

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

## *Supplement 248: DICOMweb Send*

*Prepared by:*

**DICOM Standards Committee, Working Group 27**

1812 N. Moore St, Suite 2200

Arlington, VA 22209, USA

Status: April 2026, Public Comment

Developed pursuant to DICOM Work Item 2025-05-B



## Table of Contents

Document History .....	6
Open Issues.....	6
Closed Issues .....	11
Scope and Field of Application .....	12
10 Studies Service and Resources .....	13
10.1 Overview .....	13
10.1.1 Resource Descriptions .....	13
10.1.2 Common Query Parameters .....	15
10.1.3 Common Media Types .....	15
.....	15
10.2 Conformance .....	15
10.3 Transactions Overview .....	16
10.X Send Transaction.....	19
10.X.1 Request.....	19
10.X.1.1 Target Resources.....	19
10.X.1.2 Query Parameters .....	20
10.X.1.3 Request Header Fields.....	21
10.X.1.4 Request Payload .....	21
10.X.2 Behavior .....	21
10.X.3 Response .....	21
10.X.3.1 Status Codes .....	21
10.X.3.2 Response Header Fields.....	22
10.X.3.3 Response Payload .....	22
10.Y Check Send Result Transaction .....	23
10.Y.1 Request.....	23
10.Y.1.1 Target Resource.....	23
10.Y.1.2 Query Parameters .....	23
10.Y.1.3 Request Header Fields.....	23
10.Y.1.4 Request Payload .....	23
10.Y.2 Behavior .....	23
10.Y.3 Response .....	23
10.Y.3.1 Status Codes .....	24
10.Y.3.2 Response Header Fields.....	24
10.Y.3.3 Response Payload .....	24
12 Non-Patient Instance Service and Resources.....	25
12.1 Overview .....	25
12.1.1 Resource Descriptions .....	25
12.1.2 Common Query Parameters .....	26
12.1.3 Common Media Types .....	26
12.2 Conformance .....	26
12.3 Transactions Overview .....	27
12.X Send Transaction.....	29
12.X.1 Request.....	29
12.X.1.1 Target Resources.....	29
12.X.1.2 Query Parameters .....	29
12.X.1.3 Request Header Fields.....	30
12.X.1.4 Request Payload .....	30
12.X.2 Behavior .....	30
12.X.3 Response .....	31

12.X.3.1	Status Codes .....	31
12.X.3.2	Response Header Fields .....	31
12.X.3.3	Response Payload .....	32
12.Y	Check Send Result Transaction .....	33
12.Y.1	Request .....	33
12.Y.1.1	Target Resource .....	33
12.Y.1.2	Query Parameters .....	33
12.Y.1.3	Request Header Fields .....	33
12.Y.1.4	Request Payload .....	33
12.Y.2	Behavior .....	33
12.Y.3	Response .....	33
12.Y.3.1	Status Codes .....	34
12.Y.3.2	Response Header Fields .....	34
12.Y.3.3	Response Payload .....	34
4	Symbols and Abbreviated Terms .....	35
B	Examples (Informative) .....	36
B.X1	Request to Send all Studies of a Patient to Another Server .....	36
B.X2	Flow: Handling of a Send Request .....	37
B.X3	Bi-Directional Proxy for Send Transaction .....	39
H	Capabilities Description .....	43
Annex @	Send Request Response Module .....	48
@.1	Response Message Body .....	48
C.4.2	C-MOVE Operation .....	49
C.4.2.1	C-MOVE Service Parameters .....	49
C.4.2.1.1	SOP Class UID .....	49
C.4.2.1.2	Priority	49
C.4.2.1.3	Move Destination .....	49
C.4.2.1.4	Identifier	49
C.4.2.1.4.1	Request Identifier Structure .....	49
C.4.2.1.4.2	Response Identifier Structure .....	50
C.4.2.1.5	Status	50
C.4.2.1.6	Number of Remaining Sub-Operations .....	51
C.4.2.1.7	Number of Completed Sub-Operations .....	51
C.4.2.1.8	Number of Failed Sub-Operations .....	52
C.4.2.1.9	Number of Warning Sub-Operations .....	52
C.4.2.2	C-MOVE SCU Behavior .....	52
C.4.2.2.1	Baseline Behavior of SCU .....	52
C.4.2.2.2	Extended Behavior of SCU .....	52
C.4.2.2.2.1	Relational-Retrieve .....	53
C.4.2.2.2.2	Enhanced Multi-Frame Image Conversion .....	53
C.4.2.3	C-MOVE SCP Behavior .....	53
C.4.2.3.1	Baseline Behavior of SCP .....	53
C.4.2.3.1.1	Alternative Sub-operation Mechanism Option .....	54
C.4.2.3.1.1.1	Scope .....	54
C.4.2.3.1.1.2	Interface .....	54
C.4.2.3.1.1.3	Behavior .....	55
C.4.2.3.1.1.4	Sub-operation Counters .....	55
C.4.2.3.1.1.5	Mixed-mechanism Delivery .....	55
C.4.2.3.1.1.6	Pending Responses .....	55
C.4.2.3.1.1.7	Mixed-mechanism Failure Retry .....	55
C.4.2.3.1.1.8	Backward Compatibility .....	55
C.4.2.3.1.1.9	Conformance .....	56
C.4.2.3.1.2	Proxy Option .....	56

	<u>C.4.2.3.1.2.1 Scope</u> .....	56
	<u>C.4.2.3.1.2.2 Interface</u> .....	56
	<u>C.4.2.3.1.2.3 Behavior</u> .....	56
	<u>C.4.2.3.1.2.3 Conformance</u> .....	56
	C.4.2.3.2 Extended Behavior of SCP .....	56
	C.4.2.3.2.1 Relational-Retrieve .....	57
	C.4.2.3.2.2 Enhanced Multi-Frame Image Conversion .....	57
N.1	Overview .....	58
N.1.3	DICOM Web Services .....	58
N.1.3.2	Studies Service .....	58
N.1.3.4	Non-Patient Instance Service .....	60
N.5	Service and Interoperability Description .....	61
N.5.2	DIMSE Services .....	61
N.5.2.7	Query/Retrieve Service Class .....	61
N.5.2.7.7	SCP of the Patient Root Q/R Information Model - MOVE SOP Class 61	
N.5.2.7.8	SCP of the Study Root Q/R Information Model - MOVE SOP Class 61	
N.5	Service and Interoperability Description .....	62
N.5.3	DICOM Web Services .....	62
N.5.3.2	Studies Web Service .....	62
N.5.3.2.X	<u>Send Transactions (SEND-RS)</u> .....	62
N.5.3.2.X.1	<u>User Agent</u> .....	62
N.5.3.2.X.2	<u>Origin Server</u> .....	62
N.5.3.4	Non-Patient Instance Web Service .....	62
N.5.3.4.X	<u>Send Transactions</u> .....	62
N.5.3.4.X.1	<u>User Agent</u> .....	62
N.5.3.4.X.2	<u>Origin Server</u> .....	63
N.6.3	Configuration of DICOM Web Services .....	63
N.6.3.2	Studies Web Service Configuration .....	63
N.6.3.2.X	<u>Send Transactions (SEND-RS) Configuration</u> .....	63
N.7	Network and Media Communication Details .....	64
N.7.3	Status Codes .....	64
N.7.3.3	DICOM Web Services .....	64
N.7.3.3.3	Studies Service .....	64
N.7.3.3.3.X1	<u>Send Transaction as Origin Server</u> .....	64
N.7.3.3.3.X2	<u>Send Transaction as User Agent</u> .....	65
N.7.3.3.3.X3	<u>Check Send Result Transaction as Origin Server</u> .....	65
N.7.3.3.3.X4	<u>Check Send Result Transaction as User Agent</u> .....	66
N.7.3.3.5	Non-Patient Instance Service .....	66
N.7.3.3.5.X1	<u>Send Transactions as Origin Server</u> .....	66
N.7.3.3.5.X2	<u>Send Transactions as User Agent</u> .....	67
N.7.3.3.5.X3	<u>Check Send Result Transaction as Origin Server</u> .....	67
N.7.3.3.5.X4	<u>Check Send Result Transaction as User Agent</u> .....	68

1

## Document History

2025.06	Version 00	JM	Initial version with proposed approach, document structure and content.
2025.09	Version 01	JM	Reworked comments from WG06 2025.06's meeting. Elaborated Open and Closed Issues, added sections on Examples and Conformance.
2025.11	Version 02	JM	Reworked comments from WG06 2025.09's meeting. Accepted all changes, removed resolved comments, added section on C-MOVE, resolved open comments, added new comments, added some security considerations.
2026.01	Version 03	JM	Reworked comments from WG06 2025.11's meeting. Accepted all changes, removed resolved comments, resolved open comments, removed security considerations (as that is a separate CP now).
2026.03	Version 04	JM	Reworked comments from WG06 2026.01's meeting. Accepted all changes, removed resolved comments, resolved open comments. Added two new comments.
2026.04	Version 05 Public Comment	JM	Reworked comments from WG06 2026.03 meeting; also added several open issues, and incorporated security-related texts.

2

## Open Issues

1	<p><b>Issue:</b> How to name the transactions and the supplement. What should be the colloquial name?</p> <p><b>Context:</b> In DIMSE, the operation for which a DICOMweb equivalent is to be defined is called C-MOVE. While this is a misnomer, it has been suggested to seize the opportunity – also or perhaps even especially for newcomers to DICOM – and name it properly.</p> <p>Comment: Send is a residual alternative name for C-STORE in ACR-NEMA 1985, so that is perhaps not the best name.</p> <p><b>Proposal:</b> There are several contenders: Move, Forward, Send, Provide, Deliver, Copy, .... It is currently proposed to name the supplement DICOMweb Send, and use the word Send in the transactions too. The colloquial name is proposed to be SEND-RS.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>
2	<p><b>Issue:</b> How to deal with multiple responses from the origin server to the user agent?</p> <p><b>Context:</b> DIMSE's C-MOVE allows for multiple (progress) responses by the SCP decided by the SCP. As DICOMweb is RESTful and hence transactional (request-response), how to deal with that aspect? There are several options:</p> <ul style="list-style-type: none"> <li>• <b>Combine them</b> – This follows the pattern used in the Search transaction. When a proxy of an origin server would convert a Search transaction to a C-FIND operation on an SCP, the pending results would need to be combined into one response; the proxy should also manage the limit and offset, and canceling the C-FIND when appropriate. Disadvantage: when applied to the forward transaction there would be no way to monitor progress.</li> <li>• <b>Enforce multipart responses</b> – This exploits a technical option in HTTP. When a proxy of an origin server would convert a Send transaction to a C-MOVE operation on an SCP, each pending result would need to be transformed into a part of the multi-part response, and a success/warning/failure result would yield the final part. Disadvantages:</li> </ul>

	<p>this a) does not follow existing DICOMweb interface patterns, b) does not solve the time-out issue, c) requires a somewhat more complex user agent, handling multiple HTTP parts separately instead of handling one HTTP payload.</p> <ul style="list-style-type: none"> <li>• <b>Poll for responses</b> – This follows the pattern used in the Storage Commitment service. When a proxy of an origin server would convert a Send transaction to a C-MOVE operation on an SCP, the last Pending result needs to be kept until the user agent asks for the response; at that time, a converted response can be returned. A difference with Storage Commitment is that there would be a payload (progress status) at every poll request, while Storage Commitment only provides a payload when done. Disadvantage: monitoring progress is by request of the user agent, while in DIMSE it is determined by the SCP.</li> <li>• <b>Use notifications</b> – The origin server can notify the user agent on progress, utilizing the notifications that are part of PS3.18. This follows a technical option that is currently available. When a proxy of an origin server would convert a Send Transaction to a C-MOVE, it would need to provide a subscription service for notifying the user agent, incl. extending the send transaction interface with a transaction UID. Disadvantage: requires subscription service by origin server, subscription by user agent, handling of multiple protocols by both parties.</li> <li>• <b>Long GET/long polling</b> – This follows a pattern described in RFC6202. When a proxy of an origin sever would convert a Send Transaction to a C-MOVE, it would keep the connection to the user agent open until the (first) status message that arrives is sent to the user agent, after which this – in case not all sub-operations are yet done – requests a new status. Disadvantages: yet another pattern in PS3.18 to deal with multiple messages from an SCP; seems to be old-fashioned; requires specific handling of time-outs (which are more likely).</li> </ul> <p><b>Proposal:</b> Multiple responses are handled by polling for responses, as this best follows the existing DICOMweb interface patterns (see e.g. Storage Commitment in PS3.18, Section 13). Furthermore, this approach provides a solution to the HTTP time-out issue and avoids multi-part responses.</p> <p><b>Decision:</b> [WG06-20250616] Agreed, yet reopened at WG06-20250901.</p>
4	<p><b>Issue:</b> How should the user agent identify the destination to the origin server in this transaction?</p> <p><b>Context:</b> In DIMSE, the destination of a C-MOVE is an AE Title, so in DICOMweb it would make most sense to use an HTTP endpoint (the URI of a DICOMweb Studies service that supports the Store Transaction) as destination.</p> <p>However, as an origin server can decide to use C-MOVE for its Store Transactions (see section 10.X), and a C-MOVE SCP can decide to use the Store Transaction for its sub-operations (see open issue 9 and updated section C.4.2 of PS3.4 below) the destination can, in principle, be either an AE Title or a base URI. In either case, there should be a mapping from one to another on such a server. Another option is to allow for either an endpoint or an AE Title as destination. That, however, gives the user agent the responsibility to choose, while the origin server is better positioned to do that. The chosen protocol of the sub-operation should in fact be irrelevant to the user agent.</p> <p>Using an HTTP endpoint as destination makes most sense from an API point of view: the web-world of DICOMweb is preferably mimicked into the parameters.</p> <p><b>Proposal:</b> Use HTTP endpoint as destination, unless the security considerations imply to use the AE Title instead.</p> <p><b>Decision:</b> [WG06-20250616] Agreed, yet reopened at WG06-20250901.</p>
5	<p><b>Issue:</b> Can we learn from others w.r.t. the approach to third party push use cases?</p> <p><b>Context:</b> DIMSE's C-MOVE operations allows for pushing instances to a third party: system A (the requester) asks system B (the data holder) to send certain instances to system C (the third-party destination). Do other use cases like this exist in the realm of HTTP and</p>

	<p>RESTful, and how have these been approached? That may provide input to the approach to be taken for the DICOMweb's C-MOVE equivalent.</p> <p>Some research provided us with examples like:</p> <ul style="list-style-type: none"> <li>• Cloud storage, where a client requests a cloud service to transfer data from one bucket or region to another, or even to another account.</li> <li>• Content Delivery Networks, where a system instructs a CDN to fetch and cache content from an origin server (and serve it to end users)<sup>1</sup>.</li> <li>• Genomics Data Pipeline, where a research portal requests a sequencing center to send genomics data to a bioinformatics platform.</li> <li>• Satellite Ground Station Networks, where a mission control system requests a ground station to forward telemetry data to a processing center.</li> </ul> <p><b>Proposal:</b> The two-transaction model, where there is a transaction for requesting the transfer and one for checking the progress or status, is the most common approach in the RESTful world. There are other approaches too, for instance a) utilizing webhooks (with callback URLs) instead of having to poll or b) utilizing GraphQL mutations for initiating transfers and subscriptions for status updates. As the alternative approaches are beyond what is typically done in DICOMweb, the text proposes to use the two-transaction model.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>
6	<p><b>Issue:</b> What should be the Group Number of the Attributes in the Send Request Response Module?</p> <p><b>Context:</b> Within DIMSE's C-MOVE, (intermediate) results – like number of remaining operations and list of failed SOP instances – are largely conveyed as parameters at the message exchange / command level. How should these be presented in DICOMweb, esp. can we reuse the tags of the message fields, or should we introduce new ones?</p> <p>The attribute tags of the Status, Number of Remaining Sub-operations, Number of Completed Sub-operations, Number of Failed Sub-operations, and Number of Warning Sub-operations as currently available have 0000 as Group. Although not stated literally, this Group is meant to be used in commands only. Hence, other tags with the same semantics should be created. However, this would require a proxy to re-encode these values. An exception to the implied rule could be made here, as the Send Request Response Module is not to be sent via DIMSE.</p> <p><b>Proposal:</b> It seems peculiar to introduce new tags for conveying the same, so it is proposed to use the tags that are already available.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>
7	<p><b>Issue:</b> Is there any other applicable pattern for doing requests?</p> <p><b>Context:</b> This question relates to issue 2, but this one is about a specific detail, namely that requests are to be dealt with as resources, as currently is done in Section 13, while mostly DICOMweb is about data resources.</p> <p><b>Proposal:</b> As there are several uses of resources within DICOMweb beyond that of data resources, see also Sections 11.7 (Change Workitem State) and 11.8 (Request Cancellation), and the approach is in line with that of Section 13, it does not seem wise to use yet another pattern.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>

<sup>1</sup> Actually, this is a reverse use case, as it concerns fetching data. Yet, there are three parties involved, and it can be considered a kind of 'pull from third party' instead of 'push to third party' use case that is currently not supported in DICOM.

8	<p><b>Issue:</b> Add colloquial naming for the transaction?</p> <p><b>Context:</b> Currently we have STOW-RS, WADO-RS and QIDO-RS as colloquial names for the respective Studies service transactions. Should we have something like that for Send Transactions too?</p> <p><b>Proposal:</b> Marketing terms make sense to ease communication, so it is proposed to add SEND-RS as colloquial name for this transaction.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>
9	<p><b>Issue:</b> Do we want to allow C-MOVE to be able to use Store Transactions (STOW) for its sub-operations, as we allow DICOM Send to be able to use C-STORE operations?</p> <p><b>Context:</b> Currently, C-MOVE must execute C-STORE sub-operations to get the request done. In hybrid eco-systems it may be needed to allow the SCP to use another means to handle the C-MOVE request. Use cases for this addition comprise:</p> <ul style="list-style-type: none"> <li>• A group of cooperating hospitals that allow DIMSE within a hospital, but do not allow DIMSE to cross the firewall to another hospital, while DICOMweb would be no issue.</li> <li>• For supporting easy access to studies for second opinions, a hospital wants to utilize cloud buckets from a medical cloud service. These services typically only support DICOMweb.</li> </ul> <p><b>Proposal:</b> The current text of this supplement proposes to allow this; see PS3.4, Section C.4.2.3.1.1. Alternative Sub-operation Mechanism Option. Note that here a C-MOVE SCP is required to support C-STORE operations and may support DICOMweb Store Transactions.</p> <p><b>Decision:</b> [WG-06-yyyymmdd]</p>
10	<p><b>Issue:</b> How to document support of the Send Transaction in the conformance statement?</p> <p><b>Context:</b> The Conformance Statements template needs additions for the Send Transaction. These sections should be N.5.3.2.X and N.5.3.4.X respectively. Currently, the content of these sections has been modelled after N.5.2.7.3 and N.5.2.7.4 (C-MOVE) respectively. Is this correct, or should it be modelled similar to N.5.3.2.4 (Search Transaction (QIDO-RS))?</p> <p><b>Proposal:</b> Keep as is.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>
11	<p><b>Issue:</b> Should we use URL or URI?</p> <p><b>Context:</b> Is the identification of other parties in HTTP to be done using URLs or URIs? This question is broader than the scope of this supplement.</p> <p><b>Proposal:</b> As this is broader than this supplement, this issue actually needs to be resolved elsewhere.</p> <p><b>Decision:</b> [WG06-yyyymmdd]</p>
12	<p><b>Issue:</b> Do we want to be able to create a DICOM conformant C-MOVE SCP proxy (proxying to DICOMweb Send), as we allow a DICOM conformant Send origin server proxy (proxying to DIMSE C-MOVE)?</p> <p><b>Context:</b> In hybrid environments, one may want to utilize a proxy SCP that handles C-MOVE requests by transforming C-MOVE request content to DICOMweb content, and do a DICOM Send request using that content at an applicable origin server. Currently, such proxy cannot be DICOM conformant, as C-MOVE SCPs shall, for instance, perform C-STORE sub-operations and maintain counters.</p> <p><b>Proposal:</b> The current text of this supplement proposes to allow this; see PS3.4, Section C.4.2.3.1.2. Proxy Option. Note that when claiming this option, the C-MOVE SCP is no longer required to support C-STORE sub-operations, as is does not execute sub-operations itself.</p> <p><b>Decision:</b> [WG-06-yyyymmdd]</p>

13	<p><b>Issue:</b> Do we need to require HTTPS?</p> <p><b>Context:</b> This came up during the discussion on the alternative sub-operation mechanism option in WG06. The text originally used HTTPS, and changing this triggered this issue.</p> <p><b>Proposal:</b> Do not require HTTPS. Security is orthogonal to this supplement, and will be dealt with in a separate Supplement.</p> <p><b>Decision:</b> [WG-06-yyyyymmdd]</p>
14	<p><b>Issue:</b> How to deal with cancellation in a C-MOVE SCP proxy?</p> <p><b>Context:</b> While we have excluded cancelling DICOMweb Send (see issue 3), allowing a proxy that translates C-MOVE to DICOMweb Send will need to handle C-CANCEL-MOVE-RQ, and map that to DICOMweb behavior somehow. The behavior of the SCP when receiving a C-CANCEL-MOVE is described as:</p> <p style="padding-left: 40px;">The SCP may receive a C-CANCEL-MOVE request at any time during the processing of the C-MOVE. The SCP shall interrupt all C-STORE sub-operation processing and return a status of Cancel in the C-MOVE response. The C-MOVE response with a status of Cancel shall contain the number of Completed, Failed, and Warning C-STORE sub-operations. If present, the Remaining sub-operations count shall contain the number of C-STORE sub-operations that were not initiated due to the C-CANCEL-MOVE request.</p> <p>While this text raises questions too (why are C-STORE sub-operations that have been initiated but not yet finished excluded from the counters), it is clear on what to do: interrupt the execution of sub-operations and return with status Cancel.</p> <p>A few options have been identified to deal with this in a proxy:</p> <ol style="list-style-type: none"> <li>1. Reopen closed issue 3 and do add cancellation to DICOMweb Send; this would require a new transaction, e.g. Cancel Send. The receipt of a Cancel Send request would require the DICOMweb Send origin server to interrupt all sub-operations as described above and return a proper Send Request Response Module with status Cancel.</li> <li>2. Mostly ignore the C-CANCEL-MOVE-RQ and continue to work. Yet, do return the status Cancel. While this continues to Send all matching Instances (and hence does not interrupt the sub-operations at all), this does not require a new Transaction to be defined and implemented and does not reopen the generic cancellation discussion. One could argue that continuing the sub-operations is not that bad, as the C-MOVE SCU does not know the effect of the cancellation; the C-MOVE could just have been finished.</li> <li>3. Disconnect from the DICOMweb Send origin server, returning the last known status regarding the counters. This is the sticking-your-head-in-the-sand approach, as the sub-operations will probably just continue, but no result reflecting the status can be returned, esp. regarding the counters.</li> <li>4. Intentionally do not prescribe the proxy behavior in the standard and let implementations describe their behavior in the conformance statement.</li> </ol> <p>Another option would be to not allow for DICOM conformant C-MOVE proxies as described in the text of the supplement and in issue 12.</p> <p>Feedback proposing other options is appreciated, as none of the above options are ideal.</p> <p><b>Proposal:</b> Select option 2 above, as that is the least invasive for both the standard and the implementations, still allowing DICOM conformant proxies, useful in hybrid eco-systems.</p> <p><b>Decision:</b> [WG-06-yyyyymmdd]</p>
15	<p><b>Issue:</b> Should we combine Send and Check Send Result in PS3.2, Table N.1-9?</p> <p><b>Context:</b> Table N.1-9 in PS3.2 gives an overview of the DICOMweb Services provided by &lt;product&gt;. While currently this table has no combined Transactions, it makes sense to do so as to make it easier for the reader and to prevent mistakes by the writer (allowing some resources to be available for the Send and not for the Check Send Result or vice versa). The same also holds for Table N.1-11.</p> <p><b>Proposal:</b> Combine the two into one row.</p> <p><b>Decision:</b> [WG-06-yyyyymmdd]</p>

3

### Closed Issues

3	<p><b>Issue:</b> How to deal with cancelling DICOMweb Send?</p> <p><b>Context:</b> Within DIMSE, it is possible to cancel C-MOVE, and the other basic operations C-FIND and C-GET using the C-operation-CANCEL-RQ. This has not been mimicked to DICOMweb: currently no Transaction can be cancelled. UPS-RS allows the cancellation of a workitem, but that is at application level and does not cancel a Transaction.</p> <p><b>Proposal:</b> Canceling the send request transaction will, for now, not be supported. When the need arises, a New Work Item Proposal for cancelling Transactions can be created. This would need to include cancelling Searching, Committing Storage, Sending, ...</p> <p><b>Decision:</b> [WG06-20250616] Agreed.</p>
16	<p><b>Issue:</b> How to register the tag numbers in the Send Request Response Module?</p> <p><b>Context:</b> The tags of this module have been numbered according to the proposal of issue 6. Now these numbers are not present in PS3.6, and originally this supplement proposed to add them to Table 6-1. However, this may conflict with their presence in PS3.7, Annex E. Command Dictionary.</p> <p><b>Proposal:</b> Do not add these tags to PS3.6.</p> <p><b>Decision:</b> Proposed by Rob Horn.</p>

4

5

6

## Scope and Field of Application

7 This supplement adds Send Transactions to DICOMweb's Studies and Non-Patient Instances Services to  
8 mirror the C-MOVE operation that is already available in DIMSE. The Send Transactions have been de-  
9 signed with the intention of facilitating proxies from/to DIMSE. Furthermore, it defines a C-MOVE option to  
10 allow for STOW sub-operations and an option to allow for C-MOVE SCP proxies, delegating to a DICOM-  
11 web Send origin server.

12 Security plays an important role for this supplement, as this supplement defines the three party DICOM-  
13 web Send Transaction, and provides means to cross two DICOM protocols, i.e., between DICOMweb and  
14 DIMSE, at both the operation level and the sub-operation level. Yet, DICOM considers security orthogo-  
15 nal to functionality, and hence applicable security mechanisms and considerations are described in a sep-  
16 arate supplement.

17

## Changes to NEMA Standards Publications PS 3.18

**Update first part of section 10 Studies Service and Resources as indicated below**

### 10 Studies Service and Resources

#### 10.1 Overview

The Studies Resource enables a user agent to store, retrieve, update, and search an origin server for DICOM Studies, Series, and Instances, along with their /metadata, /rendered, and /thumbnail variants; as well as Frames and Bulkdata. **It also enables a user agent to request an origin server to send DICOM Studies, Series, and Instances to another server.**

Note It is not possible to request to send selected Frames.

The Retrieve transaction of this Service is also known as WADO-RS. The Store transaction of this Service is also known as STOW-RS. The Search transaction of this Service is also known as QIDO-RS. **The Send transactions of this Service are collectively known as SEND-RS.** See Section 10.3.

#### 10.1.1 Resource Descriptions

The Studies Service manages a collection of DICOM Study resources. Each Study is organized in a hierarchy of sub-resources that correspond to the DICOM Information Model. See Section 7 “DICOM Model of the Real World” in PS3.3.

There are three top level resources:

/studies references all Studies managed by the service.

/series references all Series managed by the service.

/instances references all Instances managed by the service.

The following URI Template variables are used in resource definitions in this Section.

{study} the Study Instance UID of a Study managed by the Studies Service.

{series} the Series Instance UID of a Series contained within a Study resource.

{instance} the SOP Instance UID of an Instance contained within a Series resource.

{frames} a comma-separated list of frame numbers, in ascending order, contained within an Instance.

{bulkdataURI} an opaque URI that references a Bulkdata Value.

The Studies Service defines the following resources:

**Table 10.1-1. Resources and Descriptions**

Resource	Description
Studies Service	The Base URI of the Studies Service.
All Studies	The All Studies resource references the entire collection of Studies contained in the Studies Service.

Resource	Description
Study	The Study resource references a single Study.
Study Metadata	The Study Metadata resource references the Metadata of a Study.
Study Bulkdata	The Study Bulkdata resource references the Bulkdata of a Study.
Study Pixel Data	The Study Pixel Data resource references the Pixel Data of a Study.
Rendered Study	The Rendered Study resource references an alternate Media Type rendering of a Study.
Rendered MPR Volume Study	The Rendered MPR Volume Study resource references a multiplanar reformat rendering of a Study.
Rendered 3D Volume Study	The Rendered 3D Volume Study resource references a volume rendering of a Study.
Study Thumbnail	The Study Thumbnail resource references a thumbnail image of a Study.
Study's Series	The Study's Series resource references the collection of all Series contained in a Study.
Study's Instances	The Study's Instances resource references the collection of all Instances in a single Study.
<b><u>Study's Send Requests</u></b>	<b><u>The Study Send Requests resource references the collection of send requests associated with a single study.</u></b>
All Series	The All Series resource references the collection of all Series in all Studies contained in the Studies Service.
Series	The Series resource references a single Series.
Series Metadata	The Series Metadata resource contains the Metadata of a Series in a Study.
Series Bulkdata	The Series Bulkdata resource references the Bulkdata of a Series.
Series Pixel Data	The Series Pixel Data resource references the Pixel Data of a Series.
Rendered Series	The Rendered Series resource references an alternate Media Type rendering of a Series.
Rendered MPR Volume Series	The Rendered MPR Volume Series resource references a multiplanar reformat rendering of a Series.
Rendered 3D Volume Series	The Rendered 3D Volume Series resource references a volume rendering of a Series.
Series Thumbnail	The Series Thumbnail resource references a thumbnail image of a Series.
Series' Instances	The Series' Instances resource references the collection of all Instances in a single Series.
<b><u>Series' Send Requests</u></b>	<b><u>The Series' Send Requests resource references the collection of send requests associated with a single Series.</u></b>
All Instances	The All Instances resource references the collection of all Instances in all Series in all Studies contained in the Studies Service.
Instance	The Instance resource references a single Instance.

Resource	Description
Instance Metadata	The Instance Metadata resource contains the Metadata of an Instance.
Instance Bulkdata	The Instance Bulkdata resource references the Bulkdata of a Instance.
Instance Pixel Data	The Instance Pixel Data resource references the Pixel Data of a Instance.
Rendered Instance	The Rendered Instance resource references an alternate Media Type rendering of an Instance.
Rendered MPR Volume Instance	The Rendered MPR Volume Instance resource references a multiplanar reformat rendering of an Instance.
Rendered 3D Volume Instance	The Rendered 3D Volume Instance resource references a volume rendering of an Instance.
Instance Thumbnail	The Instance Thumbnail resource references a thumbnail image of an Instance.
<b><u>Instance's Send Requests</u></b>	<b><u>The Instance's Send Requests resource references the collection of send requests associated with a single Instance.</u></b>
Frames	The Frames resource references an ordered collection of frames in a single multi-frame Instance.
Rendered Frames	The Rendered Frames resource references an alternate Media Type rendering of an ordered collection of frames of a single multi-frame Instance.
Rendered MPR Volume Frames	The Rendered MPR Volume Frames resource references a multiplanar reformat rendering of a collection of frames.
Rendered 3D Volume Frames	The Rendered 3D Volume Frames resource references a volume rendering of a collection of frames.
Frame Thumbnail	The Frame Thumbnail resource references a thumbnail image for frames within an Instance.
Bulkdata	The Bulkdata resource contains a Bulkdata Value.

47

48 **10.1.2 Common Query Parameters**

49 ...

50 **10.1.3 Common Media Types**

51 ...

52 **10.2 Conformance**

53 An origin server claiming conformance to **one or more of** the **Retrieve-Transactions** of the Studies Service:

- 54 • shall support the Retrieve Capabilities Transaction (see Section 8.9.1);
- 55 • shall support these **Retrieve Transactions** for **all their respective** mandatory resources in Table 10.3-2.

56 ~~An origin server claiming conformance to the Store Transaction of the Studies Service:~~

- 57 ~~• shall support the Retrieve Capabilities Transaction (see Section 8.9.1);~~
- 58 ~~• shall support the Store Transaction for all mandatory resources in Table 10.3-2.~~

59 ~~An origin server claiming conformance to the Search Transaction of the Studies Service:~~

- 60 ~~• shall support the Retrieve Capabilities Transaction (see Section 8.9.1);~~  
 61 ~~• shall support the Search Transaction for all mandatory resources in Table 10.3-2.~~

62 **An origin server shall claim conformance to either both or none of the Send Transaction and the Check Send**  
 63 **Result Transaction.**

64 The user agent may support any of the transactions for any of the corresponding resources in Table 10.3-2.

65 **10.3 Transactions Overview**

66 The Studies Service consists of the transactions listed in Table 10.3-1.

67 **Table 10.3-1. Studies Service Transactions**  
 68

Transaction Name	Method	Payload		Description
		Request	Success Response	
Retrieve	GET	N/A	Instance(s), Metadata, Renderings, Pixel Data, or Bulkdata	Retrieve one or more representations of DICOM Resources.
Store	POST	Instance(s)	Store Instances Response Module	Stores one or more representations of DICOM Resources, contained in the request payload, in the location referenced by the Target Resource.
Search	GET	N/A	Result(s)	Searches the Target Resource for DICOM objects that match the search parameters and returns a list of matches in an Acceptable Media Type.
<b><u>Send</u></b>	<b><u>POST</u></b>	<b><u>N/A</u></b>	<b><u>Send Request Response Module</u></b>	<b><u>Searches the Target Resource for DICOM objects that match the search parameters and sends or starts to send these to another server.</u></b>
<b><u>Check Send Result</u></b>	<b><u>GET</u></b>	<b><u>N/A</u></b>	<b><u>Send Request Response Module</u></b>	<b><u>Gets the status and results of a Send transaction.</u></b>

69 In Table 10.3-2, the Target Resources permitted for each transaction are marked with M if support is man-  
 70 datory for the origin server and O if it is optional. A blank cell indicates that the resource is not allowed in  
 71 the transaction.

72 **Table 10.3-2. Resources by Transaction**

Resource	Retrieve	Store	Search	<b><u>Send</u></b>	<b><u>Check Send Result</u></b>
Studies Service					
All Studies		M	M		
Study	M	M			
Study Metadata	M				

<b>Resource</b>	<b>Retrieve</b>	<b>Store</b>	<b>Search</b>	<b><u>Send</u></b>	<b><u>Check Send Result</u></b>
Study Bulkdata	O				
Study Pixel Data	O				
Rendered Study	M				
Rendered MPR Volume Study	O				
Rendered 3D Volume Study	O				
Study Thumbnail	O				
Study's Series			M		
Study's Instances			M		
<b><u>Study's Send Requests</u></b>				<b><u>M</u></b>	<b><u>M</u></b>
All Series			M		
Series	M				
Series Metadata	M				
Series Bulkdata	O				
Series Pixel Data	O				
Series' Instances			M		
<b><u>Series' Send Requests</u></b>				<b><u>M</u></b>	<b><u>M</u></b>
Rendered Series	M				
Rendered MPR Volume Series	O				
Rendered 3D Volume Series	O				
Series Thumbnail	O				
All Instances			M		
Instance	M				
Instance Metadata	M				
Instance Bulkdata	O				
Instance Pixel Data	O				
Rendered Instance	M				

<b>Resource</b>	<b>Retrieve</b>	<b>Store</b>	<b>Search</b>	<b><u>Send</u></b>	<b><u>Check Send Result</u></b>
Rendered MPR Volume Instance	O				
Rendered 3D Volume Instance	O				
Instance Thumbnail	O				
<b><u>Instance's Send Requests</u></b>				<b><u>M</u></b>	<b><u>M</u></b>
Frames	M				
Rendered Frames	M				
Rendered MPR Volume Frames	O				
Rendered 3D Volume Frames	O				
Frame Thumbnail	O				
Bulkdata	M	M			

73

74 ...

75

**Add new section 10.X Send Transaction, immediately before new Section 10.Y given below**

**10.X Send Transaction**

This Transaction uses the POST method to request the sending of Studies, Series, or Instances managed by the origin server to a destination server.

Note The user agent could be the destination server of this transaction.

Together with the Check Send Result Transaction, Section 10.Y, this Transaction corresponds to the DIMSE C-MOVE Operation (see PS3.4, Section C.4.2), but not supporting cancellation.

Due to the typically long-running nature of sending Studies, Series, or Instances to another server, the origin server may choose between two alternatives to communicate about the status and outcome of the request. When the outcome of the request is immediately available, the origin server typically returns the status and outcome in the response to the request. When the outcome of the request is not available immediately, the origin server indicates that to the client in its response, and the client uses the Check Send Result Transaction. This pattern of Transactions is similar to what has been illustrated in Figure 13.1-1.

The Store sub-operations that are necessary to fulfil the send request may be executed using DIMSE or DICOMweb, or even a combination of the two. The term sub-operations will henceforth be used for both DIMSE's sub-operations and DICOMweb's sub-transactions.

**10.X.1 Request**

The request shall have the following syntax:

```
POST SP "/" {resource} "/send-requests/" {transactionUID} "?" destination {&search*} SP version CRLF
Accept: 1#media-type
*(header-field CRLF)
CRLF
```

where

destination = "destination" "=" {URL}

and where {URL} is a percent-encoded absolute URL of the destination endpoint; see [RFC 3986].

**10.X.1.1 Target Resources**

The Target Resource Path component of the Target URI specifies the collection of resources that is the target of the send request.

An origin server that is a native implementation shall support all Mandatory (M) resources specified in the Native column in Table 10.X.1-1.

An origin server that is a DIMSE Proxy implementation shall support all Mandatory (M) resources specified in the Proxy column in Table 10.X.1-1.

**Table 10.X.1-1. Send Transaction Resources**

Resource	URI Template	Native	Proxy	Query Type
All Studies Send Requests	/studies/send-requests/{transactionUID}	M	M	hierarchical
Study's Series Send Requests	/studies/{study}/series/send-requests/{transactionUID}	M	M	hierarchical

Resource	URI Template	Native	Proxy	Query Type
Study's Instances Send Requests	/studies/{study}/instances/send-requests/{transactionUID}	M	O	relational
All Series Send Requests	/series/send-requests/{transactionUID}	M	O	relational
Study Series' Instances Send Requests	/studies/{study}/series/{series}/instances/send-requests/{transactionUID}	M	M	hierarchical
All Instances Send Requests	/instances/send-requests/{transactionUID}	M	O	relational

109

110 For more information about Hierarchical Queries see Section C.4.1.3.1.1 "Hierarchical Search Method" in  
 111 PS3.4. For more information about Relational Queries see Section C.4.1.2.2.1 "Relational-Queries" in  
 112 PS3.4 and Section C.4.1.3.2.1 "Relational-Queries" in PS3.4.

113 Table 10.X.1-2 shows the resources supported by the Send transaction along with a description of the  
 114 search performed and the results returned.

115

**Table 10.X.1-2. Send Resource Descriptions**

Resource	Description
All Studies' Send Requests	Searches the entire service for Studies that match the search parameters, and sends these to the requested destination.
Study's Series' Send Requests	Searches for all Series in the specified Study that match the search parameters, and sends these to the requested destination.
Study's Instances' Send Requests	Searches for all Instances in the specified Study that match the search parameters, and sends these to the requested destination.
All Series' Send Requests	Searches the entire service for Series that match the search parameters, and sends these to the requested destination.
Study Series' Instances' Send Requests	Searches for all Instances in the specified Study and Series that match the search parameters, and sends these to the requested destination.
All Instances' Send Requests	Searches the entire service for Instances that match the search parameters, and sends these to the requested destination.

116

117 **10.X.1.2 Query Parameters**

118 The origin server shall support Query Parameters as required in Table 8.3.4-1; however, the includefield  
 119 parameter is ignored.

120 The user agent shall supply Query Parameters as required in Table 8.3.4-1; however, the includefield pa-  
 121 rameter is ignored.

122 The following sections further define details of the parameters of the search aspect of this transaction:

- 123 • Section 10.6.1.2.1 Attribute/Value Pair Requirements.

- 124 • Section 10.6.1.2.2 Search Key Types and Requirements.
- 125 • Section 10.6.1.2.3 Required Matching Attributes.
- 126 • Section 10.6.1.2.4 Optional Repository Query Attributes.

127 **10.X.1.3 Request Header Fields**

128 The origin server shall support request header fields as required in Table 10.X.1-3.

129 The user agent shall supply request header fields as required in Table 10.X.1-3.

130 Table 10.X.1-3. Request Header Fields

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

131

132 See also Section 8.4.

133 **10.X.1.4 Request Payload**

134 The request shall have no payload.

135 **10.X.2 Behavior**

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

137 The origin server shall initiate one or more sub-operations to send the matching instances to the re-  
 138 quested destination. The sub-operations are permitted to be performed using DIMSE C-STORE or DI-  
 139 COMweb Store, or even a combination of the two, at the discretion of the origin server and depending on  
 140 the capabilities of the destination.

141 Note To be able to handle the sub-operations using C-STORE operations, the origin server needs to have a  
 142 function to translate the provided endpoint to the appropriate AE Title. The definition of this function is  
 143 beyond the scope of DICOM.

144 Note It is conceivable that authentication is required to perform sub-operations. Credentials for authentication  
 145 may be coming from the origin server or the user agent, depending on the security architecture at hand.

146 **10.X.3 Response**

147 The response shall have the following syntax:

```

148 version SP status-code SP reason-phrase CRLF
149 Content-Type: media-type CRLF
150 [retry-after CRLF]
151 *(header-field CRLF)
152 CRLF
153 [payload]
```

154 **10.X.3.1 Status Codes**

155 Table 10.X.3-1 shows some common status codes corresponding to this Transaction. See also Section  
 156 8.5 for additional status codes.

157 Table 10.X.3-1. Status Code Meaning

Status	Code	Meaning
Success	200 (OK)	The origin server finished processing the send request and performing the send. It is possible some Instances matching the request were not successfully sent, e.g. due SOP

		Class incompatibility for some sub-operations. The payload describes in detail what has been achieved.
	202 (Accepted)	The origin server successfully validated the request message but has not finished processing the send request; the payload describes in detail what has been achieved to this point. The user agent is expected to follow up with a Check Send Result Transaction, described in Section 10.Y, to get updated result information.
Failure	400 (Bad Request)	The origin server cannot handle the send request because of errors in the request headers or parameters.
	409 (Conflict)	The origin server cannot handle the send request because the provided Transaction UID is already in use.
	503 (Service Unavailable)	The origin server cannot handle the send request; this may be a temporary or permanent state.

158 Note A 200 (OK) success status code should only be understood to mean that the request was successfully  
159 parsed and a Send response was returned by the origin server. The Send response may indicate that  
160 sending failed for some or even all of the matching SOP Instances.

161 **10.X.3.2 Response Header Fields**

162 The origin server shall support header fields as required in Table 10.X.3-2. All success responses shall  
163 also contain the Content Representation (see Section 8.4.2) and Payload header fields (see Section  
164 8.4.3) with appropriate values.

165 **Table 10.X.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.
Retry-After	uint	O	The number of seconds the user agent is requested to wait until starting a Check Send Result Transaction.

166 Note The Retry-After header field may be useful in conjunction with a 202 and 503 response to allow the  
167 origin server to inform the user agent about effective polling intervals.

168 **10.X.3.3 Response Payload**

169 A success response payload shall contain a Send Request Response Module. See Annex @.

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

172 **Add new section 10.Y Check Send Result Transaction, immediately after section 10.X as above**

173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205

**10.Y Check Send Result Transaction**

This Transaction uses the GET method to allow a user agent to request an origin server to provide the current result of an earlier Send Transaction, see Section 10.X.

Together with the Send Transaction, this Transaction corresponds to the DIMSE C-MOVE Operation (see PS3.4, Section C.4.2), but not supporting cancellation.

**10.Y.1 Request**

The request shall have the following syntax:

```
GET SP "/" {/resource} "/send-requests/{transactionUID} SP version CRLF
Accept: 1#media-type CRLF
*(header-field CRLF)
CRLF
```

**10.Y.1.1 Target Resource**

The Target Resource of this Transaction is a send request identified by its Transaction UID. The applicable resources are listed in Table 10.X.1-1.

**10.Y.1.2 Query Parameters**

The request has no Query Parameters.

**10.Y.1.3 Request Header Fields**

The origin server shall support Request Header Fields as required in Table 10.Y.1-1.

The user agent shall supply Request Header Fields as required in Table 10.Y.1-1.

Note The presence and values of the Check Send Result Header Fields should be the same as the Header Fields of the corresponding Send Request.

**Table 10.Y.1-1. Check Send Result 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.

**10.Y.1.4 Request Payload**

The request shall have no payload.

**10.Y.2 Behavior**

The origin server shall provide the current result status of the Send Request in an Acceptable Media Type (see section 8.7.4).

**10.Y.3 Response**

The response shall have the following syntax:

206 version SP status-code SP reason-phrase CRLF  
 207 Content-Type: media-type CRLF  
 208 [retry-after CRLF]  
 209 \*(header-field CRLF)  
 210 CRLF  
 211 [payload]

212 **10.Y.3.1 Status Codes**

213 Table 10.Y.3-1 shows some common status codes corresponding to this Transaction. See also Section  
 214 8.5 for additional status codes.

215 **Table 10.Y.3-1. Status Code Meaning**

Status	Code	Meaning
Success	200 (OK)	The origin server finished processing the send request identified by the supplied Transaction UID (see Section 10.X) and performing the send; the payload describes in detail what has been achieved.
	202 (Accepted)	The origin server has not finished processing the send request; the payload describes in detail what has been achieved at this point. The user agent is expected to follow up with a Check Send Result Transaction to get to know the entire result of the send request.
Failure	404 (Not Found)	The origin server cannot find the send request result identified by the supplied Transaction UID.
	410 (Gone)	The origin server can no longer respond to the send request result identified by the supplied Transaction UID.
	503 (Service Unavailable)	The origin server cannot handle the Check Send Result request; this may be a temporary or permanent state.

216 **Note**

- 217 1. A 200 (OK) success status code should only be understood to mean that a Check Send Result re-  
 218 sponse was returned by the origin server. The Check Send Result response may indicate that sending  
 219 failed for some or even all of the matching SOP Instances.
- 220 2. The 404 (Not Found) status code may be caused by an incorrect Transaction UID that has been sup-  
 221 plied by the user agent, or the origin server may have deleted the applicable result and not kept a rec-  
 222 ord of it.
- 223 3. The 410 (Gone) status code may be caused by the origin server having a record of the Transaction UID  
 224 but having deleted the request and/or applicable result.
- 225 4. When the 503 (Service Unavailable) status code is returned, the user agent might retry later with an-  
 226 other Check Send Result Transaction.

227 **10.Y.3.2 Response Header Fields**

228 The Response Header Fields are the same as for the Send Transaction. See Section 10.X.3.2.

229 **10.Y.3.3 Response Payload**

230 The Response Payload is the same as for the Send Transaction. See Section 10.X.3.3.

231

232 **Update first part of section 12 Non-Patient Instance Service and Resources as indicated below**

233 **12 Non-Patient Instance Service and Resources**

234 **12.1 Overview**

235 The Non-Patient Instance (NPI) Storage Service enables a user agent to retrieve, store, and search an  
 236 origin server for Instances that are not related to an individual patient. **It also enables a user agent to**  
 237 **request a server to send such Instances to another server.**

238 Note

- 239 1. Non-Patient Instances adhere to a Composite Instance IOD Information Model that does not have at its  
 240 root the Patient Information Entity representing an individual Patient.
- 241 2. "Non-patient" does not imply that there is no patient-related identifiable information in the Instances.  
 242 E.g., the Inventory IOD does include Attributes of the patient, but it does not have a Patient IE at the  
 243 root of its information model.

244 An NPI Storage Service manages a collection of resources belonging to the categories specified in Table  
 245 12.1.1-1.

246 All NPI Storage Service origin servers shall support the Retrieve Capabilities, Retrieve, and Search trans-  
 247 actions. Support for the Store transaction is optional. All NPI Storage Service user agents support one or  
 248 more of the Retrieve Capabilities, Retrieve, Store, or Search transactions.

249 **12.1.1 Resource Descriptions**

250 An NPI Service manages resources from the same NPI Category. Target URIs have the following tem-  
 251 plates:

252 /{npi-name}  
 253 /{npi-name}/{uid}  
 254 **/{npi-name}/send-requests/{transaction-uid}**

256 Where

257  
 258 npi-name = "color-palettes"  
 259 / "defined-procedure-protocols"  
 260 / "hanging-protocols"  
 261 / "implant-templates"  
 262 / "inventories"  
 263 / "inventories"  
 264 / "protocol-approvals"  
 265 uid ; is the Unique Identifier of an NPI Instance  
 266 **transaction-uid ; is the Unique Identifier of a Send Request**

267  
 268 Table 12.1.1-1 contains the templates for the NPI Resource Categories.

269 **Table 12.1.1-1. Resource Categories, URI Templates and Descriptions**

Resource Category	URI Template and Description	Corresponding IOD	Storage Class	Information Model
Color Palette	/color-palettes{/uid}	Section A.58 "Color Palette IOD" in PS3.3	Section GG "Non-Patient Object Storage Service Class" in PS3.4	Section X.1.3 "Query/Retrieve Information Model" in PS3.4

Resource Category	URI Template and Description	Corresponding IOD	Storage Class	Information Model
Defined Procedure Protocol	/defined-procedure-protocols{/uid}	Section A.82 "Procedure Protocol IODs" in PS3.3	Section GG "Non-Patient Object Storage Service Class" in PS3.4	Section HH.1.3 "Query/Retrieve Information Model" in PS3.4
Hanging Protocol	/hanging-protocols{/uid}	Section A.44 "Hanging Protocol IOD" in PS3.3	Section GG "Non-Patient Object Storage Service Class" in PS3.4	Section U.1.3 "Query/Retrieve Information Model" in PS3.4
Implant Template	/implant-templates{/uid}	Section A.61 "Generic Implant Template IOD" in PS3.3	Section GG "Non-Patient Object Storage Service Class" in PS3.4	Section BB.1.3 "Query/Retrieve Information Model" in PS3.4
Inventory	/inventories{/uid}	Section A.88 "Inventory IOD" in PS3.3	Section GG "Non-Patient Object Storage Service Class" in PS3.4	Section JJ.2 "Inventory Q/R Information Model" in PS3.4

270 The NPI SOP Classes are listed in Table GG.3-1 "Standard SOP Classes" in PS3.4.

271

272 **12.1.2 Common Query Parameters**

273 ...

274 **12.1.3 Common Media Types**

275 ...

276 **12.2 Conformance**

277 An origin server conforming to the NPI Service shall implement the Retrieve Capabilities Transaction (see  
278 Section 8.9.1).

279 The origin server shall support the transactions listed as Required in Table 12.2-1.

280 **Table 12.2-1. Required and Optional Transactions**

Transaction	Support	Section
Retrieve Capabilities	Required	Section 8.9
Retrieve	Required	Section 12.4
Store	Optional	Section 12.5
Search	Required	Section 12.6
<b><u>Send</u></b>	<b><u>Optional</u></b>	<b><u>Section 12.X</u></b>
<b><u>Check Send Result</u></b>	<b><u>Optional</u></b>	<b><u>Section 12.Y</u></b>

281 **An origin server shall claim conformance to either both or none of the Send Transaction and the Check Send**  
282 **Result Transaction.**

283 Implementations shall specify in their Conformance Statement (see PS3.2) and the Retrieve Capabilities Transaction  
284 (see Section 8.9 and Annex H):

285 • The **implementation's** role: origin server, user agent, or both.

286 • The supported resources (IODs) for each role.

287 In addition, for each supported transaction they shall specify:

288 • The supported Query Parameters, including optional Attributes, if any.

289 • The supported DICOM Media Types.

290 • The supported character sets (if other than UTF-8).

291 **12.3 Transactions Overview**

292 The NPI Service consists of the transactions listed in Table 12.3-1.

293 **Table 12.3-1. NPI Service Transactions**

Transaction	Method	Resource	Payload		Description
			Request	Success 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	/ {npi-name} / {uid}	N/A	Instance and/or Status Report	Retrieves an Instance, specified by the Target Resource in an Acceptable DICOM Media Type.
Store	POST	/ {npi-name} / {uid}	Instance(s)	Status Report	Stores one or more DICOM Instances contained in the request payload, in the location referenced by the Target URL.
Search	GET	/ {npi-name} ? {params*}	N/A	Result(s) and/or Status Report	Searches the Target Resource for Instances that match the search parameters and returns a list of matches in an Acceptable DICOM Media Type.
<b><u>Send</u></b>	<b><u>POST</u></b>	<b><u>/ {npi-name} / send-requests ? {transaction-uid}</u></b>	<b><u>N/A</u></b>	<b><u>Send Request Response Module</u></b>	<b><u>Searches the Target Resource for Instances that match the search parameters and sends or starts to send these to another server.</u></b>
<b><u>Check Send Result</u></b>	<b><u>GET</u></b>	<b><u>/ {npi-name} / send-requests ? {transaction-uid}</u></b>	<b><u>N/A</u></b>	<b><u>Send Request Response Module</u></b>	<b><u>Gets the result of a Send Transaction.</u></b>

294 The npi-name specifies the type of resource(s) contained in the payload.

295 Table 12.3-2 summarizes the Target Resources permitted for each transaction.

296

**Table 12.3-2. Resources by Transaction**

Resource	URI	Retrieve	Store	Search	<u>Send</u>	<u>Check Send Result</u>	Capabilities
NPI Service	/						X
All Instances	{/npi-name}		X	X			
Instance	{/npi-name}/{uid}	X	X				
<b><u>Send Re-quests</u></b>	<b><u>{/npi-name}/send-requests?{transaction-uid}</u></b>				<b><u>X</u></b>	<b><u>X</u></b>	

297

298

...

299 **Add new section 12.X Send Transaction, immediately before new Section 12.Y below**

300 **12.X Send Transaction**

301 This Transaction uses the POST method to request the sending of Non-Patient Instances managed by  
302 the origin server to a destination server.

303 Note The user agent could be the destination server of this Transaction.

304 Together with the Check Send Result Transaction, Section 12.Y, this Transaction corresponds to the  
305 DIMSE C-MOVE Operation (see PS3.4, Sections X.4.2, BB.4.2, HH.4.2, II.4.2, and JJ.3.2 for each kind of  
306 NPI respectively), but not supporting cancellation.

307 Due to the typically long-running nature of sending Instances to another server, the origin server may  
308 choose between two alternatives to communicate about the status and outcome of the request When the  
309 outcome of the request is immediately available, the origin returns the status and outcome in the re-  
310 sponse to the request. When the outcome of the request is not available immediately, the origin server  
311 indicates that to the client in its response, and the client uses the Check Send Result Transaction. This  
312 pattern of Transactions is similar to what has been illustrated in Figure 13.1-1.

313 The Store sub-operations that are necessary to fulfil the send request may be executed using DIMSE or  
314 DICOMweb, or even a combination of the two. The term sub-operations will henceforth be used for both  
315 DIMSE’s sub-operations and DICOMweb’s sub-transactions.

316 **12.X.1 Request**

317 The request shall have the following syntax:

318 POST SP /{npi-name}/send-requests?{transactionUID}?" destination {&search\*} SP version CRLF  
319 Accept: 1#media-type  
320 \*(header-field CRLF)  
321 CRLF

322 where

323 destination = "destination" "=" {URL}

324 and where {URL} is a percent-encoded absolute URL of the destination endpoint; see [RFC 3986].

325 **12.X.1.1 Target Resources**

326 The Target URI shall reference one of the resources shown in Table 12.X.1-1.

327 An origin server shall specify all supported npi-names in its Conformance Statement and in its response  
328 to the Retrieve Capabilities Transaction.

329 **Table 12.X.1-1. Send Transaction Resources**

Resource	URI Template	Description
All Send Requests	/{npi-name}/send-requests/{transactionUID}	Searches a collection of NPI Instances that match the search parameters and sends these to the requested destination.

330

331 **12.X.1.2 Query Parameters**

332 The user agent shall supply, and the origin server shall support, the Common Query Parameters in Sec-  
333 tion 12.1.2.

334 The origin server shall support Query Parameters as required in Table 8.3.4-1; however, the includefield  
335 parameter is ignored.

336 The user agent shall supply in the request Query Parameters as required in Table 8.3.4-1; however, the  
337 includefield parameter is ignored.

338 For each Resource Category the origin server supports, it shall support the behaviors and matching key  
339 Attributes specified in the corresponding sections in Table 12.X.1-2.

340 **Table 12.X.1-2. NPI Resource Send Attributes**

Resource Category	Behaviors and Matching Key Attributes
Color Palette	Section X.6.1.2 "Color Palette Attributes" in PS3.4.
Defined Procedure Protocol	Section HH.6.1.2 "Defined Procedure Protocol Attributes" in PS3.4.
Hanging Protocol	Section U.6.1.2 "Hanging Protocol Attributes" in PS3.4.
Implant Template	Section BB.6.1.2 "Implant Template Attributes" in PS3.4.
Inventory	Section JJ.2.2 "Inventory Q/R Information Model Attributes" in PS3.4.
Protocol Approval	Section II.6.1.2 "Protocol Approval Attributes" in PS3.4.

341

342 **12.X.1.3 Request Header Fields**

343 The origin server shall support request header fields as required in Table 12.X.1-3.

344 The user agent shall supply request header fields as required in Table 12.X.1-3.

345 **Table 12.X.1-3. Request Header Fields**

Name	Values	Usage		Description
		User Agent	Origin Server	
Accept	1#-dicom-media-type	M	M	The Acceptable Media Types of the response payload

346 See also Section 8.4.

347 **12.X.1.4 Request Payload**

348 The request has no payload.

349 **12.X.2 Behavior**

350 The origin server shall perform the search indicated by the request, using the matching behavior specified  
351 in Section 8.3.4.1.1 and in the corresponding sections in Table 8.3.4-1.

352 The rules for search results are specified in Section 8.3.4.

353 The origin server shall initiate one or more sub-operations to send the matching instances to the re-  
354 quested destination. The sub-operations are permitted to be performed using DIMSE C-STORE or DI-  
355 COMweb Store, or even a combination of the two, at the discretion of the origin server and depending on  
356 the capabilities of the destination.

357 Note To be able to handle the sub-operations using C-STORE operations, the origin server needs to have a  
358 function to translate the provided endpoint to the appropriate AE Title. The definition of this function is  
359 beyond the scope of DICOM.

360 Note It is conceivable that authentication is required to perform sub-operations. Credentials for authentication  
361 may be coming from the origin server or the user agent, depending on the security architecture at hand.

362 **12.X.3 Response**

363 The response shall have the following syntax:

```
364 version SP status-code SP reason-phrase CRLF
365 Content-Type: media-type CRLF
366 [retry-after CRLF]
367 *(header-field CRLF)
368 CRLF
369 [payload]
```

370 **12.X.3.1 Status Codes**

371 Table 12.X.3-1 shows some common status codes corresponding to this Transaction. See also Section  
372 8.5 for additional status codes.

373 **Table 12.X.3-1. Status Code Meaning**

Status	Code	Meaning
Success	200 (OK)	The origin server finished processing the send request and performing the send. It is possible some Instances matching the request were not successfully sent, e.g. due SOP Class incompatibility for some sub-operations. The payload describes in detail what has been achieved.
	202 (Accepted)	The origin server successfully validated the request message but has not finished processing the send request; the payload describes in detail what has been achieved to this point. The user agent is expected to follow up with a Check Send Result Transaction, described in Section 12.Y, to get updated result information.
Failure	400 (Bad Request)	The origin server cannot handle the send request because of errors in the request headers or parameters.
	409 (Conflict)	The origin server cannot handle the send request because the provided Transaction UID is already in use.
	503 (Service Unavailable)	The origin server cannot handle the send request; this may be a temporary or permanent state.

374 Note A 200 (OK) success status code should only be understood to mean that the request was successfully  
375 parsed and a Send response was returned by the origin server. The Send response may indicate that  
376 sending failed for some or even all of the matching SOP Instances.

377 **12.X.3.2 Response Header Fields**

378 The origin server shall support header fields as required in Table 12.X.3-2. All success responses shall  
379 also contain the Content Representation (see Section 8.4.2) and Payload header fields (see Section  
380 8.4.3) with appropriate values.

381 **Table 12.X.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.

Retry-After	uint	0	The number of seconds the user agent is requested to wait until starting a Check Send Result Transaction.
-------------	------	---	---

382 Note The Retry-After header field may be useful in conjunction with a 202 and 503 response to allow the  
383 origin server to inform the user agent about effective polling intervals.

384 **12.X.3.3 Response Payload**

385 A success response shall contain a Send Request Response Module. See Annex @.

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

388 **Add new section 12.Y Check Send Result Transaction, immediately after section 12.X above**

389 **12.Y Check Send Result Transaction**

390 This Transaction uses the GET method to allow a user agent to request an origin server to provide the  
391 current result of an earlier Send Transaction, see Section 12.X.

392 Together with that Send Transaction, this Transaction corresponds to the DIMSE C-MOVE Operation  
393 (see PS3.4, Sections X.4.2, BB.4.2, HH.4.2, II.4.2, and JJ.3.2 for each kind of NPI respectively) but not  
394 supporting cancellation.

395 **12.Y.1 Request**

396 The request shall have the following syntax:

397 GET SP /{npi-name}/send-requests/{transactionUID} SP version CRLF  
398 Accept: 1#media-type CRLF  
399 \*(header-field CRLF)  
400 CRLF

401 **12.Y.1.1 Target Resource**

402 The Target Resource of this Transaction is a send request identified by its Transaction UID. The applica-  
403 ble resources are listed in Table 12.X.1-1.

404 **12.Y.1.2 Query Parameters**

405 The request has no Query Parameters.

406 **12.Y.1.3 Request Header Fields**

407 The origin server shall support Request Header Fields as required in Table 12.Y.1-1.

408 The user agent shall supply Request Header Fields as required in Table 12.Y.1-1.

409 Note The presence and values of the Check Send Result Header Fields should be the same as those of the  
410 Header Fields of the corresponding Send Request.

411

412 **Table 12.Y.1-1. Check Send Result Header Fields**

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

413

414 See also Section 8.4.

415 **12.Y.1.4 Request Payload**

416 The request shall have no payload.

417 **12.Y.2 Behavior**

418 The origin server shall provide the current result status of the Send Request in an Acceptable Media Type  
419 (see section 8.7.4).

420 **12.Y.3 Response**

421 The response shall have the following syntax:

422 version SP status-code SP reason-phrase CRLF

423 Content-Type: media-type CRLF  
 424 [retry-after CRLF]  
 425 \*(header-field CRLF)  
 426 CRLF  
 427 [payload]

428 **12.Y.3.1 Status Codes**

429 Table 12.Y.3-1 shows some common status codes corresponding to this Transaction. See also Section  
 430 8.5 for additional status codes.

431 **Table 12.Y.3-1. Status Code Meaning**

Status	Code	Meaning
Success	200 (OK)	The origin server finished processing the send request identified by the supplied Transaction UID (see Section 12.X); the payload describes in detail what has been achieved.
	202 (Accepted)	The origin server has not finished processing the send request; the payload describes in detail what has been achieved until now. The user agent is expected to follow-up with a Check Send Result Transaction to get to know the entire result of the send request.
Failure	404 (Not Found)	The origin server cannot find the send request result identified by the supplied Transaction UID.
	410 (Gone)	The origin server can no longer provide the send request result identified by the supplied Transaction UID.
	503 (Service Unavailable)	The origin server cannot handle the Check Send Result request; this may be a temporary or permanent state.

432 Note

- 433 1. A 200 (OK) success status code should only be understood to mean that a Check Send Result re-  
 434 sponse was returned by the origin server. The Check Send Result response may indicate that sending  
 435 failed for some or even all of the matching SOP Instances.
- 436 2. The 404 (Not Found) status code may be caused by an incorrect Transaction UID that has been sup-  
 437 plied by the user agent, or the origin server may have deleted the applicable result and not kept a rec-  
 438 ord of it.
- 439 3. The 410 (Gone) status code may be caused by the origin server having a record of the Transaction UID  
 440 but having deleted the request and/or applicable result.
- 441 4. When the 503 (Service Unavailable) status code is returned, the user agent might retry later with an-  
 442 other Check Send Result Transaction.

443 **12.Y.3.2 Response Header Fields**

444 The Response Header Fields are the same as for the Send Transaction. See Section 12.X.3.2.

445 **12.Y.3.3 Response Payload**

446 The Response Payload is the same as for the Send Transaction. See Section 12.X.3.3.

447 **Update Section 4 Symbols and Abbreviated Terms**

448 **4 Symbols and Abbreviated Terms**

449 ...

450 RESTful A service implemented using the REST architecture.

451 **SEND-RS Server-Enabled Network Delivery by RESTful Services**

452 SOP Service Object Pair

453 ...

454

455 **Update Section B Examples: add new examples for the respective Send Transactions**

456 **B Examples (Informative)**

457 ...

458 **B.X1 Request to Send all Studies of a Patient to Another Server**

459 This example shows the flow of messages between the user agent and the origin server for the scenario  
 460 in which 1) the user agent requests all studies of a certain patient be sent to another server, 2) the origin  
 461 server tells the user agent to check for the result of this request later, returning the current status of the  
 462 request, 3) the user agent checks for the result, and 4) the result provided by the origin server shows that  
 463 it has sent all studies without any exceptions.

464 The patient in this example has three studies, an X-ray study with two Instances in one Series, an en-  
 465 hanced CT study with two series (each with one Instance), and an ultrasound study with one series with  
 466 one enhanced multi-frame Instance. The origin server in this example typically sends a study using study  
 467 level STOW requests, resulting in three requests: one per study. This scenario does, however, result in  
 468 five sub-operations – being the number of Instances.

469 Step 1: the user agent sends a POST request with Transaction UID 1.1.99999.20250901 for patientID  
 470 11235813, the images to be sent to the host 'another.server' whose endpoint is https://an-  
 471 other.server/stow:

```
472
473 POST /radiology/studies/send-requests/1.1.99999.20250901?destination=https%3A%2F%2Fanother.server/stow&Patien-
474 tID=11235813 HTTP/1.1
475 Host: www.hospital-stmarco
476 Accept: application/dicom+json
477
```

478 Step 2: the origin server returns its response to the request, including a Send Request Response Module.  
 479 In this scenario, there is no immediate result (return code 202 Accepted), and the server also notifies the  
 480 user agent that it ought to wait at least 300 seconds before making a follow-up request for the result. In  
 481 case where there is an immediate result, the response would be as shown below in step 4, effectively  
 482 skipping steps 2 and 3.

```
483 HTTP/1.1 202 Accepted
484 Content-Length: 247
485 Content-Type: application/dicom+json; charset=utf-8
486 Retry-After: 300
487
488 [ { "00000900": { "vr": "US", "Value": [ 65280 ] }
489   , "00001020": { "vr": "US", "Value": [ 5 ] }
490   , "00001021": { "vr": "US", "Value": [ 0 ] }
491   , "00001022": { "vr": "US", "Value": [ 0 ] }
492   , "00001023": { "vr": "US", "Value": [ 0 ] }
493 } ]
494
```

- 495 According to Table @.1-1, which defines the Send Request Response Module, this shows that:
- 496 • the request is pending – value 65280 for tag (0000,0900) is value 0xFF00 for the Status, which is, ac-  
 497 cording to PS3.4, Table 4.2-1, the Pending Service Status;
  - 498 • the Number of Remaining Sub-operations, tag (0000,1020), is 5;
  - 499 • the Number of Completed Sub-operations, tag (0000,1021), is 0;
  - 500 • the Number of Failed Sub-operations, tag (0000,1022), is 0;
  - 501 • the Number of Warning Sub-operations, tag (0000,1023), is 0.

503  
504 Step 3: after waiting 300 seconds, the suggested period of time to retry, the user agent GETs the status  
505 of the request using the same Transaction UID as the original request:

```
506 GET /radiology/studies/send-requests/1.1.99999.20250901 HTTP/1.1  
507 Host: www.hospital-stmarco  
508 Accept: application/dicom+json
```

510 Step 4: the origin server returns the response of the Check Send Result request, which contains the re-  
511 sult of the original Send Request. Note that in case the server initially responds to the POST request of  
512 step 1 with the HTTP response status code 200 (the synchronous case) the same result would be re-  
513 turned:  
514

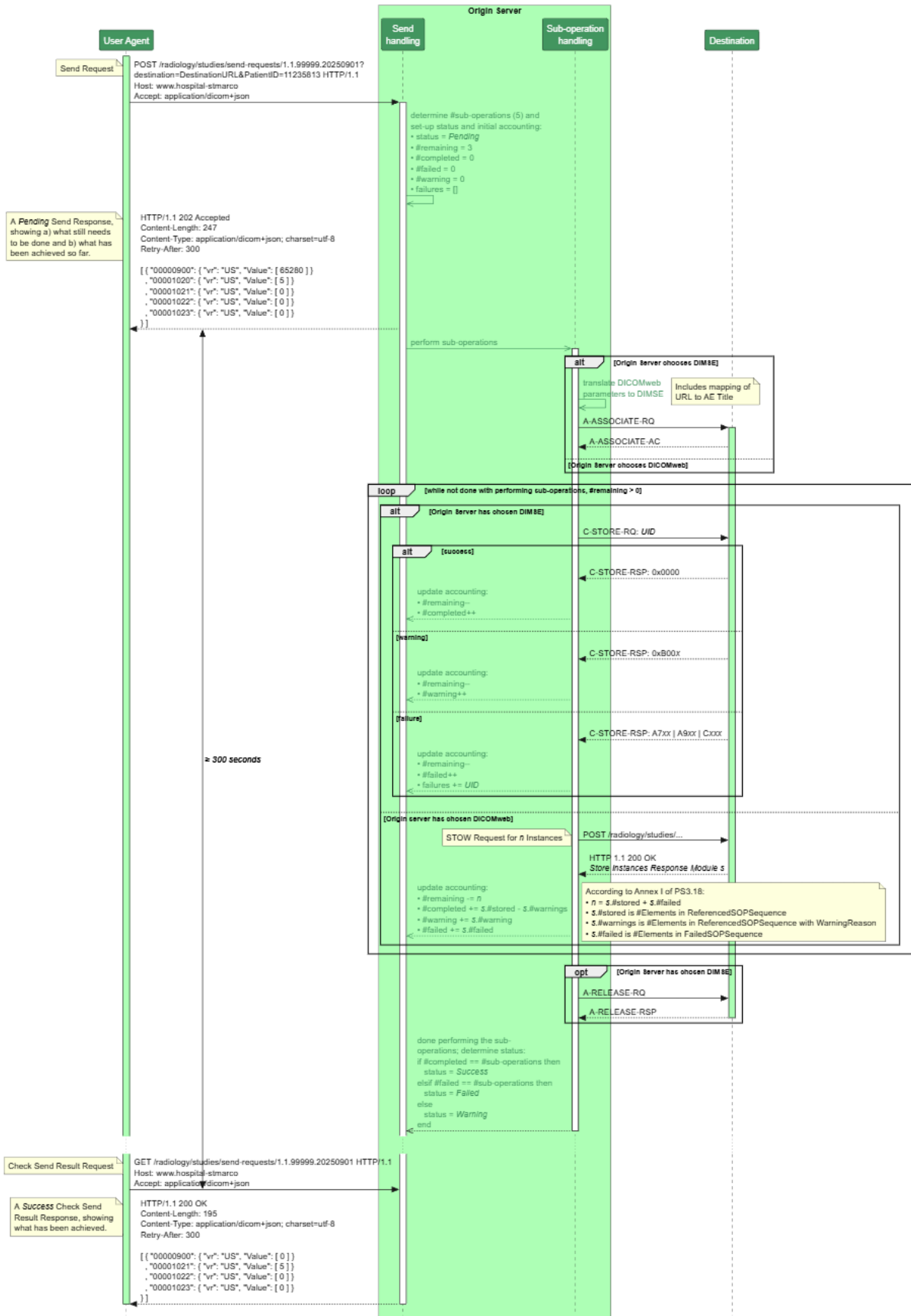
```
515 HTTP/1.1 200 OK  
516 Content-Length: 195  
517 Content-Type: application/dicom+json; charset=utf-8  
518 Retry-After: 300
```

```
519  
520  
521 [ { "00000900": { "vr": "US", "Value": [ 0 ] }  
522   , "00001021": { "vr": "US", "Value": [ 5 ] }  
523   , "00001022": { "vr": "US", "Value": [ 0 ] }  
524   , "00001023": { "vr": "US", "Value": [ 0 ] }  
525 } ]
```

526  
527 This response shows – see for details Table @.1-1 or earlier in this scenario – that the send request has  
528 been handled successfully, and that it resulted in 5 completed sub-operations, with no warnings and fail-  
529 ures. The three studies were sent, although that number is not reflected in the response.

## 530 **B.X2 Flow: Handling of a Send Request**

531  
532 The flow in Figure B.X2-1 shows how a Send Request can be handled by an origin server. In this case,  
533 the example of Annex B.X1 has been used; however, here the internals of the Origin Server are elabo-  
534 rated. It is also shown that the origin server may choose between DIMSE and DICOMweb for its sub-op-  
535 erations and it may choose to respond with one or more results. Note that multiple responses are recom-  
536 mended for long-running Transactions.  
537



539

## Figure B.X2-1 Handling a Send Request

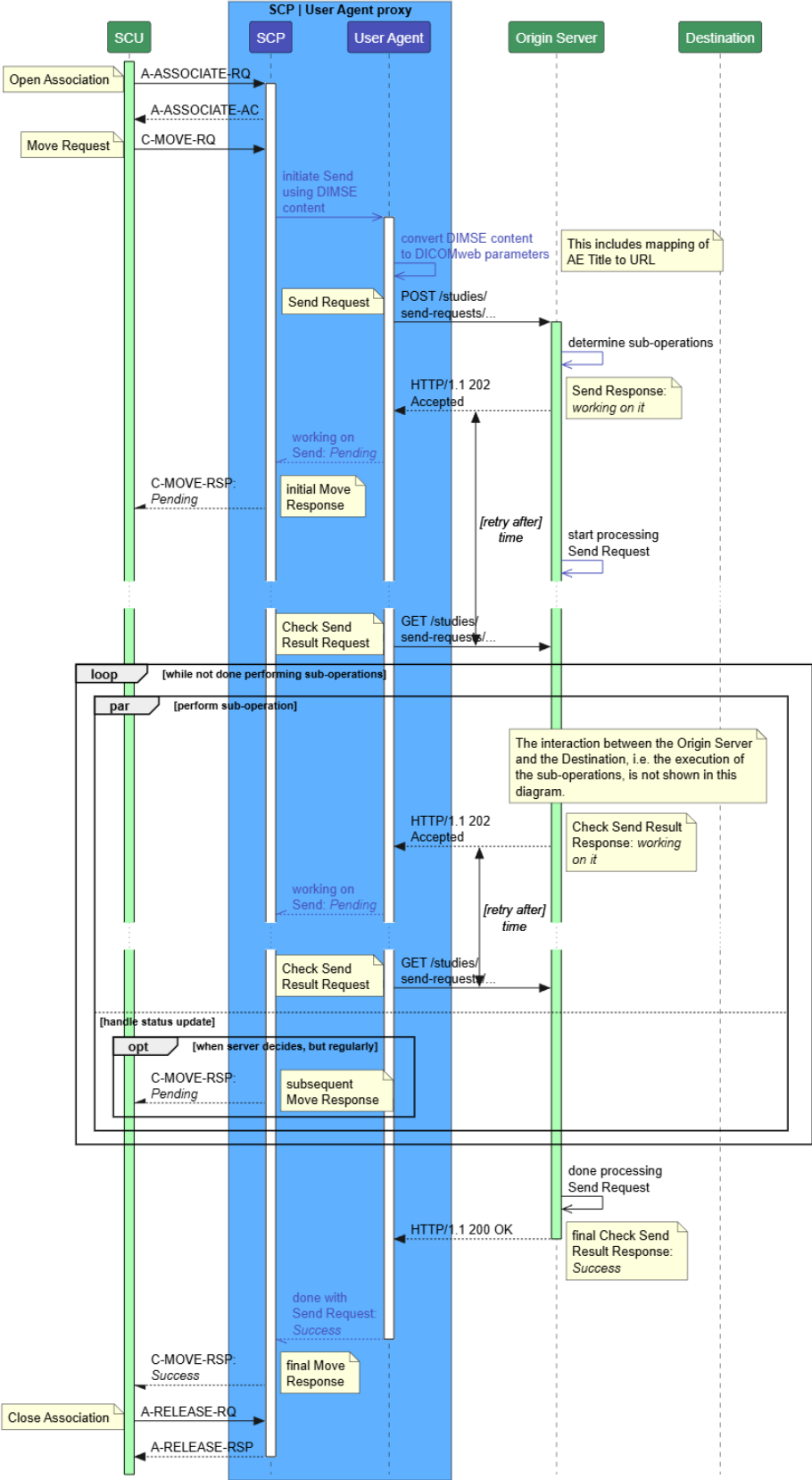
540

### **B.X3 Bi-Directional Proxy for Send Transaction**

541 The DICOMweb Send Transaction may be deployed in a hybrid environment, i.e., an environment in  
542 which both DICOMweb and DIMSE are used. In such a hybrid environment, a proxy can broker Transac-  
543 tions from one service to the other, allowing a DICOMweb Send origin server or a DIMSE C-MOVE SCP  
544 to support a mixed set of DICOMweb user agents and DIMSE SCUs.

545 DICOM does not mandate implementation of proxies; however, since they would be very useful in a hy-  
546 brid environment, the examples in this section show how this could be done. It is the implementer's re-  
547 sponsibility to match the possibly asynchronous DIMSE behavior with the polling DICOMweb behavior, for  
548 example management of Transaction UIDs.

549 Figure B.X3-1 shows how a proxy could facilitate a C-MOVE request from a DIMSE SCU to a DICOMweb  
550 origin server; the diagram does not show the sub-operations, as the emphasis is on proxying, not on the  
551 entire server behavior.

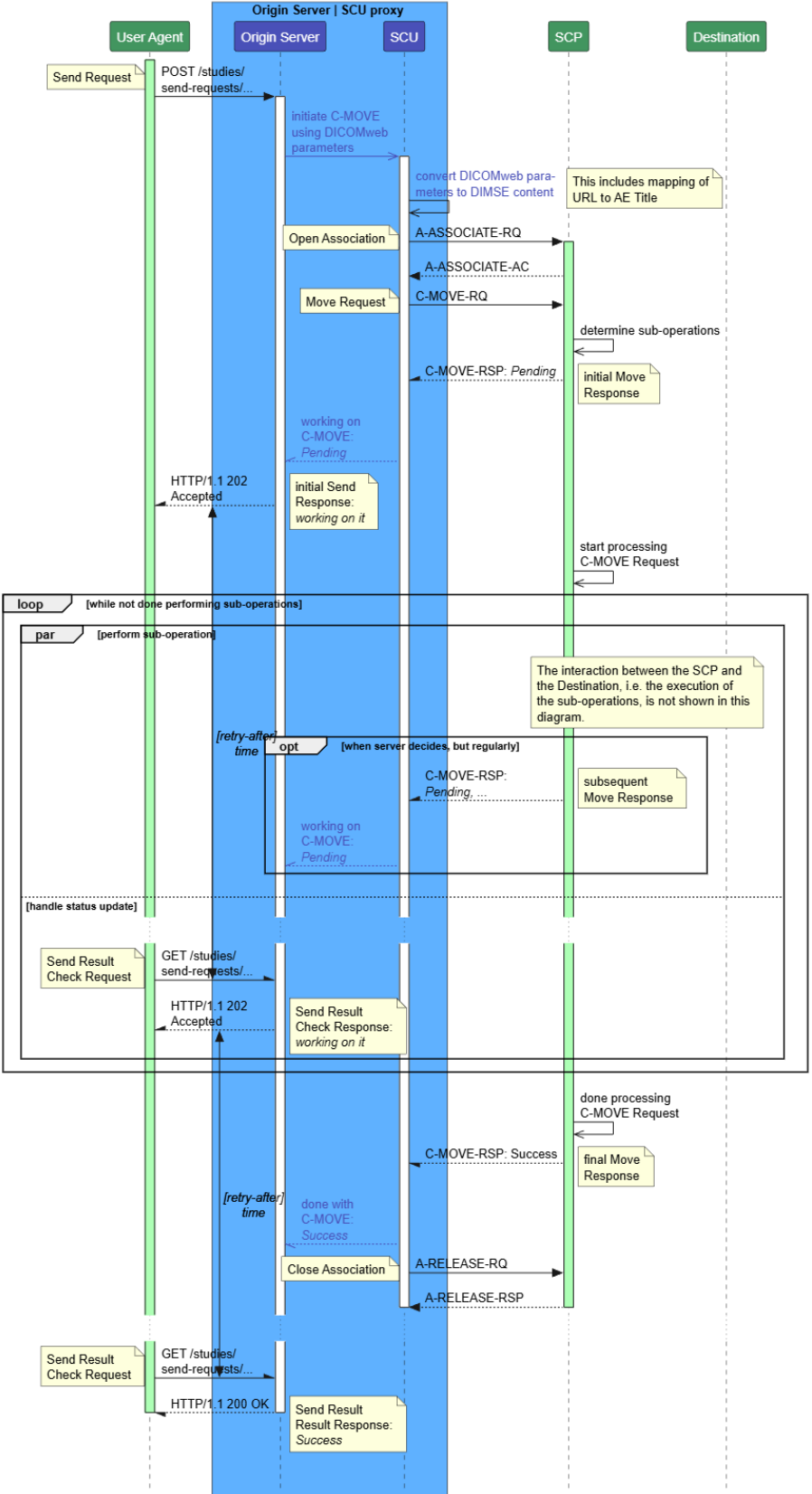


552

553

Figure B.X3-1. Send DIMSE Proxy for a DICOMweb Origin Server

554 Figure B.X3-2 shows how a proxy could facilitate a Send Request from a DICOMweb user agent to a  
555 DIMSE SCP; the diagram does not show the Sub-transactions, as the emphasis is on proxying, not on  
556 the entire server behavior. When proxying in this direction, the proxy will receive information from the  
557 SCP that it is not able to dispatch immediately to the user agent.



558

559

Figure B.X3-2. Send DICOMweb Proxy for a DIMSE SCP

560  
561

**Update Table H-1 Resources and Methods: add new resources and methods for Send Transactions**

562

## H Capabilities Description

Service	Resource	Transactions	Reference
Studies (see Section 10.1.1)			
	studies	Search for Studies Store Instances <b><u>Send Studies</u></b> <b><u>Check Send Result for Studies</u></b>	Section 10.6 Section 10.5 <b><u>Section 10.X</u></b> <b><u>Section 10.Y</u></b>
	{StudyInstance}	Retrieve Study Store Study Instances <b><u>Send Study</u></b> <b><u>Check Send Result for a Study</u></b>	Section 10.4 Section 10.5 <b><u>Section 10.X</u></b> <b><u>Section 10.Y</u></b>
	metadata	Retrieve Study Metadata	Section 10.4
	rendered	Retrieve Rendered Study	Section 10.4
	renderedmpr	Retrieve Rendered MPR Volume Study	Section 10.4
	rendered3d	Retrieve Rendered 3D Volume Study	Section 10.4
	thumbnail	Retrieve Study Thumbnail	Section 10.4
	bulkdata	Retrieve Study Bulkdata	Section 10.4
	pixeldata	Retrieve Study Pixel Data	Section 10.4
	series	Search for Study Series <b><u>Send Series</u></b> <b><u>Check Send Result for Series</u></b>	Section 10.6 <b><u>Section 10.X</u></b> <b><u>Section 10.Y</u></b>
	{SeriesInstance}	Retrieve Series <b><u>Send Series</u></b> <b><u>Check Send Result for a Series</u></b>	Section 10.4 <b><u>Section 10.X</u></b> <b><u>Section 10.Y</u></b>
	metadata	Retrieve Series Metadata	Section 10.4
	rendered	Retrieve Rendered Series	Section 10.4

Service	Resource	Transactions	Reference
	renderedmpr	Retrieve Rendered MPR Volume Series	Section 10.4
	rendered3d	Retrieve Rendered 3D Volume Series	Section 10.4
	thumbnail	Retrieve Series Thumbnail	Section 10.4
	bulkdata	Retrieve Series Bulkdata	Section 10.4
	pixeldata	Retrieve Series Pixel Data	Section 10.4
	instances	Search for Study Series Instances <b><u>Send Instances</u></b> <b><u>Check Send Result for Instances</u></b>	Section 10.4 <b><u>Section 10.X</u></b> <b><u>Section 10.Y</u></b>
	{SOPInstance}	Retrieve Instance <b><u>Send Instance</u></b> <b><u>Check Send Result for an Instance</u></b>	Section 10.4 <b><u>Section 10.X</u></b> <b><u>Section 10.Y</u></b>
	metadata	Retrieve Instance Metadata	Section 10.4
	rendered	Retrieve Rendered Instance	Section 10.4
	renderedmpr	Retrieve Rendered MPR Volume Instance	Section 10.4
	rendered3d	Retrieve Rendered 3D Volume Instance	Section 10.4
	thumbnail	Retrieve Instance Thumbnail	Section 10.4
	bulkdata	Retrieve Instance Bulkdata	Section 10.4
	pixeldata	Retrieve Instance Pixel Data	Section 10.4
	frames	N/A	N/A
	{framelist}	Retrieve Frames	Section 10.4
	rendered	Retrieve Rendered Frames	Section 10.4
	renderedmpr	Retrieve Rendered MPR Volume Frames	Section 10.4
	rendered3d	Retrieve Rendered 3D Volume Frames	Section 10.4
	thumbnail	Retrieve Frame Thumbnail	Section 10.4
	pixeldata	Retrieve Frame Pixel Data	Section 10.4
	instances	Search for Study Instances	Section 10.6
	series	Search for Series	Section 10.6

Service	Resource	Transactions	Reference
	{SeriesInstance}	N/A	N/A
	{instances}	Search for Instances	Section 10.6
	instances	Search for Instances	Section 10.6
	{BulkDataReference}	Retrieve Bulkdata	Section 10.4
Worklist (see Section 11.1.1)			
	workitems	Search for Workitem Create Workitem	Section 11.9 Section 11.4
	{Workitem}	Retrieve Workitem Update Workitem	Section 11.4 Section 11.6
	state	Change Workitem State	Section 11.7
	cancelrequest	Request Workitem Cancellation	Section 11.8
	subscribers	N/A	N/A
	{AETitle}	Subscribe Unsubscribe	Section 11.10 Section 11.11
	1.2.840.10008.5.1.4.34.5	N/A	N/A
	subscribers	N/A	N/A
	{AETitle}	Subscribe Unsubscribe	Section 11.10 Section 11.11
	suspend	Unsubscribe	Section 11.11
	1.2.840.10008.5.1.4.34.5.1	N/A	N/A
	subscribers	N/A	N/A
	{AETitle}	Subscribe Unsubscribe	Section 11.10 Section 11.11
	suspend	Suspend Worklist Subscription	Section 11.11
Non-Patient Instance (see Section 12.1.1)			
	color-palettes	N/A	N/A
	{uid}	Retrieve	Section 12.4

Service	Resource	Transactions	Reference
		Store Search <u>Send</u> <u>Check Send Result</u>	Section 12.5 Section 12.6 <u>Section 12.X</u> <u>Section 12.Y</u>
	defined-procedure-protocols	N/A	N/A
	{uid}	Retrieve Store Search <u>Send</u> <u>Check Send Result</u>	Section 12.4 Section 12.5 Section 12.6 <u>Section 12.X</u> <u>Section 12.Y</u>
	hanging-protocols	N/A	N/A
	{uid}	Retrieve Store Search <u>Send</u> <u>Check Send Result</u>	Section 12.4 Section 12.5 Section 12.6 <u>Section 12.X</u> <u>Section 12.Y</u>
	implant-templates	N/A	N/A
	{uid}	Retrieve Store Search <u>Send</u> <u>Check Send Result</u>	Section 12.4 Section 12.5 Section 12.6 <u>Section 12.X</u> <u>Section 12.Y</u>
	inventories	N/A	N/A
	{uid}	Retrieve Store Search <u>Send</u> <u>Check Send Result</u>	Section 12.4 Section 12.5 Section 12.6 <u>Section 12.X</u> <u>Section 12.Y</u>
	protocol-approvals	N/A	N/A

Service	Resource	Transactions	Reference
		Retrieve Store Search <u>Send</u> <u>Check Send Result</u>	Section 12.4 Section 12.5 Section 12.6 <u>Section 12.X</u> <u>Section 12.Y</u>
Storage Commitment Requests (see Section 13.1.1)			
	commitment-requests	Request	Section 13.4
		Result Check	Section 13.5
...			

563

564

565 **Add new Annex @ – Send Request Response Module**

566 **Annex @ Send Request Response Module**

567 **@.1 Response Message Body**

568 Table @.1-1 specifies the Attributes for of the Send Request Response Module; this is similar to what has been de-  
569 scribed for DIMSE's C-MOVE in PS3.4, Section C.4.2.1.

570 **Table @.1-1. Send Request Response Module Attributes**

Attribute Name	Tag	Type	Attribute Description
Status	(0000,0900)	1	The status of the associated sub-operations performed to this point. The status value shall be populated as described in PS3.4, Section C.4.2.3.1, using values as defined in PS3.4, Section C.4.2.1.5.  Note: DICOMweb Send does not support Cancel.
Number of Remaining Sub-operations	(0000,1020)	1C	The number of store sub-operations associated with this Send Request that are remaining. Required when the Status (0000,0900) is Pending.
Number of Completed Sub-operations	(0000,1021)	1	The number of store sub-operations associated with this Send Request that yielded Success.
Number of Failed Sub-operations	(0000,1022)	1	The number of store sub-operations associated with this Send Request that yielded Failure.
Number of Warning Sub-operations	(0000,1023)	1	The number of store sub-operations associated with this Send Request that yielded Warning.
Failed SOP Instance UID List	(0008,0058)	1C	The list of UIDs of SOP Instances for which the associated sub-operation yielded Failure. Required when Status (0000,0900) is not Pending and Number of Failed Sub-operations (0000,1022) is not zero.

571 Note Group 0000 are used in DIMSE commands and not permitted in DICOM Instances. The above  
572 Tags are used in this Send request response to match the semantics of DIMSE's C-MOVE re-  
573 sponse message.

574  
575

## Changes to NEMA Standards Publications PS 3.4

576  
577  
578

**Adapt Section C.4.2 such that DIMSE's C-MOVE operation can be used in combination with STOW-RS and SEND-RS. The entire Section is given here for reference, while only the Alternative Sub-operation Mechanism Option and Proxy Option have been added for the SCP**

### 579 C.4.2 C-MOVE Operation

580 SCUs of some SOP Classes of the Query/Retrieve Service Class may generate retrievals using the C-  
581 MOVE operation as described in PS3.7. The C-MOVE operation allows an application entity to instruct  
582 another application entity to transfer stored SOP Instances to another application entity using the C-  
583 STORE operation. Support for the C-MOVE service shall be agreed upon at Association establishment  
584 time by both the SCU and SCP of the C-MOVE in order for a C-MOVE operation to occur over the Associ-  
585 ation. The C-STORE sub-operations shall always be accomplished over an Association different from the  
586 Association that accomplishes the C-MOVE operation. Hence, the SCP of the Query/Retrieve Service  
587 Class serves as the SCU of the Storage Service Class.

588 Note

589 The application entity that receives the stored SOP Instances may or may not be the originator of the C-MOVE  
590 operation.

591 A C-MOVE request may be performed to any level of the Query/Retrieve Information Model. However, the  
592 transfer of stored SOP Instances may not be performed at this level. The level at which the transfer is per-  
593 formed depends upon the SOP Class (see Section C.6).

#### 594 C.4.2.1 C-MOVE Service Parameters

##### 595 C.4.2.1.1 SOP Class UID

596 The SOP Class UID identifies the Query/Retrieve Information Model against which the C-MOVE is to be performed.  
597 Support for the SOP Class UID is implied by the Abstract Syntax UID of the Presentation Context used by this C-MOVE  
598 operation.

##### 599 C.4.2.1.2 Priority

600 The Priority Attribute defines the requested priority of the C-MOVE operation and corresponding C-STORE sub-oper-  
601 ations with respect to other DIMSE operations being performed by the same SCP.

602 Processing of priority requests is not required of SCPs. Whether or not an SCP supports priority processing, and the  
603 meaning of the different priority levels shall be stated in the Conformance Statement of the SCP. The same priority  
604 shall be used for all C-STORE sub-operations.

##### 605 C.4.2.1.3 Move Destination

606 Move Destination specifies the Application Entity Title of the receiver of the C-STORE sub-operations.

##### 607 C.4.2.1.4 Identifier

608 The C-MOVE request shall contain an Identifier. The C-MOVE response shall conditionally contain an Identifier as  
609 required in Section C.4.2.1.4.2.

610 Note

611 The Identifier is specified as U in the definition of the C-MOVE primitive in PS3.7 but is specialized for use  
612 with this service.

##### 613 C.4.2.1.4.1 Request Identifier Structure

614 An Identifier in a C-MOVE request shall contain:

- 615 • Query/Retrieve Level (0008,0052), which defines the level of the retrieval
- 616 • Unique Key Attributes, which may include Patient ID (0010,0020), Study Instance UIDs (0020,000D), Series Instance  
617 UIDs (0020,000E), and the SOP Instance UIDs (0008,0018)

618 • Conditionally, the Attribute Query/Retrieve View (0008,0053). This Attribute may be included if Enhanced Multi-  
619 Frame Image Conversion has been accepted during Association Extended Negotiation. It shall not be included oth-  
620 erwise.

621 Specific Character Set (0008,0005) shall be present if Patient ID (0010,0020) is using a character set other than the  
622 Default Character Repertoire.

623 The Unique Keys at each level of the hierarchy and the values allowable for the level of the retrieval shall be defined  
624 in the SOP Class definition for the Query/Retrieve Information Model.

625 Note

626 1. In the non-Relational behavior, more than one entity may be retrieved if the Query/Retrieve Level is IM-  
627 AGE, SERIES or STUDY, using List of UID matching, but only Single Value Matching value may be  
628 specified for Patient ID (0010,0020).

629 2. The issuer of the Patient ID (0010,0020) is implicit; there is no provision to send the Issuer of Patient ID  
630 (0010,0021). When there is a possibility of ambiguity of the Patient ID (0010,0020) value, a STUDY level  
631 retrieval should be used instead of a PATIENT level retrieval.

632 **C.4.2.1.4.2 Response Identifier Structure**

633 The Failed SOP Instance UID List (0008,0058) specifies a list of UIDs of the C-STORE sub-operation SOP Instances  
634 for which this C-MOVE operation has failed. An Identifier in a C-MOVE response shall conditionally contain the Failed  
635 SOP Instance UID List (0008,0058) based on the C-MOVE response status value. If no C-STORE sub-operation failed,  
636 Failed SOP Instance UID List (0008,0058) is absent and therefore no Data Set shall be sent in the C-MOVE response.

637 Specific Character Set (0008,0005) shall not be present.

638 The Identifier in a C-MOVE response with a status of:

639 • Cancel, Failure, or Warning shall contain the Failed SOP Instance UID List Attribute

640 • Pending shall not contain the Failed SOP Instance UID List Attribute (no Data Set)

641 **C.4.2.1.5 Status**

642 Table C.4-2 defines the specific Status Code values that might be returned in a C-MOVE response. General Status  
643 Code values and fields related to Status Code values are defined for C-MOVE DIMSE Service in PS3.7.

644 **Table C.4-2. C-MOVE Response Status Values**

645

Service Status	Further Meaning	Status Codes	Related Fields
Failure	Refused: Out of resources - Unable to calculate number of matches	A701	(0000,0902)
	Refused: Out of resources - Unable to perform sub-operations	A702	(0000,1021) (0000,1022) (0000,1023)
	Refused: Move Destination unknown	A801	(0000,0902)
	Error: Data Set does not match SOP Class	A900	(0000,0901) (0000,0902)
	Failed: Unable to process	Cxxx	(0000,0901) (0000,0902)
Cancel	Sub-operations terminated due to Cancel Indication	FE00	(0000,1020)

Service Status	Further Meaning	Status Codes	Related Fields
			(0000,1021) (0000,1022) (0000,1023)
Warning	Sub-operations Complete - One or more Failures	B000	(0000,1021) (0000,1022) (0000,1023)
Success	Sub-operations Complete - No Failures	0000	(0000,1021) (0000,1022) (0000,1023)
Pending	Sub-operations are continuing	FF00	(0000,1020) (0000,1021) (0000,1022) (0000,1023)

646 Some Failure Status Codes are implementation specific.

647 An SCP implementation shall assign specific Failure Status Codes by replacing each 'x' symbol with a hexadecimal  
648 digit in the range from 0 to F. An SCP implementation wishing to differentiate between causes of "Failed: Unable to  
649 process" Failure Meaning shall assign those causes specific Status Code Values within valid range specified in Ta-  
650 ble C.4-2.

651 An SCU implementation shall recognize any Failure Status Code within the value range specified in Table C.4-2 as an  
652 indicator of the Failure Meaning stated in the table. There is no requirement for an SCU implementation to differentiate  
653 between specific Status Codes within the valid range.

654 **C.4.2.1.6 Number of Remaining Sub-Operations**

655 Inclusion of the Number of Remaining Sub-operations is conditional based upon the status in the C-MOVE response.  
656 The Number of Remaining Sub-operations specifies the number of Remaining C-STORE sub-operations necessary to  
657 complete the C-MOVE operation.

658 A C-MOVE response with a status of:

- 659 • Pending shall contain the Number of Remaining Sub-operations Attribute
- 660 • Cancel may contain the Number of Remaining Sub-operations Attribute
- 661 • Warning, Failure, or Success shall not contain the Number of Remaining Sub-operations Attribute

662 **C.4.2.1.7 Number of Completed Sub-Operations**

663 Inclusion of the Number of Completed Sub-operations is conditional based upon the status in the C-MOVE response.  
664 The Number of Completed sub-operations specifies the number of C-STORE sub-operations generated by the re-  
665 quested transfer that have completed successfully.

666 A C-MOVE response with a status of:

- 667 • Pending shall contain the Number of Completed Sub-operations Attribute
- 668 • Cancel, Warning, Failure, or Success may contain the Number of Completed Sub-operations Attribute

669 **C.4.2.1.8 Number of Failed Sub-Operations**

670 Inclusion of the Number of Failed Sub-operations is conditional based upon the status in the C-MOVE response. The  
671 Number of Failed sub-operations specifies the number of C-STORE sub-operations generated by the requested trans-  
672 fer that have failed.

673 A C-MOVE response with a status of:

- 674 • Pending shall contain the Number of Failed Sub-operations Attribute
- 675 • Cancel, Warning, Failure, or Success may contain the Number of Failed Sub-operations Attribute

676 **C.4.2.1.9 Number of Warning Sub-Operations**

677 Inclusion of the Number of Warning Sub-operations is conditional based upon the status in the C-MOVE response. The  
678 Number of Warning sub-operations specifies the number of C-STORE sub-operations generated by the requested  
679 transfer that had a status of warning.

680 A C-MOVE response with a status of:

- 681 • Pending shall contain the Number of Warnings Sub-operations Attribute
- 682 • Cancel, Warning, Failure, or Success may contain the Number of Warning Sub-operations Attribute

683 **C.4.2.2 C-MOVE SCU Behavior**

684 This Section discusses both the baseline and extended behavior of the C-MOVE SCU.

685 **C.4.2.2.1 Baseline Behavior of SCU**

686 An SCU conveys the following semantics with a C-MOVE request:

687 • The SCU shall supply a single value in the Unique Key Attribute for each level above the Query/Retrieve level. For  
688 the level of retrieve, the SCU shall supply a single value for one unique key if the level of retrieve is above the STUDY  
689 level and shall supply one UID, or a list of UIDs if a retrieval of several items is desired and the retrieve level is  
690 STUDY, SERIES or IMAGE. The SCU shall also supply a move destination. The move destination shall be the  
691 DICOM Application Entity Title of a DICOM Application Entity capable of serving as the SCP of the Storage Service  
692 Class.

693 • The SCU shall interpret responses to the C-MOVE with status equal to Pending during the processing of the C-  
694 STORE sub-operations. These responses shall indicate the number of Remaining, Completed, Failed, and Warning  
695 C-STORE sub-operations.

696 • The SCU shall interpret responses with a status equal to Success, Warning or Failure as final responses. The final  
697 response shall indicate the number of Successful C-STORE sub-operations and the number of Failed C-STORE  
698 sub-operations resulting from the C-MOVE operation. The SCU shall interpret a status of:

- 699 • Success to indicate that all sub-operations were successfully completed
- 700 • Warning to indicate one or more sub-operations were successfully completed and one or more sub-operations  
701 were unsuccessful or had a status of warning, or all sub-operations had a status of warning
- 702 • Failure to indicate all sub-operations were unsuccessful.

703 • The SCU may cancel the C-MOVE service by issuing a C-CANCEL-MOVE request at any time during the processing  
704 of the C-MOVE. The SCU shall interpret a C-MOVE response with a status of Cancel to indicate the transfer was  
705 canceled. The C-MOVE response with a status of Cancel shall contain the number of Completed, Failed, and Warn-  
706 ing C-STORE sub-operations. If present, the Remaining sub-operations count shall contain the number of C-STORE  
707 sub-operations that were not initiated due to the C-CANCEL-MOVE request.

708 **C.4.2.2.2 Extended Behavior of SCU**

709 Extended SCU behavior shall be negotiated at Association establishment time. If an option within the extended behavior  
710 is not agreed upon in the negotiation, then only baseline SCU behavior shall be performed with respect to that option.  
711 Extended SCU behavior includes all baseline behavior with the following option:

- 712 • Relational-retrieve

- 713 • Enhanced Multi-Frame Image Conversion

714 More than one option may be agreed upon.

#### 715 **C.4.2.2.2.1 Relational-Retrieve**

716 The C-MOVE Service with relational-retrieve removes the restriction that the SCU supply Unique Key values for levels  
717 above the Query/Retrieve level to identify an entity at the level of the retrieval. Hence, the Identifier of a C-MOVE  
718 request may transfer:

- 719 • all Composite Object Instances related to a study by only providing a Study Instance UID (0020,000D)
- 720 • all Composite Object Instances related to a series by only providing a Series Instance UID (0020,000E)

- 721 • individual Composite Object Instances by only providing a list of SOP Instance UIDs (0008,0018)

#### 722 **C.4.2.2.2.2 Enhanced Multi-Frame Image Conversion**

723 The C-MOVE Service with Enhanced Multi-Frame Image Conversion allows for selection of the default or an alternative  
724 view of the instances represented by the Information Model, and hence the retrieval of either the legacy or the converted  
725 images, together with any unconverted instances, all of which are required to be processed to maintain referential  
726 integrity within the scope of the Patient.

727 Support for Enhanced Multi-Frame Image Conversion allows the SCU to specify the Attribute Query/Retrieve View  
728 (0008,0053) in the request Identifier with a value of either "CLASSIC" or "ENHANCED".

729 If Query/Retrieve View (0008,0053) is not present in the request Identifier, then the SCU requests that the SCP provide  
730 all the requested instances it possesses, as received.

731 If Query/Retrieve View (0008,0053) is present with a value of "CLASSIC", then the SCU requests that the SCP provide  
732 all the Classic single frame Instances (converted from Enhanced multi-frame Instances if required), as well as any  
733 instances that were converted to preserve referential integrity, and any that did not need to be converted.

734 If Query/Retrieve View (0008,0053) is present with a value of "ENHANCED", then the SCU requests that the SCP  
735 provide all the Enhanced multi-frame Instances (converted from Classic single frame Instances if required), as well as  
736 any instances that were converted to preserve referential integrity, and any that did not need to be converted.

737 Note

- 738 1. The SCU may assume that no duplicate information will be provided. For example, if an entire series of  
739 single frame instances can be converted to a separate series of converted instances, a STUDY level C-  
740 MOVE will not provide both series.
- 741 2. The Query Information Model is unchanged, and the same unique keys are equally applicable to both  
742 views, except that the values for the SERIES and IMAGE level queries will be different and will depend  
743 on the converted instance content.
- 744 3. The Query/Retrieve View is still required in an IMAGE or SERIES level request Identifier, even though  
745 the requested unique key(s) are unambiguous, and the view is in a sense "redundant", because the con-  
746 version that created the requested instances may not have been executed yet. It is not permitted to specify  
747 a view that is inconsistent with the requested unique key(s).

#### 748 **C.4.2.3 C-MOVE SCP Behavior**

749 This section discusses both the baseline and extended behavior of the C-MOVE SCP.

##### 750 **C.4.2.3.1 Baseline Behavior of SCP**

751 An SCP conveys the following semantics with a C-MOVE response:

- 752 • The SCP shall identify a set of Entities at the level of the transfer based upon the values in the Unique Keys in the  
753 Identifier of the C-MOVE request. The SCP shall initiate C-STORE sub-operations for the corresponding storage  
754 SOP Instances. These C-STORE sub-operations shall occur on a different Association (that may already exist) from  
755 the C-MOVE operation. The SCP of the Query/Retrieve Service Class shall serve as an SCU of the Storage Service  
756 Class.
- 757 • The SCP shall either reuse an established and compatible Association or establish a new Association for the C-  
758 STORE sub-operations. The SCP shall initiate C-STORE sub-operations over that Association for all stored SOP

- 759 Instances related to the Patient ID, List of Study Instance UIDs, List of Series Instance UIDs, or List of SOP Instance  
760 UIDs depending on the Query/Retrieve level specified in the C-MOVE request. A sub-operation is considered a  
761 Failure if the SCP is unable to negotiate an appropriate presentation context for a given stored SOP Instance.
- 762 • Optionally, the SCP may generate responses to the C-MOVE with status equal to Pending during the processing of  
763 the C-STORE sub-operations. These responses shall indicate the Remaining, Completed, Failed, and Warning C-  
764 STORE sub-operations.
  - 765 • When the number of Remaining sub-operations reaches zero, the SCP shall generate a final response with a status  
766 equal to Success, Warning or Failure. This response shall indicate the number of Completed sub-operations, the  
767 number of Failed sub-operations, and the number of sub-operations with Warning Status. The status contained in  
768 the C-MOVE response shall contain:
    - 769 • Success if all sub-operations were successfully completed
    - 770 • Warning if one or more sub-operations were successfully completed and one or more sub-operations were unsuc-  
771 cessful or had a warning status
    - 772 • Warning if all sub-operations had a warning status
    - 773 • Failure if all sub-operations were unsuccessful
  - 774 • The SCP may receive a C-CANCEL-MOVE request at any time during the processing of the C-MOVE. The SCP  
775 shall interrupt all C-STORE sub-operation processing and return a status of Cancel in the C-MOVE response. The  
776 C-MOVE response with a status of Cancel shall contain the number of Completed, Failed, and Warning C-STORE  
777 sub-operations. If present, the Remaining sub-operations count shall contain the number of C-STORE sub-operations  
778 that were not initiated due to the C-CANCEL-MOVE request.
  - 779 • If the SCP manages images in multiple alternate encodings (see Section C.6.1.1.5.1), only one of the alternate en-  
780 codings of an image shall be included in the set of object instances retrieved by a C-MOVE request at the Patient,  
781 Study, or Series level.

782 Note

783 For retrieval of images with alternate encodings using a C-MOVE request at the Patient, Study, or Series  
784 level, the SCP may select the appropriately encoded Instance for the retrieval based on identity of the SCU,  
785 transfer syntaxes accepted in the C-STORE Association Negotiation, or other factors.

786 Note

787 If the association on which the C-MOVE operation was issued is abnormally terminated, then it will not be  
788 possible to issue any further pending responses nor a final response, nor will C-CANCEL-MOVE requests be  
789 received. The behavior of the C-MOVE SCP acting as a C-STORE SCU is undefined in this condition. Spe-  
790 cifically, whether or not any uncompleted C-STORE sub-operations continue is undefined.

791

#### 792 **C.4.2.3.1.1 Alternative Sub-operation Mechanism Option**

793 **When this Option is supported, a C-MOVE SCP may, at its discretion, satisfy the storage sub-oper-**  
794 **ations of a C-MOVE by acting as a user agent in a Store Transaction, issuing one or more Store**  
795 **requests to the specific DICOMweb endpoint determined by a local policy (e.g., by mapping the**  
796 **provided AE Title to its endpoint equivalent). See Sections 10.5 and 12.5 in PS3.18 for more infor-**  
797 **mation about the Store Transaction.**

#### 798 **C.4.2.3.1.1.1 Scope**

799 **This Option applies to all Query/Retrieve Information Model – MOVE SOP Classes.**

#### 800 **C.4.2.3.1.1.2 Interface**

801 **This option does not change the interface between the C-MOVE SCU and C-MOVE SCP. Regarding**  
802 **the destination, the C-MOVE SCU continues to supply only a Move Destination AE Title in the**  
803 **C-MOVE request; it does not supply URLs, credentials, or delivery preferences. The C-MOVE SCP**  
804 **selects the delivery mechanism, taking the capabilities of the destination into account.**

805 **C.4.2.3.1.1.3 Behavior**

806 **When the C-MOVE SCP chooses to utilize Store transactions under this Option, storage is per-**  
807 **formed using HTTP messages.**

808 **The C-MOVE SCP shall comply with section C.4.2.3 changing C-STORE operation to DICOMweb**  
809 **STOW while maintaining the same semantics and per-instance sub-operation bookkeeping.**

810 **Note Good idea to provide user identification [Rob].**

811 **C.4.2.3.1.1.4 Sub-operation Counters**

812 **The C-MOVE SCP shall maintain four counters and treat them as mutually exclusive categories for**  
813 **per-instance outcomes:**

- 814 a) **Completed = stored with no warnings;**  
815 b) **Warning = stored with warnings;**  
816 c) **Failed = not stored;**  
817 d) **Remaining = not yet attempted or not yet known.**

818 **The sum of Completed, Warning, Failed, and Remaining shall always equal the number of Compo-**  
819 **site SOP Instances matching the Identifier of the C-MOVE request.**

820 **When using Store transactions, the C-MOVE SCP shall count the per-instance outcome from the**  
821 **Store Instance Response Module (see PS3.18, Annex I) as follows:**

- 822 • **SOP Instance UIDs in the Referenced SOP Sequence (0008,1199) that have no associated**  
823 **Warning Reason (0008,1196) shall increment the Completed counter;**  
824 • **SOP Instance UIDs in the Referenced SOP Sequence (0008,1199) having an associated Warn-**  
825 **ing Reason (0008,1196) shall increment the Warning counter;**  
826 • **SOP Instance UIDs in the Failed SOP Sequence (0008,1198) shall increment the Failed counter.**

827 **All SOP Instance UIDs in the Failed SOP Sequence (0008,1198) of the Store Instance Response**  
828 **Module shall be added to the Failed SOP Instance UID List (0008,0058) of the C-MOVE Response**  
829 **Identifier.**

830 **C.4.2.3.1.1.5 Mixed-mechanism Delivery**

831 **The C-MOVE SCP may perform some sub-operations via Store transactions and others via**  
832 **C-STORE. It shall maintain a single set of counters for the entire C-MOVE.**

833 **C.4.2.3.1.1.6 Pending Responses**

834 **A C-MOVE SCP may send zero or more Pending responses during processing. When it sends a**  
835 **Pending response, it shall report the current values of the Remaining, Completed, Warning, and**  
836 **Failed counters.**

837 **C.4.2.3.1.1.7 Mixed-mechanism Failure Retry**

838 **In the event a C-STORE sub-operation yields Failure, it might be retried with a DICOMweb Store**  
839 **Transaction. Likewise, when a DICOMweb Store Transaction sub-operation yields Failure, it might**  
840 **be retried with a DIMSE C-STORE operation. Implementations may be configured to such mixed-**  
841 **mechanism failure retries.**

842 **Warnings shall not be retried. This is because Warnings are a partial Success.**

843 **C.4.2.3.1.1.8 Backward Compatibility**

844 **A C-MOVE SCP claiming this Option shall still support C-STORE sub-operations to the Move Des-**  
845 **termination AE Title.**

846 **C.4.2.3.1.1.9 Conformance**

847 **A C-MOVE SCP claiming this option shall describe in its Conformance Statement:**

- 848 • **decision logic for when it uses a C-STORE operation or a Store Transaction**
- 849 • **rules for mapping AE Titles to endpoints**
- 850 • **security controls (use of HTTPS, credential sourcing, allow-lists)**

851 **C.4.2.3.1.2 Proxy Option**

852 **When this option is supported, a C-MOVE SCP may, at its discretion, proxy the C-MOVE request**  
853 **using a Send Transaction Request.**

854 **This involves:**

- 855 • **converting the C-MOVE request content into equivalent Send Transaction content**
- 856 • **mapping the C-MOVE destination AE Title to the DICOMweb endpoint, determined by a lo-**  
857 **cal policy**
- 858 • **issuing a Send Transaction request as a user agent**
- 859 • **reformatting the results into the DIMSE format**
- 860 • **sending the DIMSE results in the C-MOVE responses**

861 **See Sections 10.X and 12.X in PS3.18 for more information about the Send transaction.**

862 **C.4.2.3.1.2.1 Scope**

863 **This Option applies to all Query/Retrieve Information Model – MOVE SOP Classes.**

864 **C.4.2.3.1.2.2 Interface**

865 **This option does not change the interface between the C-MOVE SCU and C-MOVE SCP. The**  
866 **C-MOVE SCU supplies a Move Destination AE Title in the C-MOVE request. It may supply creden-**  
867 **tials in the association negotiation. It will not supply URLs or delivery preferences. The C-MOVE**  
868 **SCP selects the delivery mechanism, taking the capabilities of the destination into account.**

869 **C.4.2.3.1.2.3 Behavior**

870 **When the C-MOVE SCP selects to utilize a Send transaction under this Option, communication**  
871 **with the Send origin server is performed using HTTP messages.**

872 **The C-MOVE SCP proxy delegates all responsibilities of the C-MOVE behavior to the Send origin**  
873 **server, including maintaining sub-operation counters (see Section C.4.2.3.1.1.4). The C-MOVE SCP**  
874 **proxy shall transform received messages into corresponding messages in the other protocol and**  
875 **forward them.**

876 **C.4.2.3.1.2.3 Conformance**

877 **A C-MOVE SCP claiming this option shall describe in its Conformance Statement:**

- 878 • **rules for mapping AE Titles to endpoints**
- 879 • **security controls (use of HTTPS, credential sourcing, allow-lists)**

880 **C.4.2.3.2 Extended Behavior of SCP**

881 Extended SCP behavior shall be negotiated at Association establishment time. If an option within the extended behavior  
882 is not agreed upon in the negotiation, then only baseline SCP behavior shall be performed with respect to that option.  
883 Extended SCP behavior includes all baseline behavior with the following option:

- 884 • Relational-retrieve
- 885 • Enhanced Multi-Frame Image Conversion

886 More than one option may be agreed upon.

887 **C.4.2.3.2.1 Relational-Retrieve**

888 The C-MOVE Service with relational-retrieve removes the restriction that the SCU supplies Unique Key values for levels  
889 above the Query/Retrieve level to help identify an entity at the level of the retrieval. Hence, the Identifier of a C-MOVE  
890 request may specify the transfer of:

- 891 • all Composite Object Instances related to a study by only providing a Study Instance UID (0020,000D)
- 892 • all Composite Object Instances related to a series by only providing a Series Instance UID (0020,000E)
- 893 • individual Composite Object Instances by only providing a list of SOP Instance UIDs (0008,0018)

894 **C.4.2.3.2.2 Enhanced Multi-Frame Image Conversion**

895 If Query/Retrieve View (0008,0053) is not present in the request Identifier, then the SCP shall identify a set of Entities  
896 at the level of the transfer based upon the values in the Unique Keys in the Identifier of the C-MOVE request that  
897 correspond to the instances it possesses, as received, and shall initiate C-STORE sub-operations for all the corre-  
898 sponding storage SOP Instances.

899 If Query/Retrieve View (0008,0053) is present with a value of "CLASSIC", then the SCP shall identify a set of Entities  
900 at the level of the transfer based upon the values in the Unique Keys in the Identifier of the C-MOVE request that  
901 correspond to the Classic single frame Instances (converted from Enhanced multi-frame Instances if required), as well  
902 as any instances that were converted to preserve referential integrity, and any that did not need to be converted, and  
903 shall initiate C-STORE sub-operations for all the corresponding storage SOP Instances.

904 If Query/Retrieve View (0008,0053) is present with a value of "ENHANCED", then the SCP shall identify a set of Entities  
905 at the level of the transfer based upon the values in the Unique Keys in the Identifier of the C-MOVE request that  
906 correspond to the Enhanced multi-frame Instances (converted from Classic single frame Instances if required), as well  
907 as any instances that were converted to preserve referential integrity, and any that did not need to be converted, and  
908 shall initiate C-STORE sub-operations for all the corresponding storage SOP Instances.

909 Note

- 910 1. The SCP will not send information that is duplicated to the C-STORE SCP. For example, if an entire  
911 series of single frame instances can be converted to a separate series of converted instances, a STUDY  
912 level C-MOVE will not send both series.
- 913 2. The C-STORE SCP will need to support the necessary SOP Classes for converted instances, otherwise  
914 the C-STORE sub-operations will fail in the normal manner and this will be reflected in the C-MOVE  
915 responses.
- 916 3. The Query Information Model is unchanged, and the same unique, required and optional keys are equally  
917 applicable to both views, except that the values for the SERIES and IMAGE level queries will be different  
918 and will depend on the converted instance content.
- 919 4. The Query/Retrieve View is still required in an IMAGE or SERIES level request Identifier, even though  
920 the requested unique key(s) are unambiguous.

921

922

## Changes to NEMA Standards Publications PS 3.2

923

**Add to N.1.3 for the Send transactions**

924

### **N.1 Overview**

925

...

926

### **N.1.3 DICOM Web Services**

927

...

928

#### **N.1.3.2 Studies Service**

929

Table N.1-9 lists details on the support of the Studies Service.

930

[Complete Table N.1-9 to indicate support for the Studies Web Service]

931

**Table N.1-9. Study Service**

Service	Transaction	Resource	User Agent	Origin Server
Studies Web Service	Retrieve Capabilities			
	<i>Retrieve (WADO-RS)</i>	Study		
		Study Metadata		
		<i>Study Bulkdata</i>		
		<i>Study Pixel Data</i>		
		Rendered Study		
		<i>Rendered MPR Volume Study</i>		
		<i>Rendered 3D Volume Study</i>		
		<i>Study Thumbnail</i>		
		Series		
		Series Metadata		
		<i>Series Bulkdata</i>		
		<i>Series Pixel Data</i>		
		Rendered Series		
		<i>Rendered MPR Volume Series</i>		
		<i>Rendered 3D Volume Series</i>		
		<i>Series Thumbnail</i>		

Service	Transaction	Resource	User Agent	Origin Server
		Instance		
		Instance Metadata		
		Instance Bulkdata		
		<i>Instance Pixel Data</i>		
		Rendered Instance		
		<i>Rendered MPR Volume Instance</i>		
		<i>Rendered 3D Volume Instance</i>		
		<i>Instance Thumbnail</i>		
		Frames		
		Rendered Frames		
		<i>Rendered MPR Volume Frames</i>		
		<i>Rendered 3D Volume Frames</i>		
		<i>Frame Thumbnail</i>		
		Bulkdata		
	<i>Search (QIDO-RS)</i>	All Studies		
		Study's Series		
		Study's Instances		
		All Series		
		Series Instances		
		All Instances		
	<i>Store (STOW-RS)</i>	All Studies		
		Study		
	<b><u>Send and Check Send Result (SEND-RS)</u></b>	<b><u>All Studies Send Requests</u></b>		
		<b><u>Study's Series Send Requests</u></b>		
		<b><u>Study's Instances Send Requests</u></b>		
		<b><u>All Series Send Requests</u></b>		

Service	Transaction	Resource	User Agent	Origin Server
		<u>Study Series' Instances Send Requests</u>		
		<u>All Instances Send Requests</u>		

932 ...

933 **N.1.3.4 Non-Patient Instance Service**

934 Table N.1-11 lists details on the support of Non-Patient Instances Service.

935 For details on the supported resource categories (e.g., Color Palette, Defined Procedure Protocol, Hanging Protocol,  
936 **or** Implant Templates, **or Protocol Approvals**), see Table N.1-1.

937 *[Complete Table N.1-11 to indicate support for the Non-Patient Instance Web Service.]*

938  
939

**Table N.1-11. Non-Patient Instance Service**

Service	Transaction	Resource	User Agent	Origin Server
Non-Patient Instances Web Service	Retrieve Capabilities			
	<i>Retrieve</i>			
	<i>Store</i>			
	<i>Search (Note)</i>			
	<u><i>Send and Check Send Result</i></u>			

940

941

942 **Add to section N.5.2 Supported DIMSE Services for C-MOVE Options**

943 **N.5 Service and Interoperability Description**

944 ...

945 **N.5.2 DIMSE Services**

946 ...

947 **N.5.2.7 Query/Retrieve Service Class**

948 ...

949 **N.5.2.7.7 SCP of the Patient Root Q/R Information Model - MOVE SOP Class**

950 As the SCP of the Study Root Q/R - Information Model - MOVE, the <Product> receives the C-MOVE-RQ and in turn  
951 uses the C-STORE-RQ sub operation to send matching SOP Instances to the Move Destination AE included in the  
952 C-MOVE-RQ.

953 *[Provide a list of Storage SOP Classes supported or reference Storage Table in Overview e.g.]*

954 *As the SCP of the Storage Service Class, all Storage SOP Classes listed in Section N.1.1 are supported.*

955 *[Describe the relationship between the incoming C-MOVE Request and the C-STORE Sub-operation, e.g., is each  
956 instance sent on one Association or is the same Association used for all instances, is this behavior configurable.]*

957 *[Describe your product behavior if a C-MOVE-CANCEL Request is received.]*

958 **[Describe whether <product> supports the Alternative Sub-operation Mechanism Option, as described in**  
959 **PS3.4, C.4.2.3.1.1. If so, provide decision logic for when it uses a C-STORE operation or a Store Transaction,**  
960 **rules for mapping AE Titles to endpoints, and security controls (use of HTTPS, credential sourcing, al-**  
961 **low-lists).]**

962 **[Describe whether <product> supports the Proxy Option, as described in PS3.4, C.4.2.3.1.2. If so, provide**  
963 **rules for mapping AE Titles to endpoints, and security controls (use of HTTPS, credential sourcing, al-**  
964 **low-lists).]**

965 **N.5.2.7.8 SCP of the Study Root Q/R Information Model - MOVE SOP Class**

966 *[If this SOP Class is supported, fill in the section as indicated in Section N.5.2.7.7.]*

967

968	<b>Add to section N.5.3 Supported DICOM Web Services for the Send Transactions</b>	
969	<b>N.5</b>	<b>Service and Interoperability Description</b>
970	...	
971	<b>N.5.3</b>	<b>DICOM Web Services</b>
972	...	
973	<b>N.5.3.2</b>	<b>Studies Web Service</b>
974	...	
975	<b><u>N.5.3.2.X</u></b>	<b><u>Send Transactions (SEND-RS)</u></b>
976	<b><u>The Studies Web Service Send Transactions are also known as SEND-RS.</u></b>	
977		
978	<b><u>N.5.3.2.X.1</u></b>	<b><u>User Agent</u></b>
979	<b><u>[Indicate whether your product supports sending matching instances to itself and/or to a different endpoint.]</u></b>	
980	<b><u>[Indicate, only when your product supports sending matching instances to itself, its behavior in case no</u></b>	
981	<b><u>Store Request is received after a specific time, e.g., &lt;Product&gt; expects to receive the Store Request in a con-</u></b>	
982	<b><u>figurable time frame after the Send Request is sent. If no Store Requests are received within this configura-</u></b>	
983	<b><u>ble timeframe, it repeats the Send Request.]</u></b>	
984	<b><u>[Indicate, only when your product supports sending matching instances to itself, whether it accepts C-</u></b>	
985	<b><u>STORE or STOW or both kinds of Store Requests. This can be done by referring to the applicable sections in</u></b>	
986	<b><u>this conformance statement.]</u></b>	
987		
988	<b><u>N.5.3.2.X.2</u></b>	<b><u>Origin Server</u></b>
989	<b><u>[Indicate whether &lt;Product&gt; performs Store sub-operations by using DIMSE (with C-STOREs) or by using</u></b>	
990	<b><u>DICOMweb (with STOW). Indicate, when applicable, how &lt;Product&gt; determines what to use and whether this</u></b>	
991	<b><u>is configurable.]</u></b>	
992	<b><u>[In case the Store sub-operations are performed using DIMSE, indicate the relationship between the incom-</u></b>	
993	<b><u>ing Send Request and the C-STORE sub-operation, e.g., is each instance sent on a separate Association or</u></b>	
994	<b><u>are all instances sent on the same Association. Indicate whether, and how, this behavior is configurable.</u></b>	
995	<b><u>Also indicate &lt;Product&gt;'s function to come to an AE Title given the URI destination.]</u></b>	
996	...	
997	<b>N.5.3.4</b>	<b>Non-Patient Instance Web Service</b>
998	...	
999	<b><u>N.5.3.4.X</u></b>	<b><u>Send Transactions</u></b>
1000	<b><u>N.5.3.4.X.1</u></b>	<b><u>User Agent</u></b>
1001	<b><u>[Indicate whether your product supports sending matching instances to itself and/or to a different endpoint.]</u></b>	
1002	<b><u>[Indicate, only when your product supports sending matching instances to itself, its behavior in case no</u></b>	
1003	<b><u>Store Request is received after a specific time, e.g., &lt;Product&gt; expects to receive the Store Request in a con-</u></b>	
1004	<b><u>figurable time frame after the Send Request is sent. If no Store Requests are received within this configura-</u></b>	
1005	<b><u>ble timeframe, it repeats the Send Request.]</u></b>	
1006	<b><u>[Indicate, only when your product supports sending matching instances to itself, whether it accepts C-</u></b>	
1007	<b><u>STORE or STOW or both kinds of Store Requests. This can be done by referring to the applicable sections in</u></b>	
1008	<b><u>this conformance statement.]</u></b>	

1009

1010 **N.5.3.4.X.2 Origin Server**

1011 *[Indicate whether <Product> performs Store sub-operations by using DIMSE (with C-STOREs) or by using*  
 1012 *DICOMweb (with STOW). Indicate, when applicable, how <Product> determines what to use and whether this*  
 1013 *is configurable.]*

1014 *[In case the Store sub-operations are performed using DIMSE, indicate the relationship between the incom-*  
 1015 *ing Send Request and the C-STORE sub-operation, e.g., is each instance sent on a separate Association or*  
 1016 *are all instances sent on the same Association. Indicate whether, and how, this behavior is configurable.*  
 1017 *Also indicate <Product>'s function to come to an AE Title given the URI destination.]*

Add to section N.6.3 Configuration of DICOM Web Services for the Send Transactions

1019 **N.6.3 Configuration of DICOM Web Services**

1020 ...

1021 **N.6.3.2 Studies Web Service Configuration**

1022 ...

1023 **N.6.3.2.X Send Transactions (SEND-RS) Configuration**

1024 **The Send Transactions are also known as SEND-RS. Table N.6-X lists configuration parameters for the Send**  
 1025 **Transactions of the Studies Web Service:**

1026 *[Remove the unsupported parameters from the local and remote configuration parameters.]*

1027 **Table N.6-X. Send Transaction Parameters**  
 1028

<b><u>Local Configuration Parameters - Send Transactions</u></b>			
<b><u>Parameter</u></b>	<b><u>Configurable</u></b>	<b><u>Default Value</u></b>	<b><u>Comments</u></b>
	<b><u>&lt;&lt;USER SERVICE FIXED&gt;&gt;</u></b>	<b><u>[If there is no default, leave blank]</u></b>	<b><u>[Provide comments or Values/ranges if appli- cable]</u></b>
<b><u>Send local Origin Server URL (Base URI)</u></b>	<b><u>FIXED</u></b>	<b><u>http://&lt;hostname&gt;:&lt;port&gt;/send</u></b>	
<b><u>Port</u></b>	<b><u>SERVICE</u></b>	<b><u>8080</u></b>	
<b><u>Secured Send local Origin Server URL (Base URI)</u></b>		<b><u>https://&lt;hostname&gt;:&lt;securedport&gt;/send</u></b>	
<b><u>Secured Port</u></b>		<b><u>8081</u></b>	
<b><u>&lt;Specific Send Trans- action parameter&gt;</u></b>			
<b><u>Remote Configuration Parameters - Send Transactions</u></b>			

<b>Local Configuration Parameters - Send Transactions</b>			
<u>Parameter</u>	<u>Configurable</u>	<u>Default Value</u>	<u>Comments</u>
<i>[Either document the number of supported remote hosts, e.g., &lt;Product&gt; supports configuration of up to &lt;X&gt; remote hosts, or state that there is no limitation other than those mandated by the operating system.]</i>			
<u>Parameter</u>	<u>Configurable</u>	<u>Default Value</u>	<u>Comments</u>
	<u>&lt;&lt;USER SERVICE FIXED&gt;&gt;</u>	<i>[If there is no default, leave blank]</i>	<i>[Provide comments or Values/ranges if applicable]</i>
<u>Send remote Origin Server URL</u>	<u>SERVICE</u>	<u>http://&lt;hostname&gt;:&lt;port&gt;/send</u>	
<u>Port</u>	<u>SERVICE</u>	<u>8080</u>	
<u>Secured Send remote Origin Server URL</u>	<u>SERVICE</u>	<u>https://&lt;hostname&gt;:&lt;securedport&gt;/send</u>	
<u>Secured Port</u>	<u>SERVICE</u>	<u>8081</u>	
<u>&lt;Specific Send Transaction parameter&gt;</u>			

1029

1030 ...

1031 **Add to section N.7.3.3 DICOM Web Services for the Send Transactions**

1032 **N.7 Network and Media Communication Details**

1033 ...

1034 **N.7.3 Status Codes**

1035 ...

1036 **N.7.3.3 DICOM Web Services**

1037 ...

1038 **N.7.3.3.3 Studies Service**

1039 ...

1040 **N.7.3.3.3.X1 Send Transaction as Origin Server**

1041 **Table N.7.3.3.3.X1-1 lists the Status Codes that an origin server supports for the Send Transaction of the Studies Service and the condition in which any of the listed Status Codes is sent.**

1042

1043 **[Describe below the condition in which the application sends the specific Status Codes in the Send Transaction response as origin server.]**

1044

1045 **Table N.7.3.3.3.X1-1. Status Codes of Origin Server for Send Transaction**

1046

<u>Status</u>	<u>Code</u>	<u>Condition</u>
<u>Success</u>	<u>200 (OK)</u>	<u>The origin server finished processing the send request</u>
	<u>202 (Accepted)</u>	<u>The origin server has not finished processing the send request yet</u>
<u>Failure</u>	<u>400 (Bad Request)</u>	<u>The origin server cannot handle the send request because of errors in the request headers or parameters</u>
	<u>409 (Conflict)</u>	<u>The origin server cannot handle the send request because the provided Transaction UID is already in use</u>
	<u>503 (Service Unavailable)</u>	<u>The origin server cannot handle the send request; this may be a temporal or permanent state</u>

1047

1048 **N.7.3.3.3.X2 Send Transaction as User Agent**

1049 **Table N.7.3.3.3.X2-1 lists the Status Codes that a user agent supports for the Send Transaction of the Studies**  
 1050 **Service and defines the application behavior, when encountering any of the listed Status Codes.**

1051 **[Describe below the behavior of the application when it receives various Status Codes in the Request Trans-**  
 1052 **action response.]**

1053  
1054

**Table N.7.3.3.3.X2-1. Status Codes of User Agent for Send Transaction**

<u>Status</u>	<u>Code</u>	<u>Behavior</u>
<u>Success</u>	<u>200 (OK)</u>	<u>Continue with completion behavior</u>
	<u>202 (Accepted)</u>	<u>Query later to get the result of the request</u>
<u>Failure</u>	<u>400 (Bad Request)</u>	<u>Report the failure</u>
	<u>409 (Conflict)</u>	<u>Retry with another Transaction UID</u>
<u>*</u>	<u>Any other code</u>	<u>Report the failure</u>

1055

1056 **N.7.3.3.3.X3 Check Send Result Transaction as Origin Server**

1057 **Table N.7.3.3.3.X3-1 lists the Status Codes that an origin server supports for the Check Send Result Transac-**  
 1058 **tion of the Studies Service and the condition in which any of the listed Status Codes is sent.**

1059 **[Describe below the condition in which the application sends the specific Status Codes in the Check Send**  
 1060 **Result Transaction response as origin server.]**

1061  
1062

**Table N.7.3.3.3.X3-1. Status Codes of Origin Server for Check Send Result Transaction**

<u>Status</u>	<u>Code</u>	<u>Condition</u>
<u>Success</u>	<u>200 (OK)</u>	<u>The origin server finished processing the send request</u>
	<u>202 (Accepted)</u>	<u>The origin server has not finished processing the send request yet</u>

<u>Status</u>	<u>Code</u>	<u>Condition</u>
<u>Failure</u>	<u>404 (Not Found)</u>	<u>The origin server cannot find the send request result</u>
	<u>410 (Gone)</u>	<u>The origin server can no longer provide the send request result</u>
	<u>503 (Service Unavailable)</u>	<u>The origin server cannot handle the check send result request; this may be a temporary or permanent state</u>

1063

1064 **N.7.3.3.3.X4 Check Send Result Transaction as User Agent**

1065 **Table N.7.3.3.3.X4-1 lists the Status Codes that a user agent supports for the Check Send Result Transaction**  
 1066 **of the Studies Service and defines the application behavior when encountering any of the listed Status Codes.**

1067 **[Describe below the behavior of the application when it receives various Status Codes in the Check Send**  
 1068 **Result Transaction response.]**

1069 **Table N.7.3.3.3.X4-1. Status Codes of User Agent for Check Send Result Transaction**  
 1070

<u>Status</u>	<u>Code</u>	<u>Behavior</u>
<u>Success</u>	<u>200 (OK)</u>	<u>Continue with completion behavior</u>
	<u>202 (Accepted)</u>	<u>Query later to get the result of the request</u>
<u>Failure</u>	<u>404 (Not Found)</u>	<u>Start all over with a send request</u>
	<u>410 (Gone)</u>	<u>Start all over with a send request</u>
<u>*</u>	<u>Any other code</u>	<u>Report the failure</u>

1071

1072 ...

1073 **N.7.3.3.5 Non-Patient Instance Service**

1074 ...

1075 **N.7.3.3.5.X1 Send Transactions as Origin Server**

1076 **Table N.7.3.3.5.X1-1 lists the Status Codes that an origin server supports for the Send Transaction of the Non-**  
 1077 **Patient Instance Service and the condition in which any of the listed Status Codes is sent.**

1078 **[Describe below the condition in which the application sends the specific Status Codes in the Send Transaction**  
 1079 **response as origin server.]**

1080 **Table N.7.3.3.5.X1-1. Status Codes of Origin Server for Send Transaction**  
 1081

<u>Status</u>	<u>Code</u>	<u>Condition</u>
<u>Success</u>	<u>200 (OK)</u>	<u>The origin server finished processing the send request</u>
	<u>202 (Accepted)</u>	<u>The origin server has not finished processing the send request yet</u>

<u>Status</u>	<u>Code</u>	<u>Condition</u>
<b><u>Failure</u></b>	<b><u>400 (Bad Request)</u></b>	<b><u>The origin server cannot handle the send request because of errors in the request headers or parameters</u></b>
	<b><u>409 (Conflict)</u></b>	<b><u>The origin server cannot handle the send request because the provided Transaction UID is already in use</u></b>
	<b><u>503 (Service Unavailable)</u></b>	<b><u>The origin server cannot handle the send request; this may be a temporal or permanent state</u></b>

1082

1083 **N.7.3.3.5.X2 Send Transactions as User Agent**

1084 **Table N.7.3.3.5.X2-1 lists the Status Codes that a user agent supports for the Send Transaction of the Non-Patient Instance Service and defines the application behavior, when encountering any of the listed Status Codes.**

1087 **[Describe below the behavior of the application when it receives various Status Codes in the Request Transaction response.]**

1089 **Table N.7.3.3.5.X2-1. Status Codes of User Agent for Send Transaction**

1090

<u>Status</u>	<u>Code</u>	<u>Behavior</u>
<b><u>Success</u></b>	<b><u>200 (OK)</u></b>	<b><u>Continue with completion behavior</u></b>
	<b><u>202 (Accepted)</u></b>	<b><u>Query later to get the result of the request</u></b>
<b><u>Failure</u></b>	<b><u>400 (Bad Request)</u></b>	<b><u>Report the failure</u></b>
	<b><u>409 (Conflict)</u></b>	<b><u>Retry with another Transaction UID</u></b>
<b><u>*</u></b>	<b><u>Any other code</u></b>	<b><u>Report the failure</u></b>

1091

1092 **N.7.3.3.5.X3 Check Send Result Transaction as Origin Server**

1093 **Table N.7.3.3.5.X3-1 lists the Status Codes that an origin server supports for the Check Send Result Transaction of the Non-Patient Instance Service and the condition in which any of the listed Status Codes is sent.**

1095 **[Describe below the condition in which the application sends the specific Status Codes in the Check Send Result Transaction response as origin server.]**

1097 **Table N.7.3.3.3.X3-1. Status Codes of Origin Server for Check Send Result Transaction**

1098

<u>Status</u>	<u>Code</u>	<u>Condition</u>
<b><u>Success</u></b>	<b><u>200 (OK)</u></b>	<b><u>The origin server finished processing the send request</u></b>
	<b><u>202 (Accepted)</u></b>	<b><u>The origin server has not finished processing the send request yet</u></b>
<b><u>Failure</u></b>	<b><u>404 (Not Found)</u></b>	<b><u>The origin server cannot find the send request result</u></b>
	<b><u>410 (Gone)</u></b>	<b><u>The origin server can no longer provide the send request result</u></b>

<u>Status</u>	<u>Code</u>	<u>Condition</u>
	<u>503 (Service Unavailable)</u>	<u>The origin server cannot handle the check send result request; this may be a temporary or permanent state</u>

1099

1100 **N.7.3.3.5.X4 Check Send Result Transaction as User Agent**

1101 **Table N.7.3.3.5.X4-1 lists the Status Codes that a user agent supports for the Check Send Result Transaction**  
 1102 **of the Non-Patient Instance Service and defines the application behavior when encountering any of the listed**  
 1103 **Status Codes.**

1104 **[Describe below the behavior of the application when it receives various Status Codes in the Check Send**  
 1105 **Result Transaction response.]**

1106 **Table N.7.3.3.5.X4-1. Status Codes of User Agent for Check Send Result Transaction**

1107

<u>Status</u>	<u>Code</u>	<u>Behavior</u>
<u>Success</u>	<u>200 (OK)</u>	<u>Continue with completion behavior</u>
	<u>202 (Accepted)</u>	<u>Query later to get the result of the send request</u>
<u>Failure</u>	<u>404 (Not Found)</u>	<u>Start all over with a send request</u>
	<u>410 (Gone)</u>	<u>Start all over with a send request</u>
<u>* _</u>	<u>Any other code</u>	<u>Report the failure</u>

1108

1109

1110

**Changes to NEMA Standards Publications PS 3.6**

1111

*No new attributes have been introduced.*

1112

1113

**Changes to NEMA Standards Publications PS 3.15**

1114

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

1115