WARC as Package Format for all Preserved Digital Material

by Eld Zierau
The Royal Library of Denmark

Except from WARC update record these are slides from
- iPres 2012 and
- PiF 2014
Motivation

Bit preservation on considerable part of digital material:
- Digitally born materials
- Substitution digitalization
- Web archive

There are many types of digital materials

We need packaging:
- Preserve link between identifier and object (incl. files)

*Explained later*

*Avoid many different package formats – Preferable one*
How to evaluate

1. Find requirements
   - Package and storage related
   - Preservation related requirements (from formats)
   - Identification related requirements

2. List of possible formats
   - AFF
   - ARC
   - BagIt
   - METS
   - RAR
   - TAR
   - WARC
   - ZIP
   - …

3. Evaluate which format fits the requirements best
List of Preservation related requirements

- Req. 1: Must be Independent of storage platform
- Req. 2: Must allow flexible packaging
- Req. 3: Must allow update records
- Req. 4: Must be standardized format
- Req. 5: Must be open
- Req. 6: Must be easy to understand
- Req. 7: Must be widely used in bit repositories
- Req. 8: Must be supported by existing tools
- Req. 9: Must be able to include digital files unchanged
- Req. 10: Must facilitate identifiers for a digital object

*Not an exhaustive list*
**Identification related requirements**

- **Requirement 10:**
  Must facilitate identifiers for a digital object

![Diagram](attachment:image.png)

*Especially a challenge when the object is ‘a file’*
Reasons for id. requirements

1. **Leave it 100% to the bit preservation solution**
   - Risk since it is *crucial* information in preservation – outsourcing of responsibility
   - Eliminate possible optimisation of packaging more files or files and metadata in the same package

2. **Naming files with the identifier**
   - file name is not part of the file itself
   - restrictions to how files are named
   - may not make same sense in the future

3. **Put identifier into files as inherited metadata**
   - knowledge of how to extract identifiers from file formats
   - would need to change original bits

4. **Wrap files and identifier in a package format**
   - requirements for the abilities of the package format
Position based like ARC and WARC

WARC is an ISO standardised enhancement of ARC

WARC/1.0
WARC-Type: warcinfo
WARC-Date: 2012-08-27T15:50:16Z
WARC-Record-ID: <urn:uuid:21d07350>
Content-Type: application/warc-fields
Content-Length: 46
application: id.kb.dk/gatekeeper/releasetest17

WARC/1.0
WARC-Type: resource
WARC-Target-URI: urn:uuid:15AE9513
WARC-Date: 2012-08-27T15:50:14Z
WARC-Record-ID: <urn:uuid:15AE9513>
Content-Type: image/tiff
Content-Length: 139803706
II*1214i
eeciRGB v2
P`pj²ÅÔå,>PcuÁÕèü$8Ma
...
Packaging the preservation metadata

WARC/1.0
WARC-Type: warcinfo
WARC-Date: 2013-01-18T19:27:59Z
WARC-Record-ID: <urn:uuid:21d07350>
Content-Type: application/warc-fields
Content-Length: 79
description: http://id.kb.dk/authorities/agents/kbDkDBIngest .html
revision: v4

WARC/1.0
WARC-Type: resource
WARC-Target-URI: urn:uuid:15AE9513
WARC-Date: 2013-01-18T19:27:59Z
WARC-Block-Digest: md5:3f349a40b0c47bb070ea6bddd2759a731
WARC-Record-ID: <urn:uuid:15AE9513>
Content-Type: image/tiff
Content-Length: 139803706
II*1214i
eećiRGB v2
P`pı² ÄÕå,>PcuÁÕëü$8Ma
...
Metadata Standards and use in Preservation

WARC/1.0
WARC-Type: metadata
WARC-Target-URI: urn:uuid:c9db2170-619c-11e2-911b-005056887b67
WARC-Date: 2013-01-18T19:27:59Z
WARC-Refers-To: <urn:uuid:15AE9513 >
WARC-Block-Digest: sha1:62cc454ef47c7d54b77f871ab1ffd3f580307414
WARC-Record-ID: <urn:uuid:c9db2170-619c-11e2-911b-005056887b67>
Content-Type: text/xml
Content-Length: 13926

<?xml version="1.0" encoding="UTF-8"?>
      xmlns="http://www.loc.gov/METS/">
...<linkingIntellectualEntityIdentifier>
   <linkingIntellectualEntityIdentifierType>UUID</linkingIntellectualEntityIdentifierType>
   <linkingIntellectualEntityIdentifierValue>41d153d1-0099-11e2-9397-005056887b67</linkingIntellectualEntityIdentifierValue>
...</linkingIntellectualEntityIdentifier>
...</mets>
Update via Preservation Metadata

- At this stage only metadata (to be implemented)
- Two ways:
  1. In preservation metadata
  2. Using a "new" update WARC-record

1. Using PREMIS – the event is an update referring back

2. WARC allows for other record types than source and metadata

In both cases use ‘concurrentTo’ as shortcut
## Package Formats - fulfilment of requirem.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Formats</th>
<th>AFF</th>
<th>ARC</th>
<th>BagIt</th>
<th>METS</th>
<th>RAR</th>
<th>Tar</th>
<th>WARC</th>
<th>ZIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Platform independent</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>2. Flexible packaging</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>3. Supports update pack.</td>
<td></td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Almost No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>4. Standardised</td>
<td></td>
<td>Little</td>
<td>No</td>
<td>So-so</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Little</td>
</tr>
<tr>
<td>5. Open</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Almost</td>
</tr>
<tr>
<td>6. Easily understandable</td>
<td></td>
<td>So-so</td>
<td>So-so</td>
<td>So-so</td>
<td>Almost</td>
<td>No</td>
<td>Little</td>
<td>Yes</td>
<td>Little</td>
</tr>
<tr>
<td>7. Widely used in BRs</td>
<td></td>
<td>No</td>
<td>So-so</td>
<td>Almost</td>
<td>So-so</td>
<td>Little</td>
<td>Yes</td>
<td>Almost</td>
<td>So-so</td>
</tr>
<tr>
<td>8. Tools available</td>
<td></td>
<td>So-so</td>
<td>Yes</td>
<td>Yes</td>
<td>So-so</td>
<td>Yes</td>
<td>Yes</td>
<td>So-so</td>
<td>Yes</td>
</tr>
<tr>
<td>9. Include files unchanged</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>10. Identifiers for files</td>
<td></td>
<td>Yes</td>
<td>So-so</td>
<td>So-so</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

### Fulfilled Levels:

- **Yes**
- **Almost**
- **So-so**
- **Little**
- **No**
Conclusion

Recommends WARC as the best suited format for long term preservation of varied digital materials.

With given requirements

Strong on:
- applying identifiers to files
- easily understandable
- one of few formally standardised formats
- extendible with record definition for updates

Weak on:
- Not well supported by tools
- Only widely used in web archiving

<table>
<thead>
<tr>
<th>Requirement</th>
<th>WARC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Platform independent</td>
<td>Yes</td>
</tr>
<tr>
<td>2. Flexible packaging</td>
<td>Yes</td>
</tr>
<tr>
<td>3. Supports update pack.</td>
<td>Yes</td>
</tr>
<tr>
<td>4. Standardised</td>
<td>Yes</td>
</tr>
<tr>
<td>5. Open</td>
<td>Yes</td>
</tr>
<tr>
<td>6. Easily understandable</td>
<td>Yes</td>
</tr>
<tr>
<td>7. Widely used in BRs</td>
<td>Almost</td>
</tr>
<tr>
<td>8. Tools available</td>
<td>So-so</td>
</tr>
<tr>
<td>9. Include files unchanged</td>
<td>Yes</td>
</tr>
<tr>
<td>10. Identifiers for files</td>
<td>Yes</td>
</tr>
</tbody>
</table>

But may change
Questions

Images of this style from digitalbevaring.dk