

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24

**Digital Imaging and Communications in Medicine (DICOM)**

*Supplement 211:  
DICOMweb application/ZIP Payload*

**DICOM Standards Committee, Working Group 27**  
1300 N. 17th Street  
Rosslyn, Virginia 22209 USA  
Developed in accordance with work item WI 2018-09-C  
VERSION: Letter Ballot Nov 11, 2020

## Table of Contents

Scope and Field of Application .....	3
Closed Questions .....	4
8.6.1 Payload Format .....	5
8.6.1.3 ZIP Payload .....	5
8.6.1.3.1 File Name and Extensions .....	5
8.6.1.3.2 BulkData URI .....	6
8.6.1.3.3 Logical Format .....	6
8.6.1.3.4 Metadata Representations .....	6
8.7 Media Types .....	7
8.7.1 Multipart Media Types .....	8
8.7.3.5 DICOM Media Type Syntax .....	8
8.7.3.5.4 ZIP Payload Media Types .....	9
8.11 Security and Privacy .....	9
<u>8.11.1 PS3.10 File Format Considerations</u> .....	10
<u>8.11.2 ZIP Format Considerations</u> .....	10
<u>8.11.2.1 Encryption</u> .....	10
10.4.1.1.2 Metadata Resources .....	i
10.4.4 Media Types .....	i
B.6 Retrieve a ZIP Payload .....	iii
F.2.2 DICOM JSON Model Object Structure .....	iv
F.2.6 BulkDataURI and BulkDataUUID .....	v

---

## 49 **Scope and Field of Application**

50 Supplement 211 adds the “application/zip” payload to the RESTful Web Services Retrieve Transactions in PS3.18. It enables  
51 retrieving all the instances in an entire DICOM study or series as a single zip payload.

52 A key use case is to enable machine learning researchers to use DICOMweb to retrieve studies, series, or other collections of  
53 images for training purposes. DICOMweb currently provides a means to retrieve an entire study or an entire series using the  
54 “multipart/related” media type, but this requires special non-browser implementations to retrieve the content because  
55 multipart/related is not currently supported by any of the major browsers.

56 Usage of this zip payload on interchange media is not addressed by this supplement.

57 When used in the context of WADO-RS, there is currently no way to request a payload with instances spanning multiple studies.

58

99 **Closed Questions**

#	Question	Notes
2	Do we want to have parameters to support anonymization?	The retrieve services may or may not support anonymization services, but that is orthogonal to the ZIP Payload packaging.
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. 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
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. We believe that mantra should still apply
16	Should we consider ZIP at the rendered level as well?	No This would be added at the same time/way as multipart payloads are handled.
17	Is it appropriate to make normative file extensions for the representations within the ZIP payload?	Table 8.6.1.3.1-1 mandates specific extensions.
28	Should this form of application/zip be supported on interchange media? e.g. CD, email, etc.	No Interchange media should be small and have a minimum number of options for parsing data. This payload can be in XML or ZIP with or without Bulkdata, all of which make media usage more problematic.

10

11

62 **Insert Reference in PS3.18 Section 2.1**

63 **[ISO/IEC 21320-1] ISO/IEC. 2015/10. 1.0. Information Technology – Document Container File. Available at**  
 64 <https://infoworks.org/notes/2013/04/n130401-13R5EoW9HY4ZvQH/n130401c11-ISO IEC 21320-1 2015.pdf>

65 **Update PS3.18 Section 8.6.1 as follows:**

66 **8.6.1 Payload Format**

67 Payloads may be in ~~either~~ single part, ZIP, or multipart format depending on the media type.

68 **Insert PS3.18 Section 8.6.1.3 as follows:**

69 **8.6.1.3 ZIP Payload**

70 A ZIP payload contains one or more representations. The media type of the payload shall be 'application/zip'. It is used as a  
 71 container for distributing a set of related DICOM SOP Instances and/or BulkData. See PS3.18 Section 8.7.3.5.4 for details of the  
 72 zip payload and PS3.18 Appendix B.5 for an example.

73 A zip payload shall not contain executable content.

74 **8.6.1.3.1 File Name and Extensions**

75 The file name shall have a file extension corresponding to its Media Type as defined by Table 8.6.1.3.1-1.

76 File names shall be unique within their directory.

77 BulkData shall be contained in the same directory as the JSON or XML file that references it, or a subdirectory thereof.

78 When BulkData is for a multi-frame image whose frames are encapsulated separately:

- 79 • The basic offset table from the Pixel Data (7FE0,0010) shall not be included.
- 80 • Each frame shall be a different file.
- 81 • The file name shall be the frame number followed by the file extension.
- 82 • The file name shall not be zero padded.
- 83 • All the frames shall be put in the same directory.
- 84 • The BulkData URI shall reference the multi-frame directory without a trailing file separator.

85 The file extension shall correspond to the media type of the representation as shown in Table 8.6.1.3.1-1.

86 **Table 8.6.1.3.1-1. Media Types and File Extensions**

Media Type	File Extension	Description
application/dicom	"dcm"	Part 10 DICOM File
application/dicom+json	"json"	Single Instance JSON Encoded DICOM
application/dicom+xml	"xml"	XML Encoded DICOM
octet/stream	"raw"	LEI Encoded BulkData
image/jpeg	"jpg"	Single Frame Images or Individual Frames within a Multi-Frame
image/dicom-rle	"rle"	
image/jls	"jls"	
image/jp2	"jp2"	
image/jpx	"jpx"	
video/mpeg2	"mpg"	MPEG2

video/mp4	"mp4"	MPEG-4
application/pdf	"pdf"	Encapsulated PDF
application/xml+cda	"cda"	Encapsulated CDA

### 8.6.1.3.2 BulkData URI

BulkData URIs within the DICOM Metadata representations (see Section 10.4.1.1.2), for any BulkData Files included in the zip payload, shall be relative references according to [RFC3986] Section 4.

For relative BulkData URIs:

- the path shall not begin with a slash or two slashes
- the path shall not contain a backslash
- the path shall not refer to a higher level in the file system hierarchy (i.e., use of ".." is not permitted)
- file name extensions corresponding to executable file types (exe, dll, etc.) are not permitted
- embedded white space is not permitted

Note:

The following are examples of valid relative URI references:

"image1.json"

"series1/image3.dcm "

"/study1/series2/bulkdata1.raw"

"study1/series2/instance3.jpg" (instance3 is a single frame)

"study1/series2/instance4" (instance4 is a multi-frame directory)

The following are examples of invalid relative URI references:

"file:///matlist.mtl" (not a relative reference)

"/matlist.mtl" (no leading slashes)

"c:/matlist.mtl" (not a relative reference)

"../matlist.mtl" (no "..")

"setup.exe" (executable extension not permitted)

"mat list.mtl" (no embedded spaces)

### 8.6.1.3.3 Logical Format

The origin server shall format the zip payload according to [ISO/IEC 21320-1].

Note:

[ISO/IEC 21320-1] allows names to be I18N, so a variety of character sets may be encountered for file names.

### 8.6.1.3.4 Metadata Representations

When a Metadata representation includes Bulkdata references, the Metadata shall include File Meta Information Attributes within the Infoset.

**Update PS3.18 Section 8.7 as follows:**

---

120 **8.7 Media Types**

121 Media types are the basis for both content negotiation and data typing of message payloads. Each PS3.18 service, and/or transaction  
122 defines the media types and associated representations that are default, required and optional.

123 The media type also specifies whether the payload contains a single representation (single part), or multiple representations  
124 (multipart or zip). ~~Multipart~~ **Multiple representation** payloads are only defined for the RESTful APIs. See Section 8.6.1.2, **Section**  
125 **8.6.1.3** and Section 10.4.3.

126 ...

127

Update PS3.18 Section 8.7.1 as follows:

### 8.7.1 Multipart Media Types

Some of the services defined in this Part of the Standard support the multipart media types [RFC2387]. The syntax is:

```
multipart-media-type = "multipart" "/" subtype *(OWS ";" OWS parameter)
```

The application/multipart-related media type is used by the RESTful services. Its syntax is:

```
multipart-related = "multipart/related"
                    OWS ";" OWS "type" "=" DQ media-type DQ
                    OWS ";" OWS "boundary" "=" boundary
                    [related-parameters]
```

#### Where

boundary ; See Section 8.6.1.2.1

```
bchar = bchar-nospace / SP
bchar-nospace = DIGIT / ALPHA / "'" / "(" / ")" / "+" / "_" / "," / "-"
               / "." / "/" / ":" / "=" / "?" / "/" / ";" / "=" / "?"
related-parameters = [";" "start" "=" cid]
                    [";" "start-info" "=" cid-list]
cid-list = cid cid-list
cid = token / quoted-string
```

The "type" parameter is required. It contains the media type of the "root" body part. It always contains the special character "/" and thus requires quote marks.

The cid is a content identifier. It should be unique for each part of the multipart message.

**In a multipart payload, the content identifier for a part containing Bulkdata shall be the UUID of the BulkData.**

Typically, the "start" and "start-info" parameters are not specified, and the "root" is the first body part.

Update PS3.18 Section 8.7.3.5 as follows:

### 8.7.3.5 DICOM Media Type Syntax

The syntax of DICOM Media Types is:

```
dicom-media-type = (dcm-singlepart / dcm-multipart / dcm-zip) [dcm-parameters]
```

#### Where

dcm-singlepart = dcm-mt-name

dcm-multipart ;see Section 8.7.3.5.1

dcm-zip ;see Section 8.7.3.5.4



166 dcm-parameters = transfer-syntax-mtp ;see Section 8.7.3.5.2

167

168 / charset-mtp;see Section 8.7.3.5.3

169

170 dcm-mt-name = dicom / dicom-xml / dicom-json ;DICOM Media Type name

171 dicom = "application/dicom"

172 dicom-xml = "application/dicom+xml"

173 dicom-json = "application/dicom+json"

174 octet-stream = "application/octet-stream"

175 All DICOM Media Types may have a Transfer Syntax parameter, but its usage may be constrained by the service for which they are  
176 used.

177 Note

178 The application/dicom+xml and application/dicom+json Media Types may have a Transfer Syntax parameter in order to  
179 specify the encoding of base64 data.

180 All DICOM Media Types may have a character set parameter, but its usage may be constrained by the service for which they are  
181 used.

182

183 **Insert PS3.18 Section 8.7.3.5.4**

#### 184 8.7.3.5.4 ZIP Payload Media Types

185 The syntax of zip payload media type is:

186 dcm-zip = "application/zip"

187 [ OWS ";" OWS "type" "=" dcm-mp-mt-name ]

188 [dcm-parameters]

189 dcm-mp-mt-name ; See 8.7.3.5.1

190 Each ZIP media type in an accept header may include a "type" parameter that specifies the media type of the primary response files  
191 included in the ZIP payload.

192 The primary response files are the direct request objects for the request, typically for a WADO-RS request these will be the DICOM  
193 instances.

194 When a zip payload is returned, the DICOM instances shall be encoded in one of the specified type parameters associated with an  
195 accept header for application/zip, defaulting to "application/dicom" if none are applicable or none specified.

196 **Modify PS3.18 Section 8.11 as follows:**

#### 197 8.11 Security and Privacy

198 It is very likely that DICOM objects contain Protected Health Information. Privacy regulations in the United States (HIPAA), Europe  
199 (GDPR), and elsewhere, require that Individually Identifiable Information be kept private. It is the responsibility of those implementing  
200 and deploying the DICOM Standard to ensure that applicable regulations for security and privacy are satisfied.

201 See, for example, [ONC Privacy Security Guide].

12 **8.11.1 PS3.10 File Format Considerations**

13 The DICOM PS3.10 File Format has security considerations that will apply whenever DICOM PS3.10 File format is used. See  
14 Section 7.5 in PS3.10.

15 **8.11.2 ZIP Format Considerations**

16 **ZIP format is known to have vulnerabilities and to be the target of malware attacks.**

17 **Implementations that process the ZIP payload should utilize appropriate defenses and safeguards such as:**

- 18 - **Virus scanners for ZIP payload**
- 19 - **Sandbox execution and processing**
- 20 - **Full format and content validation**
- 21 - **Overflow detection**

22 **Clients that store the ZIP payload for later use by other systems should consider the environments of those systems.**  
23 **This means the scanning and validation should detect attacks against at least Windows, Macintosh, and Linux operating**  
24 **systems and applications.**

5 **Conformant ZIP payloads will not contain any directly or indirectly executable content. ZIP payload validation should**  
6 **include a test for any form of executable content and consider the detection of executable content to be an indication of**  
7 **malicious content. The presence of malicious content may indicate a security breach of the source server or other**  
8 **upstream system.**

9 **8.11.2.1 Encryption**

20 **DICOM does not specify encryption for a ZIP payload.**

21 **The ZIP format includes multiple optional encryption methods. Most of these methods are no longer accepted by**  
22 **regulatory agencies. FIPS 140 prohibits most ZIP encryption options. ISO/IEC 21320-1:2015 prohibits all ZIP encryption**  
23 **options.**

225 **Update PS3.18 Section 10.4.1.1.2 as follows:**

#### 10.4.1.1.2 Metadata Resources

Table 10.4.1-2 defines the resources used to retrieve the metadata contained in Instances.

**Table 10.4.1-2. Retrieve Transaction Metadata Resources**

Resource	URI Template
Study Metadata	/studies/{study}/metadata
Series Metadata	/studies/{study}/series/{series}/metadata
Instance Metadata	/studies/{study}/series/{series}/instances/{instance}/metadata

230 The Metadata Resources are used to retrieve the DICOM instances without retrieving Bulkdata. The Metadata returned for a study, series, or instance resource includes all Attributes in the resource. For Data Elements having a Value Representation (VR) of DS, FL, FD, IS, LT, OB, OD, OF, OL, OW, SL, SS, ST, UC, UL, UN, US, and UT, the origin server is permitted to replace the Value Field of the Data Element with a Bulkdata URI. The user agent can use the Bulkdata URI to retrieve the Bulkdata. **The Bulkdata URI shall be an HTTP(S) URI. Since the BulkData URI is present, the BulkData UUID will not be present.**

235

**Update PS3.18 Section 10.4.4 as follows:**

#### 10.4.4 Media Types

The origin server shall support the media types specified as default or required in Table 10.4.4-1.

**Table 10.4.4-1. Default, Required, and Optional Media Types**

240

Media Type	Usage	Section
application/dicom	Required	Section 8.7.3.1
application/dicom+json	Default	Section 8.7.3.2
multipart/related; type="application/dicom+xml"	Required	Section 8.7.3.2
Rendered Media Types	Optional	Section 8.7.4
<b><u>application/zip</u></b>	<b><u>Optional</u></b>	<b><u>Section 8.7.3.5.4</u></b>

The origin server shall support the Transfer Syntax and Character Set media type parameters. See Section 8.7.3.5.2 and Section 8.7.3.5.3.

**Update PS3.19 Table A.1.5-2 as follows:**

245

**Table A.1.5-2. DICOM Data Set Macro**

Name	Optionality	Cardinality	Description
...			
>BulkData	C	1	<p>A reference to a blob of data that the recipient may retrieve through use of the GetData() method, a WADO-RS call or a STOW-RS call.</p> <p>Required if the DICOM Data Element represented is not zero length and an XML Infoset Value, Item, InlineBinary or PersonName element is not present.</p> <p>The provider of the data may use a BulkData reference at its discretion to avoid encoding a large DICOM Value Field as text by value in the Infoset. For example, pixel data or look up tables.</p> <p>There is a single BulkData Infoset element representing the entire Value Field, and not one per Value in the case where the Value Multiplicity is greater than one.</p> <p>Note</p> <p>E.g., a LUT with 4096 16 bit entries that may be encoded in DICOM with a Value Representation of OW, with a VL of 8192 and a VM of 1, or a US VR with a VL of 8192 and a VM of 4096 would both be represented as a single BulkData element.</p> <p>All rules (e.g., byte ordering and swapping) in PS3.5 apply.</p> <p>Note</p> <p>Implementers should in particular pay attention the PS3.5 rules regarding the value representations of OD, OF, OL and OW.</p> <p>If the BulkData has a string or text Value Representation, the value(s) of the DICOM Specific Character Set Data Element, if present, might be necessary to determine its encoding.</p>
>>uuid	C	A	<p>An identifier of this bulk data reference formatted as a UUID using the hexadecimal representation defined in ITU-T Recommendation X.667.</p> <p>Required if BulkData URI is not present. Shall not be present otherwise.</p>
>>uri	C	A	<p>The <del>HTTP(S)</del> URI for this bulk data reference.</p> <p>Required if <b>BulkData UUID is not present. the NativeDicomModel was:</b></p> <ul style="list-style-type: none"> <li><b>returned in response to a WADO-RS Retrieve Metadata request</b></li> </ul> <p>Shall not be present otherwise.</p>
>InlineBinary	C	1	<p>The Value Field of the enclosing Attribute encoded as base64.</p>

Name	Optionality	Cardinality	Description
			<p>Required if the DICOM Data Element represented is:</p> <ul style="list-style-type: none"> <li>• not zero length</li> <li>• the VR if the enclosing Attribute is either OB, OD, OF, OW, or UN</li> <li>• an XML Infoset Value or BulkData XML element is not present</li> </ul> <p>Shall not be present otherwise.</p> <p>There is a single InlineBinary Infoset element representing the entire Value Field, and not one per Value in the case where the Value Multiplicity is greater than one.</p> <p>Note</p> <p>E.g., a LUT with 4096 16 bit entries that may be encoded in DICOM with a Value Representation of OW with a VL of 8192 and a VM of 1 would be represented as a single InlineBinary element.</p> <p>All rules (e.g., byte ordering and swapping) in PS3.5 apply.</p> <p>Note</p> <p>Implementers should in particular pay attention to the PS3.5 rules regarding the value representations of OD, OF, OL and OW.</p>

**Insert PS3.18 Appendix B.5 Example ZIP Payload Layout**

**B.6 Retrieve a ZIP Payload**

250 Retrieving a study which contains 3 series and 4 instances as a zip payload. It includes one image containing two frames encoded as JPEG, and some private data. The other image is a single frame JPEG LS image.

<http://www.medical-webservice.st/dicomweb/studies/1.2.3>

with HTTP headers

Accept: application/zip; type=application/dicom+json

255 Accept: application/zip; type=octet/stream

Accept: application/zip; type=image/jls

The following table shows an example zip payload layout. The file and folder naming are exemplary and not normative.

**Table B.6-1. Example ZIP Layout**

File Path in ZIP Payload	Content Description
--------------------------	---------------------

Series1/1.2.3.json	A DICOM file containing a JSON representation of SOP Instance UID 1.2.3. Contains a BulkData URI:  BulkData/image1.jls
Series1/1.2.4.json	A DICOM file containing a JSON representation of SOP Instance UID 1.2.4. Contains BulkData URIs:  1.2.4/images  1.2.4/privateFEA31015.raw
Series1/BulkData/image1.jls  Series1/1.2.4/images/1.jpg  Series1/1.2.4/images/2.jpg  Series1/1.2.4/privateFEA31015.raw	BulkData instances
Series2/encapsulatedPDF.json	A DICOM file contain an encapsulated PDF document.  The encapsulated PDF document is encoded as inline BulkData.
keyObject1.json	A Key Object, that was chosen by the implementing system to be stored at the root level.  Referencing BulkData ./keyObject1.raw
keyObject1.raw	The BulkData for keyObject1.

260

**Update PS3.18 F.2.2 as follows:**

### F.2.2 DICOM JSON Model Object Structure

...

265 • BulkDataURI: A string encoding the ~~WADO-RS~~ URL of a bulk data item describing the Value Field of an enclosing Attribute with a VR of DS, FL, FD, IS, LT, OB, OD, OF, OL, OV, OW, SL, SS, ST, SV, UC, UL, UN, US, UT or UV (described in Section F.2.6)

270 • **BulkDataUUID: A string encoding the multipart UUID of a bulk data item describing the Value Field of an enclosing Attribute with a VR of DS, FL, FD, IS, LT, OB, OD, OF, OL, OV, OW, SL, SS, ST, SV, UC, UL, UN, US, UT or UV (described in Section F.2.6)**

•

...

**Update PS3.18 F.2.6 as follows:**

---

### F.2.6 BulkDataURI and BulkDataUUID

275 If an attribute contains a "BulkDataURI", this contains the URI of a bulk data element as defined in Table A.1.5-2 in PS3.19

**If an attribute contains a "BulkDataUUID", this contains the UUID of a bulk data element as defined in Table A.1.5-2 in PS3.19. The bulk data element will be contained in the same multipart object that contains the referencing JSON.**

280