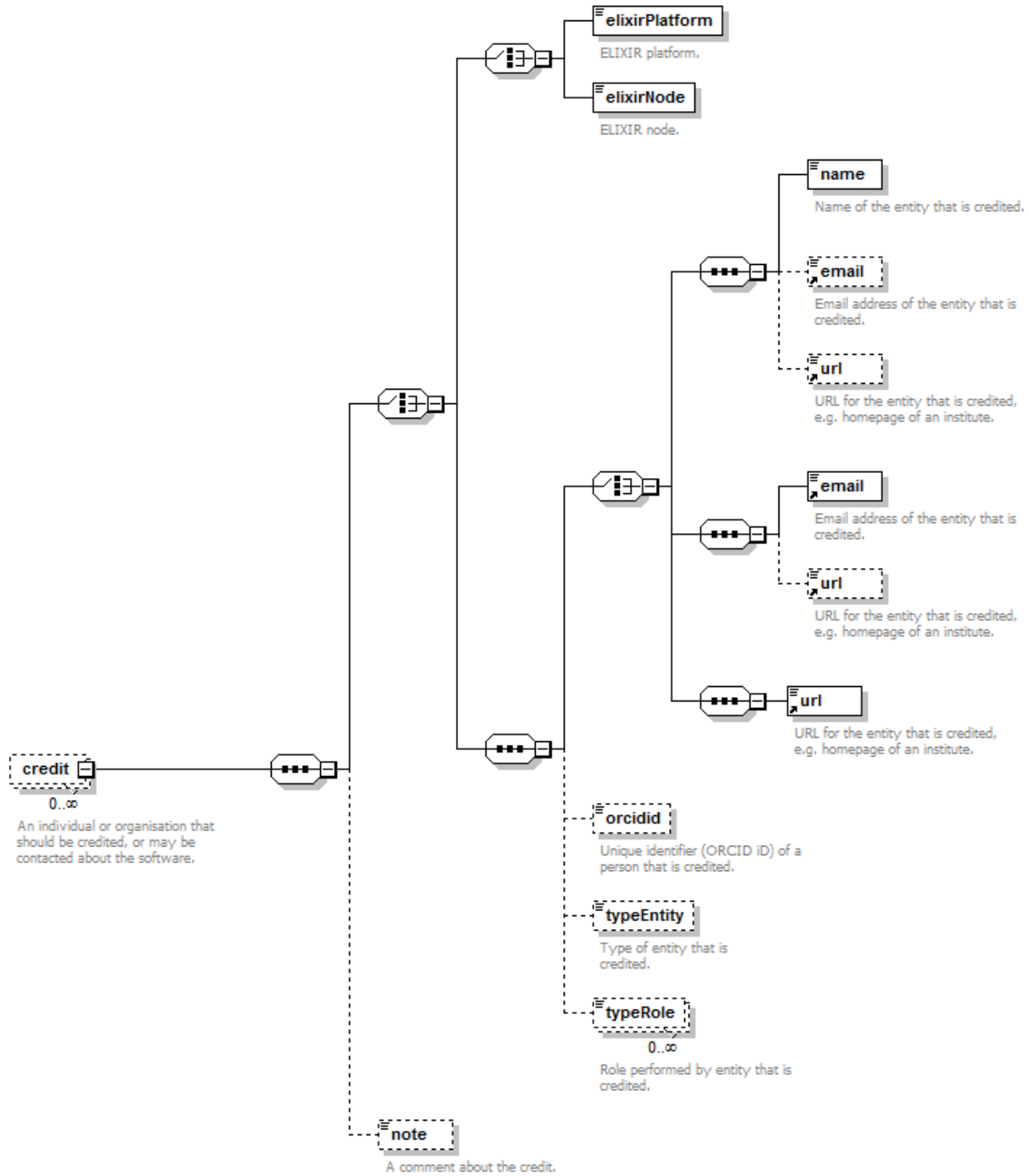


Element	Description	Type	Cardinality
biotoolsID	bio.tools ID of an existing bio.tools entry to which this software is related.	xs:token (restriction)	1 only
type	Type of relation between this and another registered software.	enum (see docs)	1 only

Note: See the [Curators Guide](#).

3.10 Credit group

Individuals or organisations that should be credited, or may be contacted about the software.



Element	Description	Type	Cardinality
name	Name of the entity that is credited.	xs:token (restriction)	0 or 1
orcidid	Unique identifier (ORCID iD) of a person that is credited.	xs:token (restriction)	0 or 1
gridid	Unique identifier (GRID ID) of an organisation that is credited.	xs:token (restriction)	0 or 1
rorid	Unique identifier (ROR ID) of an organisation that is credited.	xs:token (restriction)	0 or 1
fundrefid	Unique identifier (FundRef ID or Funder ID) of a funding organisation that is credited.	xs:token (restriction)	0 or 1
email	Email address.	email address	0 or 1
url	URL, e.g. homepage of an institute.	URL	0 or 1
tel	Telephone number.	xs:token (restriction)	0 or 1
typeEntity	Type of entity that is credited.	enum (see docs)	0 or 1
typeRole	Role performed by entity that is credited.	enum (see docs)	0 or more
note	A comment about the credit.	xs:token (restriction)	0 or 1

Note: See the [Curators Guide](#).

Controlled vocabularies

biotoolsSchema defines 17 controlled vocabularies - as standardised enumerations of terms - for description of technical software aspects (see below).

Vocabulary	Description
tool type	The type of application software.
operating system	The operating system supported by a downloadable software package, e.g. 'Linux'.
programming language	Name of programming language the software source code was written in, e.g. 'C'.
license	Software or data usage license, e.g. 'GPL-3.0'
maturity	How mature the software product is, e.g. 'Mature'.
cost	Monetary cost of acquiring the software, e.g. 'Free of charge'.
accessibility	Whether there are non-monetary restrictions on accessing an online service, e.g. 'Open access'.
link type	The type of data, information or system that is obtained when the link is resolved, e.g. 'Helpdesk'.
download type	Type of download that is linked to, e.g. 'Source code'.
documentation type	Type of documentation that is linked to, e.g. 'API documentation'.
publication type	Type of publication, e.g. 'Review'.
relation type	Type of relation between this and another registered software, e.g. 'is-NewVersionOf'.
entity type	Types of entities that may be credited, e.g. 'Person'.
entity role	Roles that may be assigned to creditable entities, e.g. 'Developer'.
ID type	Type of tool identifier, e.g. 'rrid'.
ELIXIR Platform	Name of the ELIXIR Platform that is credited, e.g. 'Tools'
ELIXIR Node	Name of the ELIXIR Node that is credited, e.g. 'Norway'
ELIXIR Community	Name of relevant ELIXIR (or associated) community, e.g. 'Galaxy'

4.1 Tool type

The type of application software: a discrete software entity can have more than one type, e.g. “Command-line tool, Web application”

Type	Description
Bioinformatics portal	web site providing a platform/portal to multiple resources used for research in a focused area, including biological databases, web applications, training resources and so on.
Command-line tool	A tool with a text-based (command-line) interface.
Database portal	A Web site providing a portal to a biological database, typically allowing a user to browse, deposit, search, visualise, analyse or download data.
Desktop application	A tool with a graphical user interface that runs on your desktop environment, <i>e.g.</i> on a PC or mobile device.
Library	A collection of components that are used to construct other tools. bio.tools scope includes component libraries performing high-level bioinformatics functions but excludes lower-level programming libraries.
Ontology	A collection of information about concepts, including terms, synonyms, descriptions etc.
Plug-in	A software component encapsulating a set of related functions, which are not standalone, <i>i.e.</i> depend upon other software for its use, <i>e.g.</i> a Javascript widget, or a plug-in, extension add-on etc. that extends the function of some existing tool.
Script	A tool written for some run-time environment (<i>e.g.</i> other applications or an OS shell) that automates the execution of tasks. Often a small program written in a general-purpose languages (<i>e.g.</i> Perl, Python) or some domain-specific languages (<i>e.g.</i> sed).
SPARQL endpoint	A service that provides queries over an RDF knowledge base via the SPARQL query language and protocol, and returns results via HTTP.
Suite	A collection of tools which are bundled together into a convenient toolkit. Such tools typically share related functionality, a common user interface and can exchange data conveniently. This includes collections of stand-alone command-line tools, or Web applications within a common portal.
Web application	A tool with a graphical user interface that runs in your Web browser.
Web API	An application programming interface (API) consisting of endpoints to a request-response message system accessible via HTTP. Includes everything from simple data-access URLs to RESTful APIs.
Web service	An API described in a machine readable form (typically WSDL) providing programmatic access via SOAP over HTTP.
Workbench	An application or suite with a graphical user interface, providing an integrated environment for data analysis which includes or may be extended with any number of functions or tools. Includes workflow systems, platforms, frameworks etc.
Workflow	A set of tools which have been composed together into a pipeline of some sort. Such tools are (typically) standalone, but are composed for convenience, for instance for batch execution via some workflow engine or script.

4.2 Operating system

The operating system supported by a downloadable software package, e.g. “Linux”

Maturity	Description
Linux	All flavours of Linux/UNIX operating systems.
Windows	All flavours of Microsoft Windows operating system.
Mac	All flavours of Apple Macintosh operating systems (primarily Mac OS X).

4.3 Programming language

Name of programming language the software source code was written in, e.g. “C”

See the [technical docs](#) for a list of valid terms.

4.4 License

Software or data usage license, e.g. “GPL-3.0”

See the [technical docs](#) for a complete list of valid terms. Noteworthy terms are below:

License	Description
Proprietary	Software for which the software’s publisher or another person retains intellectual property rights - usually copyright of the source code, but sometimes patent rights.
Freeware	Proprietary software that is available for use at no monetary cost. In other words, freeware may be used without payment but may usually not be modified, re-distributed or reverse-engineered without the author’s permission.
Not licensed	Software which is not licensed and is not proprietary.
Other	Software under license not currently supported by biotoolsSchema.

4.5 Maturity

How mature the software product is, e.g. “Mature”

Maturity	Description
Emerging	Nascent or early release software that may not yet be fully featured or stable.
Mature	Software that is generally considered to fulfill several of the following: secure, reliable, actively maintained, fully featured, proven in production environments, has an active community, and is described or cited in the scientific literature.
Legacy	Software which is no longer in common use, deprecated by the provider, superseded by other software, replaced by a newer version, is obsolete etc.

4.6 Cost

Monetary cost of acquiring the software, e.g. “Free of charge (with retritions)”

Cost	Description
Free of charge	Software which available for use by all, with full functionality, at no monetary cost to the user.
Free of charge (with restrictions)	Software which is available for use at no monetary cost to the user, but possibly with limited functionality, usage restrictions, or other limitations.
Commercial	Software which you have to pay to access.

4.7 Accessibility

Whether there are non-monetary restrictions on accessing an online service, e.g. “Open access”

Accessibility	Description
Open access	An online service which is available for use to all, but possibly requiring user accounts / authentication.
Open access (with restrictions)	An online service which is available for use to all, but possibly with some usage limitations and other restrictions.
Restricted access	An online service which is available for use to a restricted audience, e.g. members of a specific institute.

4.8 Link type

The type of data, information or system that is obtained when the link is resolved, e.g. “Mailing list”

Link type	Description
Discussion forum	Online forum for user discussions about the software.
Galaxy service	An online service providing the tool through the Galaxy platform.
Helpdesk	A phone line, web site or email-based system providing help to the end-user of the software.
Issue tracker	Tracker for software issues, bug reports, feature requests etc.
Mailing list	Mailing list for the software announcements, discussions, support etc.
Mirror	Mirror of an (identical) online service.
Software catalogue	Some registry, catalogue etc. other than bio.tools where the tool is also described.
Repository	A place where source code, data and other files can be retrieved from, typically via platforms like GitHub which provide version control and other features, or something simpler, e.g. an FTP site.
Social media	A website used by the software community including social networking sites, discussion and support fora, WIKIs etc.
Service	An online service (other than Galaxy) that provides access (an interface) to the software.
Technical monitoring	Information about the technical status of a tool.
Other	Other type of link for software - the default if a more specific type is not available.

4.9 Download type

Type of download that is linked to, e.g. “Binaries”

Download type	Description
API specification	File providing an API specification for the software, e.g. Swagger/OpenAPI, WSDL or RAML file.
Biological data	Biological data, or a web page on a database portal where such data may be downloaded.
Binaries	Binaries for the software; compiled code that allow a program to be installed without having to compile the source code.
Command-line specification	File providing a command line specification for the software.
Container file	Container file including the software.
Icon	Icon of the software.
Screenshot	Screenshot of the software.
Source code	The source code for the software, that can be compiled or assembled into an executable computer program.
Software package	A software package; a bundle of files and information about those files, typically including source code and / or binaries.
Test data	Data for testing the software is working correctly.
Test script	Script used for testing whether the software is working correctly.
Tool wrapper (CWL)	Tool wrapper in Common Workflow Language (CWL) format for the software.
Tool wrapper (galaxy)	Galaxy tool configuration file (wrapper) for the software.
Tool wrapper (tav-erna)	Taverna configuration file for the software.
Tool wrapper (other)	Workbench configuration file (other than taverna, galaxy or CWL wrapper) for the software.
VM image	Virtual machine (VM) image for the software.
Downloads page	Web page summarising general downloads available for the software.
Other	Other type of download for software - the default if a more specific type is not available.

4.10 Documentation type

Type of documentation that is linked to, e.g. "Citation instructions"

Documentation type	Description
API documentation	Human-readable API documentation.
Citation instructions	Information on how to correctly cite use of the software; typically which publication(s) to cite, or something more general, e.g. a form of words to use.
Code of conduct	A set of guidelines or rules outlining the norms, expectations, responsibilities and proper practice for individuals working within the software project.
Command-line options	Information about the command-line interface to a tool.
Contributions policy	Information about policy for making contributions to the software project.
FAQ	Frequently Asked Questions (and answers) about the software.
General	General documentation.
Governance	Information about the software governance model.
Installation instructions	Instructions how to install the software.
Quick start guide	A short guide helping the end-user to use the software as soon as possible.
User manual	Information on how to use the software, tailored to the end-user.
Release notes	Notes about a software release or changes to the software; a change log.
Terms of use	Rules that one must agree to abide by in order to use a service.
Training material	Online training material such as a tutorial, a presentation, video etc.
Other	Some other type of documentation not listed in biontoolsSchema.

4.11 Publication type

Type of publication, e.g. “Primary”

Publication type	Description
Primary	The principal publication about the tool itself; the article to cite when acknowledging use of the tool.
Method	A publication describing a scientific method or algorithm implemented by the tool.
Usage	A publication describing the application of the tool to scientific research, a particular task or dataset.
Benchmarking study	A publication which assessed the performance of the tool.
Review	A publication where the tool was reviewed.
Other	A publication of relevance to the tool but not fitting the other categories.

4.12 Relation type

Type of relation between this and another registered software, e.g. “isNewVersionOf”

Relation type	Description
isNewVersionOf	The software is a new version of an existing software, typically providing new or improved functionality.
hasNewVersion	(inverse of above)
uses	The software provides an interface to or in some other way uses the functions of other software under the hood, e.g. invoking a command-line tool or calling a Web API, Web service or SPARQL endpoint to perform its function.
usedBy	(inverse of above)
includes	A workbench, toolkit or workflow includes some other, independently available, software.
includedIn	(inverse of above)

4.13 Entity type

Type of entity that is credited, e.g. "Person"

Entity type	Description
Person	Credit of an individual.
Project	Credit of a community software project not formally associated with any single institute.
Division	Credit of or a formal part of an institutional organisation, e.g. a department, research group, team, etc
Institute	Credit of an organisation such as a university, hospital, research institute, service center, unit etc.
Consortium	Credit of an association of two or more institutes or other legal entities which have joined forces for some common purpose. Includes Research Infrastructures (RIs) such as ELIXIR, parts of an RI such as an ELIXIR node etc.
Funding agency	Credit of a legal entity providing funding for development of the software or provision of an online service.

4.14 Entity role

Role performed by entity that is credited, e.g. "Developer"

Role	Description
Developer	Author of the original software source code.
Maintainer	Maintainer of a mature software providing packaging, patching, distribution etc.
Provider	Institutional provider of an online service.
Documentor	Author of software documentation including making edits to a bio.tools entry.
Contributor	Some other role in software production or service delivery including design, deployment, system administration, evaluation, testing, documentation, training, user support etc.
Support	Provider of support in using the software.
Primary contact	The primary point of contact for the software.

4.15 ID type

Type of tool identifier, e.g. "rrid"

Role	Description
doi	Digital Object Identifier of the software assigned (typically) by the software developer or service provider.
rrid	Research Resource Identifier as used by the NIH-supported Resource Identification Portal (https://scicrunch.org/resources).
cpe	Common Platform Enumeration (CPE) identifier as listed in the CPE dictionary (https://cpe.mitre.org/dictionary/).
biotoolsCURIE	bio.tools CURIE (secondary identifier).

4.16 ELIXIR Platform

Name of the ELIXIR Platform that is credited, e.g. “Tools”

ELIXIR Platform	Description
Data	ELIXIR Data Platform
Tools	ELIXIR Tools Platform
Compute	ELIXIR Compute Platform
Interoperability	ELIXIR Interoperability Platform
Training	ELIXIR Training Platform

4.17 ELIXIR Node

Name of the ELIXIR Node that is credited, e.g. “Norway”

ELIXIR Node
Belgium
Czech Republic
Denmark
EMBL
Estonia
Finland
France
Germany
Greece
Hungary
Ireland
Israel
Italy
Luxembourg
Netherlands
Norway
Portugal
Slovenia
Spain
Sweden
Switzerland
UK

4.18 ELIXIR Community

Name of relevant ELIXIR (or associated) community, e.g. "Galaxy"

ELIXIR Node
3D-BioInfo
Federated Human Data
Galaxy
Human Copy Number Variation
Intrinsically Disordered Proteins
Marine Metagenomics
Metabolomics
Microbial Biotechnology
Plant Sciences
Proteomics
Rare Diseases

5.1 Governance

biotooolsSchema is a community-driven project governed under the leadership of the [French ELIXIR node](#) (Jacques van Helden, Joint Head of Node) in collaboration with other national [ELIXIR nodes](#) and partners beyond ELIXIR. biotooolsSchema development is supported by the French ELIXIR node, and has benefited from the voluntary contributions of many individuals since its inception.

Development on the ground is led by French ELIXIR Node staff and affiliates (Hervé Ménager, Jon Ison and Alban Gaignard), in close collaboration with the ELIXIR Tools Platform and other partners within and beyond the [ELIXIR infrastructure](#).

If you'd like to get involved with the project please contact us via [GitHub](#).

5.2 Main authors & contributors

- Jon Ison (CNRS, Institut Français de Bioinformatique, FR) - *lead developer*
- Hans-Ioan Ienasescu (University of Copenhagen, DK)
- Emil Rydza (formerly University of Copenhagen, DK)
- Piotr Chmura (University of Copenhagen, DK)
- Kristoffer Rapacki (formerly CBS-DTU, DK)
- Alban Gaignard (L'institut du Thorax, FR)
- Veit Schwämmle (University of Southern Denmark, DK)
- Jacques van Helden (Aix-Marseille Université, FR)
- Matúš Kalaš (University of Bergen, NO)
- Hervé Ménager (Institut Pasteur, FR)

5.3 Contributors

Thanks to the many people who have contributed - if you're not listed below, please let us know!

- Dmitry Repchevski (BCS, ES)
- Piotr Wojciech Dabrowski (RKI, DE)
- Bjoern Gruening (University of Freiburg, DE)
- Chris Morris (STFC, UK)
- Frederik Coppens (ELIXIR-BE)
- Heinz Stockinger (ELIXIR-CH)
- Helen Parkinson (EMBL-EBI, UK)
- Julie McMurry (EMBL-EBI, UK)
- Magnus Palmblad (LUMC, NL)
- Michael Crusoe (University of California)
- Niall Beard (University of Manchester)
- Rodrigo Lopez (EMBL-EBI)
- Delegates of the many BioMedBridges and ELIXIR [workshops and meetings](#).

CHAPTER 6

License

biotoolsSchema is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License ([CC BY-SA 4.0](#)).

We recommend, however, that while biotoolsSchema is being actively maintained by its authors, substantial derived work, major modifications *etc.* are consulted with the developers beforehand at the time of consideration, and consistent solutions are sought in collaboration.