Digital Imaging and Communications in Medicine (DICOM)

Supplement 246: DICOMweb Modality Workflow Services

Prepared by:

DICOM Standards Committee, Working Group 27

1300 N. 17th Street, Suite 900

Rosslyn, Virginia 22209 USA

Status: March 2025, Public Comment

Developed pursuant to DICOM Work Item 2023-10-C

Table of Contents

Document History

2024.08	Version 00	JM, DK	Initial version with proposed document structure and content.
2024.11	Version 01	JM, DK	Extended with observations; alternative approach included.
2025.01	Version 02	JM, DK	Alternative approach promoted, extended with conformance.
2025.01	Version 03	JM	Split into two services; reworked comments from WG06.
2025.01	Version 04	DK	Added several examples.
2025.03	Version 05	JM, DK	Added diagrams for bi-directional proxies. More examples. Reworked comments from WG06.

Open Issues

2	**Context**: The notified parties of the MPPS notification service as specified in PS3.4, F.9. **Issue**: This service does not specify how the MPPS Notification SCP knows what SCUs to notify on MPPS changes. **Proposal**: Do not change the current way this behavior is specified; just add one or more notes to make clear that this aspect is something beyond the standard, and that a conceivable way to achieve this would be configuring the SCP with the SCUs to be notified. **Decision**: [WGxx: YYYY-MM-DD] None yet.
3	**Context**: MPPS notifications. **Issue**: In HTTP there is no way for an origin server to open a connection to a user agent. Therefore, MPPS notifications as present in DIMSE cannot be mimicked in DICOMweb. There are several ways to deal with this in DICOMweb: 1. Do not allow for DICMweb MPPS notifications at all; 2. Have user agents always open a WebSocket pipeline as defined in Section 8.10, without knowing whether they will be notified or not (see issue 2); 3. Use a subscription mechanism like is done in UPS(-RS), including global subscriptions (applicable for all performed procedures, not only specific ones). 4. Use HTTP/2 Server Push mechanism (suggested in a WG27 meeting). 5. Do not allow for DICOMweb MPPS notifications at all but instead elaborate on the pattern used in IHE's Scheduled Workflow integration profile, where an intermediate party (broker-like) forwards MPPS updates to interested parties; ensure that such behavior is mentioned in the conformance statement. **Proposal**: Go for the fifth option, where an MPPS origin server can be a MPPS user agent when forwarding MPPS updates when received. **Decision**: [WGxx, YYYY-MM-DD] None yet.
4	**Context**: HTTP method for updating an MPPS. **Issue**: Updating an MPPS, i.e. making a partial change to it, requires a DICOMweb transaction with an HTTP method. The HTTP Patch method is a request for making partial changes to an existing resource and therefore seems most appropriate. However, the approach in DICOMweb for updating a resource is set by UPS-RS, and this utilizes the POST method for making an update to a Workitem (PS3.18, Section 11.6.1) and introduces a new resource for changing a Workitem's state which uses PUT for that (PS3.18, 11.7.1). Using PATCH for MPPS updates would therefore be against the approach used in UPS-RS but would be in line with HTTP semantics. **Proposal**: Go for using the PATCH method for MPPS updates and create a separate CP for changing UPS-RS' approach to keep DICOMweb architecturally consistent. Even though

Commented [JM1]: 5. Mirroring/forwarding as in IHE SWF is an option. Make clear whether you do, as a server, in the conformance statement.

Commented [JM2]: Also for global subscriptions like in UPS.

	the latter would require breaking changes, no UPS-RS implementations are known, so no harm is done. Furthermore, using PATCH will align DICOMweb's transactions with HTTP semantics. **Decision**: [WGxx, YYYY-MM-DD] None yet.
5	**Context**: Partial updates to MPPS sequences. **Issue**: DIMSE does not allow for partial updates of sequences within an MPPS (PS3.4, F7.2.2.2). With modalities that create Series that can contain over 100K instances and that may update the MPPS regularly (multiple N-SETs), this may be wasting bandwidth considerably, as each time the entire Referenced Image Sequence within the Performed Series Sequence is to be sent, including all that was already sent before. Another approach to this could be diverting from DIMSE's MPPS N-SET semantics and allowing for updates of this and similar sequences in the MPPS. As the MPPS N-SET was created in a time where Series would never have so many instances, it was not a big problem to send over the same information repeatedly. However, this has changed. **Proposal**: Although the concern is valid, it is not clear whether there are many occurrences of modalities that update the MPPS *repeatedly*, even though there could be merit in doing so, for instance for showing progress. It is therefore proposed to not include this in this supplement and write a CP for this behavior when needed, including DIMSE. **Decision**: [WGxx, YYYY-MM-DD] None yet.
6	**Context**: The name of the service returning modality scheduled procedure steps. **Issue**: In DIMSE this service is formally called the Basic Worklist Management Service (see PS3.4, Annex K) and is colloquially called the Modality Worklist Service. However, it might be clearer when its name shows that it is the counterpart of the Modality Performed Procedure Step Service. **Proposal**: Baptize this DICOMweb service the Modality Scheduled Procedure Step Service. **Decision**: [WGxx, YYYY-MM-DD] None yet.

Closed Issues

1	**Context**: The description of the work item proposal talked about adding the Modality Worklist and the Modality Performed Procedure Step services to DICOMweb, in principle based on the existing DICOMweb Worklist service. This was expected to boil down to creating an informative annex and any normative changes needed if gaps are discovered. **Issue**: It proved very hard, if not impossible, to map MWL/MPPS to UPS, as, among other things, the two serve different purposes. **Proposal**: Create new Modality Workflow Services and Resources based on the MWL/MPPS DIMSE model instead of basing them on UPS-RS. **Decision**: [WG06: 2025-01-14] Agreed with proposal.

Scope and Field of Application

This supplement defines the means to perform modality workflow management in DICOMweb. Modality workflow services enable a user agent to use and create workflow-related resources on an origin server. They are an extension to the existing DICOMweb services, providing RESTful interfaces to the Modality Worklist (MWL) and Modality Performed Procedure Step (MPPS) services that are already available in DIMSE. The modality workflow services have been designed with the intention of facilitating proxies from/to DIMSE.

Changes to NEMA Standards Publications PS 3.18

Add new section Y Modality Scheduled Procedure Step Service and Resources, immediately before section X below

Y Modality Scheduled Procedure Step Service and Resources

Y.1 Overview

The Modality Scheduled Procedure Step Service enables a user agent to search for Scheduled Procedure Steps, and entities related to these steps, intended to be performed on an imaging modality. It corresponds to the DIMSE Modality Worklist (MWL) service as defined in Annex K of PS3.4 and has the same semantics.

Y.1.1 Resource Descriptions

The Modality Scheduled Procedure Step Service provides access to a collection of Modality Scheduled Procedure Steps, defined as the resource given in Table Y.1.1-1.

Table Y.1.1-1. Resources, URI Templates and Descriptions

Resource	URI Template	Description
Modality Scheduled Procedure Steps	/modality-scheduled-procedure-steps	The collection of Modality Scheduled Procedure Steps managed by the origin server.

Y.1.2 Common Query Parameters

The origin server shall support Query Parameters as required in Table Y.1.2-1.

The user agent shall supply in the request Query Parameters as required in Table Y.1.2-1.

Table Y.1.2-1. Common Query Parameters

Name	Value	Usage		Section
		User Agent	Origin Server	
Accept	media-type	O	M	Section 8.3.3.1
Accept-Charset	charset	O	M	Section 8.3.3.2

Y.1.3 Common Media Types

The origin server shall support the media types specified as Default or Required in Table Y.1.3-1.

Table Y.1.3-1. Default, Required, and Optional Media Types

Media Type	Usage	Section
application/dicom+json	Default	Section 8.7.3.2
application/dicom+xml	Required	Section 8.7.3.2
multipart/related; type="application/dicom+json"	Required	Section 8.7.3.2
multipart/related; type="application/dicom+xml"	Required	Section 8.7.3.2

Y.2 Conformance

An origin server conforming to the Modality Scheduled Procedure Step Service shall support the Retrieve Capabilities Transaction (see Section 8.9.1).

An origin server conforming to the Modality Scheduled Procedure Step Service shall support the Transactions listed as Required in Table Y.2-1 and may support Transactions listed as Optional.

Table Y.2-1. Required and Optional Transactions

Transaction	Support	Section
Retrieve Capabilities	Required	Section 8.9
Search	Required	Section Y.4

Implementations shall specify in their Conformance Statement (see PS3.2) and the Retrieve Capabilities Transaction the supported Transactions and the implementations' role: origin server, user agent, or both.

In addition, for each supported Transaction they shall specify:

- The supported Query Parameters, including optional Attributes, if any.
- The supported DICOM Media Types.
- The supported character sets (if other than UTF-8).

Y.3 Transactions Overview

The Modality Scheduled Procedure Step Service consists of the Transactions listed in Table Y.3-1.

Table Y.3-1. Modality Scheduled Procedure Step Service Transactions

Transaction Name	Method	Payload		Description
		Request	Success Response	
Search	GET	none	dataset according to PS3.4, Table K.6-1	Searches for Modality Scheduled Procedure Steps

Table Y.3-2 lists the Modality Scheduled Procedure Step Service Transactions and their corresponding DIMSE Operations used in MWL.

Table Y.3-2. Mapping of Modality Scheduled Procedure Step Service Transactions and DIMSE Operations

Transaction	Operation	Reference	DIMSE Service
Search	Query Worklist	PS3.4, K.4	C-FIND

Note As in DIMSE, the Transactions do *not* provide a complete CRUDL interface for the respective resource. For instance, it is not possible to create Modality Scheduled Procedure Steps using DICOM, neither with DIMSE, nor with DICOMweb. What DICOM *does* provide is access to scheduled procedure steps at the level required for modalities.

Y.4 Search Transaction

This Transaction searches the Modality Scheduled Procedure Steps for scheduled procedure steps that match the specified Query Parameters and returns a list of matching scheduled procedure steps. Each scheduled procedure step in the returned list includes return Attributes specified in the request. The Transaction corresponds to the DIMSE MWL C-FIND Operation (see PS3.4, Section K.4.1).

Commented [JM3]: Flip from right to left.

Commented [JM4R3]: Possibly create CPs for adding this kind of tables sections 10 and 11 too.

Commented [JM5R3]: Possibly move to X.3, with next table, below overview table.

Commented [JM6R3]: Table has been flipped and moved. Still open: CP.

Y.4.1 Request

The request shall have the following syntax:

```
GET SP /modality-scheduled-procedure-steps?{&match*}{&includefield}{&fuzzymatching}{&offset}{&limit} SP version CRLF
Accept: 1#media-type CRLF
*(header-field CRLF)
CRLF
```

Y.4.1.1 Target Resources

The Target Resource for this Transaction is the Modality Scheduled Procedure Steps.

Y.4.1.2 Query Parameters

The origin server shall support Query Parameters as required in Table 8.3.4-1.

The user agent shall supply in the request Query Parameters as required in Table 8.3.4-1.

Y.4.1.3 Request Header Fields

The origin server shall support header fields as required in Table X.4.1-1.

The user agent shall supply in the request header fields as defined in Table X.4.1-1.

Table X.4.1-1. Request Header Fields

Name	Values	Usage		Description
		User Agent	Origin Server	
Accept	media-type	M	M	The Acceptable Media Types of the response payload.

See also Section 8.4.

Y.4.1.4 Request Payload

The request shall have no payload.

Y.4.2 Behavior

The origin server shall perform a search according to the requirements specified in Section 8.3.4.

For each matching modality scheduled procedure step, the origin server shall include in the results:

- All Attributes in Table K.6-1 “Attributes for the Modality Worklist Information Model” in PS3.4 with a Return Key Type of 1 or 2.
- All Attributes in Table K.6-1 “Attributes for the Modality Worklist Information Model” in PS3.4 with a Return Key Type of 1C or 2C for which the conditional requirements are met.
- All other Attributes passed as match parameters that are supported by the origin server as either matching or return Attributes.
- All other Attributes passed as includefield parameter values that are supported by the origin server as return Attributes.

Y.4.3 Response

The response shall have the following syntax:

```
version SP status-code SP reason-phrase CRLF
CRLF
[payload]
```

Commented [JM7]: Q for WG06: What about 2C? (The text in this section has been copied and adapted from 11.9.2, and there 2C is not mentioned either. That is peculiar.)

Commented [JM8R7]: Go ahead, but see whether there are JSON issues with empty sequences. WG20. Annex F.

Commented [JM9R7]: As far as I can see there are no issues in JSON with empty Sequences. When stripping the Sequence example in Table F.3.1-1 I get:

“ggggeeee” : { “vr” : “SQ”
, “Value” : [ ]
}

which is perfectly alright, not mapping an empty Sequence to *null* (which would be wrong). Is also states in F.2.2 that empty sequence items are represented by empty objects, which maps to the above.

104 **Y.4.3.1 Status Codes**

105 Table Y.4.3-1 shows some common status codes corresponding to this transaction. See also Section 8.5
106 for additional status codes.

107 **Table Y.4.3-1. Status Code Meaning**

Status	Code	Meaning
Success	200 (OK)	The origin server returns the matching results.
	204 (No Content)	The origin server has no matching results.
Failure	400 (Bad Request)	The origin server cannot handle the search request because of errors in the request headers or parameters.
	413 (Payload Too Large)	The origin server cannot return the results, as their combined size exceeds the maximum payload size supported. The user agent may repeat the request with paging or with a narrower query to reduce the size.
	503 (Service Unavailable)	The origin server cannot handle the query; this may be a temporary or permanent state.

108

109 **Y.4.3.2 Response Header Fields**

110 The origin server shall support header fields as required in Table Y.4.3-2.

111 **Table Y.4.3-2. Response Header Fields**

Name	Values	Origin Server Usage	Description
Content-Type	media-type	C	See section 8.4.2.
Content-Encoding	encoding	C	See section 8.4.2.
Content-Length	Uint	C	See section 8.4.3.

112

113 All success responses shall also contain the Content Representation (see Section 8.4.2) and Payload
114 header fields (see Section 8.4.3) with appropriate values.

115 **Y.4.3.3 Response Payload**

116 A success response shall contain a dataset according to PS3.4, Table K.6-1 supplied in an Acceptable
117 Media Type. See Section 8.7.5.

118 A failure response payload may contain a Status Report describing any failures, warnings, or other useful
119 information.

Add new section X Modality Performed Procedure Step Service and Resources, immediately after section Y above

X Modality Performed Procedure Step Service and Resources

X.1 Overview

The Modality Performed Procedure Step Service enables a user agent to report progress on Performed Procedure Steps as executed by imaging modalities. This service corresponds to the DIMSE Modality Performed Procedure Step (MPPS) service as defined in Annex F of PS3.4 and has the same semantics. However, Notifications, as defined in PS3.4, Annex F.9, are not supported by this service. Instead, to be able to achieve notification-like behavior, it is recommended to mimic the approach taken in IHE's Scheduled Workflow integration profile [IHE RAD TF-1], where the Actor Modality Performed Procedure Step Manager forwards the creation and updating of Modality Performed Procedure Steps to other Actors that are interested in progress.

X.1.1 Resource Descriptions

There is one resource defined by this service:

MPPS A dataset containing the Attributes specified in Table F.7.2-1 "Modality Performed Procedure Step SOP Class N-CREATE, N-SET and Final State Attributes" in PS3.4.

In the Modality Performed Procedure Step Service, an MPPS is identified by an MPPS UID, which corresponds to the SOP Instance UID used in the PS3.4 MPPS Service, see e.g. Section F.7.2.1.2.

The following URI Template variables are used in the definitions of the resources throughout Chapter X.

{mppsUID} The UID of the MPPS.

The Modality Performed Procedure Step Service manages a number of MPPSs; its resources are given in Table X.1.1-1.

Table X.1.1-1. Modality Workflow Service Resource Descriptions

Resource	URI Template	Description
Modality Performed Procedure Step	/modality-performed-procedure-steps/{mppsUID}	A single Modality Performed Procedure Step.

X.1.2 Common Query Parameters

The origin server shall support Query Parameters as required in Table X.1.2-1.

The user agent shall supply in the request Query Parameters as required in Table X.1.2-1.

Table X.1.2-1. Common Query parameters

Name	Value	Usage		Section
		User Agent	Origin Server	
Accept	media-type	O	M	Section 8.3.3.1
Accept-Charset	charset	O	M	Section 8.3.3.2

Commented [JM10]: Use the template of 11 and 12, and see what fits best. Also, add CPs for the other sections.

Commented [JM11R10]: Template of 11 seemed most appropriate. Used here, no CPs for other sections yet.

Commented [JM12]: There might be a CP on this one changing the language of this, so adapt to this change when applicable.

Commented [JM13R12]: @WG06: what is the CP number?

X.1.3 Common Media Types

The origin server shall support the media types specified as Default or Required in Table X.1.3-1.

Table X.1.3-1. Default, Required, and Optional Media Types

Media Type	Usage	Section
application/dicom+json	Default	Section 8.7.3.2
application/dicom+xml	Required	Section 8.7.3.2
multipart/related; type="application/dicom+json"	Required	Section 8.7.3.2
multipart/related; type="application/dicom+xml"	Required	Section 8.7.3.2

Commented [JM14]: Also CP for section 13.1.3.

Commented [JM15R14]: Mailed to David as a minor.

X.2 Conformance

An origin server conforming to the Modality Performed Procedure Step Service shall support the Retrieve Capabilities Transaction (see Section 8.9.1). Furthermore, it shall support the transactions listed as Required in Table X.2-1 and may support Transactions listed as Optional. The support of the Subscribe and Unsubscribe transactions is mutually dependent.

Table X.2-1. Required and Optional Transactions

Transaction	Support	Section
Retrieve Capabilities	Required	Section 8.9
Create	Required	Section X.4
Update	Required	Section X.5
Retrieve	Optional	Section X.6

Implementations shall specify in their Conformance Statement (see PS3.2) and the Retrieve Capabilities Transaction the supported Transactions and the implementations' role: origin server, user agent, or both.

In addition, for each supported Transaction they shall specify:

- The supported Query Parameters, including optional Attributes, if any.
- The supported DICOM Media Types.
- The supported character sets (if other than UTF-8).

X.3 Transactions Overview

The Modality Workflow Service consists of the Transactions listed in Table X.3-1.

Table X.3-1. Modality Performed Procedure Step Services Transactions

Transaction Name	Method	Payload		Description
		Request	Success Response	
Create	PUT	dataset according to PS3.4, Table F.7.2-1 (N-CREATE)	none	Creates a new Modality Performed Procedure Step

Transaction Name	Method	Payload		Description
		Request	Success Response	
Update	PATCH	dataset according to PS3.4, Table F.7.2-1 (N-SET)	none	Updates the target Modality Performed Procedure Step
Retrieve	GET	none	dataset according to PS3.4, Table F.8.2-1	Retrieves the target Modality Performed Procedure Step

In Table X.3-2, the Target Resources permitted for each transaction are marked with M if support is mandatory for the origin server and O if it is optional. A blank cell indicates that the resource is not allowed in the transaction.

Table X.3-2. Resources by Transaction

Resource	Create	Update	Retrieve		
MPPS	M	M	O		
Subscription					

Table X.3-3 lists the Modality Performed Procedure Step Service Transactions that have a corresponding DIMSE Operation in DIMSE MPPS.

Table X.3-3. Mapping of Modality Performed Procedure Step Service Transactions and DIMSE Operations

Transaction	Operation	Reference	DIMSE Service
Create	Create MPPS Instance	PS3.4, F.7.2.1	N-CREATE
Update	Set MPPS Information	PS3.4, F.7.2.2	N-SET
Retrieve	Get MPPS Information	PS3.4, F.8.2.1	N-GET

Note As in DIMSE, the Transactions do *not* provide a complete CRUDL interface for the respective resources. For instance, it is not possible to list all Modality Performed Procedure Steps using DICOM, neither with DIMSE, nor with DICOMweb. What DICOM *does* provide is access to performed procedure steps at the level required for modalities.

X.4 Create Transaction

This Transaction creates a Modality Performed Procedure Step with the given Attributes. It corresponds to the DIMSE MPPS N-CREATE Operation (see PS3.4, Section F.7.2.1).

X.4.1 Request

The request shall have the following syntax:

```
PUT SP /modality-performed-procedure-steps/{mppsUID} SP version CRLF
Accept: 1#media-type CRLF
*(header-field CRLF)
CRLF
```

Commented [JM16]: Rob will send a text about this one.

Commented [JM17R16]: A text has been posted 2024.01.18. There are worries, and yet no conclusive argument to not include the PATCH.
It was understood that an updated version of a CP was to be shared. We will wait a bit.

Commented [JM18]: Flip from right to left.

Commented [JM19R18]: Possibly create CPs for adding this kind of tables sections 10 and 11 too.

Commented [JM20R18]: Possibly move to X.3, with next table, below overview table.

Commented [JM21R18]: Done, except for CP, but that is already included in the copied comment in Section Y.

Commented [JM22]: Operation text should reflect what is mentioned in K and F.

Commented [JM23R22]: It does.

Commented [JM24]: Q to WG06: should we provide a list of operations / methods that are *not* supported? (For reasons of clarity.)

Commented [JM25R24]: Is table X.3-2 sufficient? Note that that one is not complete, as deletion is not covered.

Commented [JM26R24]: There was no conclusive answer from WG06, but the current approach seems sufficient to me.

payload

X.4.1.1 Target Resource

The Target Resource of this transaction is an individual Modality Performed Procedure Step identified by its MPPS UID.

X.4.1.2 Query Parameters

The request has no Query Parameters.

X.4.1.3 Request Header Fields

The origin server shall support Request Header Fields as required in Table X.4.1-1.

The user agent shall supply Request Header Fields as required in Table X.4.1-1.

Table X.4.1-1. Request Header Fields

Name	Values	Usage		Description
		User Agent	Origin Server	
Accept	media-type	M	M	The Acceptable Media Types of the response payload.

See Section 8.4.

X.4.1.4 Request Payload

The request payload shall be present and shall contain one representation consistent with the Content-Type header field. The representation shall conform to Media Types described in Section 8.7.3 DICOM Media Type Sets. The payload shall conform to Section 8.6 Payloads.

The request payload shall contain the Modality Performed Procedure Step attributes with which the user agent requests the origin server to create a Modality Performed Procedure Step resource, according to PS3.4, Table F.7.2-1, requirement type N-CREATE (SCU).

Commented [JM27]: Look at 11.4.1.4 for language. Lines 162 and 164 suggests there are two things in the payload, namely an MPPS and a dataset.

Commented [JM28R27]: Continue to look at those sections for the sections below.

Commented [JM29R27]: Resolved by removing some text. Still have to check other occurrences.

X.4.2 Behavior

The origin server shall create a Modality Performed Procedure Step identified by the provided MPPS UID and filled with the provided attributes in the payload.

X.4.3 Response

The response shall have the following syntax:

version SP status-code SP reason-phrase CRLF
CRLF
[payload]

X.4.3.1 Status Codes

Table X.4.3-1 shows some common status codes corresponding to this transaction. See also Section 8.5 for additional status codes.

Table X.4.3-1. Status Code Meaning

Status	Code	Meaning
Success	200 (OK)	The origin server has created the requested Modality Performed Procedure Step with the provided attributes.
Failure	400 (Bad Request)	The origin server cannot handle the create request because of errors in the request headers or parameters.

	409 (Conflict)	The origin server cannot create the target Modality Performed Procedure Step because the provided Modality Performed Procedure Step UID is already in use.
	503 (Service Unavailable)	The origin server cannot handle the creation of the Modality Performed Procedure Step; this may be a temporal or permanent state.

X.4.3.2 Response Header Fields

The origin server shall support header fields as required in Table X.4.3-2.

Table X.4.3-2. Response Header Fields

Name	Values	Origin Server Usage	Description
Content-Type	media-type	C	See section 8.4.2.
Content-Encoding	encoding	C	See section 8.4.2.
Content-Length	uint	C	See section 8.4.3.

All success responses shall also contain the Content Representation (see Section 8.4.2) and Payload header fields (see Section 8.4.3) with appropriate values.

X.4.3.3 Response Payload

A success response shall have no payload.

A failure response payload may contain a Status Report describing any failures, warnings, or other useful information.

X.5 Update Transaction

This Transaction sets Attributes of an existing Modality Performed Procedure Step. It corresponds to the DIMSE MPPS N-SET Operation (see PS3.4, Section F.7.2.2).

X.5.1 Request

The request shall have the following syntax:

```
PATCH SP /modality-performed-procedure-steps/{mppsUID} SP version CRLF
Accept: 1#media-type CRLF
*(header-field CRLF)
CRLF
payload
```

X.5.1.1 Target Resources

The Target Resource of this transaction is an individual Modality Performed Procedure Step identified by its MPPS UID.

X.5.1.2 Query Parameters

The request has no Query Parameters.

Commented [JM30]: Is it needed to refer to Common Query Parameters?

Commented [JM31R30]: It is currently not consistent in PS3.18.
For instance Section 10.5.1.2 Query Parameters states “The Store transaction has no Query Parameters.” and nothing more. So propose to keep it like it is now.

X.5.1.3 Request Header Fields

The origin server shall support Request Header Fields as required in Table X.5.1-1.

The user agent shall supply Request Header Fields as required in Table X.5.1-1.

Table X.5.1-1. Request Header Fields

Name	Values	Usage		Description
		User Agent	Origin Server	
Accept	media-type	M	M	The Acceptable Media Types of the response payload.

See also Section 8.4.

X.5.1.4 Request Payload

The request payload shall be present and shall contain one representation consistent with the Content-Type header field. The representation shall conform to Media Types described in Section 8.7.3 DICOM Media Type Sets. The payload shall conform to Section 8.6 Payloads.

The request payload shall contain the Modality Performed Procedure Step attributes with which the user agent requests the origin server to update a Modality Performed Procedure Step resource, according to PS3.4, Table F.7.2-1, requirement type N-SET (SCU).

X.5.2 Behavior

The origin server shall update the Modality Performed Procedure Step identified by the provided MPPS UID with the provided attributes in the payload.

X.5.3 Response

The response shall have the following syntax:

version SP status-code SP reason-phrase CRLF
CRLF
[payload]

X.5.3.1 Status Codes

Table X.5.3-1 shows some common status codes corresponding to this transaction. See also Section 8.5 for additional status codes.

Table X.5.3-1. Status Code Meaning

Status	Code	Meaning
Success	200 (OK)	The origin server has updated the Modality Performed Procedure Step with the provided attributes.
Failure	400 (Bad Request)	The origin server cannot handle the update request because of errors in the request headers or parameters.
	404 (Not Found)	The origin server has no knowledge about the target Modality Performed Procedure Step.
	409 (Conflict)	The origin server cannot update the target Modality Performed Procedure Step, for instance because the changes provided are incompatible with the data of the target Modality Performed Procedure Step.
	410 (Gone)	The origin server knows that the target Modality Performed Procedure Step did exist but has been deleted.
	503 (Service Unavailable)	The origin server cannot handle the creation of the Modality Performed Procedure Step; this may be a temporal or permanent state.

Note When it is requested that attributes are to be updated while these have not been made available at creation time, a 409 (Conflict) can be returned; this is the case when PS3.4, Table F.7.2-1 specifies that these attributes should have been made available at creation time.

X.5.3.2 Response Header Fields

The origin server shall support header fields as required in Table X.5.3-2.

Table X.5.3-2. Response Header Fields

Name	Values	Origin Server Usage	Description
Content-Type	media-type	C	See section 8.4.2.
Content-Encoding	encoding	C	See section 8.4.2.
Content-Length	uint	C	See section 8.4.3.

All success responses shall also contain the Content Representation (see Section 8.4.2) and Payload header fields (see Section 8.4.3) with appropriate values.

X.5.3.3 Response Payload

A success response should have no payload.

A failure response payload may contain a Status Report describing any failures, warnings, or other useful information.

X.6 Retrieve Transaction

This Transaction retrieves an existing Modality Performed Procedure Step. It corresponds to the MPPS DIMSE N-GET Operation (see PS3.4, Section F.8.2.1).

X.6.1 Request

The request shall have the following syntax:

```
GET SP /modality-performed-procedure-steps/{mppsUID}{?includefield*} SP version CRLF
Accept: 1#media-type CRLF
*(header-field CRLF)
CRLF
```

X.6.1.1 Target Resources

The Target Resource of this transaction is an individual Modality Performed Procedure Step identified by its MPPS UID.

X.6.1.2 Query Parameters

The origin server shall support the includefield Query Parameter. This specifies the Attributes that shall be included in the response. The value is either a comma-separated list of attributes, or the single keyword "all", which means that all available attributes of the object should be included in the response..

```
includefield = *("includefield" "=" (1#attribute / "all") )
```

There may be one or more includefield parameters; however, if a parameter with the value of "all" is present, then other includefield parameters shall not be present.

The includefield parameter corresponds to DIMSE's PS3.4, Table F.8.2-1 "Modality Performed Procedure Step Retrieve SOP Class N-GET Attributes".

The user agent may supply includefield Query Parameters as described above.

X.6.1.3 Request Header Fields

The origin server shall support Request Header Fields as required in Table X.6.1-1.

The user agent shall supply Request Header Fields as required in Table X.6.1-1.

Table X.6.1-1. Request Header Fields

Name	Values	Usage		Description
		User Agent	Origin Server	
Accept	media-type	M	M	The Acceptable Media Types of the response payload.

See Section 8.4.

X.6.1.4 Request Payload

The request shall have no payload.

X.6.2 Behavior

If the Modality Performed Procedure Step exists on the origin server, the attributes of this as specified in the includefield shall be returned in an Acceptable Media Type (see Section 8.7.4). When the includefield is absent, all attributes shall be returned.

X.6.3 Response

The response shall have the following syntax:

version SP status-code SP reason-phrase CRLF
CRLF
[payload]

X.6.3.1 Status Codes

Table X.6.3-1 shows some common status codes corresponding to this transaction. See also Section 8.5 for additional status codes.

Table X.6.3-1. Status Code Meaning

Status	Code	Meaning
Success	200 (OK)	The origin server returned the target Modality Performed Procedure Step.
Failure	400 (Bad Request)	The origin server cannot handle the retrieve request because of errors in the request headers or parameters.
	404 (Not Found)	The origin server has no knowledge about the target Modality Performed Procedure Step.
	410 (Gone)	The origin server knows that the target Modality Performed Procedure Step did exist but has been deleted.
	503 (Service Unavailable)	The origin server cannot handle the retrieval of the target Modality Performed Procedure Step; this may be a temporal or permanent state.

X.6.3.2 Response Header Fields

The origin server shall support header fields as required in Table X.6.3-2.

342 **Table X.6.3-2. Response Header Fields**

Name	Values	Origin Server Usage	Description
Content-Type	media-type	C	See section 8.4.2.
Content-Encoding	encoding	C	See section 8.4.2.
Content-Length	uint	C	See section 8.4.3.

343

344 All success responses shall also contain the Content Representation (see Section 8.4.2) and Payload
345 header fields (see Section 8.4.3) with appropriate values.

346 **X.6.3.3 Response Payload**

347 A success response has a payload containing the requested Modality Performed Procedure Step in the
348 Selected Media Type.

349 A failure response payload may contain a Status Report describing any failures, warnings, or other useful
350 information.

351

352

Commented [JM32]: Check whether this is done consistently. (11.4.3.3).

Commented [JM33R32]: Is in line with e.g. 11.5.3.3.

Update Section 2 Normative References: add [IHE RAD TF-1]

…

2.3 Other References

…

[FHIR Access Denied] HL7. . FHIR Security - Access Denied Response Handling. http://hl7.org/fhir/security.html#AccessDenied.

[IHE RAD TF-1] Integrating the Healthcare Enterprise (IHE). Radiology Technical Framework Volume 1. http://www.ihe.net/uploadedFiles/Documents/Radiology/IHE_RAD_TF_Vol1.pdf.

[IHE RAD TF-~~Vol~~-2] Integrating the Healthcare Enterprise (IHE). Radiology Technical Framework Volume 2. http://www.ihe.net/uploadedFiles/Documents/Radiology/IHE_RAD_TF_Vol2.pdf.

…

Commented [JM34]: This way of referring to Technical Frameworks of IHE is used elsewhere in the standard, and hence should be used here too.
Note to the editor: please update the four references in PS3.18 to [IHE RAD TF-2] to match this aligned format.

Update Section 4 Symbols and Abbreviated Terms: add CRUDL, MPPS, MWL, and UPS

4 Symbols and Abbreviated Terms

…

ABNF	Augmented Backus-Naur Form. See [RFC5234] and [RFC7405].
CRUDL	**Create, Read, Update, Delete, List; basic operations/actions on objects.**
DICOM	Digital Imaging and Communications in Medicine

…

JSON	JavaScript Object Notation
MPPS	**Modality Performed Procedure Step service. See PS3.4, Annex F.**
MWL	**Modality Worklist service; colloquial name for the Basic Worklist service. See PS3.4, Annex K.**
QIDO-RS	Query based on ID for DICOM Objects by RESTful Services

…

UID	Unique (DICOM) Identifier
UPS	**Unified Procedure Step service. See PS3.4, Annex CC.**
UPS-RS	Unified Procedure Step by RESTful Services

…

Update Section 8.1.1 Request Message Syntax by removing unused methods and adding PATCH

…

method = "CONNECT" / "DELETE" / "GET" / "HEAD" / "OPTIONS" / **"PATCH" /** "POST" / "PUT"

…

8.1.1.1 Method

The request method is one of the HTTP methods, such as CONNECT, DELETE, GET, HEAD, OPTIONS, **PATCH,** POST, **and** PUT. See [RFC7230] Section 4.

Update Section B Examples: add new examples for the Modality Workflow Service

B Examples (Informative)

…

B.X1 Searching for Modality Scheduled Procedure Steps using JSON Media Type

This example illustrates a request to retrieve the scheduled procedure steps for a scheduled station: CTSCANNER, start date: 20250101 and modality: CT, where the results are to be returned in JSON. Also, the number of returned results is limited to 20 and the results are requested to contain all available tags. The offset of the returned results is set to 0.

```
GET /radiology/modality-scheduled-procedure-steps/?00400100.00400010=CTSCANNER
&00400100.00400002=20250101&00400100.00080060=CT &limit=20&offset=0&includefield=all HTTP/1.1
Host: www.hospital-stmarco
Accept: application/dicom+json
```

An example of a successful response to the above request is given below:b

```
HTTP/1.1 200 OK
Content-Length: 1191
Content-Type: application/dicom+json; charset=utf-8
…
[ {
  , …
  , "00100010": { "vr": "PN", "Value": [{ "Alphabetic": "Doe^Sally" }] }
  , "0020000D": { "vr": "UI", "Value": ["1.2.250.1.59.40211.3000000809041250108230000000 4"] }
  , "00401001": { "vr": "SH", "Value": ["P-ID-22"] }
  , …
  , "00400100": { "vr": "SQ", "Value":
    [ { "00400002": { "vr": "DA", "Value": ["20250101"] }
      , "00400007": { "vr": "LO", "Value": ["Specials^04a_HeadCTA"] }
      , "00400009": { "vr": "SH", "Value": ["PS-ID-23"] }
      , "00400010": { "vr": "SH", "Value": ["CTSCANNER"] }
      , …
      }
    , { "00400002": { "vr": "DA", "Value": ["20250101"] }
      , "00400007": { "vr": "LO", "Value": ["Specials^04a_SpineCTA"] }
      , "00400009": { "vr": "SH", "Value": ["PS-ID-24"] }
      , "00400010": { "vr": "SH", "Value": ["CTSCANNER"] }
```

Commented [JM35]: Make the removal of CONNECT and HEAD a separate CP.

Commented [JM36]: Should be the same as the one below (or the other way around).

Commented [JM37]: Reorder parameters to be in the same order as in the text.

```
      , ...
      }
    , ...
    ] }
  , ...
  }
, ...
]
```

The response returns two scheduled procedure steps for Sally Doe, one for the head and the other one for the spine. The attributes are according to PS3.4, Table_K.6-1 “Attributes for the Modality Worklist Information Model”:

- Patient's Name (0010,0010);
- Study Instance UID (0020,000D);
- Requested Procedure ID (0040,1001);
- Scheduled Procedure Step Sequence (0040,0100);
  - Scheduled Procedure Step Description (0040,0007);
  - Scheduled Station Name (0040,0010);
  - Scheduled Procedure Step Start Date (0040,0002);
  - Scheduled Procedure Step ID (0040,0009).

B.X2 Creating a Modality Performed Procedure Step using JSON Content Type

This example illustrates a request to create a modality performed procedure step using JSON. The intention is to mark it in the state: “IN PROGRESS”. This is a continuation of the previous example as given in B.X1, where the Patient’s Name (0010,0010), Study Instance UID (0020,000D), Scheduled Procedure Step Description (0040,0007), and Requested Procedure ID (0040,1001) have been taken over from the received modality scheduled procedure step, and the Performed Procedure Step Status (0040,0252), Performed Procedure Step ID (0040,0253), Accession Number (0008,0050), and Scheduled Procedure Step ID (0040,0009) have been added as “IN PROGRESS”, 1.2.250.1.59.40211.12345678.987654, 1, and “PS-ID-23” respectively, some of which in the Scheduled Step Attributes Sequence (0040,0270).

```
PUT /radiology/modality-performed-procedure-steps/ 1.2.250.1.59.40211.12345678.987654 HTTP/1.1
Host: www.hospital-stmarco
Content-Type: application/dicom+json
...
{
...
, "00100010": { "vr": "PN", "Value": [{ "Alphabetic": "Doe^Sally" }] }
, "00400242": { "vr": "SH", "Value": ["CTSCANNER"] }
, "00400252": { "vr": "CS", "Value": ["IN PROGRESS"] }
, "00400253": { "vr": "SH", "Value": ["1.2.250.1.59.40211.12345678.987654"] }
, ...
, "00400270": { "vr": "SQ", "Value":
  [ { "00080050": { "vr": "SH", "Value": ["1"] }
    , "0020000D": { "vr": "UI", "Value": ["1.2.250.1.59.40211.3000000809041250108230000000 4"] }
    , "00400007": { "vr": "LO", "Value": ["Specials^04a_HeadCTA"] }
    , "00400009": { "vr": "SH", "Value": ["PS-ID-23"] }
    , "00401001": { "vr": "SH", "Value": ["P-ID-22"] }
    ...
    }
  , ...
  ] }
, ...
}
...
```

Commented [JM38]: Elaborate on the content of the request here in the text.

Commented [JM39]: Change that to legal UID. Also below.

Commented [JM40]: Also make up another UID here. ACME incorporated. PS3.17.

A successful response to the request will be:

```
HTTP/1.1 200 OK
```

B.X3 Updating a Modality Performed Procedure Step with Produced Image Data using JSON Content Type

This example illustrates an HTTP request for updating a modality performed procedure step using JSON. The intention is to record the newly created instances as part of the Referenced Image Sequence (0008,1140) during the ongoing acquisition of images on the modality.

This example is a continuation of the previous example as given in B.X2, working on the same MPPS with UID 1.2.250.1.59.40211.12345678.987654. It adds a Performed Series Sequence (0040,0340), which contains:

- a Series Description (0008,103E) with value "Head 1.50 Hr64 ax";
- Gregory House as the performing physician (Performing Physician's Name (0008,1050));
- a Referenced Image Sequence (0008,1140) with two items that have the same Referenced SOP Class UID (0008,1150), namely "1.2.840.10008.5.1.4.1.1.2", which is CT Image, and have distinct Referenced SOP Instance UIDs (0008,1155);
- a Protocol Name (0018,1030) with value "Special^99a_HeadCTA";
- a Series Instance UID (0020,000E): "1.2.250.1.59.40211.197132.300000200407183228403000000007".

```
PATCH /radiology/modality-performed-procedure-steps/1.2.250.1.59.40211.12345678.987654 HTTP/1.1
Host: www.hospital-stmarco
Content-Type: application/dicom+json
…
{…
, "00400340": { "vr": "SQ", "Value":
  [ { "0008103E": { "vr": "LO", "Value": ["Head 1.50 Hr64 ax"] }
    , "00081050": { "vr": "PN", "Value": [{ "Alphabetic": "House^Gregory" }] }
    , "00081140": { "vr": "SQ", "Value":
      [ { "00081150": { "vr": "UI", "Value": ["1.2.840.10008.5.1.4.1.1.2"] }
        , "00081155": { "vr": "UI", "Value": ["1.2.250.1.59.40211.197132.300000200407183228403000520"] }
        , …
        }
      , { "00081150": { "vr": "UI", "Value": ["1.2.840.10008.5.1.4.1.1.2"] }
        , "00081155": { "vr": "UI", "Value": ["1.2.250.1.59.40211.197132.300000200407183228403000521"] }
        , …
        }
      , …
      ] }
    , …
    , "00181030": { "vr": "LO", "Value": ["Special^99a_HeadCTA"] }
    , "0020000E": { "vr": "UI", "Value": ["1.2.250.1.59.40211.197132.300000200407183228403000007"] }
    , …
    }
  , …
  ] }
, …
}
…
```

A successful response to the request will be:

Commented [JM41]: This example is about recording newly created instances. What needs to be here (on the ellipsis).

Commented [JM42R41]: Nothing is mandatory (see PS3.4, Table F.7.2-1), so the ellipses can be empty.

```
HTTP/1.1 200 OK
```

Note Even though the Modality Performed Procedure Step is updated, sequences within it, like the Performed Series Sequence (0040,0340) and the Referenced Image Sequence (0008,0140) are to be given in their entirety, as required in DIMSE; updates to these are not allowed. See PS3.4, section F.7.2.2.2.

B.X4 Completing a Modality Performed Procedure Step using JSON Content Type

This example illustrates an example of an HTTP request for completing a modality performed procedure step. It is a continuation of the previous example as given in B.X3, working on the same MPPS with UID 1.2.250.1.59.40211.12345678.987654. Here, the mandatory Performed Procedure Step End Date (0040,0250) and Performed Procedure Step End Time (0040,0251) are added, and the Performed Procedure Step Status (0040,0252) is set to “COMPLETED”.

```
PATCH /radiology/modality-performed-procedure-steps/1.2.250.1.59.40211.12345678.987654 HTTP/1.1
Host: www.hospital-stmarco
Content-Type: application/dicom+json
…
{ …
, "00400250": { "vr": "DA", "Value": ["20200101"] }
, "00400251": { "vr": "TM", "Value": ["1300"] }
, "00400252": { "vr": "CS", "Value": ["COMPLETED"] }
, …
}
…
```

A successful response to the request will be:

```
HTTP/1.1 200 OK
```

B.X5 Retrieving a Modality Performed Procedure Step using JSON Media and Content Type

Here we have two examples, the first returning all available attributes, and the second returning a specified selection of attributes.

B.X5.1 Return All Attributes

This example illustrates a request to retrieve an existing modality performed procedure step in JSON returning all attributes. It is a culmination of the previous examples as given in B.X2-B.X4 in which all attributes that have been added are returned here.

```
GET /radiology/modality-performed-procedure-steps/ 1.2.250.1.59.40211.12345678.987654?includefield=all HTTP/1.1
Host: www.hospital-stmarco
Accept: application/dicom+json
```

A successful response to the request will be:

```
HTTP/1.1 200 OK
Content-Length: 2191
Content-Type: application/dicom+json; charset=utf-8
…
[ {
…
  , "00100010": { "vr": "PN", "Value": [{ "Alphabetic": "Doe^Sally" }] }
  , "00400242": { "vr": "SH", "Value": ["CTSCANNER"] }
```

Commented [JM43]: We could decide that DICOMweb has different semantics here (e.g. all entries of sequences are seen as additions), but this needs to be approved, as this has some repercussions, e.g. what to do with duplicates? Or with inconsistencies?

Commented [JM44R43]: Make the text declarative. “Updates of …”

Commented [JM45]: Final state requirements? Refer to them.

```
 , "00400252": { "vr": "CS", "Value": ["COMPLETED"] }
 , "00400253": { "vr": "SH", "Value": ["1.2.250.1.59.40211.12345678.987654"] }
 , …
 , "00400270": { "vr": "SQ", "Value":
  [ { "00080050": { "vr": "SH", "Value": ["1"] }
    , "0020000D": { "vr": "UI", "Value": ["1.2.250.1.59.40211..3000000809041250108230000000 4"] }
    , "00400007": { "vr": "LO", "Value": ["Specials^04a_HeadCTA"] }
    , "00400009": { "vr": "SH", "Value": ["PS-ID-23"] }
    , "00401001": { "vr": "SH", "Value": ["P-ID-22"] }
    …
    }
  , …
  ] }
 , …
 , "00400340": { "vr": "SQ", "Value":
  [ { "0008103E": { "vr": "LO", "Value": ["Head 1.50 Hr64 ax"] }
    , "00081050": { "vr": "PN", "Value": [{ "Alphabetic": "House^Gregory" }] }
    , "00081140": { "vr": "SQ", "Value":
     [ { "00081150": { "vr": "UI", "Value": ["1.2.840.10008.5.1.4.1.1.2"] }
       , "00081155": { "vr": "UI", "Value": ["1.2.250.1.59.40211.197132.3000002004071832284030000520"] }
       , …
       }
     , { "00081150": { "vr": "UI", "Value": ["1.2.840.10008.5.1.4.1.1.2"] }
       , "00081155": { "vr": "UI", "Value": ["1.2.250.1.59.40211.197132.3000002004071832284030000521"] }
       , …
       }
     , …
     ] }
    , …
    , "00181030": { "vr": "LO", "Value": ["Special^99a_HeadCTA"] }
    , "0020000E": { "vr": "UI", "Value": ["1.2.250.1.59.40211.197132.3000002004071832284030000007"] }
    , …
    }
 , …
 ] }
 , …
} ]
```


The attributes are according to PS3.4, Section F.8.

Commented [JM46]: Please hyperlink this reference.

B.X5.2 Returning Specific Attributes Only

This example illustrates a request to retrieve an existing modality performed procedure step in JSON returning specific attributes only, in this case the Patient's Name (0010,0010), the Performed Procedure Step Status (0040,0252), and the Performed Station Name (0040,0242).

```
GET /radiology/modality-performed-procedure-steps/
1.2.250.1.59.40211.12345678.987654?includefield=00100010,00400252,00400242 HTTP/1.1
Host: www.hospital-stmarco
Accept: application/dicom+json
```

A successful response to the request will be:

```
HTTP/1.1 200 OK
Content-Length: 289
Content-Type: application/dicom+json; charset=utf-8

[ { "00100010": { "vr": "PN", "Value": [{ "Alphabetic": "Doe^Sally" }] }
  , "00400242": { "vr": "SH", "Value": ["CTSCANNER"] }
  , "00400252": { "vr": "CS", "Value": ["COMPLETED"] }
} ]
```

Note The order of the attributes in the result is different than that of the request, as the JSON result needs to provide the attributes in ascending order (see Section F.2.2). Such ordering is not required for the query parameters.

B.X6 Bi-directional Proxies for Searching the Modality Scheduled Procedure Steps

The DICOMweb Modality Scheduled Procedure Step Service may be deployed in a hybrid environment, i.e. an environment in which both DICOMweb and DIMSE are used. In such a hybrid environment, a proxy can broker transactions from one service to the other, allowing a DICOMweb origin server or a DIMSE SCP to support workflow primitives for a mixed set of DICOMweb user agents and DIMSE SCUs.

DICOM does not require an implementation of proxies; however, since they would be very useful in a hybrid environment, the examples in this section show how this could be done.

Figure B.X6-1 shows how a proxy could facilitate a request for searching modality scheduled procedure steps from a DIMSE SCU to a DICOMweb origin server.

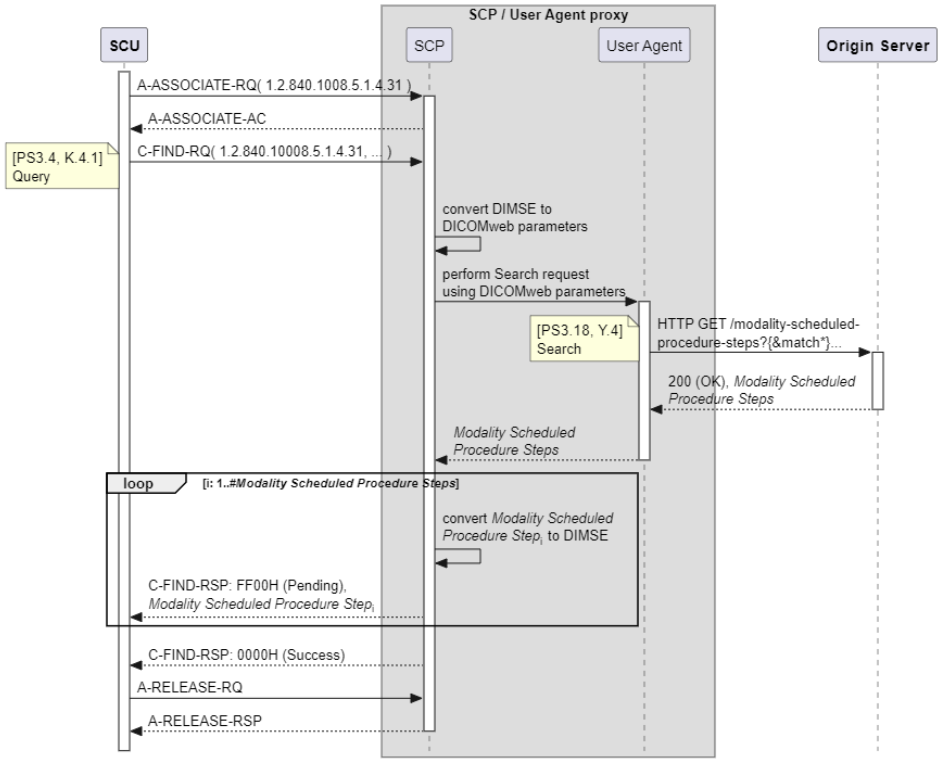

Figure B.X6-1. Modality Scheduled Procedure Step DIMSE Proxy for DICOMweb Origin Server

Figure B.X6-2 shows how a proxy could facilitate a request for searching modality scheduled procedure steps from a DICOMweb user agent to a DIMSE SCP.

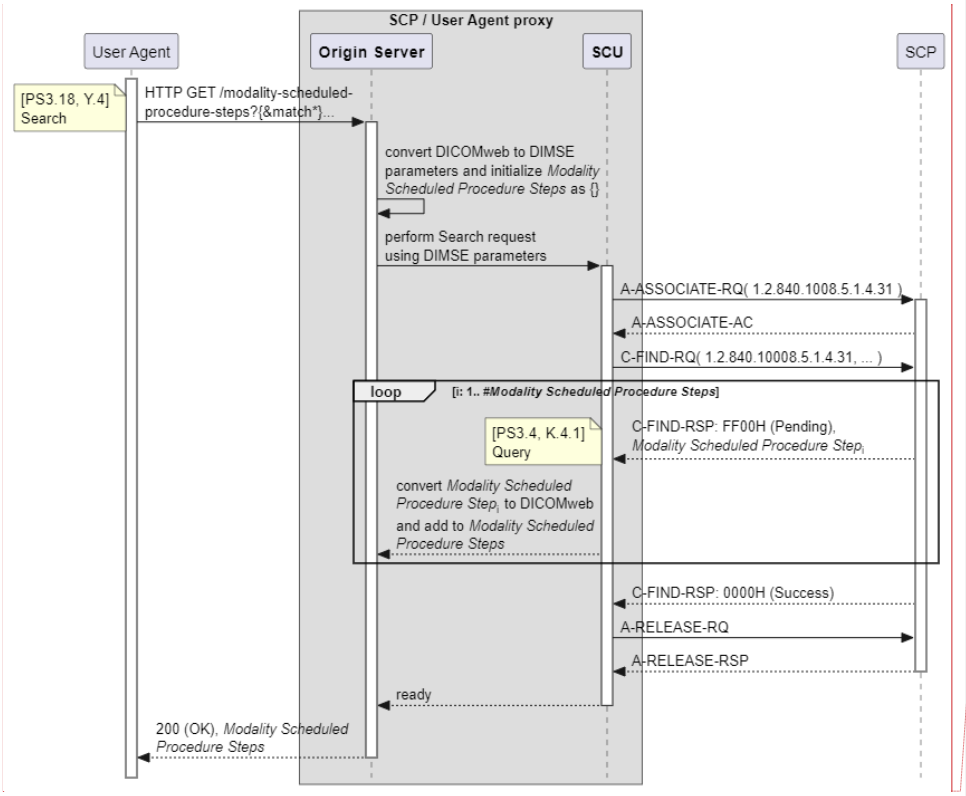

Figure B.X6-2. Modality Scheduled Procedure Step DICOMweb Proxy for DIMSE SCP

B.X7 Bi-directional Proxies for Managing a Modality Performed Procedure Step

The DICOMweb Modality Performed Procedure Step Service may be deployed in a hybrid environment, i.e. an environment in which both DICOMweb and DIMSE are used. In such a hybrid environment, a proxy can broker transactions from one service to the other, allowing a DICOMweb origin server or a DIMSE SCP to support workflow primitives for a mixed set of DICOMweb user agents and DIMSE SCUs.

DICOM does not require an implementation of proxies; however, since they would be very useful in a hybrid environment, the examples in this section show how this could be done.

B.X7.1 Create

Figure B.X7.1-1 shows how a proxy could facilitate a request for creating a modality performed procedure step from a DIMSE SCU to a DICOMweb origin server.

Commented [JM47]: Name of proxy is incorrect. Also for others below.

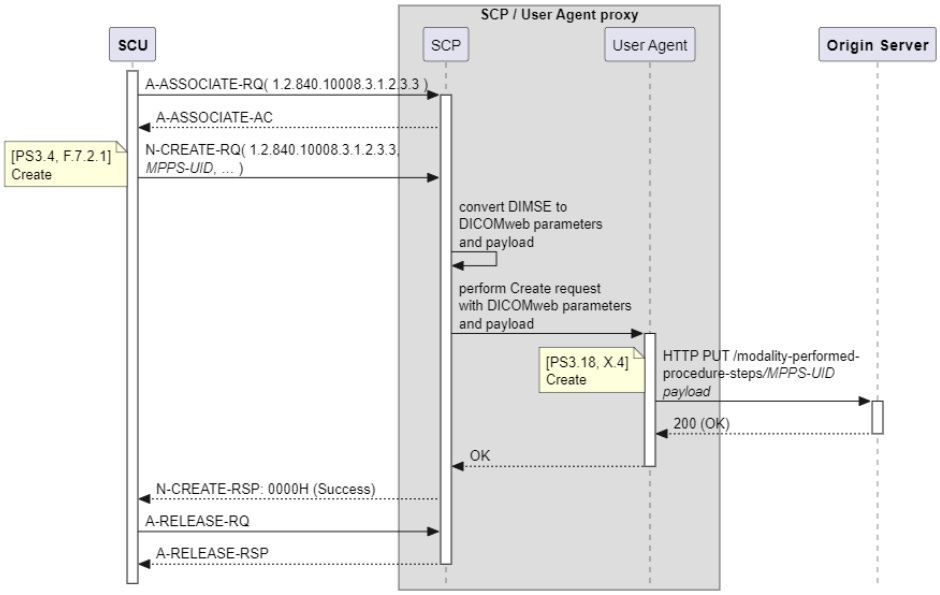

Figure B.X7.1-1. MPPS Create DIMSE Proxy for DICOMweb Origin Server

Figure B.X7.1-2 shows how a proxy could facilitate a request for creating a modality performed procedure step from a DICOMweb user agent to a DIMSE SCP.

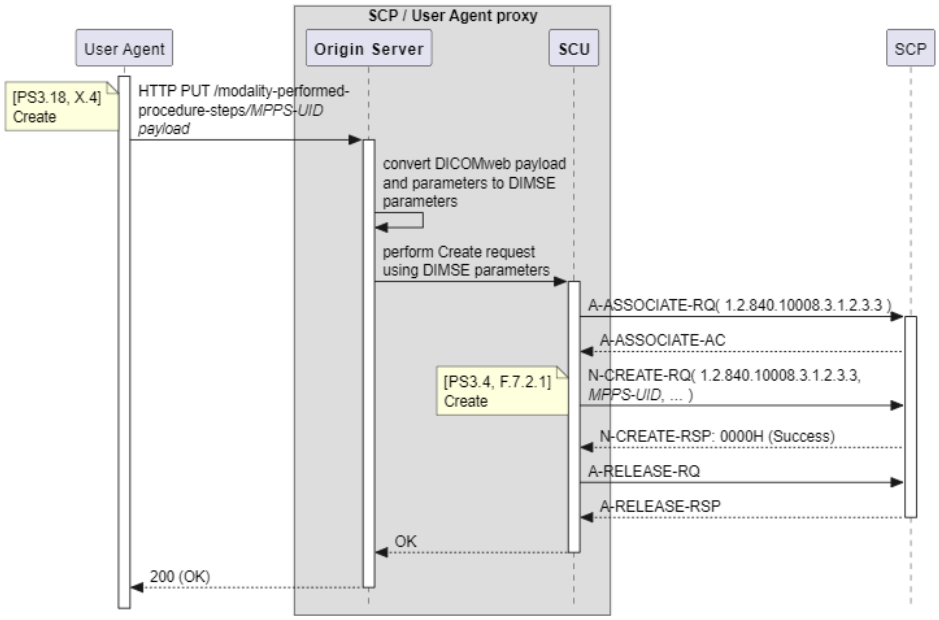

Figure B.X7.1-2. MPPS Create DICOMweb Proxy for DIMSE SCP

B.X7.2 Update

Figure B.X7.2-1 shows how a proxy could facilitate a request for updating a modality performed procedure step from a DIMSE SCU to a DICOMweb origin server.

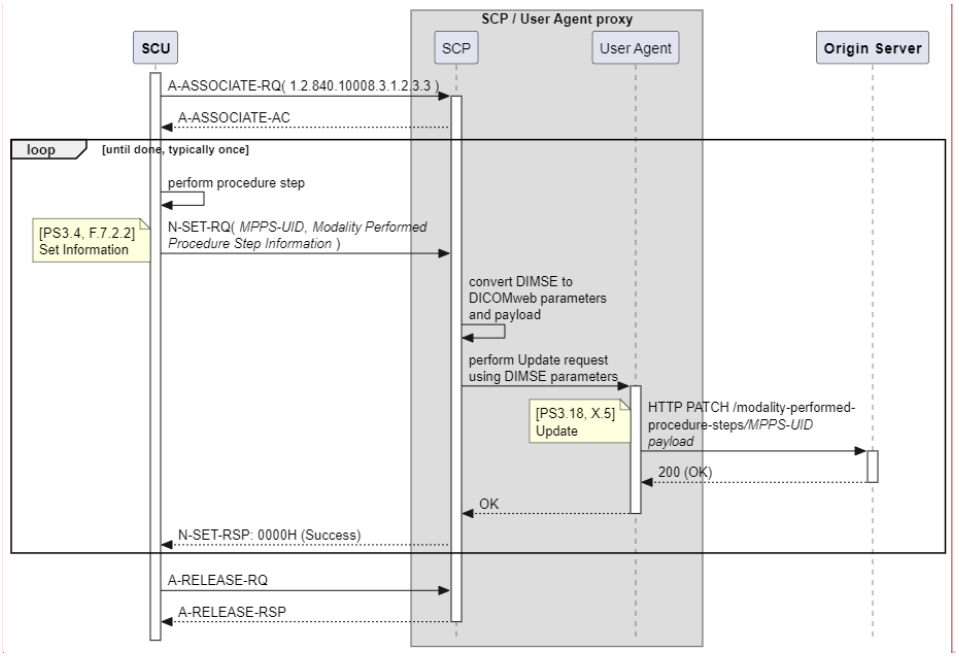

Figure B.X7.2-1. MPPS Update DIMSE Proxy for DICOMweb Origin Server

Figure B.X7.2-2 shows how a proxy could facilitate a request for updating a modality performed procedure step from a DICOMweb user agent to a DIMSE SCP.

Commented [JM48]: Remove looping in both examples.

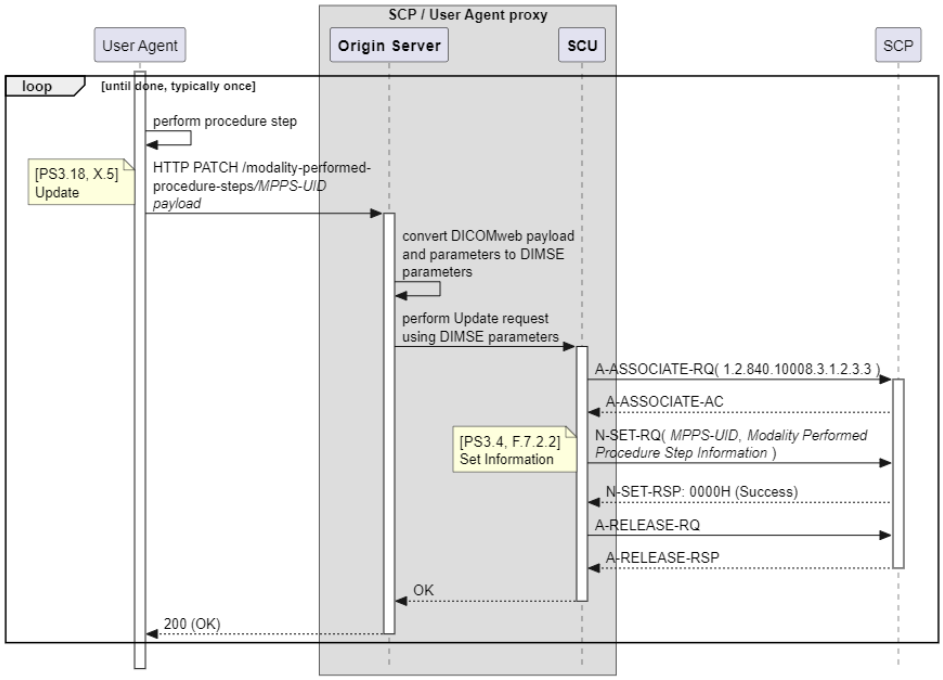

Figure B.X7.2-2. MPPS Update DICOMweb Proxy for DIMSE SCP

B.X7.3 Retrieve

Figure B.X7.3-1 shows how a proxy could facilitate a request for retrieving a modality performed procedure step from a DIMSE SCU to a DICOMweb origin server.

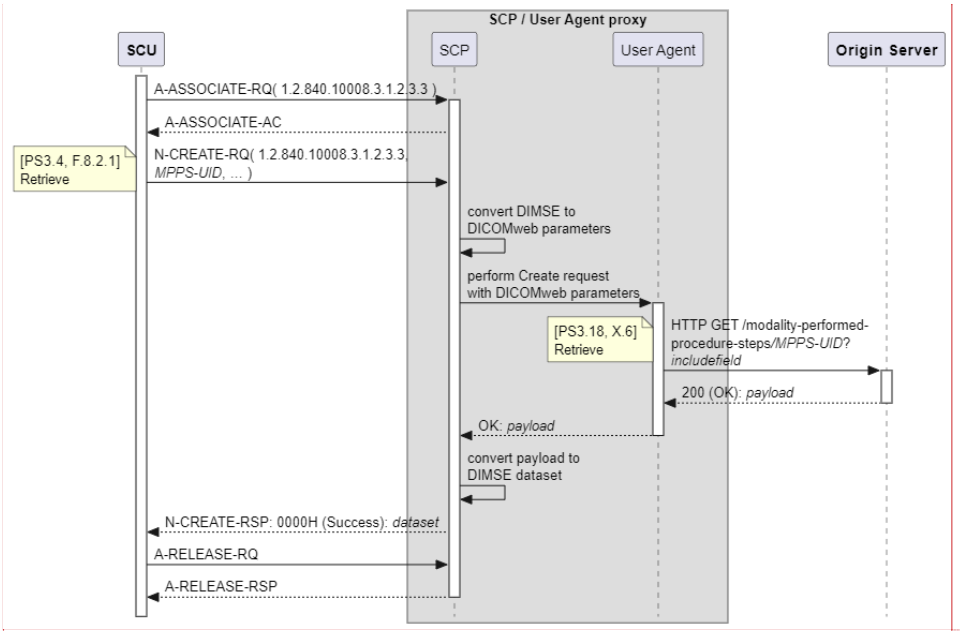

Figure B.X7.2-1. MPPS Retrieve DIMSE Proxy for DICOMweb Origin Server

Figure B.X7.4-2 shows how a proxy could facilitate a request for retrieving a modality performed procedure step from a DICOMweb user agent to a DIMSE SCP.

Commented [JM49]: Should be the N-GET. Also in example below.

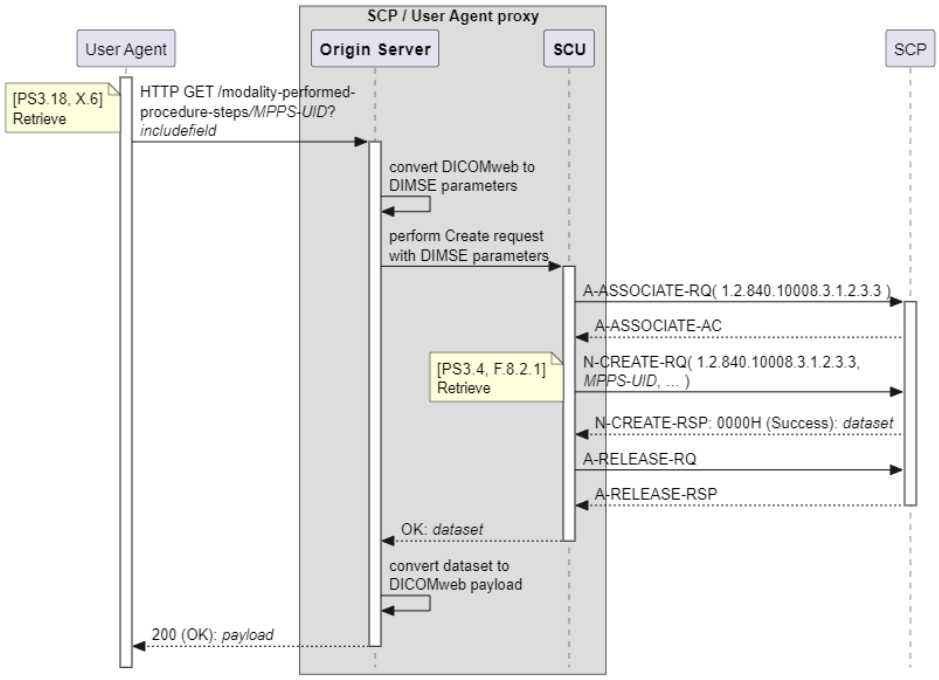

Figure B.X7.2-2. MPPS Retrieve DICOMweb Proxy for DIMSE SCP

690 ***Update Table H-1 Resources and Methods: add new resources and methods for Modality***
691 ***Workflow Services***

692 # H Capabilities Description

Service	Resource	Transactions	Reference
…			
Storage Commitment Requests (see Section 13.1.1)			
	commitment-requests	Request	Section 13.4
		Result Check	Section 13.5
Modality Scheduled Procedure Step Service (see section Y.1.1)			
	modality-scheduled-procedure-steps	**Search**	**Section Y.4**
Modality Performed Procedure Step Service (see Section X.1.1)			
	modality-performed-procedure-steps	**Create**	**Section X.4**
		Update	**Section X.5**
		Retrieve	**Section X.6**

693

Changes to NEMA Standards Publications PS 3.2

Add new sections to N.1.3 for the Modality Scheduled and Performed Procedure Step Services

N.1 Overview

…

N.1.3 DICOM Web Services

…

N.1.3.Y Modality Scheduled Procedure Step Service

Table N.1.3.Y-1 lists details on the support of the Modality Scheduled Procedure Step Service.

[Complete Table N.1.3.Y-1 to indicate support for the Modality Scheduled Procedure Step Web Service]

Table N.1.3.Y-1 Modality Scheduled Procedure Step Service

Service	Transaction	Resource	User Agent	Origin Server
Modality Scheduled Procedure Step Service	*Search*	modality-scheduled-procedure-steps		

N.1.3.X Modality Performed Procedure Step Service

Table N.1.3.X-1 lists details on the support of the Modality Performed Procedure Step Service.

[Complete Table N.1.3.X-1 to indicate support for the Modality Performed Procedure Step Web Service]

Table N.1.3.X-1 Modality Performed Procedure Step Service

Service	Transaction	Resource	User Agent	Origin Server
Modality Performed Procedure Step Service	*Create*	modality-performed-procedure-steps		
	Update			
	Retrieve			

[When supporting both the Origin Server and User Agent roles, indicate whether creation and update of MPPSs on the Origin Server side is mirrored on the User Agent side by selecting one of the two texts below. When only supporting one of the roles, remove the texts below.]

This system does not mirror the creation and updating of MPPSs on the Origin Server side to the User Agent side.

This system mirrors the creation and updating of MPPSs on the Origin Server side to the User Agent side.

Add a new subsections on the Modality Scheduled and Performed Procedure Step Services to section N.5.3 Supported DICOM Web Services

N.5 Service and Interoperability Description

…

N.5.3 Supported DICOM Web Services

…

N.5.3.Y Modality Scheduled Procedure Step Web Service

This section provides details regarding the Modality Scheduled Procedure Step Web Service. For an overview of supported Transactions and resources see Table N.1.3.Y-1 Modality Scheduled Procedure Step Service.

N.5.3.Y.1 Search Transaction – Modality Scheduled Procedure Step Service

[If your system does not support the Modality Scheduled Procedure Step Web Service Search Transaction, you can indicate that this section is not applicable and remove the subsections below.]

N.5.3.Y.1.1 User Agent

The Search Transaction user agent can request resources listed in Table N.5.3.Y.1.1-1.

[List the supported resources for your Modality Scheduled Procedure Step Search Transaction user agent. Remove the non-supported resources rows. Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.Y.1.1-1: Resources for Search Transaction – User Agent

Resource	Comments
	See Resources path in table Y.1.1-1 in PS3.18
modality-scheduled-procedure-steps	

The Search Transaction user agent supports Header Fields listed in Table N.5.3.Y.1.1-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.Y.1.1-2: Header Fields for Search Transaction – User Agent

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json (Default)* *application/dicom+xml* *multipart/related; type="application/dicom+json"* *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.Y.1.2 Origin Server

The Search Transaction origin server receives GET requests to search for modality scheduled procedure steps.

The user agent specifies the Target Resource as part of the URI and the acceptable Content-Type in the HTTP header (i.e. XML or JSON).

The URI is composed by a Base URI: see Base URI for the origin server in Section N.6.3.Y.

The Request Transaction origin server supports resources listed in Table N.5.3.Y.1.2-1.

[Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.Y.1.2-1: Resources for Search Transaction – Origin Server

Resource	Comments
	See Resources path in Table Y.1.1-1 in PS3.18
modality-scheduled-procedure-steps	

The Search Transaction origin server supports Header Fields listed in Table N.5.3.Y.1.2-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.Y.1.2-2: Header Fields for Search Transaction – Origin Server

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json* *application/dicom+xml* *multipart/related; type="application/dicom+json"* *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.X Modality Performed Procedure Step Web Service

This section provides details regarding the Modality Performed Procedure Step Web Service. For an overview of supported Transactions and resources see Table N.1.3.X-1 Modality Performed Procedure Step Service.

N.5.3.X.1 Create Transaction – Modality Performed Procedure Step Service

[If your system does not support the Modality Performed Procedure Step Web Service Create Transaction, you can indicate that this section is not applicable and remove the subsections below.]

N.5.3.X.1.1 User Agent

The Create Transaction user agent can request to create resources listed in Table N.5.3.X.1.1-1.

[List the supported resources for your Modality Performed Procedure Step Create Transaction user agent. Remove the non-supported resources rows. Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.X.1.1-1: Resources for Create Transaction – User Agent

Resource	Comments
	See Resources path in table X.1.1-1 in PS3.18
modality-performed-procedure-steps	

The Create Transaction user agent supports Header Fields listed in Table N.5.3.X.1.1-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.X.1.1-2: Header Fields for Create Transaction – User Agent

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json (Default)* *application/dicom+xml* *multipart/related; type="application/dicom+json"* *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.X.1.2 Origin Server

The Create Transaction origin server receives PUT requests to create a modality performed procedure step.

The user agent specifies the Target Resource as part of the URI and the acceptable Content-Type in the HTTP header (i.e. XML or JSON).

The URI is composed by a Base URI: see Base URI for the origin server in Section N.6.3.X.

The Create Transaction origin server supports resources listed in Table N.5.3.X.1.2-1.

[Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.X.1.2-1: Resources for Create Transaction – Origin Server

Resource	Comments
	See Resources path in Table X.1.1-1 in PS3.18
modality-performed-procedure-steps	

The Create Transaction origin server supports Header Fields listed in Table N.5.3.X.1.2-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.X.1.2-2: Header Fields for Create Transaction – Origin Server

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json* *application/dicom+xml* *multipart/related; type="application/dicom+json"*	

	multipart/related; type="application/dicom+xml"	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.X.2 Update Transaction – Modality Performed Procedure Step Service

[If your system does not support the Modality Performed Procedure Step Web Service Update Transaction, you can indicate that this section is not applicable and remove the subsections below.]

N.5.3.X.2.1 User Agent

The Update Transaction user agent can request to update resources listed in Table N.5.3.X.2.1-1.

[List the supported resources for your Modality Performed Procedure Step Update Transaction user agent. Remove the non-supported resources rows. Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.X.2.1-1: Resources for Update Transaction – User Agent

Resource	Comments
	See Resources path in table X.1.1-1 in PS3.18
modality-performed-procedure-steps	

The Update Transaction user agent supports Header Fields listed in Table N.5.3.X.2.1-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.X.2.1-2: Header Fields for Update Transaction – User Agent

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json (Default)* *application/dicom+xml* *multipart/related; type="application/dicom+json"* *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.X.2.2 Origin Server

The Update Transaction origin server receives PATCH requests to update a modality performed procedure step.

The user agent specifies the Target Resource as part of the URI and the acceptable Content-Type in the HTTP header (i.e. XML or JSON).

The URI is composed by a Base URI: see Base URI for the origin server in Section N.6.3.X.

The Update Transaction origin server supports resources listed in Table N.5.3.X.2.2-1.

[Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.X.2.2-1: Resources for Update Transaction – Origin Server

Resource	Comments
	See Resources path in Table X.1.1-1 in PS3.18
modality-performed-procedure-steps	

The Update Transaction origin server supports Header Fields listed in Table N.5.3.X.2.2-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.X.2.2-2: Header Fields for Update Transaction – Origin Server

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json* *application/dicom+xml* *multipart/related; type="application/dicom+json"* *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.X.3 Retrieve Transaction – Modality Performed Procedure Step Service

[If your system does not support the Modality Performed Procedure Step Web Service Retrieve Transaction, you can indicate that this section is not applicable and remove the subsections below.]

N.5.3.X.3.1 User Agent

The Retrieve Transaction user agent can request to retrieve resources listed in Table N.5.3.X.3.1-1.

[List the supported resources for your Modality Performed Procedure Step Retrieve Transaction user agent. Remove the non-supported resources rows. Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.X.3.1-1: Resources for Retrieve Transaction – User Agent

Resource	Comments
	See Resources path in table X.1.1-1 in PS3.18
modality-performed-procedure-steps	

The Retrieve Transaction user agent supports Header Fields listed in Table N.5.3.X.3.1-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.X.3.1-2: Header Fields for Retrieve Transaction – User Agent

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json (Default)* *application/dicom+xml*	

	multipart/related; type="application/dicom+json" *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

N.5.3.X.3.2 Origin Server

The Retrieve Transaction origin server receives GET requests to retrieve a modality performed procedure step.

The user agent specifies the Target Resource as part of the URI and the acceptable Content-Type in the HTTP header (i.e. XML or JSON).

The URI is composed by a Base URI: see Base URI for the origin server in Section N.6.3.X.

The Retrieve Transaction origin server supports resources listed in Table N.5.3.X.3.2-1.

[Fill in information on your implementation in the Comments column when necessary.]

Table N.5.3.X.3.2-1: Resources for Retrieve Transaction – Origin Server

Resource	Comments
	See Resources path in Table X.1.1-1 in PS3.18
modality-performed-procedure-steps	

The Retrieve Transaction origin server supports Header Fields listed in Table N.5.3.X.3.2-2.

[List the supported Header Fields and their supported Values. Fill in information on your implementation in the "Comments" column when necessary.]

Table N.5.3.X.3.2-2: Header Fields for Retrieve Transaction – Origin Server

Header Field	Supported Values	Comments
Content-Type	*application/dicom+json* *application/dicom+xml* *multipart/related; type="application/dicom+json"* *multipart/related; type="application/dicom+xml"*	
Content-Length		*[If Content-Encoding is not present]*
Content-Encoding		*[If Content-Length is not present]*

Add a new subsection on the Modality Workflow Services to section N.7.3.3 DICOM Web Services.

N.7 Network and Media Communication Details

…

N.7.3 Status Codes

…

N.7.3.3 DICOM Web Services

…

N.7.3.3.Y Modality Scheduled Procedure Step Service

N.7.3.3.Y.1 Search Transaction as Origin Server

Table N.7.3.3.Y.1-1 lists the Status Codes that an origin server supports for the Search Transaction of the Modality Workflow Service and the condition in which any of the listed Status Codes is sent.

[Describe below the condition in which the application sends the specific Status Codes in the Search Transaction response as origin server.]

Table N.7.3.3.Y.1-1: Status Codes of Origin Server for Search Transaction

Status	Code	Condition
Success	*200 (OK)*	*Copy table from Section Y when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
	503 (Service Unavailable)	

N.7.3.3.Y.2 Search Transaction as User Agent

Table N.7.3.3.Y.2-1 lists the Status Codes that a user agent supports for the Search Transaction of the Modality Workflow Service and defines the application behavior, when encountering any of the listed Status Codes.

[Describe below the behavior of the application when it receives various Status Codes in the Search Transaction response]

Table N.7.3.3.Y.2-1: Status Codes of User Agent for Search Transaction

Status	Code	Behavior
Success	*200 (OK)*	*Copy table from Section Y when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
*	*Any other code*	

N.7.3.3.X Modality Performed Procedure Step Service

N.7.3.3.X.1 Create Transaction as Origin Server

Table N.7.3.3.X.1-1 lists the Status Codes that an origin server supports for the Create Transaction of the Modality Workflow Service and the condition in which any of the listed Status Codes is sent.

[Describe below the condition in which the application sends the specific Status Codes in the Create Transaction response as origin server.]

Table N.7.3.3.X.1-1: Status Codes of Origin Server for Create Transaction

Status	Code	Condition
Success	*200 (OK)*	*Copy table from Section X when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
	503 (Service Unavailable)	

N.7.3.3.X.2 Create Transaction as User Agent

Table N.7.3.3.X.2-1 lists the Status Codes that a user agent supports for the Create Transaction of the Modality Workflow Service and defines the application behavior, when encountering any of the listed Status Codes.

[Describe below the behavior of the application when it receives various Status Codes in the Create Transaction response]

Table N.7.3.3.X.2-1: Status Codes of User Agent for Create Transaction

Status	Code	Behavior
Success	*200 (OK)*	*Copy table from Section X when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
*	*Any other code*	

N.7.3.3.X.3 Update Transaction as Origin Server

Table N.7.3.3.X.3-1 lists the Status Codes that an origin server supports for the Update Transaction of the Modality Workflow Service and the condition in which any of the listed Status Codes is sent.

[Describe below the condition in which the application sends the specific Status Codes in the Update Transaction response as origin server.]

Table N.7.3.3.X.3-1: Status Codes of Origin Server for Update Transaction

Status	Code	Condition
Success	*200 (OK)*	*Copy table from Section X when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
	503 (Service Unavailable)	

N.7.3.3.X.4 Update Transaction as User Agent

Table N.7.3.3.X.4-1 lists the Status Codes that a user agent supports for the Update Transaction of the Modality Workflow Service and defines the application behavior, when encountering any of the listed Status Codes.

[Describe below the behavior of the application when it receives various Status Codes in the Update Transaction response]

Table N.7.3.3.X.4-1: Status Codes of User Agent for Update Transaction

Status	Code	Behavior
Success	*200 (OK)*	*Copy table from Section X when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
*	*Any other code*	

N.7.3.3.X.5 Retrieve Transaction as Origin Server

Table N.7.3.3.X.5-1 lists the Status Codes that an origin server supports for the Retrieve Transaction of the Modality Workflow Service and the condition in which any of the listed Status Codes is sent.

[Describe below the condition in which the application sends the specific Status Codes in the Retrieve Transaction response as origin server.]

Table N.7.3.3.X.5-1: Status Codes of Origin Server for Retrieve Transaction

Status	Code	Condition
Success	*200 (OK)*	*Copy table from Section X when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
	503 (Service Unavailable)	

N.7.3.3.X.6 Retrieve Transaction as User Agent

Table N.7.3.3.X.6-1 lists the Status Codes that a user agent supports for the Retrieve Transaction of the Modality Workflow Service and defines the application behavior, when encountering any of the listed Status Codes.

[Describe below the behavior of the application when it receives various Status Codes in the Retrieve Transaction response]

Table N.7.3.3.X.6-1: Status Codes of User Agent for Retrieve Transaction

Status	Code	Behavior
Success	*200 (OK)*	*Copy table from Section X when reviewed*
	202 (Accepted)	
Failure	*400 (Bad Request)*	
	409 (Conflict)	
*	*Any other code*	

Changes to NEMA Standards Publications PS 3.4

Update section F.9 with a note of how an SCP knows what SCUs to notify about changes

F.9 Modality Performed Procedure Step Notification SOP Class

The Modality Performed Procedure Step Notification SOP Class is intended for those Application Entities requiring notifications of Modality Performed Procedure Step's changes in state.

An Application Entity may choose to take some actions based upon a notification or request for information but is in no way required to do so.

Note

…

2. The terms IS and PACS used in the previous example are provided for clarification purposes only. This document does not define nor constrain the purpose or role of any IS, PACS or acquisition Application Entity conforming to this Service Class Specification.
3. **It is beyond the scope of the specification to define how the SCP knows about what SCUs to notify about changes. A conceivable way would be to make this a configuration item of the SCP.**

…

Commented [JM50]: Make this a separate CP.

939 ## Changes to NEMA Standards Publications PS 3.6

940 ***There are no new attributes to be added to table 6-1 of section 6.***

941 ## Changes to NEMA Standards Publications PS 3.15

942 ***There are no new attributes to be added to table E.1-1 of annex E.***

943