Mailbag Specification

1. Introduction

1.1 Purpose

The purpose of the Mailbag Specification is to define a set of file layout conventions to allow for the preservation of email in multiple file formats in a consistent and structured package that may be created and acted upon by computer software. The Mailbag Specification is an extension of RFC 8493, The BagIt File Packaging Format, hereafter referred to as Bagit. All mailbags MUST conform to RFC 8493 as valid "bags."

No single email format satisfies all needs for long-term preservation. To ensure a wide variety of future use cases, email should be preserved in multiple formats. A "mailbag" is suitable for storing and transferring these multiple formats in a manner that they will remain connected, consistent, and actionable.

The extension of the Bagit specification allows for the use of existing Bagit implementations and workflows while also utilizing Bagit's strong integrity assurances through its use of checksum algorithms. Similar to Bagit, the Mailbag Specification is designed to provide just enough structure to accomplish its purpose while also enabling further extension to meet local practices.

1.2 Status of this Document

Mailbag Working Meeting Pre-Release

1.3 License

1.4 Requirements

This document is designed as an extension of The BagIt File Packaging Format v1.0 [RFC8493] and all mailbags must comply with that specification.

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

The key words “MAY”, “MUST”, “MUST NOT” ,“RECOMMENDED”, “OPTIONAL”, “SHOULD”, and “SHOULD NOT” in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.5 Terminology

This document relies on the terminology defined in the Bagit specification section 1.3.

The following additional terms have precise definitions as used in this document:

derivative: An additional email format created prior to, or during, the creation of a mailbag. For example, a tool may create multiple derivative formats for email, such as EML, PDF, or WARC during the creation of a mailbag.

email folder: A grouping of email created by an email account user to arrange and manage email. Inbox, Sent Mail, Drafts, and Trash are common default email folders, and email users also commonly create custom folders. Email folders can contain other email folders.

email serialization format: One of a number of formats for storing email on a computer file system. Common email serialization formats are MBOX, EML, PST, or MSG, and email may also be serialized using PDF, WARC, or other file formats.

format subdirectories: a set of optional directories in the payload of a mailbag. Format subdirectories are named using the lower case file extension of the email serialization format they contain.

Message-ID: An identifier for email messages generated by the client program sending the email. Message-ID is included in email headers and typically available in email formats such as MBOX, PST, EML, and MSG.

Message-Identifier: An identifier assigned to each email message during the creation of a mailbag. Message-Identifiers are unique within a mailbag and must be valid filenames

in both Unix-based and Windows operating systems. Thus they cannot contain characters such as: <, >, :, ", /, \, |, ?, and *.

source: The original capture method or serialization format of email prior to the creation of a mailbag. This could be the IMAP protocol, or serialization formats such as MBOX, EML, PST, etc. It is important to note that the source may not be the email account's original form, but only the format that existed prior to packaging as a mailbag.

2. Overview of a Mailbag

A mailbag is a set of directories and files with a common structure defined by this document and the Bagit specification. A mailbag is used to manage email serialized from an email account in multiple formats. Multiple email accounts are expected to require multiple mailbags.

A mailbag MUST be a valid Bagit bag with these additional requirements:

1. The payload directory of a mailbag MUST contain at least one of a set of format subdirectories named after each of the serialization formats it contains in lower case.
2. A mailbag MUST contain a bag-info.txt file containing a number of required and optional fields.
3. A mailbag MUST contain an additional tag file named "mailbag.csv". This file must be a valid Comma-separated values (CSV) file and list each message with an identifier, columns to connect messages to their different included formats, and basic email header information.

A mailbag MAY also contain a few optional elements:

1. The payload directory of a mailbag MAY contain a subdirectory labeled "attachments" which contains files attached to email messages.
2. A mailbag may contain two additional tag files, named "folders_not_retained.txt" and "messages_not_retained.txt", used to document email folders and messages not retained prior to packaging into a mailbag.
3. A Bagit tag manifest file is also RECOMMENDED.

2.1 Example of a mailbag

```
<base directory>/
        |
        +-- bagit.txt
        |
        +-- bag-info.txt
```

```
|
+-- mailbag.csv
|
+-- manifest-<algorithm>.txt
|
+-- tagmanifest-<algorithm>.txt
|
+-- data/
    |
    +-- mbox/
    |    +-- [payload files]
    +-- pdf/
    |    +-- [payload files]
    +-- warc/
    |    +-- [payload files]
    +-- attachments/
         +-- [Message-Identifier]/
         |    +-- [payload files]
         +-- [Message-Identifier]/
         |    +-- [payload files]
         …
```

3. Identifiers

Mailbags use identifiers to maintain connections between individual email messages in different formats and filesystem locations. The commonly-used identifier for email messages is Message-ID as described in RFC2392. However, there are cases when the Message-ID header is empty in email serialization formats and Message-IDs also commonly contain characters that are incompatible with filesystems. Thus, mailbags contain an additional Message-Identifier for each message that are unique within a mailbag. Message-Identifiers are REQUIRED for each message and must be valid as filenames in both Unix-based and Windows operating systems. Thus they cannot contain characters such as:

```
< > : " / \ | ? *
```

Sequential numbers MAY be used as Message-Identifiers. Implementations can map Message-Identifiers to Message-IDs using the mailbag.csv tag file as defined in Section 5.3.

4. Payload Structure

The payload directory of a mailbag MUST contain subdirectories named after each of the serialization formats it contains in lower case. At least one payload subdirectory is required.

4.1 Format subdirectories

A mailbag payload MUST contain at least one of the following set of optional format subdirectories.

- mbox
- pst
- msg
- eml
- pdf
- warc

If present, each of these format subdirectories MUST contain email in the format designated by the name of the subdirectory. Format subdirectories MUST be in lower case and may contain either email in the source format or derivative formats. At least one payload subdirectory will be designated as the source format in a REQUIRED Mailbag-Source field in the bag-info.txt tag file.

4.1.1 Additional subdirectories within format subdirectories

Format subdirectories MAY contain additional subdirectories that represent the folder structure of the source email account. It is RECOMMENDED to use additional subdirectories when an email account uses folders meaningfully to preserve the original arrangement of the email account. However, a mailbag may not contain every email folder originally present in the email account. In this case, it is RECOMMENDED to document which folders were not retained in a tag file named “folders_not_retained.txt”, as defined in Section 5.4.2.

```
<base directory>/
|
+-- bagit.txt
|
+-- bag-info.txt
|
+-- mailbag.csv
|
+-- manifest-<algorithm>.txt
|
+-- tagmanifest-<algorithm>.txt
|
+-- data/
    |
    +-- mbox/
     |    +-- [payload files]
```

```
+-- eml/
|     +-- [payload files]
+-- pdf/
|     +-- [payload files]
+-- warc/
      +-- [payload files]
```

4.1.3 Format subdirectories example with additional subfolders

This example demonstrates multiple different uses for additional subfolders. Here, the eml format subdirectory uses email folders as additional subdirectories, the pdf Format Subdirecory uses senders as additional subdirectories, and the warc format subdirectory uses recipients as additional subdirectories.

```
<base directory>/
|
+-- bagit.txt
|
+-- bag-info.txt
|
+-- mailbag.csv
|
+-- manifest-<algorithm>.txt
|
+-- tagmanifest-<algorithm>.txt
|
+-- data/
    |
    +-- mbox/
    |     +-- All mail Including Spam and Trash.mbox
    +-- eml/
    |     +-- Inbox
    |     |     +-- [payload files]
    |     +-- Sent Mail
    |     |     +-- [payload files]
    |     +-- Junk
    |           +-- [payload files]
    +-- pdf/
    |     +-- Inbox
    |     |     +-- [payload files]
    |     +-- Sent Mail
    |     |     +-- [payload files]
    |     +-- Junk
    |           +-- [payload files]
```

```
    +-- warc/
        +-- Inbox
        |    +-- [payload files]
        +-- Sent Mail
        |    +-- [payload files]
        +-- Junk
             +-- [payload files]
```

4.2 Attachment Subdirectory

A mailbag payload directory MAY contain a subdirectory labeled “attachments”. The Attachment Subdirectory OPTIONALLY contains attachment files extracted from email messages. It is RECOMMENDED to extract attachments and place them in the attachment subdirectory when creating derivative formats that do not contain embedded attachments, such as PDF and/or WARC derivatives.

If present, the Attachment Subdirectory MUST contain additional subdirectories named with the Message-Identifier of the email where the contained attachments were extracted from. If present, these Message-Identifier subdirectories SHOULD contain files attached to the corresponding email message.

4.2.1 Attachment Subdirectory Example

```
<base directory>/
    |
    +-- bagit.txt
    |
    +-- bag-info.txt
    |
    +-- mailbag.csv
    |
    +-- manifest-<algorithm>.txt
    |
    +-- tagmanifest-<algorithm>.txt
    |
    +-- data/
        |
        +-- mbox/
        |    +-- [payload files]
        +-- eml/
        |    +-- [payload files]
        +-- pdf/
```

```
|    +-- [payload files]
+-- warc/
|    +-- [payload files]
+-- attachments/
     +-- [Message-Identifier]/
     |    +-- [payload files]
     +-- [Message-Identifier]/
     |    +-- [payload files]
     ...
```

4.3. Payload File Names

4.3.1 Message File Names

It is RECOMMENDED to maintain the original filenames for email messages in source formats. If the email source uses single files for individual emails, such as EML or MSG formats, derivative formats should also use the original filenames with a new file extension to reflect the change in format. File extensions for WARC files MAY denote if compression is used. For cases when the source format does not use single files for individual emails, such as IMAP, MBOX, and PST, derivative formats with single files for individual emails, such as EML, PDF, or WARC, MUST use the emails Message-Identifier as filenames.

4.3.2 Attachment File Names

It is RECOMMENDED to maintain the original filenames for attachments.

4.3.3 File and Directory Name Interoperability issues

Mailbag implementations SHOULD follow the interoperability recommendations in Bagit specification section 6.1.1.3.

Attachment filenames provide the potential for interoperability issues, as creating a Mailbag in a Windows environment may attempt to handle filenames that were originally created in a POSIX environment.

4.3.4 Filename Example with MBOX as source

<base directory>/

```
|
+-- bagit.txt
|
+-- bag-info.txt
|
+-- mailbag.csv
|
+-- manifest-<algorithm>.txt
|
+-- tagmanifest-<algorithm>.txt
|
+-- data/
    |
    +-- mbox/
    |    -- All mail Including Spam and Trash.mbox
    +-- pdf/
    |    -- 8sPf2WLSpyp65KLcYNgpX5.pdf
    |    -- LVGxWjUABLVB5dep5hZTiZ.pdf
    |    -- 9SXvpbe2AtgvedwZJCfWF4.pdf
    |       ...
    +-- warc/
    |    -- 8sPf2WLSpyp65KLcYNgpX5.warc.gz
    |    -- LVGxWjUABLVB5dep5hZTiZ.warc.gz
    |    -- 9SXvpbe2AtgvedwZJCfWF4.warc.gz
    |       ...
    +-- attachments/
          +-- 8sPf2WLSpyp65KLcYNgpX5/
          |    -- Image1.jpg
          |    -- Image2.jpg
          +-- 9SXvpbe2AtgvedwZJCfWF4/
               -- packageList.pdf
```

4.3.5 Filename Example with EML as source

```
<base directory>/
|
+-- bagit.txt
|
+-- bag-info.txt
|
+-- mailbag.csv
|
+-- manifest-<algorithm>.txt
```

```
|
+-- tagmanifest-<algorithm>.txt
|
+-- data/
    |
    +-- eml/
     |    -- A free and open internet.eml
     |    -- A simple pledge.eml
     |    -- A year of organizing.eml
     |       ...
    +-- pdf/
     |    -- A free and open internet.pdf
     |    -- A simple pledge.pdf
     |    -- A year of organizing.pdf
     |       ...
    +-- warc/
     |    -- A free and open internet.warc
     |    -- A simple pledge.warc
     |    -- A year of organizing.warc
     |       ...
    +-- attachments/
          +-- 8sPf2WLSpyp65KLcYNgpX5/
               +-- Image1.jpg
               +-- Image2.jpg
          +-- 9SXvpbe2AtgvedwZJCfWF4/
               +-- packageList.pdf
```

5. Tag Files

A mailbag MUST contain the required Bagit tag files as specified in Bagit 2.1, including a bagit.txt bag declaration and a payload manifest. In addition to the Bagit minimum requirements, a mailbag MUST contain a bag-info.txt metadata file and an additional mailbag.csv comma separated values (CSV) file containing identifiers and a set of commonly used email headers for each message.

It is also RECOMMENDED for a mailbag to contain a tag manifest file as specified in Bagit 2.2.1. A mailbag MAY also contain two additional tag files, named “folders_not_retained.txt” and “messages_not_retained.txt” which, if present, MUST document email folders and messages that were not retained prior to, or during, packaging, and thus are not present in the payload directory. A mailbag MAY also contain additional tag files that are not defined by this document as specified in Bagit 2.2.4.

5.1 Example of tag files in a mailbag

```
<base directory>/
        |
        +-- bagit.txt
        |
        +-- bag-info.txt
        |
        +-- mailbag.csv
        |
        +-- manifest-<algorithm>.txt
        |
        +-- tagmanifest-<algorithm>.txt
        |
        +-- folders_not_retained.txt
        |
        +-- messages_not_retained.txt
        |
        +-- data/
              +-- [payload files]
```

5.2 Bag-info.txt

As valid bags, mailbags retain the rules and reserved elements as specified in Bagit 2.2.2. A mailbag MUST contain the set of required Reserved Metadata Fields listed below, including the “Bagging-Date” and “External-Identifier'' fields that are reserved in Bagit 2.2.2. A mailbag MAY also contain any fields in the following set of Optional Metadata Fields. Possible values are quoted and listed in brackets. These values are case sensitive and should be included exactly as they appear. Mailbags do not expect the order of bag-into.txt fields to be significant. All of both the Required Metadata Fields and the Optional Metadata Fields MUST NOT be repeated.

5.2.1 PREMIS Mapping

A mailbag is not expected to contain a full PREMIS implementation, but does attempt to document information that can be mapped to PREMIS fields. A mailbag expects to be mapped to two PREMIS events, one for the initial capture of email and another for its packaging into a mailbag.

5.2.2 Required Metadata Fields

Bag-Type: “Mailbag”
Denotes that a bag is a mailbag.

Mailbag-Source: [“imap”, “mbox”, “eml”, “pst”, “pdf”, “warc”]
Denotes the protocol or serialization format which

Original-Included: [“True”, “False”]
Denotes whether the original source format, such as an MBOX or set of EML files are included in original form. If an included MBOX file does not contain the same content as the original, such as if messages or email folders were excluded, the value MUST be “false”. Similarly if the email was packaged using the IMAP protocol, the value MUST be “false”. Important to note that this does not refer to the email prior to capture, but the state of the email prior to packaging.

Package-Date:
Timestamp denoting when a mailbag was packaged. MUST conform to RFC3339, such as 1937-01-01T12:00:27.87+00:20. Mapped to PREMIS package event, eventDateTime field.

Bagging-Date:
Date (YYYY-MM-DD) denoting when a mailbag was packaged. Less-precise duplicate of Package-Date for compatibility with Bagit implementations.

External-Identifier:
An identifier for the mailbag as a whole supplied by the agent creating the mailbag.

Mailbag-Agent:
Description of the agent packaging email into a mailbag. Mapped to PREMIS package event, eventDetail@program.

Mailbag-Agent-Version:
Version of the agent packaging email into a mailbag. Mapped to PREMIS package event, eventDetail@version.

5.2.3 Optional Metadata Fields

Mailbag-Source-Filename:
The original filename of prior to packaging into a mailbag. This field is useful if the source was an account-level serialization, such as an MBOX or PST file. This field SHOULD NOT be used if the source was IMAP or a message-level serialization, such as EML or MSG.

Capture-Date:

Timestamp denoting when the email included in a mailbag was originally captured. May be similar to Package-Date if a mailbag was packaged using the IMAP protocol. MUST conform to RFC3339, such as 1937-01-01T12:00:27.87+00:20. Mapped to PREMIS capture event, eventDateTime field.

Capture-Agent:
A string field describing the agent used to capture the email included in a mailbag. Mapped to PREMIS capture event, eventDetail@program.

Capture-Agent-Version:
A string field describing the version of the agent used to capture the email included in a mailbag. Mapped to PREMIS capture event, eventDetail@version.

IMAP-User:
The username used to connect using the IMAP protocol. IMAP-User SHOULD be used if a mailbag was packaged using IMAP.

IMAP-Host:
The host used to connect using the IMAP protocol. IMAP-Host SHOULD be used if a mailbag was packaged using IMAP.

MBOX-Format-Details:
A string field describing the nature of MBOX file(s) included in a mailbag.

MBOX-Agent:
A string field describing the agent used to create, read, or manipulate MBOX file(s) during the packaging of a mailbag.

MBOX-Agent-Version:
A string field describing the version of the software agent used to create, read, or manipulate MBOX file(s) during the packaging of a mailbag.

EML-Format-Details:
A string field describing the nature of EML file(s) included in a mailbag.

EML-Agent:
A string field describing the agent used to create, read, or manipulate EML file(s) during the packaging of a mailbag.

EML-Agent-Version:
A string field describing the version of the software agent used to create, read, or manipulate EML file(s) during the packaging of a mailbag.

PDF-Format-Details:

A string field describing the nature of PDF file(s) included in a mailbag.

PDF-Agent:
A string field describing the agent used to create, read, or manipulate PDF file(s) during the packaging of a mailbag.

PDF-Agent-Version:
A string field describing the version of the software agent used to create, read, or manipulate PDF file(s) during the packaging of a mailbag.

WARC-Format-Details:
A string field describing the nature of WARC file(s) included in a mailbag.

WARC-Agent:
A string field describing the agent used to create, read, or manipulate WARC file(s) during the packaging of a mailbag.

WARC-Agent-Version:
A string field describing the version of the software agent used to create, read, or manipulate WARC file(s) during the packaging of a mailbag.

5.2.4 Example bag-info.txt files

Example 1 (MBOX source and PDF & WARC derivatives):

Bag-Type: Mailbag
Mailbag-Source: MBOX
Mailbag-Source-Filename: All mail Including Spam and Trash.mbox
Original-Included: True
Package-Date: 2021-05-05T10:33:54+00:00
Bagging-Date: 2021-05-05
External-Identifier: 6700b2a1-f7fb-48d1-92b3-4ba1f5fc88d0
Mailbag-Agent: mailbag
Mailbag-Agent-Version: 0.0.1
Capture-Date: 2016-11-16T09:58:31+00:00
Capture-Agent: Gmail Export
MBOX-Format-Details: MBOXO
MBOX-Software-Agent: mailbox
MBOX-Software-Version: 0.4
PDF-Format-Details: PDF/A
PDF-Software-Agent: wkhtmltopdf
PDF-Software-Version: 0.12.6
WARC-Format-Details: WARC 1.1
WARC-Software-Agent: wget
WARC-Software-Version: 1.21.1

Example 2 (IMAP source and MBOX & EML & PDF & WARC derivatives):

Bag-Type: Mailbag
Mailbag-Source: IMAP
Original-Included: False
Package-Date: 2021-05-04T18:16:58+00:00
Bagging-Date: 2021-05-04
External-Identifier: 944efc3e-d9df-40ad-8b87-fbb120241ddb
Mailbag-Agent: mailbag
Mailbag-Agent-Version: 0.0.1
Capture-Date: 2021-05-04T18:16:23+00:00
Capture-Agent: imaplib
MBOX-Format-Details: MBOXO
MBOX-Software-Agent: mailbox
MBOX-Software-Version: 0.4
EML-Format-Details: EML
EML-Software-Agent: email
EML-Software-Version: 4.0.2
PDF-Format-Details: PDF
PDF-Software-Agent: pyPdf
PDF-Software-Version: 1.13
WARC-Format-Details: WARC 1.1
WARC-Software-Agent: warcio
WARC-Software-Version: 1.7.4

Example 3 (PDF source and no derivatives):

Bag-Type: Mailbag
Mailbag-Source: PDF
Original-Included: True
Package-Date: 2021-04-14T13:33:42+00:00
Bagging-Date: 2021-04-14
External-Identifier: 64bd6983-b0d0-4cd5-9521-d04ee8da3dae
Mailbag-Agent: mailbag
Mailbag-Agent-Version: 0.0.1
PDF-Format-Details: PDF
PDF-Software-Agent: wkhtmltopdf
PDF-Software-Version: 0.12.6

Example 4 (PST source and MBOX & PDF & WARC derivatives):

Bag-Type: Mailbag
Mailbag-Source: PST
Mailbag-Source-Filename: email_export.pst
Original-Included: True
Package-Date: 2020-03-08T22:31:12+00:00

Bagging-Date: 2021-03-08
External-Identifier: 08c91329-12a2-4360-b60a-0075aba3f6fd
Mailbag-Agent: mailbag
Mailbag-Agent-Version: 0.0.1
PST-Format-Details: 7.2
PST-Software-Agent: libpff
PST-Software-Version: 20161119
MBOX-Format-Details: MBOXO
MBOX-Software-Agent: mailbox
MBOX-Software-Version: 0.4
PDF-Format-Details: PDF
PDF-Software-Agent: pyPdf
PDF-Software-Version: 1.13
WARC-Format-Details: WARC 1.1
WARC-Software-Agent: warcio
WARC-Software-Version: 1.7.4

5.3 mailbag.csv

A Mailbag MUST contain an additional tag file named “mailbag.csv.” This file MUST be a CSV file that complies with RFC4180 and lists each message included in a mailbag. The mailbag.csv file MUST contain column headers from the set below, MUST use a comma (,) as a delimiter, MUST use double quotes (“) as quote characters for all fields, and MUST NOT use an additional escape character. The mailbag.csv file provides the necessary structure for mailbag implementations to connect multiple serialization formats for each message, along with their Message-Identifier, relative location within an account’s email folders, and commonly used email headers.

5.3.1 mailbag.csv column headers

A mailbag.csv MUST contain the following column headers in the order that they are listed:

- Message-Identifier
- Message-ID
- Email-Folder
- Original-Filename
- Date
- From
- To
- Cc
- Bcc

- Subject
- Content-Type

5.3.2 mailbag.csv columns

Message-Identifier:

An identifier created for each email message as it is packaged into a mailbag as described by Section 3.

Message-ID:

The original unique identifier for an email message created and/or managed by the email account prior to its packaging into a mailbag. This identifier is commonly contained within an email's Message-ID header and is described in RFC2392.

Email-Folder:

The relative filepath of a message within an email account's folder structure. The format of the filepath MUST comply with the rules for Bagit manifests as defined in Bagit 2.1.3.

Original-Filename:

The original filename for the email message if it was serialized to an individual file prior to packaging within a mailbag. This is useful if the email source was an EML or PDF file, and the original filename was retained as described by Section 4.3.1. If the message does not have an original filename, such as if the email source was MBOX file, the value of the Original-Filename column MUST be an empty string.

Date:

The content of an email message's Date header. The value MUST be an empty string if either the header has no value or is not present.

From:

The content of an email message's From header. The value MUST be an empty string if either the header has no value or is not present.

To:

The content of an email message's To header. The value MUST be an empty string if either the header has no value or is not present.

Cc:

The content of an email message's Cc header. The value MUST be an empty string if either the header has no value or is not present.

Bcc:

The content of an email message's Bcc header. The value MUST be an empty string if either the header has no value or is not present.

Subject:
The content of an email message's Subject header. The value MUST be an empty string if either the header has no value or is not present.

Content-Type:
The content of an email message's Content-Type header. The value MUST be an empty string if either the header has no value or is not present.

5.4 folders_not_retained.txt and messages_not_retained.txt

A mailbag MAY contain two additional tag files used to denote that part of an email account was not retained prior to, or during, packaging. Each line of a messages_not_retained.txt file must contain the Message-ID for a message not present in a mailbag. Each line of a folders_not_retained.txt file must contain an email folder not contained in a mailbag. When email folders not retained were included within another email folder, these folders MUST be expressed as a filepath relative to the account level. The format of the filepath MUST comply with the rules for Bagit manifests as defined in Bagit 2.1.3.

5.4.1 Example messages_not_retained.txt

a6e213b376f34b4f839059e4cc19f2f8@jerrynadler.com
1709911313.283722191466906290194.JavaMail.app@rbg23.atlis1
c2617427190b4d0a9d45e8f741d92041@carolynmaloney.com
20160623144948.9714F608DE@233elwb03.blackmesh.com
9656357.20160623150335.576bfa478e89a7.63874426@mail1.mcsignup.com
...

5.4.2 Example folders_not_retained.txt

Drafts
Listservs/A&A
Listservs/ERS
Trash
...

6. References

[RFC2119]
Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", IETF, March 1997.
DOI 10.17487/RFC2119
URL: https://tools.ietf.org/html/rfc2119

[RFC2392]
Levinson, E., “Content-ID and Message-ID Uniform Resource Locators,” IETF, August 1998.
DOI 10.17487/RFC2392
URL: https://tools.ietf.org/html/rfc2392

[RFC3339]
Klyne, G, Newman, C., “Date and Time on the Internet: Timestamps,” IETF, July 2002.
DOI 10.17487/RFC3339
URL: https://tools.ietf.org/html/rfc3339

[RFC8174]
Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", IETF, May 2017.
DOI 10.17487/RFC8174
URL: https://tools.ietf.org/html/rfc8174

[RFC8493]
Kunze, J., Littman, J., Madden, E., Scancella, J., Adams, C., “The BagIt File Packaging Format (V1.0),” IETF. October 2018.
DOI 10.17487/RFC8493
URL: https://tools.ietf.org/html/rfc8493