Project Title	European Science Cluster of Astronomy & Particle physics ESFRI research Infrastructure
Project Acronym	ESCAPE
Grant Agreement No	824064
Instrument	Research and Innovation Action (RIA)
Topic	Connecting ESFRI infrastructures through Cluster projects (INFRA-EOSC-4-2018)
Start Date of Project	04.02.2019
Duration of Project	42 Months
Project Website	https://projectescape.eu/

D3.7 - License, provenance and metadata guidelines for the software and service repository

Work Package	WP3 OSSR
Lead Author (Org)	Thomas Vuillaume, LAPP (CNRS)
Contributing Author(s) (Org)	Enrique Garcia, LAPP (CNRS)
Due Date	31.12.2020 - M24
Date	29.01.2021
Version	1.0

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

Dissemination Level

X	PU: Public
	PP: Restricted to other programme participants (including the Commission)
	RE: Restricted to a group specified by the consortium (including the Commission)
	CO: Confidential, only for members of the consortium (including the Commission)

Versioning and contribution history

Version	Date	Authors and Contributors	Notes
0.1	09.11.2020	Thomas Vuillaume (LAPP, CNRS)	Setup document and plan
0.2	17.12.2020	Enrique Garcia, Thomas Vuillaume (LAPP, CNRS)	First draft
0.3	23.12.2020	Jutta Schnabel (FAU)	Discussion on first draft
0.4	15.01.2021	ESCAPE WP3	First internal WP3 revision
0.5	21.01.2021	Enrique Garcia, Thomas Vuillaume (LAPP, CNRS)	Terminology adjustments, formatting issues
0.6	22.01.2021	ESCAPE WP3	Include inputs to second draft
1.0	29.01.2021	Enrique Garcia (LAPP), Harro Verkouter (JIVE), Kay Graf (FAU), Mark Kettenis (JIVE), Jutta Schnabel (FAU), Mathieu Servillat (OBSP), Christian Tacke (GSI), Thomas Vuillaume (LAPP)	Final editing session

Disclaimer

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

Table of Contents

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

1. Introduction

With the implementation of the ESCAPE *Open-source Scientific Software and Service Repository* (OSSR)[1] as conceptionally proposed in deliverable D3.3[2], partners of the particle physics, astroparticle and astronomy communities are able to aggregate their software in a common place. However, this alone is not sufficient to answer the *FAIR* (Findable, Accessible, Interoperable and Reusable) principles[3,4]. These statements - that govern open data publications - list the licensing of published resources like data, software and supplementary material as one of the key principles. This ensures legal security for the user of the resources, and is therefore considered an essential part of the "re-usability"-requirement of FAIR data. In other words, any software should be assigned a well-chosen license (FAIR principle R1.1)[5]. Also, a minimum amount of provenance information needs to be provided together with the software, as well as a metadata file so that users and services can find it, understand the status and integrate it in their respective environments.

In this document, we propose the open licenses, the metadata schema and the minimum provenance information that any software should contain to be included in the OSSR. The provenance for the OSSR is discussed and a unified metadata implementation (that contains the minimal provenance information) is proposed, using the *CodeMeta* standard[6]. We explain the reasoning behind these choices. Finally, we expose guidelines and an example of a project following these principles in the OSSR, thus providing guidance for future partners and projects.

2. Licences, authorship and copyright

2.1. Open Licenses

It is a common misunderstanding that not including a license in a project makes the software open-source and available to all. Actually, any package, library, source code or any other form of software that is supposed to be interoperable and reusable will always require a license, as all rights are protected by the applicable national copyright law. This generally prohibits or largely restricts the reuse of the software, especially regarding redistribution and inclusion of (or parts of) the source code or libraries in public software projects, as is the case in the OSSR. Thus, through the application of a license, the reuse of the software can be granted to any user and or limited by the conditions introduced in the license in a legally safe manner.

[1] https://zenodo.org/communities/escape2020
[2] https://projectescape.eu/sites/default/files/ESCAPE_D3.3_Conceptual%20Design%20Report.pdf
[3] https://www.go-fair.org/fair-principles/
[4] Although used as guideline here, the FAIR principles for data might not be fully valid in the regime of software and services, as the *Report from the EOSC Executive Board Working Group (WG) Architecture Task Force (TF) SIRS* points out.
[5] https://www.go-fair.org/fair-principles/r1-1-metadata-released-clear-accessible-data-usage-license/
[6] https://codemeta.github.io/

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

2.2. Authorship and Copyright

The moment any creative work is created, the holder of copyright is generally the creator, and for collaborative works all authors jointly. The following exception must be emphasized: the copyright of contractually developed software belongs to the employer (see EU Directive 91/250/EEC[7]). Ancillary copyright can generally be contractually transferred.

As most software developed in the OSSR context is created by employees of the various institutes and contributors, the contractual regulations in employment contracts or funding and collaborative agreements have to be considered when determining the copyright holder. In addition to that, third-party contributions are common in software development, the authors of which would keep the copyright to their contribution. This necessitates diligent management of the copyright when distributing software.

3. License types

Licensing is a subject that starts to be commonly treated and discussed in software and astronomy related conferences. As an example, at the last *ADASS XXX* online conference[8], a specific session on licensing was scheduled[9], showing an increasing use, interest and understanding by the community. Also, within the ESCAPE community, a *Workshop on Open Science Software Lifecycles* (WOSSL)[10] has been organised by WP3, during which provenance and licensing issues were presented and discussed. The outcome and summary were compiled within a live webpage[11] allowing further discussions.

In general terms, there are two kinds of software licenses: **proprietary** and **open source**. The first ones require the explicit permission of the author(s) for use, modification and distribution. Therefore, this kind of licenses are out of the scope of the ESCAPE and EOSC projects. We do not recommend that software with a proprietary license is included into the ESCAPE open-source ecosystem, if no solid justification for the use of such a license is provided, as this would prevent its Interoperability and Reusability, pillars of the FAIR principles.

Among the open-source (OS) licenses, there are three general types; **permissive, weak copyleft** and **strong copyleft** (described below). A full list of the existing open-source licenses can be found in the following hyperlinks[12,13], not detailed in this document for readability purposes.

[7] https://en.wikipedia.org/wiki/Computer_Programs_Directive
[8] https://adass2020.es/
[9] https://schedule.adass2020.es/adass2020/talk/DUE8F7/
[10] https://indico.in2p3.fr/event/21698/
[11] https://escape2020.pages.in2p3.fr/wp3/wossl/
[12] https://spdx.org/licenses/
[13] https://opensource.org/licenses/alphabetical

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

The *Open Source Initiative*[14] defines a permissive license as a non-copyleft license, thus a license that guarantees the freedom of use, modification, redistribution and that permits proprietary derivative works[15]. On the contrary, a copyleft license is a license that allows derivative works but obliges to use the same license as the original work[16]. Common examples of permissive licenses are the MIT and the BSD-3 Clause licenses[17]. An example of weakly permissive is the LGPLv3+ (GNU Lesser General Public License or later). Last, the prime example of a copy-left license is the GPLv3 license (GNU General Public License). It should be noted that copy-left or lesser copy-left licenses generally enforce the use of open licenses on all derived software, spreading the spirit of open software, while on the other hand they can limit the interoperability of the software, generating a conflict with the FAIR criterion.

To conclude, the current license challenges lie in;

1. choosing the appropriate (open-source) license for a project, a software, source code or any other supplementary material.
2. to be license compliant when creating a product from a combination of various parent projects with different licenses. What it is called derivative software (see below).

License compatibility has to be ensured by the contributors, as compliance with licensing standards is required by the OSSR terms of use and should be a point to be stressed on the OSSR guidelines. Furthermore, this verification must come from the contributor side.

License compatibility matrices can be found on the web[18,19], and specifying all the possible combinations is not possible in this document. An example of how to integrate a BSD-3 Clause license library into a MIT licensed project can be found at the ESCAPE template project[20], and information resources on licensing are provided by OSSR[21].

4. OSSR Software products

Software and services in the OSSR can be contributed in different forms - as software repository that contains source code and compilation instructions, or as software image for the creation of a container of pre-compiled software to be run as service. In both cases, the software is most likely built on external libraries that themselves carry a license. The licenses used for OSSR products must therefore follow the respective conditions for these products and be handled accordingly by the party offering their software through the OSSR (contributor).

[14] https://opensource.org/
[15] https://opensource.org/faq#permissive
[16] https://opensource.org/faq#copyleft
[17] The Apache 2.0 license is another common example; it should be underlined, however, that its patent retaliation clause imposes additional restrictions and that it is incompatible with the GPLv2 license.
[18] https://en.wikipedia.org/wiki/License_compatibility
[19] https://en.wikipedia.org/wiki/Comparison_of_free_and_open-source_software_licences
[20] https://gitlab.in2p3.fr/escape2020/wp3/template_project_escape/-/blob/master/LICENSE
[21] As starting point, a knowledge base has been created to collect guidelines about licensing: https://escape2020.pages.in2p3.fr/wp3/licensing

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

- **Source code**: Software source code can be offered through OSSR as code repository. The software is entered by the contributor with permission of the copyright holder(s) and assigned a license that meets FAIR requirements.
- **Derived software**: Software is generally created including compiled libraries of third-party software. The license conditions of the included libraries have to be checked for compliance and suitability for redistribution by the contributor.
- **Software container images**: The assembly of compiled software (and other products like data) in software container images as copyright protected act requires the assigning of a license to the software image as such. The contributor must therefore ensure that not only the content of a software container image carries all necessary licenses, but also the image itself is assigned a license.
- **ESCAPE software products**: For software products developed in the context of the ESCAPE projects, the contributors can be more explicitly advised to choose licenses to foster FAIR principles, and according guidelines will be made available.

5. Provenance Information in the OSSR

Provenance (*information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness*[22]) is a vast word, and depending on the context its meaning and implications can change. The level and granularity of the provenance information can widely vary, from providing a single contact point, to a detailed path or mapping of each step, stage, actor and tools used in the resulting product (as given by a complete git log of a source code development for example).

Similar to licensing, provenance also starts to be discussed and treated in software and astronomy conferences, e.g. in the scheduled provenance session at the ADASS XXX conference[23], and its proceeding[24]. Within the ESCAPE community, an ESCAPE workshop on provenance (in WP4), was also organised[25].

The intention within the ESCAPE WP3, as well as the result of this deliverable, is to define the minimum amount of provenance that should be provided within the publications and entries that will be published into the OSSR, rather than defining a full provenance model. This provenance will be referred hereafter as "provenance representation for the OSSR".

In an exercise of simplicity, the minimum provenance metadata needed for software should be;

- The list of Authors,
- The list of Contributors and the kind of contribution (Editor, Maintainer, Sponsor, Data curator ...),

[22] https://www.ivoa.net/documents/ProvenanceDM/
[23] https://schedule.adass2020.es/adass2020/talk/P3FK9U/
[24] https://arxiv.org/abs/2101.08691
[25] https://indico.in2p3.fr/event/21913

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

- The version of the software.

As it can be seen in the following section, this information is included into the OSSR metadata representation that we propose with this document.

Again, the verification that the entry to be published into the OSSR contains the metadata compliant with the proposed schema must come from the contributor side.

6. Metadata for the OSSR

To answer the FAIR principles, software annotation by metadata is essential. Without it, software is not Findable nor Accessible, and hardly Interoperable and Reusable. In this section, we define the OSSR metadata schema including provenance data, summing the information that must be provided alongside a software product in the OSSR.

Note that this metadata schema is the reflection of our current understanding of the ESCAPE environment and is subject to evolution with the combination of ESCAPE services that might require extra information to be provided in order to use the software correctly, e.g. in common workflows combining different software and/or service solutions. The metadata is divided into two categories: required and recommended. Changes to the definition and classification of the items might occur to meet requirements for future developments in the ESCAPE project.

Required

- Title
- Authors: list of authors
- Contact (maintainer):
  - name: could be a person or an entity (e.g. E. Garcia, ESCAPE or CTA Observatory)
  - email or other contact channel
- Short description of the software/service
- License

Recommended

- Provide a persistent identifier (such as a Digital Object Identifier (doi), if already available)
- Publication date (if already published elsewhere)
- URL to the (live) development repository
- Type of publication: source code, compiled binaries, container, image
- Grant/funding
- Contributors
- References
- Programming language
- External dependencies (including matching versions)
- Operating system
- Compilation environment

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

- Hardware requirements
  - General use case (HPC, server, local desktop)
  - CPU, RAM, HDD/SSD requirements
  - GPU requirements
  - FPGA
- Data Type (inputs/outputs)
- Access
  - Open
  - Closed
  - Restricted
  - Embargoed
- Documentation language
- Keywords: to be selected in a defined list of keywords, see below

Keywords list
This list of keywords is the reflection of the ESCAPE environment and is subject to evolution. A provider may use as many keywords as desired, however we recommend adding the relevant keywords from this list to facilitate software integration and findability among projects and partners. Additional controlled vocabularies and a categorisation of the keyword list will be added at a later stage.

- CTA
- LSST
- LOFAR
- SKA
- EGO-Virgo
- KM3NeT
- ELT
- EST
- HL-LHC
- FAIR
- CERN
- ESO
- JIVE
- VO
- EOSC
- ESO
- Astronomy
- Astroparticle physics
- Particle physics

6.1. Relation between provenance, metadata and license

Even though comprehensive provenance, metadata and open license are not sufficient to ensure the FAIR principles, they are essential pillars. Without proper provenance and metadata, software will not be findable, not accessible, and hardly interoperable. Without an open license, it will not be interoperable or reusable. It appears also crucial that the license be stated in the metadata to ease reusability of the software.

6.2. Metadata implementation

The compulsory information that any new entry/resource must contain when it is uploaded to Zenodo, and especially the ESCAPE Zenodo community, might seem a good solution to be

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

used as the OSSR metadata. Even more when we ourselves realise that any entry published in the repository can be exported into different metadata schemas and/or common citing formats (e.g. *DataCite*, *Dublin Core*, *BibTeX*, *Schema.org* ...). However, the information that Zenodo requires is not enough for the ESCAPE needs. The described entries that can be exported are only the mandatory entries plus some of the optional ones.

One the contrary, there are various metadata standards contexts that satisfy the whole OSSR key metadata proposed in this document. A metadata context defines the metadata schema to be followed, whereas the schema is the organization and structure that the metadata will follow. In other words, the schema will define the syntax as well as the keys to be used in the metadata file.

Beside the context, there exists the metadata encoding; that is the programming language in which the file will be written. JSON-LD and XML are the two most common languages used for metadata files. While the former is more human-readable, the latter is more machine oriented.

The *CodeMeta* project[26] proposes a schema context focused on providing the minimal metadata schema for scientific software and code. It is based on the *Schema.org* schema[27] and it was extended by the same project. This means that the CodeMeta format uses a "type" of the schema.org schema (the *SoftwareSourceCode* type) and it extends it by providing additional properties/terms that are useful in the context of software description. The reasons to choose the CodeMeta project schema and implement it into the OSSR ecosystem are various;

1. It completely fulfils the OSSR provenance description,
2. As community effort, it is supported by the Software Heritage[28] foundation and the Astrophysics Source Code Library (ASCL)[29],
    a. as well as by Zenodo, although not implemented yet.
3. It is commonly used in the software development community (as seen at[30]), and it is a context oriented specifically to describe source code,
4. It can be easily extended with any other schema.org type (like any property/term within the *SoftwareApplication*, *DataSet*, *Thing*... types),
5. Vocabularies as crosswalks to other ontologies are available.

Using the CodeMeta metadata standard and including a *codemeta.json* file (as described in the guidelines section 7) would allow that any other external service, project, user or entity that can harvest and read a CodeMeta context file, will be actually able to fulfil all the four FAIR principles, i.e., find, access, interoperate and reuse the cited entity. Moreover, a detailed and complete metadata file would also contain most of the minimum desired metadata provenance information for the OSSR (with the exception of the *release notes*, see below). **Because of the above mentioned reasons, we have decided to use and implement the CodeMeta context into the OSSR and its ecosystem**.

[26] https://codemeta.github.io/
[27] https://schema.org/
[28] https://www.softwareheritage.org/
[29] https://ascl.net/
[30] https://schedule.adass2020.es/adass2020/talk/DUE8F7/

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

All the CodeMeta schema terms are listed in https://codemeta.github.io/terms, and as mentioned above, the schema can be extended[31] by using any term from the schema.org schema[32]. It must be also stressed that the same CodeMeta project provides a *codemeta.json* file generator[33] that allows the creation of a metadata file in a quick way. The web application also allows the import and validation of *codemeta.json* files. Finally, the implemented version is the CodeMeta Schema v2.0[34].

The fact that the CodeMeta schema can be extended is an extremely important point. In the whole ESCAPE and EOSC ecosystem, the OSSR will need to interact with other services, and likely with external ones. To date, CodeMeta fulfils the OSSR needs, but there is an ongoing discussion about how to connect to other services, in particular with the ESCAPE ESFRI Science Analysis Platform (ESAP). Here are some of the points currently under investigation;

- How to describe an (analysis) pipeline using the CodeMeta schema?
  - Is the schema itself enough to provide all the needed information to the analysis platform?
  - If not, the CodeMeta context must be extended to correctly describe the pipeline and ensure for reusability.
- Is the CodeMeta context complete enough to describe the content of the Jupyter notebooks in a project?
  - Should these notebooks be included as part of the same parent publication into the OSSR; or should they be provided in another, though linked, resource?
  - Should the notebooks be considered part of a source code/resource, or part of a pipeline?
- Similar questions appear for the software provided as or within containers;
  - Is it the CodeMeta context enough to describe these this kind of digital resource?
  - What information must be provided to implement their provenance? Should the container receipt be included as a compulsorily part of the repository entry?

These requirements and questions will be refined with the development of the ESAP and its connection to the OSSR. As already can be foreseen, some of these requirements will be very specific to ESCAPE services and/or have not been addressed by provenance implementation available to date (and to the best of our knowledge). The possibility to extend the CodeMeta context thus appear as essential for the future developments of provenance, which is bound to evolve with time and with the supported services.

Up to now it has been shown that using a CodeMeta metadata file to describe the resources within the OSSR works for the ESCAPE ecosystem (except for the ongoing discussion points mentioned above). Including a metadata file in a resource will describe the publication within the OSSR for the rest of ESCAPE services as well as for any other EOSC external service that can interpret the schema.

[31] https://codemeta.github.io/developer-guide/#extending-the-codemeta-context
[32] https://schema.org/docs/full.html
[33] https://codemeta.github.io/codemeta-generator/
[34] https://raw.githubusercontent.com/codemeta/codemeta/2.0/codemeta.jsonld

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

However, while testing the described implementation, we realised about a missing block in the development platform-repository connection (GitLab-Zenodo). Zenodo, despite being supporters of the CodeMeta project, cannot ingest a *codemeta.json* file and use the same to include the information into a new entry or new version. To solve this missing link, a package was developed and implemented into the ESCAPE libraries that are used for the continuous integration of a resource into the Zenodo repository[35,36], as presented and described in the deliverable 3.3. The developed package (`codemeta2zenodo`) maps the fields of the *codemeta.json* file and creates the Zenodo-compliant corresponding metadata file; a *.zenodo.json* file. This way, by including a single CodeMeta metadata file, a digital resource can be findable either from the analysis platform or the repository, and - at the same time - it will be possible to automatically deploy it into the Zenodo repository thanks to the CI pipeline (`ZenodoCI`) and the developed mapping library. Again, and closing the circle, the *.zenodo.json* file cannot be used as the metadata file to be fetched by other services, as e.g. the EOSC portal, because its lack of information about the resource - despite the possibility of being exported into multiple contexts and formats.

A last point should be mentioned for clarification. The *codemeta.json* metadata file is not intended for scientific citation, whereas there are specific metadata files and objects that already satisfy this purpose. The *codemeta.json* is **purely intended to fulfil the FAIR principles**. Scientific citation is out of the scope of the OSSR onboarding implementation, although we would fully recommend the use of standards like the *CITATION.cff* citation standard file[37], for example.

6.3. Future ESCAPE provenance and implementation with other WPs

The current metadata for the OSSR and its implementation has been proposed to describe the basic provenance of source codes and derived software and allow services, machines and humans to find and reuse this software. This implementation will surely evolve with the ESCAPE needs in the future. One known caveat in the current metadata is a way to connect the source code with their associated software containers, images, workflows, or final ESCAPE software products.

The evolution of ESCAPE provenance and metadata will come with the evolution and maturation of its services and their needs. The proposal in this document is not a complete provenance model, however it establishes the starting point for this final model that should be able to describe in detail software products with container images and/or full workflows. This starting point is the suggestion of the kind of license that any future OSSR entry should contain, together with stabilising and standardizing the minimum metadata and provenance information that must be provided with it.

In order not to stop the onboarding process, and at the same time continue developing links with the different ESCAPE WPs, all the OSSR entries would need to be internally categorized temporarily. This would allow to rapidly find the entries whose metadata likely need to be

[35] https://gitlab.in2p3.fr/escape2020/wp3/codemeta2zenodo
[36] https://gitlab.in2p3.fr/escape2020/wp3/zenodoci
[37] https://citation-file-format.github.io/

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

extended (the identified cases are described in the previous section), and adapt their provenance model later on.

7. Guidelines and rules of participation to the ESCAPE OSSR

The following guidelines indicate how to provide software to the OSSR and under what conditions, in particular regarding licensing and provenance. Integration of a new entry in the OSSR is submitted to a curation phase that will validate its compliance to the following guidelines.

1. Any digital resource provided into the OSSR **must contain a license**.
    a. It most provide a machine-readable description of the license.
    b. The application of the FAIR principles to the products offered in the OSSR results in the necessity to assign licenses to all relevant parts, including software code, containerized and bundled software and metadata.

2. We **strongly recommend** to provide an **open-source** license. The choice of a **permissive** open-source license is **advised**.
    a. The Terms of Use for the OSSR require contributors/providers to ensure that all copyright is respected when offering software products through the OSSR. Thus, compatibility between the licenses of derivative or combined works must be checked before the on-boarding into the OSSR and done **by the contributor/provider side**.
    b. The choices of proprietary licenses must be justified.
    c. If no license is assigned yet, it is recommended to use either the MIT or BSD-3 Clause license.
    d. The metadata file itself is considered to be offered under a CC0 license[38].
    e. For contributed software, guidelines are collected and offered to the contributors to manage the licenses in their software product[39].
    f. Final ESCAPE open science products, as well as container images, are expected to follow these guidelines too.

3. We **strongly recommend** that any software provided to the OSSR incorporates a metadata file that describes the software to be uploaded. The use of the CodeMeta project schema, and therefore the addition of a ***codemeta.json*** **metadata** file into the root directory of the resource is recommended for the OSSR.
    a. The metadata file can be generated at https://codemeta.github.io/codemeta-generator/.
    b. The **minimum metadata information** to be added into any digital resource is the one required to create a new upload or a new version within Zenodo (see *Metadata for the OSSR* section). However, we **strongly recommend** to create a metadata file as complete as possible.
    c. In relation with the previous point, it is recommended to provide a summary of user-visible changes between versions (e.g. provide a 'release note' with every

[38] Creative Commons Public domain dedication: https://creativecommons.org/publicdomain/zero/1.0/
[39] As starting point, a knowledge base has been created to collect guidelines about licensing: https://escape2020.pages.in2p3.fr/wp3/licensing

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

new software release as well as fill this field within the *codemeta.json*) together with the OSSR software product or its documentation. This is essential provenance information and helps fulfilling the FAIR principles.

d. The verification that the entry contains the minimum provenance information defined in this document is the responsibility of the **software provider**.
e. To date, the only ZenodoCI compatible metadata context file is the *codemeta.json* schema file.
f. Providing a *codemeta.json* file with your software, in concurrence with the use of the ZenodoCI will allow the provider to define all software metadata using a single model and implementation. The integration of the software into the OSSR, as well as the use of other services (such as the production of a container for your software) will be based on this metadata.

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.