

5

10

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

*Supplement 194: RESTful Non-Patient Instance Storage*

15

20 **DICOM Standards Committee, Working Group 27: Web Technologies**

1300 N. 17th Street Suite 900

Rosslyn, Virginia 22209 USA

VERSION: Letter Ballot

Developed in accordance with work item 2015-12-C.

25

**Closed Issues**

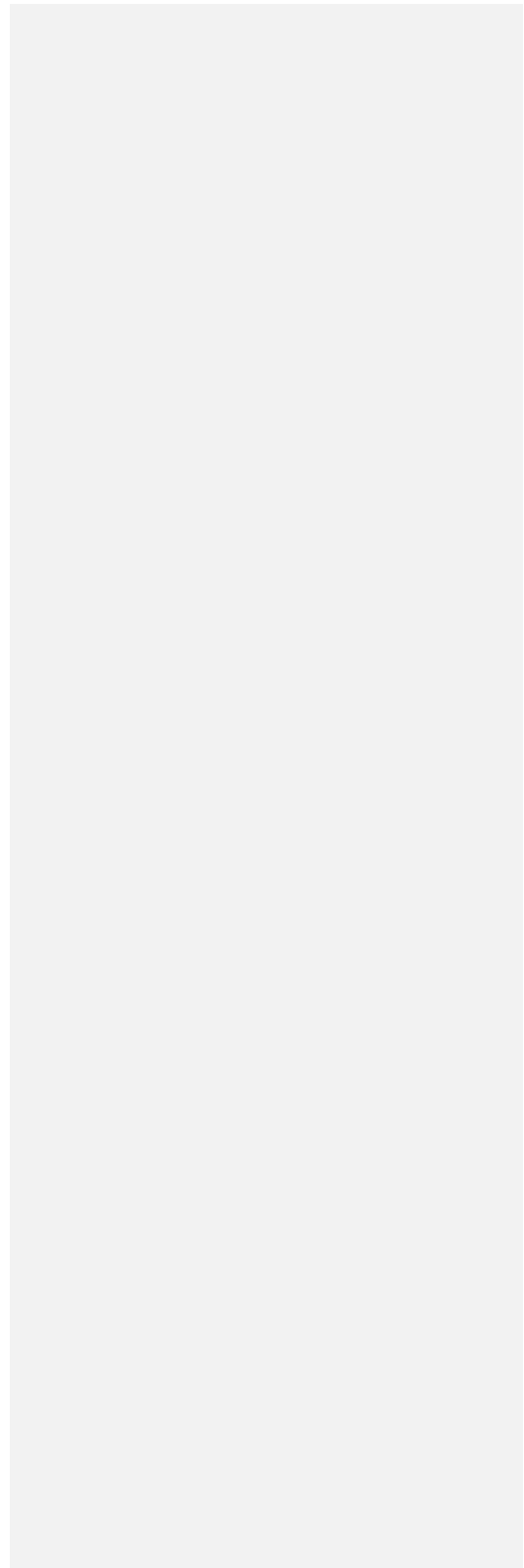
#	Description
1	Should the service allow retrieving multiple instances in the same transaction? <b>No</b> , for symmetry
2	Should the service all storing multiple instances in the same transaction? <b>Yes</b> , for symmetry.
3	Do we want accept and charset query parameters? <b>Yes</b> , for symmetry with other services.
4	Do we need to support any transfer syntax other than EVRLE? <b>Yes</b> , for symmetry (deflate).
5	Is the distinction between Metadata and Bulkdata necessary? <b>Yes</b> , for symmetry and because some instances could be large (e.g. implants).
6	Should any other IODs be included? <b>No</b> .
7	Do we want/need a notification API for create, retrieve, update, events? Maybe for some, such as procedure protocols (need longer lived notifications), or could use search? <b>No</b> , not yet.
8	Should this service support only DICOM Media Types? <b>Yes</b> .
9	Do any of these IODs have Transfer Syntaxes that are specific to them? <b>No</b> , except 'deflate' and EVRLE which applies to all of them.
10	Is there any reason to Search inside individual Instances? <b>No</b> .
11	Should all DICOM Media Types ('application/dicom', 'application/dicom+xml', 'application/dicom+json') be supported? <b>Yes</b> .
12	Should we define the attributes that must be supported for search or leave them unspecified? Should each NPI IOD define its own Search attributes? What does CP 1550 do? <b>No</b> . Use the attributes defined in PS3.4.
13	Should support for the Store transaction be required? <b>No</b> , for symmetry with Studies service
14	The sections on status codes will be updated as part of Supplement 183: Web Services Re-Documentation
15	Should support for 'application/dicom' media type be required. <b>Yes</b> .
16	Should there be one /implant-templates or three different roots? <b>Yes</b> .
17	Should the Information Model name be the root or should it have /npis root before the Information Model name. For example: /npis/color-palette, or /color-palette Each NPI resource should be its own root.
18	If any service implements Store should all services have to implement it. <b>No</b> .
19	Should ETag and Last-Modified header fields be required or optional? Suggested answer: required. Maybe only if Store is supported. Neither. They should be handled in a CP
20	Should we leave the term Status Details in the document and define it in a general sense? The Status Details CP is not ready, but this would require something similar as required by HTTP. <b>Yes</b> .

**Table of Contents**

30	1	Scope and Field of Application	5
	6.X	RS Non-Patient Instance (NPI) Storage Service	5
	6.X.1	Conformance	5

	6.X.2	Media Types	5
	6.X.3	Resources	5
35	6.X.4	General Query Parameters	6
	6.X.5	Transactions	6
	6.X.5.1	Retrieve Capabilities Transaction	7
	6.X.5.1.1	Request	7
	6.X.5.1.1.1	Resource	7
40	6.X.5.1.1.2	Query Parameters	7
	6.X.5.1.1.3	Request Header Fields	7
	6.X.5.1.1.4	Request Payload	7
	6.X.5.1.2	Behavior	7
	6.X.5.1.3	Response	7
45	6.X.5.1.3.1	Status Codes	7
	6.X.5.1.3.2	Response Header Fields	7
	6.X.5.1.3.3	Response Payload	8
	6.X.5.1.4	Media Types	8
	6.X.5.2.5	Conformance	8
50	6.X.5.2	Retrieve Transaction	8
	6.X.5.2.1	Request	8
	6.X.5.2.1.1	Resources	8
	6.X.5.2.1.2	Query Parameters	8
	6.X.5.2.1.3	Request Header Fields	8
55	6.X.5.2.1.4	Request Payload	8
	6.X.5.2.2	Behavior	9
	6.X.5.2.3	Response	9
	6.X.5.2.3.1	Status Codes	9
	6.X.5.2.3.2	Response Header Fields	9
60	6.X.5.2.3.3	Response Payload	9
	6.X.5.2.4	Media Types	9
	6.X.5.2.5	Conformance	9
	6.X.5.3	Store Transaction	10
	6.X.5.3.1	Request	10
65	6.X.5.3.1.1	Resources	10
	6.X.5.3.1.2	Query Parameters	10
	6.X.5.3.1.3	Request Header Fields	10
	6.X.5.3.1.4	Request Payload	10
	6.X.5.3.2	Behavior	10
70	6.X.5.3.3	Response	10
	6.X.5.3.3.1	Status Codes	11
	6.X.5.3.3.2	Response Header Fields	11
	6.X.5.3.3.3	Response Payload	11
	6.X.5.3.4	Media Types	11
75	6.X.5.3.5	Conformance	11
	6.X.5.4	Search Transaction	12
	6.X.5.4.1	Request	12
	6.X.5.4.1.1	Resources	12
	6.X.5.4.1.2	Query Parameters	12
80	6.X.5.4.1.3	Request Header Fields	12
	6.X.5.4.1.4	Request Payload	12
	6.X.5.4.2	Behavior	13
	6.X.5.4.3	Response	13
	6.X.5.4.3.1	Status Codes	13
85	6.X.5.4.3.2	Response Header Fields	13

6.X.5.4.3.3	Response Payload	13
6.X.5.4.4	Media Types	13
6.X.5.4.5	Conformance	13



# 1 Scope and Field of Application

This supplement defines Restful Services (RS) for retrieving, storing, and searching for non-patient related IODs such as hanging protocols, color palettes, procedure protocols, etc.

95 The transactions defined for this service are very similar to those defined for the RS Studies Service. They allow a user agent to retrieve, store, and search for non-patient related IODs from an origin server in DICOM Media Types.

Security is beyond the scope of the RESTful services defined in this supplement. However, generic Web security mechanisms are fully compatible.

Add the following section to Part 18, Section 6:

## 100 6.X RS Non-Patient Instance (NPI) Storage Service

The RS Non-Patient Instance (NPI) Storage Services define a set of RESTful transactions that enable a user agent to retrieve, store, and search an origin server for non-patient related instances in DICOM Media Types.

### 6.X.1 Conformance

105 An implementation conforming to the RS Non-Patient Instance Storage Service shall specify all NPI resources it supports.

It shall also document all supported transactions, and whether it plays the role of origin server or user agent, or both.

For each transaction supported, an implementation shall document in its Conformance Statement the resources supported (see PS3.4), and any other conformance requirements defined by the transaction.

110 Implementations conforming to the RS Non-Patient Instances Storage Service shall support the Retrieve Capabilities, Retrieve, and Search transactions for all supported NPI resources. Implementations may support the Store transaction.

### 6.X.2 Media Types

For the Retrieve Capabilities transaction see Section 6.8.

For all other NPI transactions the origin server shall support DICOM Media Types. See Section 6.1.1.8.

115 **Table 6.X.2-1: Default, Required, and Optional Media Types**

Media Type	Usage
application/dicom	Required
application/dicom+json	Default
application/dicom+xml	Optional

### 6.X.3 Resources

An NPI Service manages one or more collections of NPI resources. Each collection contains Instances from one IOD. The target resource URIs have the following templates:

120 /  
 /{type}  
 /{type}/{uid}

Where

125 type = "color-palettes"  
 =/ "defined-procedure-protocols"  
 =/ "hanging-protocols"  
 =/ "implant-template" "/" "generic"

130      =/ "implant-template" "/" "assembly"  
           =/ "implant-template" "/" "groups"  
           uid           ; is the Unique Identifier of an NPI Instance

The NPI Resource Name is the name of a collection resource that corresponds to a single IOD.

Table 6.X.3-1 contains the names and templates for the NPI resources. It also includes the PS3.3 Section in which the corresponding IOD is defined.

**Table 6.X.3-1: Resources, URI Templates and Descriptions**

Resource Name	URI Template and Description	IOD	Storage Class	Information Model
Service	/			
Color Palette	/color-palettes{/uid}	PS3.3, Annex A.58	PS3.4, Annex XX	PS3.4, Annex X.1.3
Defined Procedure Protocol	/defined-procedure-protocols{/uid}	Sup 121 (PS3.3, Annex A.X1.2)	Sup 121 PS.3.4, Annex XX	Sup121 (PS3.4, Annex BB.6.1)
Hanging Protocol	/hanging-protocols{/uid}	PS3.3, Annex A.44	PS3.4, Annex XX	PS3.4, Annex U.1.3
Implant Template	/implant-template/{type}/{uid}	PS3.3, Annex A.61	PS3.4, Annex XX	PS3.4, Annex BB.1.3
Generic Implant Template	/implant-template/assembly{/uid}			
Implant Assembly Template	/implant-template/generic{/uid}			
Implant Template Group	/implant-template/group{/uid}			

Commented [JP1]: Update when resources are final

135      **6.X.4      General Query Parameters**

All NPI transactions support the Accept (see Section 6.1.1.5) and Character Set (see Section 6.1.2.2) Query Parameters.

**6.X.5      Transactions**

The NPI Service defines the transactions specified in the following table:

**Table 6.X.5-1: NPI Service Transactions**

140

Transaction	Method	Resource	Payload		Description
			Request	Response	
Retrieve Capabilities	OPTIONS	/	N/A	Capabilities Description	Retrieves a description of the capabilities of the NPI Service, including transactions, resources, query parameters, etc.
Retrieve	GET	{type}/{uid}	N/A	Instance and/or Status Details	Retrieves an Instance, specified by the target resource in an Acceptable DICOM Media Type.
Store	POST	/npis/{type}/{uid}	Instance(s)	Status Details	Stores one or more DICOM Instances, contained in the request payload, in the location referenced by the target resource URL.
Search	GET	/npis/{type} ?{params*}	N/A	Result(s) and/or Status Details	Searches the target resource for Instances that match the search parameters and returns a list of matches in an Acceptable Media Type.

Table 6.X.5-2 shows the target resources permitted for each transaction.

**Table 6.X.5-2 Resources by Transaction**

Resource	URI	Retrieve	Store	Search	Capabilities
NPI Service	/				X
All Instances	/{type}		X	X	

Instance	/{type} / {uid}	X	X		
----------	-----------------	---	---	--	--

### 6.X.5.1 Retrieve Capabilities Transaction

145 The Retrieve Capabilities transaction retrieves a machine-readable description of the NPI service implemented by an origin server. The response contains a machine-readable Capabilities Description document. The Capabilities Description document describes the transactions, resources, representations, etc. that are supported by the service(s).

#### 6.X.5.1.1 Request

150 The Retrieve Capabilities request uses the OPTIONS method and has the following format:

```
OPTIONS SP / SP version CRLF
Accept: 1#media-type CRLF
*(header-field CRLF)
CRLF
```

#### 6.X.5.1.1.1 Resource

The target resource for this transaction is the Base URI ("/") of the service.

#### 6.X.5.1.1.2 Query Parameters

See Section 6.X.4.

#### 6.X.5.1.1.3 Request Header Fields

160 Table 6.X.5-3 shows the most common Mandatory, Conditional, and Optional header fields for this transaction.

Table 6.X.5-3: Request Header Fields

Header Fields	Value	Usage	Requirements
Accept	media-range	M	
Accept-Charset	1#charset	O	

#### 6.X.5.1.1.4 Request Payload

The request has no payload.

#### 6.X.5.1.2 Behavior

165 The origin server shall return a machine-readable description of its capabilities in an Acceptable Media Type.

#### 6.X.5.1.3 Response

The format of the response is as follows:

```
version SP status-code SP reason-phrase CRLF
Content-Type: media-type CRLF
*(header-field CRLF)
CRLF
payload
```

#### 6.X.5.1.3.1 Status Codes

A success response shall have a status code of 200 (OK).

175 A failure response shall have a 400 or 500 level status code.

#### 6.X.5.1.3.2 Response Header Fields

Table 6.X.5-4: Response Header Fields

Header Field	Value	Usage	Requirements
Content-Type	media-type	M	
Content-Length	uint	C	Required if no transfer coding has been applied. Shall not be present otherwise.

Transfer-Encoding	encoding	C	Required if a transfer coding has been applied. Shall not be present otherwise.
-------------------	----------	---	---

### 6.X.5.1.3.3 Response Payload

180 A success response shall have a payload containing a Capabilities Description document in the Selected Media Type. The Capabilities Description document shall describe the service in as much detail as possible.

A failure response shall have a payload describing the error.

### 6.X.5.1.4 Media Types

See Section 6.X.2.

### 185 6.X.5.2.5 Conformance

An implementation of an NPIS Service shall support the Retrieve Capabilities transaction.

The response to the Retrieve Capabilities transaction and the Conformance Statement shall document:

- Implementations role: origin server, user agent, or both
- Supported IODs for each role
- 190 • Supported DICOM Media Types
- Supported character sets (if other than UTF-8)

## 6.X.5.2 Retrieve Transaction

195 The Retrieve DICOM transaction retrieves the target resource in a DICOM Media Type. If non-DICOM Media Types are present in the Acceptable Media Types of the request, the origin server shall return a 400 (Bad Request) response with an appropriate Status Details document in the payload.

Commented [JP2]: Keep or remove?

### 6.X.5.2.1 Request

The Retrieve request has the following syntax:

```
200 GET SP /{type}/{uid} SP version CRLF
Accept: 1#dicom-media-type CRLF
[If-None-Match: entity-tag CRLF]
*(header-field CRLF)
CRLF
```

### 6.X.5.2.1.1 Resources

Table 6.X.5-5 shows the resources and URI Templates supported by the Retrieve DICOM transaction.

205 An origin server shall specify all supported resources in its conformance statement and in its response to the Retrieve Capabilities transaction.

Table 6.X.5-5: Resources and URI Templates

Resource	URI Template
Instance	/ {type} / {uid}

### 6.X.5.2.1.2 Query Parameters

See Section 6.X.4.

### 210 6.X.5.2.1.3 Request Header Fields

Table 6.X.5-6 shows the most common Mandatory, Conditional, and Optional header fields for this transaction.

Table 6.X.5-6: Request Header Fields

Header Fields	Value	Usage	Requirements
Accept	media-range	M	
Accept-Charset	1#charset	O	

### 6.X.5.2.1.4 Request Payload



The request shall have no payload.

### 215 6.X.5.2.2 Behavior

The origin server locates the target resource and returns it in an Acceptable DICOM Media Type. If the resource cannot be located, an appropriate Status Details response shall be returned.

### 6.X.5.2.3 Response

The response has the following syntax:

220 version SP status-code SP reason-phrase CRLF  
 Content-Type: dicom-media-type CRLF  
 \*(header-field CRLF)  
 CRLF  
 payload

### 225 6.X.5.2.3.1 Status Codes

The response shall have an appropriate status code. Table 6.X.5-7 contains the most common status codes for this transaction.

**Table 6.X.5-7: Status Codes**

Status	Code	Phrase	Description
Success	200	OK	Indicates that all instances were successfully retrieved.
	304	Not Modified	Indicates that the user agent's current representation is up to date, so no payload was returned. This status code shall only be returned for a Conditional Retrieve request containing an If-None-Match header field.
Failure	See [RFC7231, Section 6< <a href="https://tools.ietf.org/html/rfc7231#section-6">https://tools.ietf.org/html/rfc7231#section-6</a> >]		

### 6.X.5.2.3.2 Response Header Fields

230 **Table 6.X.5-8: Request Header Fields**

Header Field	Value	Usage	Requirements
Content-Type	dicom-media-type	M	
Content-Length	uint	C	Required if no transfer coding has been applied. Shall not be present otherwise.
Transfer-Encoding	encoding	C	Required if a transfer coding has been applied. Shall not be present otherwise.

### 6.X.5.2.3.3 Response Payload

A success response shall have a payload containing the DICOM instances specified by the target resource in an Acceptable DICOM Media Type. The payload may be single part or multipart depending on the media type.

235 If there are any errors or warnings encountered by the origin server, the response shall contain a Status Details document describing them.

### 6.X.5.2.4 Media Types

See Section 6.X.2.

### 6.X.5.2.5 Conformance

An implementation of an NPIS Service shall support the Retrieve Capabilities transaction.

240 The response to the Retrieve Capabilities transaction and the Conformance Statement shall document:

- Implementations role: origin server, user agent, or both
- Supported IODs for each role
- Supported DICOM Media Types
- Supported character sets (if other than UTF-8)

### 245 6.X.5.3 Store Transaction

This transaction requests that the origin server store the representations of the NPIs contained in the request payload so that they may be retrieved in the future using the Instance UIDs.

The Store Service only supports DICOM Media Types.

#### 6.X.5.3.1 Request

250 Transactions in this service use the POST method. The request syntax is:

```
POST SP /{type} {/uid} SP version CRLF
Content-Type: dicom-media-type CRLF
*(header-field CRLF)
CRLF
255 payload
```

#### 6.X.5.3.1.1 Resources

The target URL shall reference either the All Instances or Instance resource. Table 6.X.5-9 shows the resources and URI Templates supported by the Store DICOM transaction.

Table 6.X.5-9: Resources and URI Templates

Resource	URI Template	Description
All Instances	/{type}	Stores representations of a set of Instances.
Instance	/{type} / {uid}	Stores a representation of a single Instance with a UID equal to uid.

#### 260 6.X.5.3.1.2 Query Parameters

See Section 6.X.4.

#### 6.X.5.3.1.3 Request Header Fields

Table 6.X.5-10: Store Request Header Fields

Header Field	Value	Usage	Requirements
Content-Type	dicom-media-type	M	
Accept	dicom-media-type	M	
Content-Length	uint	C	Required if no transfer coding has been applied. Shall not be present otherwise.
Transfer-Encoding	encoding	C	Required if a transfer coding has been applied. Shall not be present otherwise.

#### 6.X.5.3.1.4 Request Payload

265 The request payload shall be present and shall contain one or more representations in the DICOM Media Type specified by the Content-Type header field.

#### 6.X.5.3.2 Behavior

The origin server stores the representations contained in the request payload so that they may be retrieved later using the Retrieve DICOM transaction.

270 Before storing the representations, the origin server may coerce the values of data elements.

If any element is coerced, the Original Attribute Sequence (0400,0561) (see PS3.3, Section C.12.1) shall be included in the stored DICOM instances, and the response shall describe the modifications.

#### 6.X.5.3.3 Response

The response shall have the following syntax:

```
275 version SP status-code SP reason-phrase CRLF
*(header-field CRLF)
CRLF
payload
```

280 The response shall contain an appropriate status code and a payload containing a Status Details document in an Acceptable Media Type. The Status Details document shall describe the success, warning, or failure status for each Instance contained in the request payload.

#### 6.X.5.3.3.1 Status Codes

The response shall have an appropriate status code. Table 6.X.5-11 contains the most common status codes for this transaction.

285

**Table 6.X.5-11: Common Status Codes**

Status Code	Description
200 (OK)	Indicates that the origin server has successfully received, processed, and stored at least one of the representations contained in the request payload. The response shall include a Status Details document detailing the disposition of all representations contained in the request.
202 (Accepted)	Indicates that the origin server successfully received and validated the request message, but may not have validated, processed, or stored the representations in the request payload. The response shall include a Status Details document detailing the disposition of all representations contained in the request. The user agent can use a Query or Retrieve transaction at a later time to determine if the request has completed. Alternatively, if the Service supports notifications, and the user agent has a subscription for newly created resources, it will receive an event notification when the transaction has completed.
409 (Conflict)	Indicates that the origin server did not store any of the representations contained in the request payload because of representation specific errors. For example, unsupported IOD, or Study Instance UID mismatch.
415 (Unsupported Media Type)	Indicates that the origin server does not support the media type specified in the Content-Type header field of the request, and none of the representations contained in the request were processed or stored

#### 6.X.5.3.3.2 Response Header Fields

**Table 6.X.5-12: Store Response Header Fields**

Header Field	Value	Usage	Requirements
Content-Type	media-type	M	
Content-Length	uint	C	Required if no transfer coding has been applied. Shall not be present otherwise.
Transfer-Encoding	encoding	C	Required if a transfer coding has been applied. Shall not be present otherwise.
Warning	text	C	If there are any conditions that a user agent should be aware of, it is recommended that a DICOM Status Code and descriptive reason as defined in Section 6.6.1.3.2.1.

#### 6.X.5.3.3.3 Response Payload

290 The response payload shall contain a description of any additions, modifications, or deletions to the stored representations.

#### 6.X.5.3.4 Media Types

See Section 6.X.2.

#### 6.X.5.3.5 Conformance

295 If an implementation supports the Store transaction, the response to the Retrieve Capabilities transaction and the Conformance Statement shall document:

- Implementations role: origin server, user agent, or both
- Supported IODs for each role
- Supported DICOM Media Types
- Supported character sets (if other than UTF-8)

300 **6.X.5.4 Search Transaction**

The Search transaction searches the collection of NPI Instances contained in the target resource. A successful response shall contain details of the matching Instances. The search criteria are specified in the query parameters. Each match includes the default and requested attributes from the matching Instance.

**6.X.5.4.1 Request**

305 The Search service uses the GET method and has the following syntax:

```
GET SP /{type} {?parameter*} SP version CRLF
Accept: 1#dicom-media-type CRLF
*(header-field CRLF)
CRLF
```

310 **6.X.5.4.1.1 Resources**

The target URI shall reference the All Instances resource. Table 6.X.5-11 shows the resources and URI Templates supported by the Search transaction. An origin server that supports the Search transaction shall support the resources specified in Table 6.X.5-13.

**Table 6.X.5-13: Resources and URI Templates**

Resource	URI Template	Description
All Instances	/{type}	Searches the set of resources with different Instance UIDs.

315 **6.X.5.4.1.2 Query Parameters**

The search transaction supports the Query Parameters defined in Section 6.X.4. It also supports the search parameters defined in Section 6.7.1.1.

The search parameters in the query component of the target URL specify the matching criteria, the attribute values to be returned, and the results to be returned. The URI template for the query parameters is:

320 {?search\*} = “?” {&match\*} {&include\*} {&offset} {&limit}

**6.X.5.4.1.2.1 Attributes and Behaviors**

For any Resource Names the origin server supports, it shall support the behaviors and matching key attributes as specified in the corresponding Information Model sections in Table 6.X.5-14.

**Table 6.X.5-14: NPI Resource Search Attributes**

Resource Name	Information Model
Color Palette	PS3.4, Annex X.1.3
Defined Procedure Protocol	Sup121 (PS3.4,Annex BB.6.1)
Hanging Protocol	PS3.4, Annex U.1.3
Implant Template	PS3.4, Annex BB.1.3
Generic Implant Template	
Implant Assembly Template	
Implant Template Group	

**Commented [JP3]:** Add rest of implant templates when decision on resource names is made.

325 Each implementation shall specify any additional search parameters, attributes, and behaviors it supports in its conformance statement and in its response to the Retrieve Capabilities transaction.

**6.X.5.4.1.3 Request Header Fields**

**Table 6.X.5-15: Search Request Header Fields**

Header Field	Value	Usage	Requirements
Accept	dicom-media-type	M	

**6.X.5.4.1.4 Request Payload**

330 The request has no payload.

**6.X.5.4.2 Behavior**

The origin server shall perform the search indicated by the request, using the matching behavior specified in the corresponding Information Model, and return a response containing the search results, or an appropriate failure response.

335 The matching behavior and paging rules are specified in Section 6.7.1.2.1.

**6.X.5.4.3 Response**

A success response shall have a status code of 200 (OK) and a payload containing the search results in the Selected Media Type.

A failure response should contain a payload describing the error(s) encountered.

**340 6.X.5.4.3.1 Status Codes**

The response shall have an appropriate status code. Table 6.X.5-16 contains the most common status codes for this transaction.

**Table 6.X.5-16: Common Status Codes**

Status Code	Description
200 (OK)	Indicates that the origin server has successfully received, search, and found at least one of the resources matching the request.
204 (No Content)	Indicates that the origin server has successfully received, search, and found no resources matching the request.
409 (Conflict)	Indicates that the origin server did not return any search results because the request was not well formed.
415 (Unsupported Media Type)	Indicates that the origin server does not support any of the Acceptable Media Types.

**345 6.X.5.4.3.2 Response Header Fields****Table 6.X.5.17: Search Response Header Fields**

Header Field	Value	Usage	Requirement
Content-Type	media-type	M	
Content-Length	uint	C	Required if no transfer coding has been applied. Shall not be present otherwise.
Transfer-Encoding	encoding	C	Required if a transfer coding has been applied. Shall not be present otherwise.

**6.X.5.4.3.3 Response Payload**

The response payload will contain the search results in the Selected Media Type.

350 The origin server shall support the behaviors and return attributes as specified in the corresponding Information Model sections in Table 6.X.5-13.

**6.X.5.4.4 Media Types**

See Section 6.X.2.

**6.X.5.4.5 Conformance**

An implementation of an NPIS Service shall support the Search transaction.

355 The response to the Retrieve Capabilities Transaction and the Conformance Statement shall document whether the implementation plays the role of origin server, user agent, or both.

The Conformance Statement for an origin server shall include the following:

- The supported resources
- Optional Attributes supported

- Any additional behaviors

