WARC implementation guidelines

Contribution from WARC usage task force

Status: Version 1.0

Author: Clément Oury

Date of issue: 27/01/2009

Number of pages: 24

Table of Contents

1 Objectives and organization

1.1 Objectives and scope

Since May 15, 2009, memory institutions and other digital archiving organizations can use a standardized way to store and preserve documents harvested from the web: the WARC file format, officially released as ISO 28500:2009.

WARC is an extension of the ARC format, which has been used since 1996 by the Internet Archive and then by most members of the International Internet Preservation Consortium (IIPC). These institutions recognized the need to extend the ARC format to add new possibilities, notably the recording of HTTP request headers, the recording of arbitrary metadata, the allocation of an identifier for every contained file, the management of duplicates and of migrated records, and the segmentation of the records. WARC files are intended to store every type of web content, whether retrieved by HTTP or another protocol.

International standardization is a critical step towards the wide adoption of the WARC format. It ensures its public availability, its stability – a standardized format is not implementation specific – and at least its maintenance and evolution. The end of this ISO process is a great beginning, but there is still work to do.

Indeed, WARC standard only gives generic rules: it specifies how to write valid WARC files so that they will be recognized and used by different tools. Very few recommendations have been given on the way to write WARC files in specific cases and circumstances.

Basically, the question is now: "We have got a format, how are we going to use it?". Beyond the standard's generic rules, there is a need for guidelines or recommendations on what to do when multiple design choices are offered by the standard. The objective of this document is therefore to gather advice and best practice to help institutions designing and creating WARC files for collection management, access, preservation, and interoperability with collections from different institutions.

This document is not an explanation of the WARC standard, an amendment or an evolution of the WARC standard, nor a technical documentation of tools that produce WARC files.

The intended audience is engineers who develop tools that produce WARC files, and practitioners who use the standard in real-life situations.

This document is called "WARC implementation guidelines".

1.2 Document management and evolution

This document is based on WARC experts brainstorming and on first developer's experiences towards WARC. It has been written by a task force gathering people actively involved in the writing of the standard, and in the development of the tools – all of them belonging to IIPC member institutions.

Members of the group were:

- Sara Aubry, Gildas Illien, Clément Oury, Bibliothèque nationale de France
- Gina Jones, Library of Congress
- John Kunze, California Digital Library
- Tue Larsen, Netarchive.dk
- Julien Masanès, European Archive
- Mark Middleton, Mark Williamson, Hanzo Archives
- Gordon Mohr, Brad Tofel, Steve, Internet Archive
- Gordon Paynter, National Library of New Zealand

This document necessary reflects a **work in progress**. In fact, some recommendations come from experience of developers and users of different crawling engines (for example recommendations on filenames), but most of them are thoughts and anticipations on future issues.

There is indeed still little user's experience towards the new capabilities offered by WARC. These implementation guidelines will need to be maintained and updated through times. Some parts of the document may be amended according to new user experiences. Additional problems or questions – not imagined yet – will probably be addressed and solved in the future.

Version	Date	Nature of update
0.1	29/07/2009	First draft released following an expert meeting held in Paris on June 29, 2009.
0.2	21/09/2009	Second draft released, taking into account written comments and a conference call held on September 15, 2009.
0.2.1	25/09/2009	Additions made to the section dedicated to record identification

1.0	27/01/2010	1st validated version. It takes into account comments from Gordon Paynter and from IIPC Preservation Working Group members. The document form and expression was also amended.

1.3 Organization

The document structure reflects the main circumstances in which WARC files may be created. Therefore, it does not follow the actual presentation order of the standard, but makes a functional breakdown instead.

A first section contains "general recommendations" that apply in all circumstances.

The following sections consider all functional cases when WARC files may be generated. These are grouped according to three main functions:

- Web harvesting;
- Data packaging;
- Operations on WARC files.

For each functional case, recommendations are provided on the way WARC records as well as WARC fields within WARC records should be written. Examples of records may be provided.

1.4 Related documents

1.4.1 WARC Standard

This document does not replace the WARC standard; it should be read as a companion document. To avoid redundancy references to the standard are made throughout these guidelines.

You may purchase the official published version on ISO website: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44717.

A similar version is available on the netpreserve forum: http://www.netpreserve.org/forum/viewtopic.php?f=70&t=386. For copyright reasons, access to this version is restricted to IIPC members, as IIPC experts largely contributed to the writing of the standard. This version must not be distributed outside of the IIPC.

On the other hand, a draft version is freely available on BnF website: http://bibnum.bnf.fr/WARC/index.html. It corresponds to the version released by IIPC experts before the ISO editorial board added modifications. From a technical point of view, this draft is similar to the published standard, but the forms of the two documents are different.

1.4.2 Other documents

A general introduction to the WARC format and to those guidelines has been done during the IIPC conference on "Active Solutions for Preserving Internet Content", held in San Francisco on October 7: http://www.netpreserve.org/events/program.php.

Finally, references are made to the latest draft of the IIPC metadata document (v3). It is available on the Preservation Working Group wiki: http://www.netpreserve.org/wiki/images/5/56/Iipc_metadatav3a_447.xls. Note that this document is only a work in progress and may be updated (access restricted to IIPC members only).

2 General recommendations

2.1 Scope

This section is dedicated to recommendations that apply in all cases where WARC files are produced.

2.2 File naming

WARC file names should represent a first set of information on the content and the origin of the files. For this reason they should strictly follow carefully defined conventions.

On the other hand, institutions may be compelled in the future, for many possible reasons, to change the name of their files. Therefore they should not put information in the file names that will not be recorded elsewhere.

Note that the filename is stored in WARC-Filename field of the Warcinfo record. The content of this WARC-Filename field should not change even when the name of the WARC file is changed as it records the original name of the file.

Recommendation n°1: Institutions should create and maintain their own naming conventions.
Recommendation n°2: Information available in the file names should also be recorded elsewhere to allow institutions to be free to change the names of their files if needed.
Recommendation n°3: The content of the WARC-Filename field of the Warcinfo record should not be changed.

Annex C of the WARC standard recommend designing WARC file names according to the following scheme[1]:

Prefix-Timestamp-Serial-Crawlhost.warc.gz

[1] Annex C of WARC standard also recommends to IIPC institutions to begin their filenames with "IIPC". This recommendation supposes that an internal identification system within IIPC has been defined. As this system does not exist yet, this recommendation can not be followed for now.

"Prefix is an abbreviation usually reflective of the project or crawl that created this file. Time-stamp is a 14-digit GMT time-stamp indicating the time the file was initially begun. Serial is an increasing serial number within the process creating the files, often (but not necessarily) unique with regard to the prefix. Crawlhost is the domain name or IP address of the machine creating the file".

In this rule, prefix is the only part of the naming where there is a human intervention. We recommend to record two possible kind of information in the prefix:

1. identification of institution / customer

2. identification of additional information about the internal instance ran for producing the file, e.g. identification of the job /collection (job ID, customer ID, collection name…).

3. identification of the nature of the WARC file, when it has not been produced by a crawling operation (e.g. repackaging operation or conversion from an ARC file; see examples below in sections 4.2.2 and 5.4.3).

Using a serial number is often necessary to avoid collision between ARC files. Timestamp may not be sufficient if the clock of the machine has been changed (e.g. after a crash).

Several institutions had already chosen a naming strategy for their ARC files conforming to the recommendations given in these guidelines. See as an example, the name of this ARC file produced by IA for its third French domain crawl:

BNF-CRAWL-003-20071013163428-02639-crawling04.us.archive.org.arc.gz

- BNF-CRAWL-003: indicates that the ARC file belongs to the third broad crawl made for BnF

- 20071013163428: is the creation date timestamp

- 02639: is the serial number

- crawling04.us.archive: is the name of the crawler

- arc.gz: is the extension

The naming scheme of ARC files generated at the National Library of New Zealand with the Web Curator Tool can be another example:

NLNZ-TI1179651-20091006011055-00000-kaiwae-z11.arc or
NLNZ-TI1179651-20091006011055-00001-kaiwae-z11.arc

- NLNZ: The National Library of New Zealand

- TI1179651: The ARC belongs to Web Curator Tool Target Instance number 1179651

- 20091006011055: Timestamp
- 00000 or 00001: The ARC sequence number for this job
- kaiwae-z11: The server where the crawler was running
- arc: the extension

In combination, the Target Instance number (1179651) and ARC sequence number (00000, 00001) are enough to avoid filename collisions.

Note: recommendations on the design of internal crawl instance IDs might be later given by the working group.

2.3 Record identification

2.3.1 General

There are two common ways to identify a WARC record. Historically, the record's Target-URI and record creation date ("URL+Timestamp") have been used. The URL+Timestamp can help end-users discover and access records while navigating within WARC- or ARC-based web archives.

However, it leaves the possibility of collision when two records with the same URL were harvested on the same GMT date (problem which may occur in federated archives involving collections coming from different institutions). That is the reason why the indication of the "collection" is frequently associated to this URL+Timestamp combination, where "collection" is an identifier for the institution or collection that holds the resource. For example, in

http://web.archive.org/web/20000301030523/http://www.natlib.govt.nz/

the prefix "http://web.archive.org/web/" effectively acts as a collection identifier for the IA web archive.

URL+Timestamp identification will not extend easily to WARC records that lack a meaningful Target-URI, or is hardly applicable when several WARC records share the same Target-URI (for example when a request, a response and a metadata record have the same Target-URI).

That is the reason why there is another way to identify a WARC record: the mandatory WARC-Record-ID, which addresses these problems. It offers the most general identification of a WARC record for data management and records preservation. A very large number of unique IDs is typically required (e.g., billions for a large collection building initiative spanning years), so an appropriate identifier generation method should be considered.

2.3.2 Identifier Generation

The *generation* method refers to the technical means for creating potentially billions of unique strings.

The generator might be based on a simple sequential counter or on software such as Noid [http://www.cdlib.org/inside/diglib/noid/noid.pdf] or libraries based on the UUID standard UUID [http://www.ietf.org/rfc/rfc4122.txt]. The Noid tool uses a set of counters; it can generate short strings, append check digits, and reserve or recycle strings. UUID software libraries don't require setting up or maintaining counters. Both Noid and UUID can generate billions of identifiers and both rely for global uniqueness on organizationally maintained registries (Noid on a Name Assigning Authority Number (NAAN) registry maintained by the CDL, and UUID on vendor registries of assigned MAC addresses, which themselves rely on an Organizationally Unique Identifier (OUI) registry maintained by the IEEE).

Whatever means is used, the generated IDs should be highly unique so as not to defeat the purpose of the WARC-Record-ID. It is critical that the method scale to create strings that are unique at least across the assigning institution. In addition, it is *strongly* recommended that these strings be embedded in a service context that represents them as globally (not just locally) unique. Global uniqueness provides the best foundation for interoperation among different institutional collections, such as a federation archives.

2.3.3 Identifier Service Context

The identifier *service context* is extra information, such as an identifier scheme label, that is added to the generated string before using it as a record ID. For example, here are four WARC record IDs based on the string "b123456789k987654321p2" (which could have been generated by UUID or Noid):

1. urn:uuid:b123456789k987654321p2
   This is a URI from the URN scheme and that depends on the UUID's promise of global uniqueness, which is based on the generating computer's MAC address, time of day, and random numbers. In the current Internet, URIs that begin with "urn" are not directly *actionable* (will not be converted by widely available web-aware software into the web address of the referenced content), so in its present form this WARC record ID will not be suitable for access.

2. http://hdl.handle.net/10.130/b123456789k987654321p2
   This is a URI that implicitly embeds a Handle [http://www.handle.net/overviews/handle-syntax.html] scheme identifier. It depends on the Handle registry having a unique entry for organization 10.130 and on that organization's ability to generate and assign unique strings. Because this URI begins with "http", it sets up the expectation that web access is or once was possible. Access with this Handle involves *resolution* through the global Handle system infrastructure (entered via hdl.handle.net), followed by a final

web redirection to the referenced content. Assigning Handles requires maintaining a local Handle server and may require annual fees to CNRI (Corporation for National Research Initiatives).

3. http://n2t.net/10130/b123456789k987654321p2
   A URI that begins with "http://n2t.net/" implicitly embeds identifiers intended for redirection via a small web server located at the host n2t.net. This is the consortially owned N2T (Name to Thing) resolver maintained at CDL and replicated globally for high availability. Similar to a Handle, this URI depends on the NAAN registry having a unique entry for organization 10130 and on that organization's ability to generate and assign unique strings. Like a Handle, it sets up the expectation that web access is or once was possible, but access with this URI involves just a web redirection to the referenced content. The N2T resolver can work with embedded identifiers from the ARK, URN, Handle, and DOI schemes.

4. http://n2t.net/ark:/10130/b123456789k987654321p2
   An actionable URI for which the path part begins "ark:/" is an Archival Resource Key (ARK). This kind of URI sets up the expectation that three kinds of access are or once were possible: (a) to the content, (b) to its metadata record (by appending '?'), and (c) to the archival commitment to it (by appending '??'). Supporting three kinds of access may not be appropriate for all digital content. This particular ARK also enjoys the benefits of an N2T-based URI.

With today's Internet, embedding an identifier in a URL is required to create actionable identifiers. In making a choice, an institution may prefer identifiers consistent with those used for its other digital assets. Thus the identifiers generation system should be a pluggable component of the WARC writing tools.

Recommendation n°4: For the near future, use Collection+URL+Timestamp combination for harvested files identification when providing access to external users.
Recommendation n°5: Use WARC record identifiers for internal management and preservation purposes.
Recommendation n°6: The chosen system of WARC record identifiers should generate identifiers unique within and outside the institution collection.
Recommendation n°7: WARC writing tools should enable institutions to choose their own records identification system.

2.4 Recording of processing information

2.4.1 General

WARC files are intended to be self-descriptive. Therefore, they should contain all information on **why** and **how** they have been created, including configuration files, log files, etc. In the reference model for and Open archival information system (OAIS), this kind of information is called context and provenance information.

The WARC standard defines two levels where processing information may be added: the record level and the WARC file level.

For each level of granularity, different types of metadata are relevant:

- At the WARC record level, it may be the seed URL that led to the discovery and the harvesting of a response record. As another example, processing information of a conversion record may be the name of the tool that migrated the payload from one file format to another.

- At the WARC file level, processing information may be the description of the crawler that created the WARC file. Processing information of a WARC file generated from an ARC file may be the name of the original ARC file.

For each level of granularity, there are several locations to put processing information.

- At the WARC record level, information may be located in **record headers** (e.g. for a response record, a reference to the corresponding request record) and in one or several **metadata records** (e.g. a metadata record indicating its seed URL).

- At the WARC file level, information may be located in the **warcinfo record**, in **metadata records** or in **resource records**.

A problem arises when it comes to information related to a set of WARC files (generated by a same crawl instance). Where should be located configuration and logs files describing this crawl instance?

We may also find at the crawl instance level some content/collection information, for example information that explains why a job was generated (e.g.: this seed is in this job because it was flagged in my curator tool as being part of the "election crawl").

To record context and provenance information that pertain to a higher level than the one of the WARC file, this document recommends using the same locations as for information pertaining to the WARC file level: warcinfo, resource and metadata records.

2.4.2 Use of warcinfo records for processing information

It is recommended to record in warcinfo:

- information about the crawling institution or customer;
- information about the crawler, and generally technical and organizational information on the job;
- origin of the file: e.g. created during a crawl, a format conversion, a repackaging operation;
- some collection/content information which reveals the purpose of the file.

Warcinfo contains therefore high-level configuration information, which is available before the writing of the WARC file begins. Typically, information conveniently recorded in file names must also be recorded in warcinfo.

Note: warcinfo should only record information gathered at harvesting time and that will not change. Information subject to later changes and updates (e.g. collection description used for access, such as cataloging information) should not be recorded there. It may be recorded in another WARC file or in external locations.

Note on information redundancy within warcinfo records.

It is recommended in the standard to put a warcinfo record at the beginning of each WARC file. So, when a single crawl instance generates several WARC files, the information recorded in all warcinfo records will be quite similar.

This redundancy should not be viewed as a problem, as WARC files are supposed to be self-descriptive, and they should not need to refer to other WARC files.

On the other hand, information that does not need to be duplicated should be recorded in the resource records described below.

At last, all WARC records, whatever their record type may be (except warcinfo records), should have a WARC-Warcinfo-ID field. This information is critical to recall the origin of each WARC record.

2.4.3 Use of resource records for processing information

Some information cannot be recorded in warcinfo records, as it is released at the end of a crawl process: notably configuration, report and log files. These files should be located in resource records, as it is explained in the standard: "A resource record, with a

synthesized Target-URI, may also be used to archive other artifacts of a harvesting process inside WARC files".

We recommend using a separate resource record to store each individual file (i.e. each individual report file, log file… see examples below).

Resource records shall have a Content Type field that indicates the MIME type of the recorded configuration file. A Target-URI is mandatory for each resource record. Institutions need to design their own URI scheme. No recommendations are given yet in these implementations guidelines.

The Danish NetarchiveSuite tool may be proposed as an example. It generates ARC files where all configuration, report and log files of a single job are copied as separate ARC records.

Each record has an URI following the scheme:

metadata://netarkivet.dk/crawl/[file_type]/[filename]?[heritrixVersion]&[harvestid]&[jobid]

For example:

metadata://netarkivet.dk/crawl/logs/crawl.log?heritrixVersion=1.14.3&harvestid=6&jobid=41

It is recommended that all resource records containing processing information files are stored in a specific WARC file (that may be called a "metadata WARC file"). On the other hand, institutions that perform small-scale crawls (where all harvested files of a same crawl instance will generally be recorded in a single WARC file) may locate them in the same WARC file as the harvested data.

For large-scale crawls, it may be necessary to generate several metadata WARC files. In fact, records containing process information may be very big; crawl logs notably may be larger than 1 Gb. In this case, two solutions are possible:

- record the crawl log in a WARC file larger than the theoretical limit of 1 Gb;
- segment the crawl log and record it in several WARC files.

A third solution would be to truncate the crawl log after the maximum size has been reached. This solution is not recommended.

2.4.4 Association of records to the crawl instance level

To associate these resource records to the level of the crawl instance, it is recommended – at the end of the crawl process – to write a metadata record that lists the warcinfo IDs of all WARC files produced by a crawl instance. As warcinfo records

contain the original name of the WARC files in which they are contained, this metadata record will stand as a kind of manifest of the crawl.

Secondly, each resource record containing processing information will be linked to this "manifest" metadata record.

Three examples of records are given below to explain how they should be designed (important fields had been set in bold types).

1. "Manifest" metadata record

```
WARC/1.0
WARC-Type: metadata
WARC-Date: 2006-09-19T17:20:24Z
Content-Length: 2000
WARC-Warcinfo-ID: <urn:uuid:d7ae5c10-e6b3-4d27-967d-34780c58ba39>
WARC-Record-ID: <urn:uuid:16da6da0-bcdc-49c3-927e-57494593b943>
Content-Type: application/warc-fields
WARC-Block-Digest: sha1:T7AFNTKDO92MSS7ZENMFZY6ND8RG9DL5

[The block contains the list of the Warcinfo-IDs of all WARC files
created during the crawl instance, including the metadata WARC
file(s)]
```

2. Resource records

```
WARC/1.0
WARC-Type: resource
WARC-Date: 2006-09-19T17:20:24Z
Content-Length: 29365
WARC-Warcinfo-ID: <urn:uuid:d7ae5c10-e6b3-4d27-967d-34780c58ba39>
WARC-Record-ID: <urn:uuid:16da6da0-bcdc-49c3-927e-57494593dddd>
Content-Type: application/xml
WARC-Block-Digest: sha1:VXT4AF5BBZVHDYKNC2CSM8TEAWDB6CH8
WARC-Payload-Digest: sha1:VXT4AF5BBZVHDYKNC2CSM8TEAWDB6CH8
WARC-Concurrent-To: <urn:uuid:16da6da0-bcdc-49c3-927e-57494593b943>

[The block contains the crawler configuration]


WARC/1.0
WARC-Type: resource
WARC-Date: 2006-09-19T17:20:24Z
Content-Length: 5321654
WARC-Warcinfo-ID: <urn:uuid:d7ae5c10-e6b3-4d27-967d-34780c58ba39>
WARC-Record-ID: <urn:uuid:70653950-a77f-b212-e434-7a7c6ec909ef>
Content-Type: text/plain
WARC-Block-Digest: sha1:T7HXETFVA92MSS7ZENMFZY6ND6WF7KB7
WARC-Payload-Digest: sha1:T7HXETFVA92MSS7ZENMFZY6ND6WF7KB7
WARC-Concurrent-To: <urn:uuid:16da6da0-bcdc-49c3-927e-57494593b943>

[The block contains the crawl log]
```

In both examples, the "WARC-Concurrent-To" fields refer to the record-ID of the manifest metadata record (proposed as first example). Note that a specific MIME type for manifest metadata records might be proposed by the working group.

Recommendation n°8: The following system should be used to record processing information at the record, container file and crawl instance level:

Level of information	Location	Examples
WARC record	Record headers	Reference to the request that caused the sending of the record
	Metadata record concurrent to the record	Seed URL that led to the harvesting of the record
WARC file	File name	Institution ID, crawl instance ID, timestamp…
	Warcinfo record at the beginning of the file	Institution ID, crawl instance ID, crawler information, content information that reveal the purpose of the file
	Metadata record(s) concurrent to the file's warcinfo record	Filedesc of the original ARC file (see 4.2.4 below)
Crawl instance	File name	Institution ID, crawl instance ID, timestamp…
	Warcinfo record at the beginning of all files generated by crawl instance	Institution ID, crawl instance ID, crawler information, content information that reveal the purpose of the file
	"Manifest" metadata record concurrent to warcinfo records of all files of the crawl instance	List of all WARC files produced by the crawl instance
	Resource record(s) concurrent to the manifest metadata record	Configuration, report, log files Additional collection information explaining reasons to launch a crawl

Recommendation n°9: Use a WARC-Warcinfo-ID field in every WARC record (except warcinfo records).

Recommendation n°10: Use a WARC-Filename field in every warcinfo record.

3 Web harvesting

3.1 Scope

This section is dedicated to recommendations on how to write WARC files during a web harvesting session. Basically, it describes how should behave the WARC writing module of a crawler.

3.2 Record types generated during web harvesting

Typically, the harvesting of a file on the web may generate two different sets of records.

1. A request record containing the crawler's request; a response record containing the server's response (http response + payload); and an optional metadata record.

2. A request record containing the crawler's request; a revisit record if the crawler chose not to record the harvested file because it was a duplicate; and an optional metadata record.

In both cases, all the records share the same WARC-date (see WARC standard, p. 6) and the same Target-URI, and are linked together using WARC-Concurrent-To relationships. A revisit record is also linked to the original response record using WARC-Refers-To field.

3.3 Use of metadata records in web harvesting

3.3.1 General

Different kind of information may be located in the metadata records concurrent-to response records. You may find examples of relevant information in the draft IIPC metadata document.

For now, these implementation guidelines only study the case of payload information generated during the harvesting process. Other examples and cases may be added later.

3.3.2 Recording payload information generated during the harvesting process

Metadata records may be used to record additional information on the harvested payload. This information will typically be format information generated by identification or characterization tools (see below, section 5.2). If the identification/characterization is made "on the fly", that is if the operation is part of the harvesting process, output information should be described in metadata records concurrent-to each corresponding record. The reliable MIME type (and, if possible, the version number) of the payload should be written in the "WARC-Identified-Payload-Type" field.

These metadata records may be in the same WARC file as their corresponding records or another file if the previous one is full. These records should also contain references to the tools that performed the identification or characterization processes.

Note that format information generation is currently a very time-consuming process, which is likely to be limited to small-scale collections. See below, section 5.2, for recommendations in the case where operations are done afterwards (after the end of the harvesting process).

Recommendation n°11: Payload information computed during the harvesting process should be recorded in metadata records, with a concurrent-to field that indicates the ID of the record containing the corresponding payload. The name of the format and if possible its version number should be recorded in the WARC-Identified-Payload-Type field of the record containing the corresponding payload.

4 Data packaging

4.1 Scope

This section is dedicated to recommendations on how to package in WARC files data that are not directly harvested on the Web (but that may have been originally published on the Web). As there is still very few users experience towards this topic, most of this section refers to ARC/WARC conversion, that how to package in WARC files data previously contained in ARC files.

4.2 ARC to WARC conversion

4.2.1 Number of ARC and WARC files

As a first principle, it is recommended to convert one ARC file into one WARC file. Even though the standard size of ARC files (100 Mb) is ten times smaller than the one of WARC files (1 Gb), merging 10 ARC files in one single WARC file would make disappear the original ARC files organization. Moreover, it would make the operation really more complex for ARC to WARC conversion tools.

Recommendation n°12: Maintain a 1 ARC file = 1 WARC file relationship for ARC to WARC conversions.

4.2.2 Converted WARC file names

There are typically two ways of generating the WARC filename:

1. use the original ARC filename and add a prefix indicating that the WARC file come from a conversion operation;
2. use the naming scheme defined in the annex C of the standard, where:
    - prefix notably indicates that the WARC file come from a conversion operation;
    - timestamp is the WARC file creation date;
    - serial is a serial number;
    - crawlhost is the name of the machine where conversion process occurred.

Both solutions are consistent with the recommendations expressed in section 2.1

The first solution is better to indicate the content and the origin of the file.

The second solution is better for practical operations (to identify which WARC file has been created from which machine).

There is no agreement within the working group on which is the better solution.

4.2.3 WARC-Date field

Another question arises when it comes to the date of a converted WARC record. The standard states that the WARC-date field of a record should be filed with a timestamp that "shall represent the instant that data capture for record creation began".

In a conversion process, is the record creation date the conversion date or the original harvesting date?

For practical reasons, it seems preferable to use the original harvesting date. Access tools will indeed use the WARC date field to display the harvesting date to end-users.

Note that a different behavior should be adopted for payload migration: according to the standard, the WARC-date of a conversion record is the date of the creation of the new record, that is when the migration occurred. There is indeed a great difference between converting a file from a container format to another, and migrating the format of this file.

Recommendation n°13: The WARC-date of a converted WARC record should be the same as the one of the original ARC record.

To record the date when the ARC to WARC conversion occurred, the converted record should be linked to the warcinfo record describing the conversion process, thanks to its WARC-Warcinfo-ID field.

4.2.4 ARC filedesc record

The filedesc of an ARC file is a special case record giving some information on the crawl operation itself. It is located at the beginning of each ARC file.

It may seem logical to convert this filedesc in a warcinfo record, as this record type has been designed to play the role of the former filedesc. However, the warcinfo record of a converted WARC file should rather describe the conversion process (and it is not possible to have two warcinfo records within the same WARC file).

So it is recommended to create for each converted record a warcinfo describing the conversion process, and to locate the original filedesc in a metadata record concurrent to this warcinfo record. To be consistent with the previous recommendation, the WARC-Date of this metadata record should be the original date (that is the date of the creation of the filedesc).

Recommendation n°14: The filedesc of an ARC file should be located in a metadata record concurrent to the warcinfo record of the converted WARC file.

4.3 Packaging web data to WARC

This section is dedicated to recommendations on how to package into WARC files data originally published on the Web and copied offline afterwards.

It includes notably recording HTTrack captures, wget captures, or web files back-up, into WARC files.

This section is currently empty; however some recommendations expressed for ARC to WARC conversion (notably on WARC dates) should apply. Other recommendations may be given in the future.

4.4 Packaging non-web data to WARC

Several institutions use ARC file format to store and manage non web resources, for example digitized books or electronic journals. Note that the WARC standard was primarily intended to specify a format storing resources from mainstream Internet application layer protocols (HTTP, DNS, FTP).

Therefore it has been decided not to include recommendations on the packaging of non-web data into WARC files in the scope of these WARC implementation guidelines.

5 Operations on WARC files

5.1 Scope

This section is dedicated to recommendations on how to write WARC records or files generated by operations performed on previously existing WARC files. These operations on WARC files may include ingest processes, data exchange, and preservation operations.

5.2 Payload identification and characterization

During a harvest session, very little information is obtained on the format of the harvested files. The most common information is the MIME type of the file, sent by the server, and this is sometimes incorrect.

Institutions may want to get more information on the format of the files they have collected, in order to provide access (to know the reading software they have to deploy

on their web archives access platform) or perform preservation operations (migration, choice of emulation solutions).

They may also use several tools to identify, validate or characterize the format of the payloads contained in WARC files[2]: for example Droid or the Unix "File" command for identification, Jhove for validation and characterization. This kind of operation will most commonly be made on payloads of response, resource, conversion or continuation records.

Identification and characterization can be performed during of after the harvest. See section 3.3.2 for recommendations on operations performed as part of the harvesting process.

If the operation is made after the harvest on existing WARC files, a distinction should be made:

- if the output is not supposed to be kept for the long term by the institution, it should not be written in a WARC file (but recorded in an external and temporary location);

- if this information is supposed to last and be useful for long term management and documentation of the collection, it should be recorded in a separate WARC file, in metadata records referring-to each corresponding record. Information should also be given on the tool(s) that performed the identification/characterization processes.

Format information contained in WARC files may be replicated in external locations (e.g. METS or PREMIS files), at the institution's choice. No recommendation is given on the use of external locations for format information.

Recommendation n°15: Payload format information computed after the end of an harvesting process should be recorded in metadata records , with a refers-to field that indicates the ID of the record containing the corresponding payload.

5.3 Virus checking

Archives may consider viruses contained in payloads to be a threat to their own computing system or that of their users. They may use therefore an AV tool to check

[2] Stephen Abrams, Sheila Morrissey and Tom Cramer distinguish four possible operations: identification (discovering in what format is encoded the file, and in what version), validation (determining conformance of the file to the theoretical requirements of the format), features extraction (reporting the intrinsic properties of a digital object) and assessment (determining the level of acceptability of a digital object for a specific use on the basis of locally-defined policies). See http://www.bl.uk/ipres2008/ipres2008-proceedings.pdf.

their collections contained in WARC files. Several successive levels of treatments are then possible:

1. Record the information that a WARC record is infected. This information should be located in a metadata record (referring to the original record) identifying the AV tool used and the name of the virus.

2. Give order to the access tool not to access the infected file.

3. Heal each infected payload separately and generate a conversion record to store the healed payload. Make a refers-to link between the healed and the infected payload.

Each institution may choose until what level of treatment it wants to go.

It is recommended to avoid deleting the original infected payload, unless the institution's security policy strictly forbid holding of viruses. Deleting a harvested file goes against the goal of maintaining the collection's integrity and authenticity. Moreover, changing a single record will force the institution to re-index the whole WARC file.

5.4 WARC files repackaging

5.4.1 Repackaged record IDs

This section is dedicated to WARC repackaging operations. A repackaging operation consists in copying a set of WARC records from an original WARC file, to a new WARC file. The copied records may be chosen according to different filtering rules. Such operations may be useful for QA operations (sampling), format migration (extracting records whose payload is in a specific format to migrate it in another format), or to deliver a set of records to a third party.

It is recommended that a repackaged record is considered a new record. Indeed, the WARC-Warcinfo-ID field of a repackaged record is different from the one of the original record, so the records are different. As a consequence, a repackaged record should have a new record-ID.

Other fields, including the WARC-Date field, should not change.

5.4.2 Link with previous record

The repackaged record should be linked to the previous one. Two solutions are possible:

1. Generating a metadata record "concurrent-to" the repackaged record and that "refers-to" the original record OR

2. Using in the repackaged record a new WARC field: "WARC-previous-record-ID" (that contains the ID of the original record).

Even though the first solution is possible, the second one is recommended.

If a record generated by a repackaging operation is repackaged one more time, a third (or greater) generation record is created. Two alternative solutions may be recommended, according to the repackaging operation goals:

- the "WARC-previous-record-ID" field contains the ID of the last repackaged record OR

- the former "WARC-previous-record-ID" field (or fields) is kept and a new "WARC-previous-record-ID" field is written, so that the list of all intermediate records will be kept. Note that the order of the "WARC-previous-record-ID" headers will not necessarily be meaningful as the WARC standard states that "named field may appear in any order" (section 4, page 3).

5.4.3 Repackaged WARC file names

The name of a repackaged WARC file should follow the naming scheme defined in the annex C of the standard:

  - prefix should notably indicate that the WARC file come from a repackaging operation;

  - timestamp should be the new WARC file creation date;

  - serial should be a serial number;

  - crawlhost should be the name of the machine where repackaging process occurred.

Recommendation n°16: A repackaged WARC record should be considered a new record.
Recommendation n°17: A new field "WARC-previous-record-ID" should be added to the list of named fields in a repackaged record to make a reference to the original record. This field should be repeatable.
Recommendation n°18: A repackaged WARC record should have the same header fields as the original record, expect for WARC-Record-ID, WARC-Warcinfo-ID and possibly "WARC-previous-record-ID".