

Supplement 211: DICOMWEB SUPPORT FOR RETRIEVE VIA APPLICATION/ZIP

PREPARED BY BRAD GENEREAUX, ON BEHALF OF DICOM WORKING GROUP 27
JANUARY 7, 2019

Introduction / Scope

- Add “application/zip” as an option in the DICOMweb WADO-RS specification, to retrieve an entire DICOM study or series as a single packaged file
- Primary use case is to enable researchers in analytics / machine learning users of DICOMweb who want to retrieve studies / series / collections of images for training purposes
 - DICOMweb currently provides a means to retrieve an entire study or an entire series using a “multipart/related” content type; but this requires special non-browser implementations to retrieve the content (multipart/related is not supported by any browsers; it would require special programmatic interventions); and it provides the objects separately (as opposed to a bundle) making them more difficult to work with on disk / importing into their machine learning tooling as they are streamed into individual files
- Secondary use case is for consumer use of DICOM objects, as providing a ZIP file of DICOM content to a patient or caregiver is more familiar than providing them with a multipart/related response to be programmatically interpreted

Examples

Retrieve the full study:

```
GET https://server.com/wado/studies/2.16.124.113543
```

```
Accept: application/zip
```

Response: A DICOM ZIP file containing all of the DICOM instances of this study (with a DICOMDIR)

Retrieve an individual series:

```
GET https://server.com/wado/studies/2.16.124.113543/series/2.16.678
```

```
Accept: application/zip
```

Response: A DICOM ZIP file containing all of the DICOM instances for this series (with a DICOMDIR)

Specify specific transfer syntax:

```
GET https://server.com/wado/studies/2.16.124.113543
```

```
Accept: application/zip; transfer-syntax=1.2.840.10008.1.2.4.70
```

Response: A DICOM ZIP file containing all of the DICOM instances of this study (with a DICOMDIR) with a specific syntax

Retrieve a full study by URL only:

```
GET https://server.com/wado/studies/2.16.124.113543?accept=application/zip
```

Response: A DICOM ZIP file containing all of the DICOM instances of this study (with a DICOMDIR)

Resolved Discussions/Questions (Thus Far)

- (1, 2, 7) Is “zip” a “DICOM media type”, a rendered media type, or something else? / Would we consider supporting the DICOM media type already defined in Part 12 section V? / Do we want to include support for DICOMDIR structures, or make it the default (or only) format?
 - Yes, it is a DICOM media type and this supplement will use what is defined in http://dicom.nema.org/medical/dicom/current/output/html/part12.html#chapter_V
 - We will use existing DICOM Zip specification, which uses DICOMDIR
 - Supports existing implementations already using DICOM Zip (one implementer confirmed); committee recognizes this may not be “ideal” but the cost doesn’t justify the benefit of the change; note recipients have no requirements to do anything with the DICOMDIR
- (3a) Do we want to have parameters to support anonymization?
 - No; ; anonymize not supported by WADO-RS, and not directly related to this supplement and thus will not be included as part of this supplement
 - Recipients can anonymize the data after receipt; or data could be pre-anonymized. It just isn’t in this step of the pipeline
- (6) Do we want to allow support for tar or rar, be silent on it, or explicitly disallow it?
 - The specification describes application/zip; any other packaging formats will not be specified
- (8) Can we support retrievals that are completely specified in a URL (and not require Accept header to be passed)?
 - Yes; this is already supported in PS3.18 6.1.1.5 (query parameter “accept”) http://dicom.nema.org/medical/dicom/current/output/html/part18.html#sect_6.1.1.5

Discussions/Questions with a Proposed Position

- (3b) Do we want to have parameters to support ZIP encryption?
 - We believe so - ZIP encryption is useful for data "at rest" once the client retrieves the ZIP file (it is irrelevant for data "in transfer" as HTTPS encrypts traffic)
 - Encrypting at the client is less efficient than specifying at time of request
- (4) Do we need to specify any details regarding the transfer syntax / media types of the contained files?
 - We believe so, using the same structure/methodology as what multipart/related supports
- (5) Do we want to specify a different RESTful resource (e.g., /packaged)?
 - No, we don't believe that's necessary; it's essentially a transformation on the same path as other WADO-RS services
 - As with any of the REST services DICOMweb offers, the design tended to follow the principles of REST Uniform Interface (https://en.wikipedia.org/wiki/Representational_state_transfer#Uniform_interface) and Clean URLs (https://en.wikipedia.org/wiki/Clean_URL); ZIP is just a transformed representation of a DICOM study and it is acceptable and encouraged by REST community to specify it in this way
 - In line with how other APIs work and is more discoverable and what's expected
- (11) Do we need to update the reference to ZIP as a standard? The reference standard is currently <https://support.pkware.com/display/PKZIP/APPNOTE>
 - No, in as far as we can tell, this remains the correct URL to reference
- (13) What happens if a client accepts both multipart/related and application/zip and are "weighted" equally?
 - HTTP protocol specifies it is a server's decision to choose then, and we believe that mantra should still apply

Discussions/Questions to be Resolved

- (10) Are there conceivable limits to the ZIP file size itself? What about DICOMDIR limits? Do we need to limit the # of items in the root directory?
 - This problem would apply to any implementation of DICOM Zip – solutions would apply beyond web; maybe we should make a note in “notes” of any upper limits that is useful for implementers
 - Does anyone at WG-6 remember any enhanced object size discussions that might be relevant?
- (9) Should there be a requirement for a “README” to indicate “where it came from” and other metadata?
 - Might be something for IHE / PDI, whoever composed the ZIP; see https://www.ihe.net/uploadedFiles/Documents/Radiology/IHE_RAD_TF_Vol3.pdf, in particular section 4.47.4.1.2 to see if anything is worth mirroring, with specifics about the README in 4.47.4.1.2.2.2
- (12) Are there security concerns about popping a URL into Chrome for example?
 - We need to think through if there are any concerns; e.g., is there special behavior if someone tries to download but not authenticated
- (15) Should we consider ZIP at the instance level, or just at the study / series level?
- (16) Should we consider ZIP at the rendered level as well?
 - November 2018 discussions suggested this should be in scope (meets need for algorithm developers with JPG libraries)
 - December 2018 discussions suggested this should be removed from scope (many “gotchas” in what people would need to specify)
- (17) Do we need a chapter in PS3.12 Appendix V to describe any specific behavior for network transport / on the wire for ZIP files?
 - To be determined still