Research Data Repository Interoperability WG Final Recommendations

Abstract

This document represents the final outcome of the RDA Research Data Repository Interoperability Working Group (WG) and has been developed between July 2017 and March 2018 by the WG's members. It contains recommendations for achieving research data repository platform interoperability on a level needed to realize the use cases described in the WG's Case Statement [1]. In order to reach this goal, two steps have been taken by the WG: The definition of an exchange format and the description of functional requirements needed in order to exchange data in the defined format.

1. Introduction

Exchanging digital content between research data repository platforms in a machine-operable way is even today, mainly due to the vast amount of different platforms, versions and underlying data models, a challenging task. During discussions that took place in the first phase of this working group we were facing a huge interest in this topic and especially in a final outcome that can be easily adopted for the majority, if not for all, research data repository platforms. However, we also got an impression of why this topic has not been addressed with a broader focus, yet, reflected in the primer document [2] which was the first deliverable of this WG. One can see, especially in the platform capability matrix [3], which is also part of the primer document, that there are not many commonalities between the evaluated platforms with regard to standard models, interfaces or generic tools. Only OAI-PMH and tendencies towards the support of DataCite stood out in the final matrix, and both focus only on metadata.

On the other side, there are also upcoming standards and specifications like ResourceSync [4], SWORDv3 [5] or DataCrate [6] trying to cope with the interoperable exchange of digital content between different platforms. The latter two are still in their specification phase. Therefore, contacts have been established, there is an agreement to align efforts, but it was just too early to consider the one or the other into this WG's recommendations.

The third promising candidate in terms of research data repository interoperability is ResourceSync. Its main focus is on synchronizing resources via the network using Web standards like HTTP and the Sitemap protocol [7]. Unfortunately, with these constraints, ResourceSync is out of scope for this WG as the effort for implementing ResourceSync, which has, according to our capability matrix, to be done for all listed platforms, seems to be significant and only applicable to HTTP-accessible platforms. Furthermore, ResourceSync support would only be available for new versions of a respective platform and not for all the existing platforms out there not able or willing to update to a new revision.

Purpose and Scope

With the aforementioned side conditions it was quite clear how the purpose and the scope of the final recommendation should look. The option of providing/recommending a specification or generic API, or something like ResourceSync, which has to be implemented for each adopting platform, was discarded. Instead, the active WG members agreed to strive for a common exchange format which can be written/read by custom tools specific to platforms and in some cases even to platform versions. Some major platforms and initiatives have undertaken promising efforts on dealing with content exchange in a similar way (see Appendix 1), but with a very narrowed focus on one respective platform or data model. However, this WG's recommendations should align with these efforts.

This document aims for recommending a generic exchange format providing a self describing structure as well as an external, publicly accessible description that can be used for validation purposes. The adoption barrier should be as low as possible and existing efforts (see Appendix 1) should be taken into account as far as possible. The exchange format should strongly rely on well accepted standards and should support cross system exchange at least to a basic extent. However, it is not in the scope of this document to describe an exchange format that would allow content to be exchanged across systems in a lossless way. Rather, the goal is to combine a well adoptable exchange format together with a minimal set of standardized metadata, which can be expected to be present within the exchanged content and can be evaluated by the consumer. Furthermore, the focus on research data, which can be expected to be of huge volume in some cases, should be taken into account by the envisioned recommendation.

Use Case Description

The primary use case targeted in this document is the migration/replication of digital objects between a source and a destination research data repository platform where:

a) Source and destination support a fully compatible data model and ontologies, or
b) source and destination are using the same platform with a different data model and/or ontologies, e.g. due to version differences, or
c) source and destination are using different repository platforms with a different data model and ontologies.

The envisioned recommendation should enable all three scenarios, where scenario (a) should be realizable with a minimum level of information loss, whereas scenario (c) should be supported by utilizing a very common model for representing digital objects between a wide range of different repository platforms.

2. Definitions

BagPack: Stands for ‘BagIt-based Packaging Format’ and is used in the course of this document to name packages created according to this recommendations.

Digital Object: A single managed entity in an instance of a research data repository platform consisting of data and all associated metadata.

3. BagPack Format Description

- *BagPacks* **MUST** be valid bags according to the BagIt specification [8].
- In addition to the elements required by the BagIt specification, *BagPacks* **MUST** contain one or more metadata files located at

  metadata/<identifier>.<extension>

  relative to the bag root, where <identifier> is the identifier of a well known metadata standard, such as "datacite" and <extension> is the filename extension corresponding to the file format, such as "xml", "rdf", or "ttl". Sample names and extensions are listed in the following table:

Name	Identifier	Extension	Comment
Datacite Metadata Schema	datacite	xml	Mandatory for compliant *BagPacks*, recommended version 4 or newer
Dublin Core	dc	xml, rdf, ttl...	Extension must match the contained content format
OAI-ORE	oai-ore	xml, jsonld, ttl...	Extension must match the contained content format
(Tabular) Data Package	(tabular-)data-package	json	
Common European Research Information Format (CERIF)	cerif	xml	
Metadata Encoding & Transmission Standard	mets	xml	

- This naming scheme and location **MAY** be used for all kinds of accompanying metadata in the bag, e.g. provenance information in ProvONE or access information following the WebACL standard.
- Adopters of this recommendation **MAY** add arbitrary platform-specific metadata files in the metadata folder, even not following the aforementioned naming scheme, e.g. to ensure a lossless transfer of digital objects between identical source and destination platforms. However, a receiver of the *BagPack* is not obliged to interpret any of these additional metadata files and **MUST** not reject the *BagPack* if it contains unknown or differently named metadata files.
- All *BagPacks* **MUST** contain a metadata file datacite.xml with metadata following the DataCite Metadata Schema (version 4+).
- The content of datacite.xml **MUST** be valid according to the DataCite standard [9]. In particular, the mandatory properties **MUST** be provided and the recommended properties **SHOULD** be provided if available.
- The content of datacite.xml **MAY** be valid according to the DataCite schema. If this is not the case, e.g. due to a missing DOI, the bag **SHOULD** not be rejected.
- If the *BagPack* contains more than one digital object, e.g. a collection of them, it **MAY** contain additional metadata files

  metadata/<identifier>-<objectid>.<extension>.

  In this case, the files metadata/<identifier>.<extension> contain the metadata describing the *BagPack* as a whole, while the files metadata/<identifier>-<objectid>.<extension> contain metadata describing the individual object, <objectid> being an identifier for this object.

- Single object-related metadata **SHOULD** provide appropriate properties defining the relation between individual objects, according to the metadata standard being used, such as RelatedIdentifier in the case of DataCite.
- Metadata files located at metadata/ qualify as "Other Tag Files" according to the BagIt specification. They **SHOULD** be listed in a file tagmanifest-<alg>.txt contained in the *BagPack*. Otherwise, implementations **SHOULD** ignore them according to the BagIt specification.
- For each *BagPack* there **SHOULD** exist a machine readable BagIt profile [10] describing the expected structure and content of the *BagPack*. A sample profile can be found in section 4.1
- The according profile **MUST** be referenced by a property BagIt-Profile-Identifier inside bag-info.txt.
- Referenced profiles **SHOULD** be documented in a web-accessible location, e.g. at the RDA Research Data Repository Interoperability WG GitHub Repository [11].
- The profile **MAY** be additionally stored at /metadata/profile/profile.json within the *BagPack* for preservation purposes.
- Existing profiles **SHOULD** be considered for reuse instead of creating a new one with similar or identical content.
- Profiles **SHOULD** be versioned.

3.1. Sample Profile Description

{
"BagIt-Profile-Info":{
“BagIt-Profile-Identifier":"https://raw.githubusercontent.com/RDAResearchDataRepositoryInteropWG/bagit-profiles/master/generic/0.1/profile.json",
"Source-Organization":"rd-alliance.org",
"Contact-Name":"Research Data Repository Interoperability WG",
"Contact-Email":"rda-rdrinterop-wg@rda-groups.org",
"External-Description":"Base BagIt profile recommended by the RDA Research Data Repository Interoperability WG.",
"Version":"0.1"
},
	/*Basic profile information containing the publicly accessible location of the profile as well as version and contact information.
	*/

"Bag-Info":{
"Bagging-Date":{
"required":true
},
"Contact-Phone":{
"required":false
},
"Source-Organization": {
"required": false
},
"Contact-Name": {
"required": false
},
"Contact-Email":{
"required":true
},
"External-Identifier":{
"required":false
},
	/*Definition of mandatory fields that MUST be part of the bag’s bag-info.txt as well as of optional fields that SHOULD be part of bag-info.txt. Besides bagging data and contact information there are also information for identification (e.g. External-Identifier that may contain a persistent identifier) and information for

"External-Description":{
"required":true
},
"Bag-Size":{
"required":true
},
"Payload-Oxum":{
"required":true
},
"Source-Identifier":{
"required":false
}
},
	transfer validation support (e.g. Payload-Oxum)
	*/

"Manifests-Required":["sha256"],	/*Required manifests.*/
"Allow-Fetch.txt":true, "Serialization":”optional”, "Accept-Serialization":["application/zip", "application/tar", "application/tar+gzip"],	/*Allowed features, e.g. fetching missing files listed in fetch.txt, serialization of the bag (e.g. zip) and allowed mime types for serialization.*/
"Accept-BagIt-Version":["0.97"],	/*List of supported versions BagIt versions. */
"Tag-Manifests-Required":["sha256"],	/*Set tag manifest files to be required. */
"Tag-Files-Required":["metadata/datacite.xml"] }	/*Required tag file containing datacite metadata according to these recommendations. */

Listing 1: Base profile description

4082975711093941170f02e0d4ca777b4af78c8c39a19fb06ef920c831753590 metadata/datacite.xml	/*Recommended tag manifest file containing the SHA256 checksum and the path of manifest files. In the example, only the recommended metadata in datacite.xml qualifies as tag file. */

Listing 2: Compliant tagmanifest-sha256.txt

BagIt-Version: 0.97 Tag-File-Character-Encoding: UTF-8	/*bagit.txt containing bag version and file encoding information. */

Listing 3: Compliant bagit.txt

Bagging-Date: 2017-08-29 Contact-Phone: +49 721 608-24042 Source-Organization: Research Data Alliance Contact-Name: Thomas Jejkal Contact-Email: thomas.jejkal@kit.edu External-Identifier: 10.15497/RDA00020 External-Description: Packaging containing the Research Data Repository Interoperability WG primer document. Bag-Size: 135 KB Payload-Oxum: 131643.1 Source-Identifier: RDA File Repository BagIt-Profile-Identifier: https://raw.githubusercontent.com/RDAResearchDataRepositoryInteropWG/bagit-profi↲l es/master/generic/0.1/profile.json	/*bag-info.txt compliant to the profile above. */

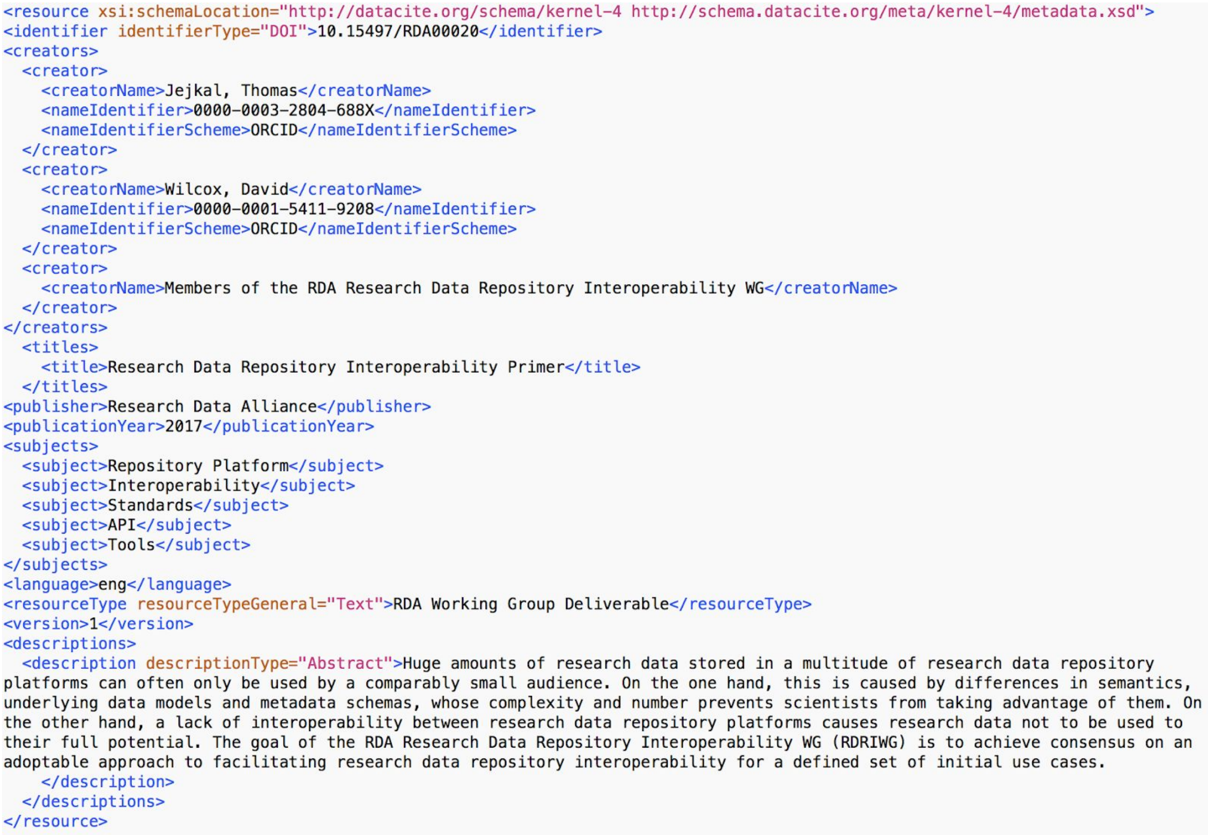
Listing 4: Compliant bag-info.txt

70cc12fb6207ff00b64d280450b7c82916d7326c39ab0385754ec25480ea63a8 data/primer_final.pdf	/*Compliant manifest file including the valid SHA256 checksum. */

Listing 5: Compliant manifest-sha256.txt

https://www.rd-alliance.org/system/files/ResearchDataRepositoryInteroperabilityPrimerfinal.pdf 131643 data/primer_final.pdf	/*Compliant fetch file including the download URL, octet count and target payload filename. */

Listing 6: Compliant fetch.txt



```xml
<resource xsi:schemaLocation="http://datacite.org/schema/kernel-4 http://schema.datacite.org/meta/kernel-4/metadata.xsd">
<identifier identifierType="DOI">10.15497/RDA00020</identifier>
<creators>
  <creator>
    <creatorName>Jejkal, Thomas</creatorName>
    <nameIdentifier>0000-0003-2804-688X</nameIdentifier>
    <nameIdentifierScheme>ORCID</nameIdentifierScheme>
  </creator>
  <creator>
    <creatorName>Wilcox, David</creatorName>
    <nameIdentifier>0000-0001-5411-9208</nameIdentifier>
    <nameIdentifierScheme>ORCID</nameIdentifierScheme>
  </creator>
  <creator>
    <creatorName>Members of the RDA Research Data Repository Interoperability WG</creatorName>
  </creator>
</creators>
  <titles>
    <title>Research Data Repository Interoperability Primer</title>
  </titles>
<publisher>Research Data Alliance</publisher>
<publicationYear>2017</publicationYear>
<subjects>
  <subject>Repository Platform</subject>
  <subject>Interoperability</subject>
  <subject>Standards</subject>
  <subject>API</subject>
  <subject>Tools</subject>
</subjects>
<language>eng</language>
<resourceType resourceTypeGeneral="Text">RDA Working Group Deliverable</resourceType>
<version>1</version>
<descriptions>
  <description descriptionType="Abstract">Huge amounts of research data stored in a multitude of research data repository
platforms can often only be used by a comparably small audience. On the one hand, this is caused by differences in semantics,
underlying data models and metadata schemas, whose complexity and number prevents scientists from taking advantage of them. On
the other hand, a lack of interoperability between research data repository platforms causes research data not to be used to
their full potential. The goal of the RDA Research Data Repository Interoperability WG (RDRIWG) is to achieve consensus on an
adoptable approach to facilitating research data repository interoperability for a defined set of initial use cases.
    </description>
  </descriptions>
</resource>
```

Listing 7: Simple datacite.xml example

```
Example_bag
|--metadata
|  |--datacite.xml
|--data                                    //empty, payload primer_final.pdf is fetched
|--bagit-info.txt
|--bagit.txt
|--fetch.txt
|--manifest-sha256.txt
|--tagmanifest-sha256.txt
```

Listing 8: Sample bag

4. Adoption Guidelines

4.1. Export

Before starting with the adoption of these recommendations for a new platform, this document should be carefully read to determine if the requirements given by these recommendations could be met. Furthermore, getting familiar with the BagIt Specification [8] as well as with the BagIt Profiles Specification [10] is highly recommended. Afterwards, the GitHub repository of the WG [11] is a good starting point, on the one hand to check for existing adoptions for the corresponding target platform, on the other hand to go through existing BagIt Profiles to find an existing profile that can be reused for the own adoption. For the majority of use cases, the generic profile available at [12] should be sufficient. If an adoption has special requirements for the profile one should formulate them according the the BagIt Profiles specification in a new profile and send a pull request to the GitHub repository in order to make the profile publicly available.

> Currently, profile inheritance is not supported by the BagIt Profiles specification. Therefore, if an adopter creates a new profile at least the requirements of the generic profile (and all other profiles that should be additionally supported) must be merged manually into the new profile.

While creating a new profile the adopter should mainly think about which metadata attributes should be mandatory in the bag and which tag files, e.g. additional metadata files stored in the 'metadata' folder relative to the bag root, besides 'metadata/datacite.xml', must be present.

> Please keep in mind that 'metadata/datacite.xml' MUST be listed in the 'Tag-Files-Required' section of every profile compliant to this recommendations. Furthermore, the metadata attribute 'BagIt-Profile-Identifier' is mandatory according to the BagIt Profiles Specification.

After finding or creating an appropriate profile the implementation may continue with the exporter. It is up to the adopter in which way the exporter is realized. It might be realized as a standalone command line tool as well as tightly coupled to the repository platform, e.g. by a web-accessible interface. There are existing implementations of BagIt available (see section 5) reducing the effort for creating bags compliant to the specification.

While writing the exporter one should consider including as much information as possible in the form of additional tag files (preferably) following the recommended naming conventions described in section 3, even if these tag files can only be evaluated by the same platform. The main goal should be to create *BagPacks* which can be used to realize a possibly lossless import into the same platform later on. For all other platforms, the importer may decide to use the one or the other tag file in addition to datacite.xml in order to extract metadata beyond the basic set of DataCite metadata. Adoptions may also decide to structure the payload in the 'data' folder according to the platform's needs. However, according to the BagIt Specification the content of 'data' should be treated as semantically opaque.

Now, to follow these recommendations, the mandatory tag file 'metadata/datacite.xml' should be created. Therefore, existing metadata elements from the platform metadata model should be used in order to fill at least the mandatory elements according to the DataCite standard [9].

According to the DataCite standard providing an identifier of type DOI is mandatory for successful validation against the DataCite schema. If a digital object exported into a *BagPack* has no valid DOI assigned, this document recommends using one of the machine operable codes listed by the DataCite standard, e.g. (:none) or (:tba).

Depending on the library used to create the bag mandatory metadata elements may now be added, as well as manifest entries for all payload and tag files. Finally, the bag may or may not be serialized if serialization is allowed by the utilized profile. Here, the adopter should keep in mind, that according to the BagIt specification a serialized bag should contain a folder having the same name as the serialized file containing the actual bag, e.g. a serialized *BagPack* named ‘my_package.zip has to contain a folder named ‘my_package’ containing the ‘data’ and ‘metadata’ subfolders as well as all bag-specific files.
If not serialized, the final *BagPack* should look as follows:

```
BagPack
|--metadata (must contain at least datacite.xml, may contain additional metadata tag files)
|  |--datacite.xml
|--data  (may contain payload or is empty if payload is entirely fetched)
|--bagit.txt (contains BagIt version and character encoding information)
|--bagit-info.txt (should contain all mandatory attributes required by the profile)
|--fetch.txt (Required if using fetching, must be enabled in the profile)
|--manifest-sha256.txt (or e.g. manifest-sha512.txt if sha512 digest is required by the profile)
|--tagmanifest-sha256.txt (or e.g. tagmanifest-sha512.txt if sha512 digest is required by the profile)
```

4.2. Import

Importing a *BagPack* should start with a conformance check of the received bag using the contained profile information. This allows a fail fast if mandatory elements are missing before e.g. fetching data or starting to create content within the target platform. In this phase, the importer may also check if it is familiar with the profile and if the *BagPack* contains platform specific content allowing a possibly lossless import.
Afterwards, according to the BagIt specification, all payload not present in ‘data’ folder should be fetched before checksums validation. If fetching and checksum validation succeeds, the actual ingest into the target platform can start. Otherwise, the *BagPack* should be rejected.

This recommendation does not encourage the schema validation of the mandatory tag file ‘metadata/datacite.xml’ against the official DataCite schema. At least, a failed validation should not result in rejecting the *BagPack*. This is because most *BagPacks* will contain unpublished research data in various states and will therefore not be able to have a DOI assigned, which is mandatory according to the DataCite schema.

For obtaining information for the actual ingest, the adopter should have checked in the first phase, whether the *BagPack* contains platform-specific or other familiar content. If this is the case, a possibly lossless ingest can be attempted including the semantical evaluation of the payload. If the only well-known source of information the *BagPack* provides is DataCite metadata, the importer should try to map as many DataCite elements as possible to elements of the platform model. At this point, experiences gained from the previously implemented mapping from the platform model into DataCite, as it was required for implementing the exporter, can be useful. Afterwards, the ingest of the payload will finalize the import.

5. Other Resources

Bagger Profiles:
https://github.com/LibraryOfCongress/bagger/[...]/gov/loc/repository/bagger/profiles
AVpreserve's *Exactly* tool:
https://www.avpreserve.com/avpsresources/tools/
BagIt Implementations:
https://confluence.ucop.edu/display/Curation/BagIt
BagIt Java Library:
https://github.com/LibraryOfCongress/bagit-java

References

[1] Case Statement of the Research Data Repository Interoperability WG, Jejkal, T. and Wilcox, D., https://www.rd-alliance.org/group/researc[..]erability-wg/case-statement/research-data-repository

[2] Primer Document, Members of the Research Data Repository Interoperability WG, http://dx.doi.org/10.15497/RDA00020

[3] Repository Platform Capability Matrix, Members of the Research Data Repository Interoperability WG, https://docs.google.com/spreadsheets/d/1n876pzOU[...]hsaCssI/edit?usp=sharing

[4] ResourceSync Framework Specification (ANSI/NISO Z39.99-2017),
http://www.openarchives.org/rs/1.1/resourcesync

[5] SWORDv3,
http://swordapp.org/swordv3/

[6] Research Data Crate,
https://github.com/UTS-eResearch/datacrate

[7] sitemaps.org,
http://www.sitemaps.org/protocol.html

[8] The BagIt File Packaging Format (V0.97),
https://tools.ietf.org/html/draft-kunze-bagit-14

[9] DataCite Metadata Schema,
https://schema.datacite.org/

[10] BagIt Profiles,
https://github.com/ruebot/bagit-profiles

[11] RDA Research Data Repository Interoperability WG GitHub Repository:
https://github.com/RDAResearchDataRepositoryInteropWG

[12] Generic BagIt Profile,
https://github.com/RDAResearchDataRepositoryInteropWG/bagit-profiles/blob/master/generic/0.1/profile.json

Appendix

Sample Bagit-based Packages

Fedora

```
Example_bag
|-- bag-info.txt
|-- bagit.txt
|-- data
|   |-- fcrepo
|   |   |-- rest
|   |   |   |-- basic
|   |   |   |   |-- image
|   |   |   |   |   |-- fcr%3Ametadata.ttl
|   |   |   |   |-- image.binary
|   |   |   |-- basic.ttl
|   |   |-- rest.ttl
|-- manifest-sha1.txt
|-- tagmanifest.sha1.txt
```

DataONE

Source: https://releases.dataone.org/online/api-documentation-v2.0/design/DataPackage.html

```
Example_bag
|--bagit.txt
|--data
|  |---data-file-1.csv
|--oai-ore.txt
|--pid-mapping.txt                //doi://10.xxxx/AA/TG43 data/data-file-1.csv
|--manifest-sha1.txt
```

Data Conservancy

Source: https://wiki.library.jhu.edu/display/DCSDOCPKG/4.+DC+Packaging+Specification

```
Example_bag
|--META-INF
|  |--org.dataconservancy.packaging
|     |--state
|        |--pkgState.bin
|--bagit.txt
|--data
```

```
|  |--bin
|     |--Documents
|        |--data-file-1.csv
|  |--obj
|       |--Documents.ttl
|       |--data-file-1.csv.ttl
|--oai-ore.txt
|--pid-mapping.txt                    //doi://10.xxxx/AA/TG43 data/data-file-1.csv
|--manifest-sha1.txt
```

Research Object BagIt Archive

Source: https://github.com/ResearchObject/bagit-ro

```
Example_bag
|--data
|  |--README.md
|  |--analyse.py
|  |--numbers.csv
|  |--results.txt
|--metadata
|  |--annotations
|     |--numbers.jsonld
|  |--provenance
|     |--results.prov.jsonld
|  |--manifest.json
|--bagit-info.txt
|--bagit.txt
|--fetch.txt
|--manifest-sha1.txt
|--tagmanifest-sha1.txt
```

DARIAH-DE Repository

Source: https://repository.de.dariah.eu/1.0/dhcrud/21.11113/0000-000B-CAC4-4/bag
(https://dx.doi.org/10.20375/0000-000B-CAC4-4)

```
Example_bag
|--data
|  |--01_metadata.ttl
|  |--02_adm.ttl
|  |--03_tech.xml
|  |--data.txt
|--bagit-info.txt
|--bagit.txt
|--manifest-md5.txt
|--tagmanifest-md5.txt
```