

DICOMweb Modality Services

WG27

First Read

August 2024

WORK ITEM 2023-10-C – DICOMWEB MODALITY SERVICES

Introduction

The DICOM Standard defines several services. Two of these are targeted towards modalities, namely the Modality Worklist service (see [PS3.4, Annex K](#), more specifically [K.6.1](#)) and the Modality Performed Procedure Step service (see [PS3.4, Annex F](#), more specifically [F.7-F.9](#)). Currently, these services are defined using DIMSE.

Limitations of Current Standard

Both the Modality Worklist service and the Modality Performed Procedure Step service are not yet available in DICOMweb. This limits a) the uptake of DICOMweb for modalities and b) the support of workflow services for modalities that are (intended to be) part of a web-based ecosystem.

Description of Proposal

Add the Modality Worklist and the Modality Performed Procedure Step services to DICOMweb, in principle based on the existing DICOMweb Worklist service (UPS-RS; see [PS3.18, section 11](#)). This would boil down to creating an informative annex and any normative changes needed if gaps are discovered.



PROGRESS

Done since last meeting

- Added slide on context of DICOMweb Modality Services
- Completed detailed sequence charts on SCP / User Agent proxy (happy flow) – in presentation
- Detailed many more sequence charts and started mapping of data
- Created initial document
- Identified several open issues

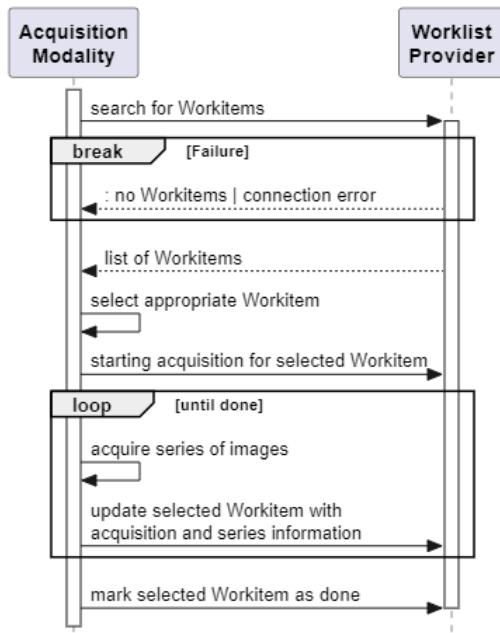
Agenda

- Review new slides and initial document, and draw conclusions

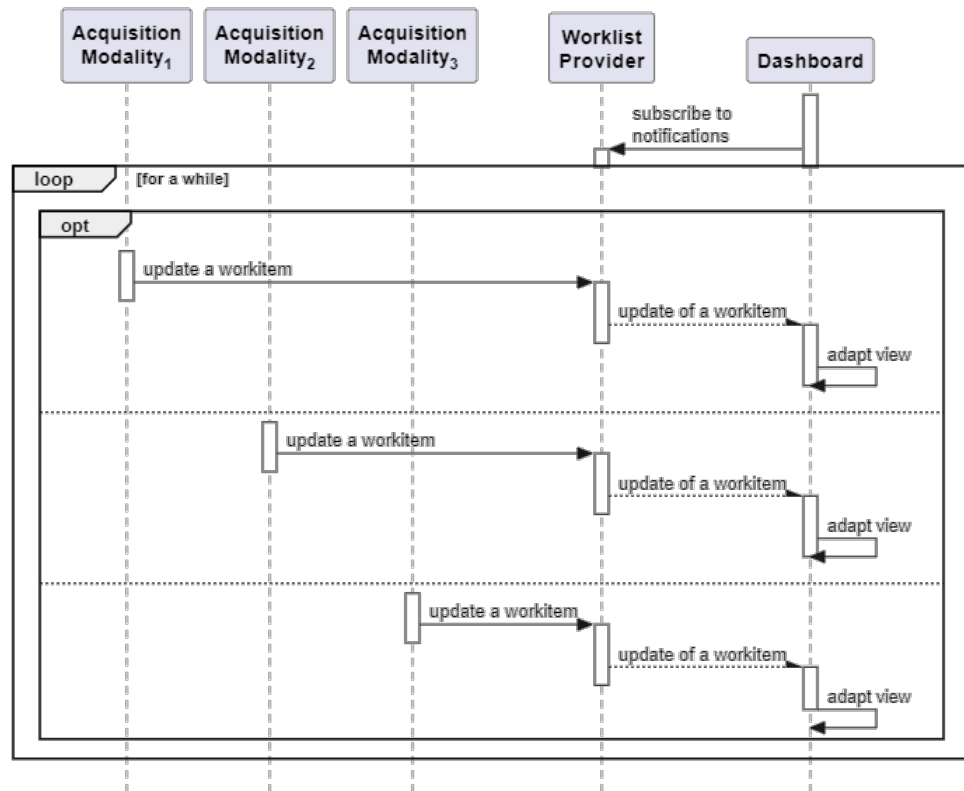


SCENARIOS SKETCHING THE CONTEXT

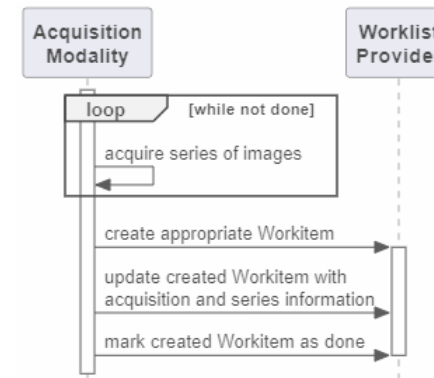
Scheduled Acquisition Workflow



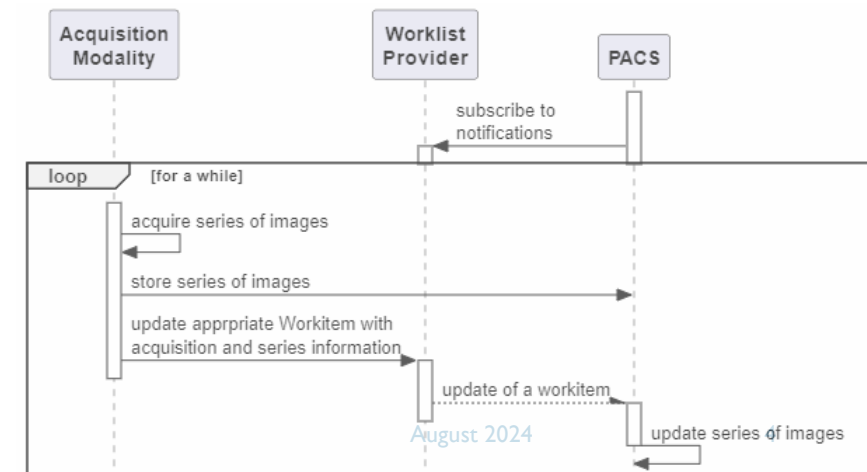
Acquisition Workflow Monitoring



Encounter-Based Acquisition Workflow

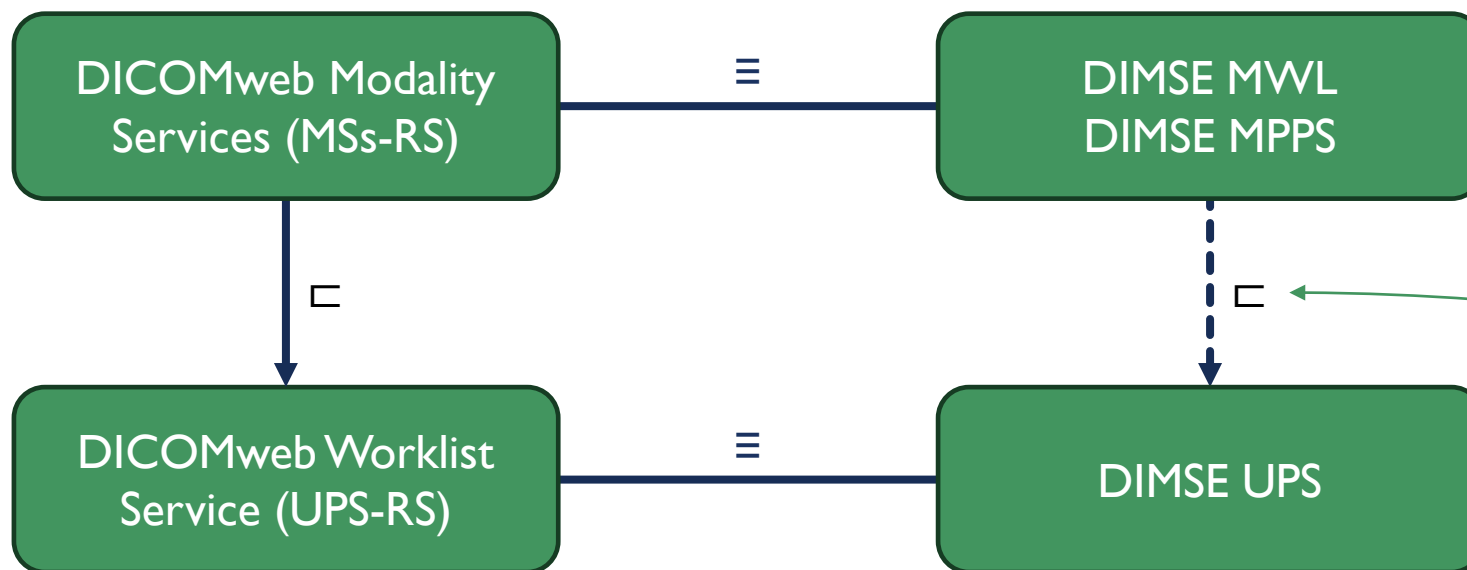


Reviewing Workflow (PS3.4, F.9, note 1)



OVERVIEW – RELATIONS BETWEEN SPECIFICATIONS

Symbol	Meaning
≡	Equivalence
⊃	Based-on



Implicit mapping from MWL and MPPS to UPS. Where is this mapping or do we need to create it?

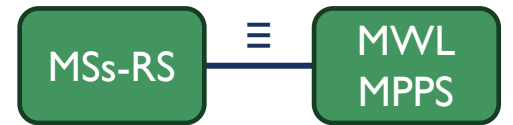
OVERVIEW – PRIMITIVES OF DICOMWEB MODALITY SERVICES

1. Get Applicable Workitems
2. Claim and Prepare Workitem
3. Report Progress on Workitem
4. Change Workitem State
5. Retrieve Workitem
6. Get Notified initialize

“The modality should inform the Information System as soon as possible that the performance of the Procedure Step has been started by sending the N-CREATE Service Request. ... Some of the Attribute Values are already known at the beginning of the Procedure Step, they are required to be sent in the N-CREATE command.”
Note in PS3.4, F.7.2.1

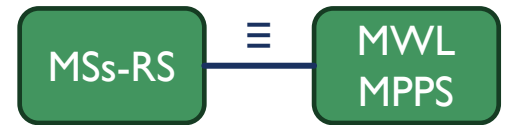
OVERVIEW – PRIMITIVES OF DICOMWEB MODALITY SERVICES (MAPPED)

- | | | | |
|----|---|---|--|
| 1. | Get Applicable Workitems _{MSs-RS} | ≡ | Query _{MWL} |
| 2. | Claim and Prepare Workitem _{MSs-RS} | ≡ | Create _{MPPS} |
| 3. | Report Progress on Workitem _{MSs-RS} | ≡ | Set Information _{MPPS} |
| 4. | Change Workitem State _{MSs-RS} | ≡ | Set Information _{MPPS} |
| 5. | Retrieve Workitem _{MSs-RS} | ≡ | Get Information _{MPPS} |
| 6. | Get Notified _{MSs-RS} | ≡ | Receive Event Notification _{MPPS} |



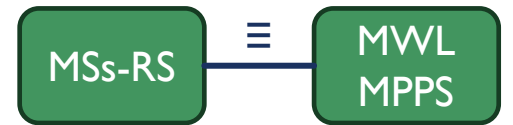
OVERVIEW – PRIMITIVES OF DICOMWEB MODALITY SERVICES (MAPPED)

1. Get Applicable Workitems_{MSs-RS} ≡ C-FIND_{MWL}
2. Claim and Prepare Workitem_{MSs-RS} ≡ N-CREATE_{MPPS}
3. Report Progress on Workitem_{MSs-RS} ≡ N-SET_{MPPS}
4. Change Workitem State_{MSs-RS} ≡ N-SET_{MPPS}
5. Retrieve Workitem_{MSs-RS} ≡ N-GET_{MPPS}
6. Get Notified_{MSs-RS} ≡ N-EVENT-REPORT_{MPPS}



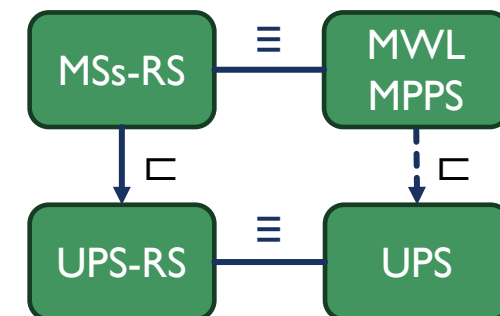
OVERVIEW – PRIMITIVES OF DICOMWEB MODALITY SERVICES (MAPPED)

1. Get Applicable Workitems_{MSs-RS} ≡ Query_{MWL}
2. Claim and Prepare Workitem_{MSs-RS} ≡ Create_{MPPS}
3. Report Progress on Workitem_{MSs-RS} ≡ Set Information_{MPPS}
4. Change Workitem State_{MSs-RS} ≡ Set Information_{MPPS}
5. Retrieve Workitem_{MSs-RS} ≡ Get Information_{MPPS}
6. Get Notified_{MSs-RS} ≡ Receive Event Notification_{MPPS}



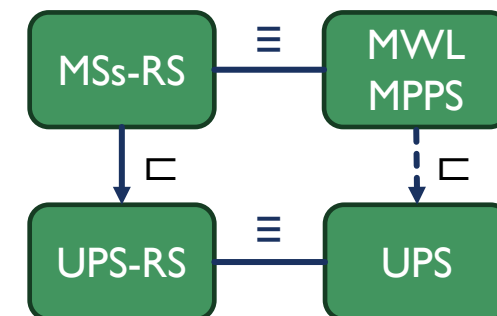
OVERVIEW – PRIMITIVES OF DICOMWEB MODALITY SERVICES (MAPPED)

- | | |
|---|---|
| <ol style="list-style-type: none"> 1. Get Applicable Workitems_{MSs-RS} <ul style="list-style-type: none"> ☐ Search_{UPS-RS} 2. Claim and Prepare Workitem_{MSs-RS} <ul style="list-style-type: none"> ☐ Change Workitem State_{UPS-RS}
[& Update Workitem_{UPS-RS}] 3. Report Progress on Workitem_{MSs-RS} <ul style="list-style-type: none"> ☐ Update Workitem_{UPS-RS} 4. Change Workitem State_{MSs-RS} <ul style="list-style-type: none"> ☐ Change Workitem State_{UPS-RS} 5. Retrieve Workitem_{MSs-RS} <ul style="list-style-type: none"> ☐ Retrieve Workitem_{UPS-RS} 6. Get Notified_{MSs-RS} <ul style="list-style-type: none"> ☐ <...>_{UPS-RS} | <ul style="list-style-type: none"> ≡ Query_{MWL} ≡ Search_{UPS} ≡ Create_{MPPS} ≡ Change State_{UPS} ≡ Set Information_{UPS} ≡ Set Information_{MPPS} ≡ Set Information_{UPS} ≡ Set Information_{MPPS} ≡ Change State_{UPS} ≡ Get Information_{MPPS} ≡ Get Information_{UPS} ≡ Receive Event Notification_{MPPS} ≡ Report a Change_{UPS} |
|---|---|



OVERVIEW – PRIMITIVES OF DICOMWEB MODALITY SERVICES (MAPPED)

- | | | |
|----|--|----------------------------------|
| 1. | Get Applicable Workitems _{MSs-RS} | ≡ C-FIND _{MWL} |
| | ☐ GET /workitems?{match}...UPS-RS | ≡ C-FIND _{UPS} |
| 2. | Claim and Prepare Workitem _{MSs-RS} | ≡ N-CREATE _{MPPS} |
| | ☐ PUT /workitems/{workitem}/stateUPS-RS | ≡ N-ACTION _{UPS} |
| | [& POST /workitems/{workitem}?...UPS-RS] | ≡ N-SET _{UPS} |
| 3. | Report Progress on Workitem _{MSs-RS} | ≡ N-SET _{MPPS} |
| | ☐ POST /workitems/{workitem}?...UPS-RS | ≡ N-SET _{UPS} |
| 4. | Change Workitem State _{MSs-RS} | ≡ N-SET _{MPPS} |
| | ☐ PUT /workitems/{workitem}/stateUPS-RS | ≡ N-ACTION _{UPS} |
| 5. | Retrieve Workitem _{MSs-RS} | ≡ N-GET _{MPPS} |
| | ☐ GET /workitems/{workitem}UPS-RS | ≡ N-GET _{UPS} |
| 6. | Get Notified _{MSs-RS} | ≡ N-EVENT-REPORT _{MPPS} |
| | ☐ <websocket call>UPS-RS | ≡ N-EVENT-REPORT _{UPS} |



DIMSE NOTIFICATIONS VS DICOMWEB NOTIFICATIONS

For modality services, SCPs can send update messages

- PS3.4, F.9.2 Notifications [of MPPS]

“The Application Entity that claims conformance as an SCU to this SOP Class shall be permitted to receive the **following notification**. The Application Entity that claims conformance as an SCP to this SOP Class shall be capable of providing the notifications defined in Table F.9.2-1.” Q: Can notification indeed be unsolicited/unsubscribed?

- PS3.4, CC.2.4 Report a Change in UPS Status (N-EVENT-REPORT)

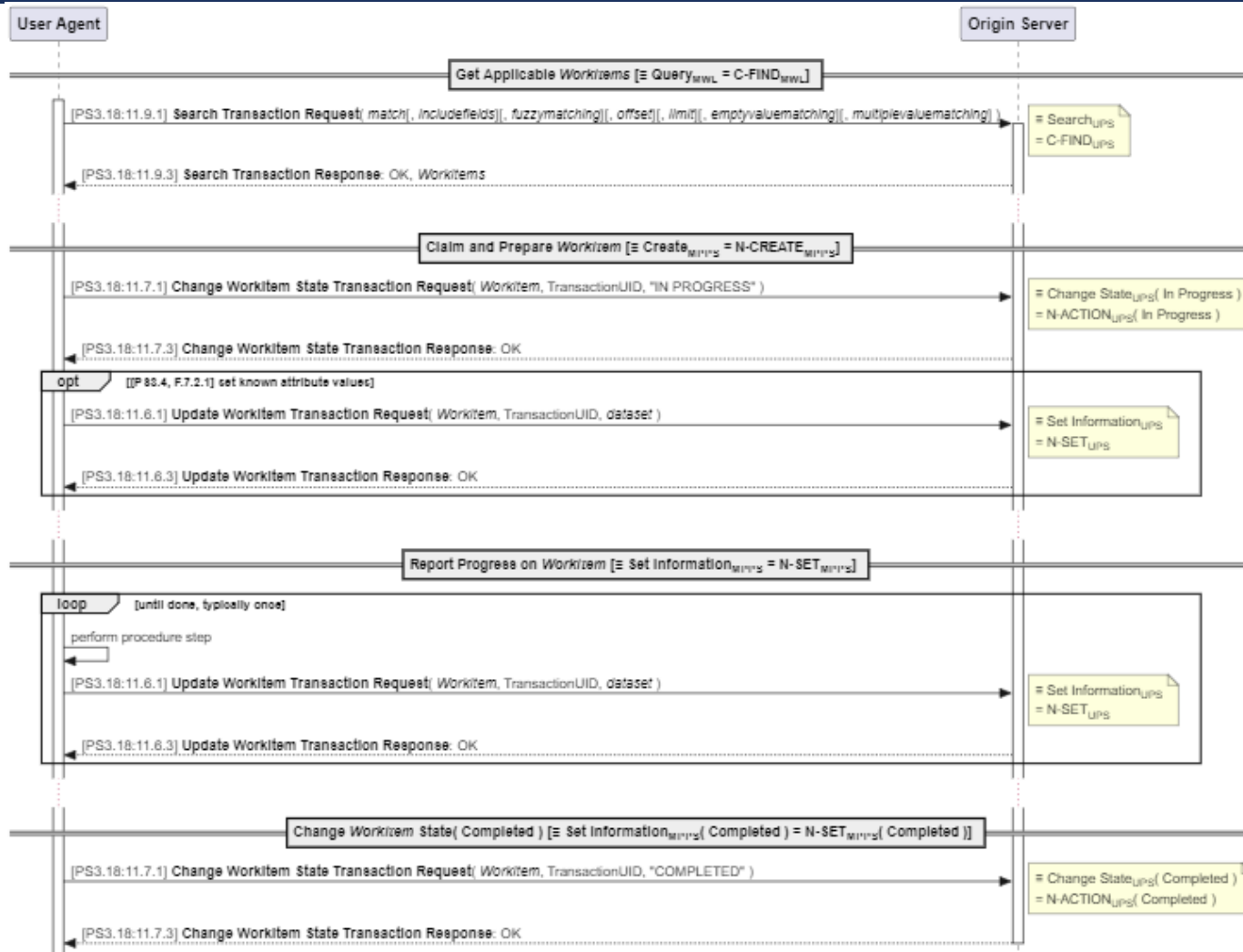
“This operation allows an SCP to notify an SCU of a change in state of a UPS instance or a change in state of the SCP itself. This operation shall be invoked by the SCP through the DIMSE N-EVENT-REPORT Service.”

PS3.18, 8.10: “Some RESTful services support Notifications.” Applicable transactions for MSs-RS:

- `SubscribeUPS-RS` `POST /workitems/{resource}/...` ≡ `Subscribe to Receive UPS Event ReportsMPPS` `N-ACTIONMPPS`
- `UnsubscribeUPS-RS` `DELETE {/resource}` ≡ `Unsubscribe from Receiving UPS Event ReportsMPPS` `N-ACTIONMPPS`
- `SuspendUPS-RS` `POST {/resource}` ≡ `Suspend Global SubscriptionMPPS` `N-ACTIONMPPS`

Q: Resuming indeed not possible?

DICOMWEB MODALITY SERVICES – HAPPY FLOW

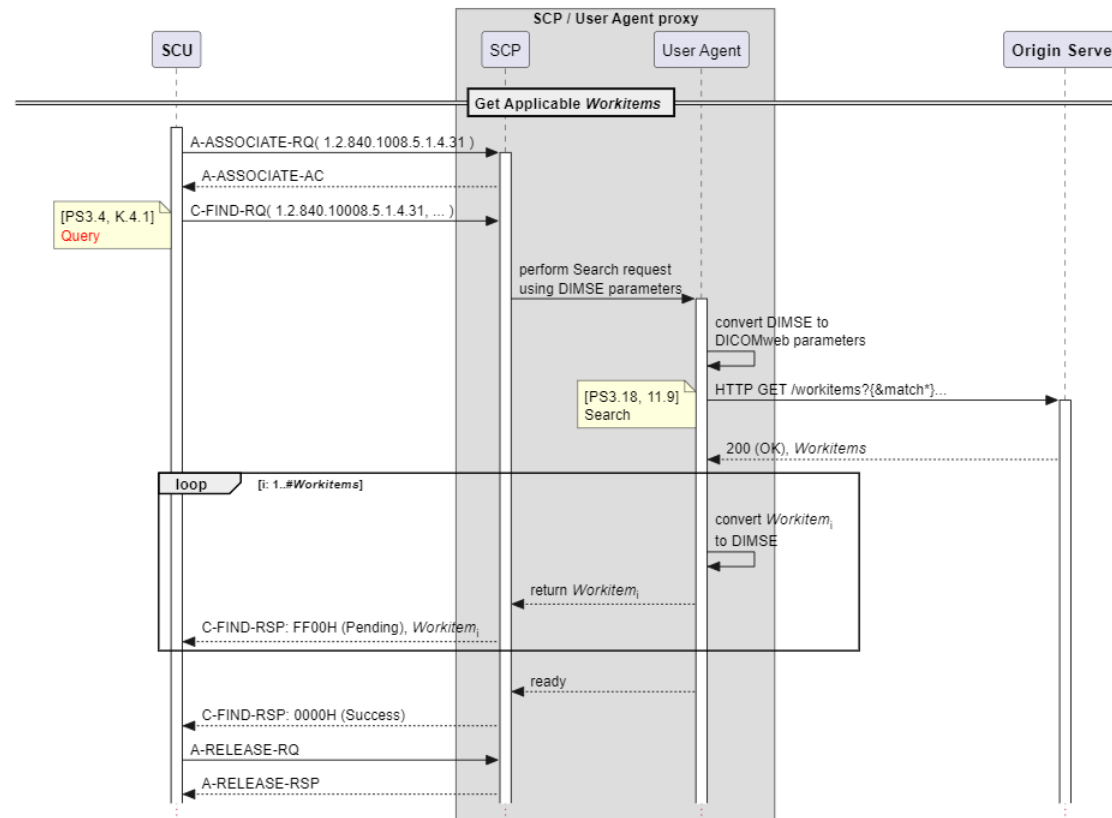


DICOMWEB PROXIES WITH DIMSE

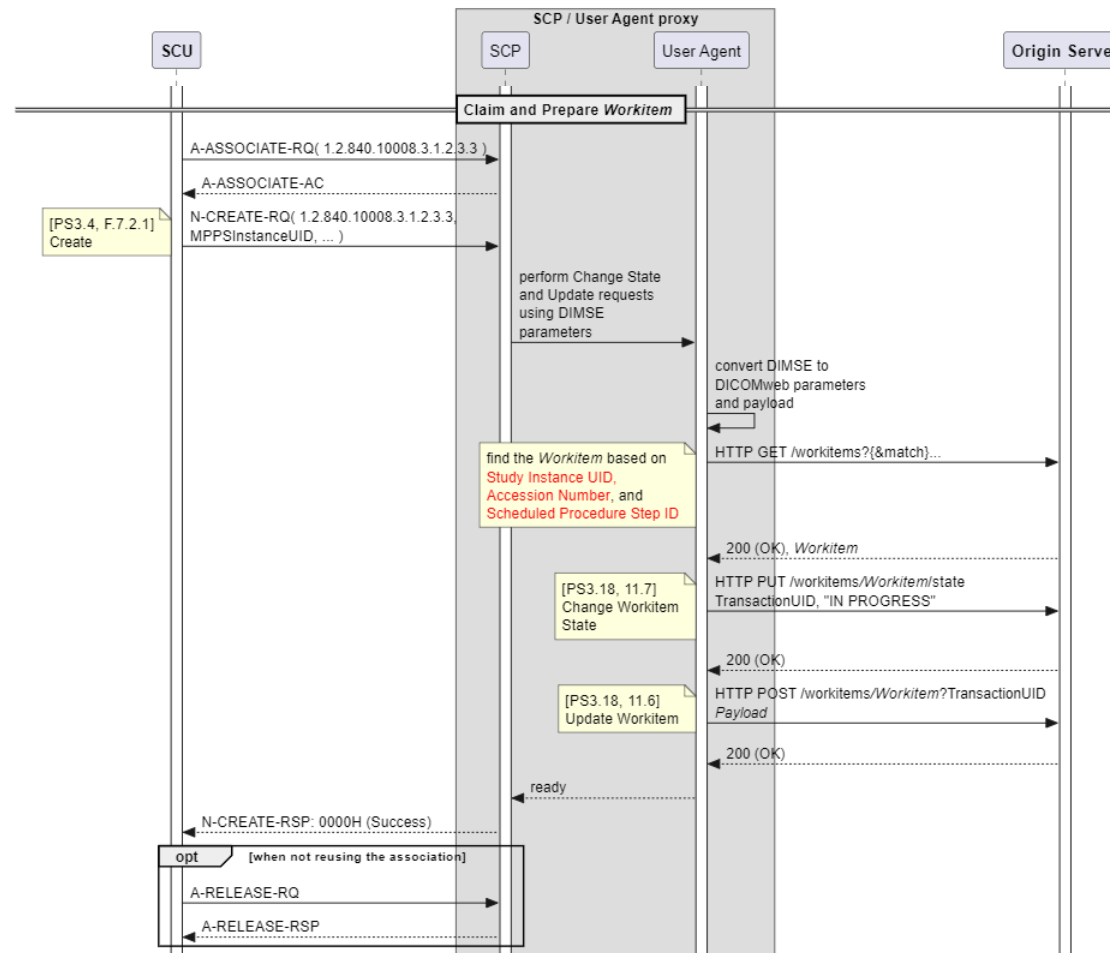
Observation

- In DIMSE, the modality must copy information from the MWL to the MPPS. This is no longer needed when using UPS-RS, as (some of) the applicable information *is* already in the workitem.
 - So, when a DIMSE Modality wants to use an Origin Server for Modality Services, the job of the proxy is relatively easy, namely it can ignore some of the provided data.
 - The other way around, when a DICOMweb Modality wants to connect to an SCP for MWL and/or MPPS, the proxy must provide information, as the Modality will not provide it.

SCP / USER AGENT PROXY – MWL WITH UPS-RS

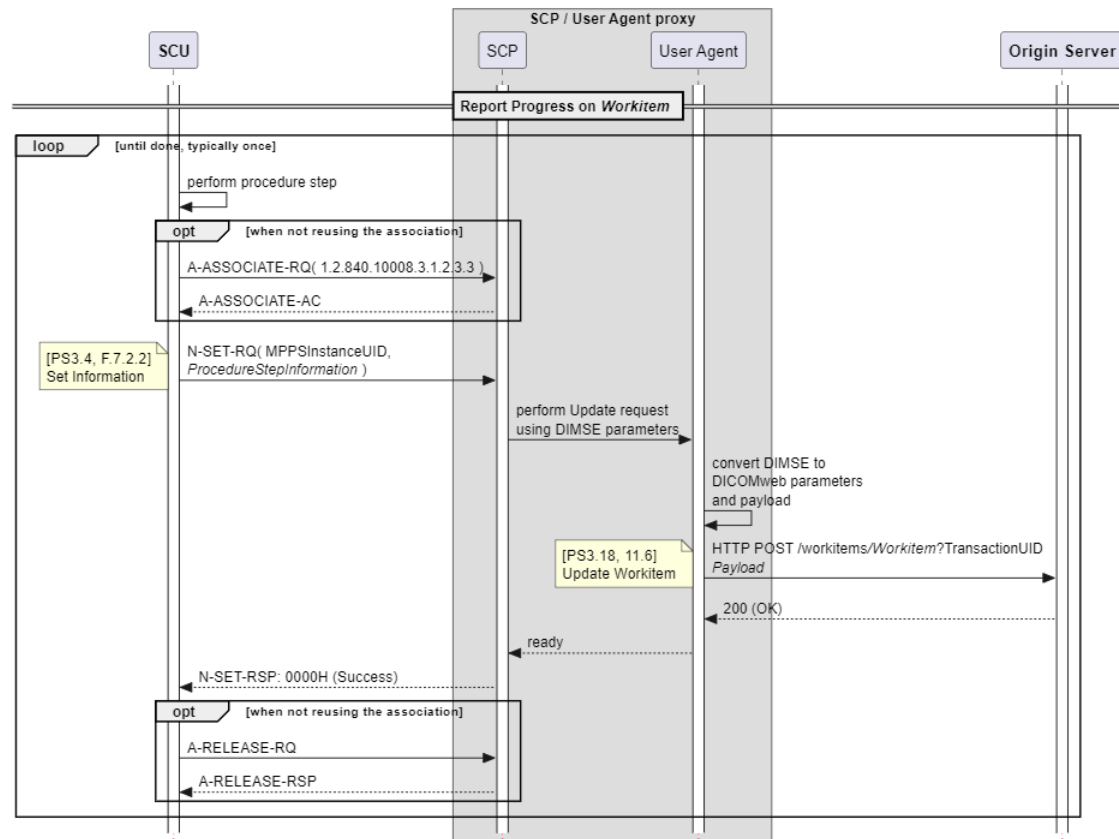


SCP / USER AGENT PROXY – MPPS INITIALIZATION WITH UPS-RS



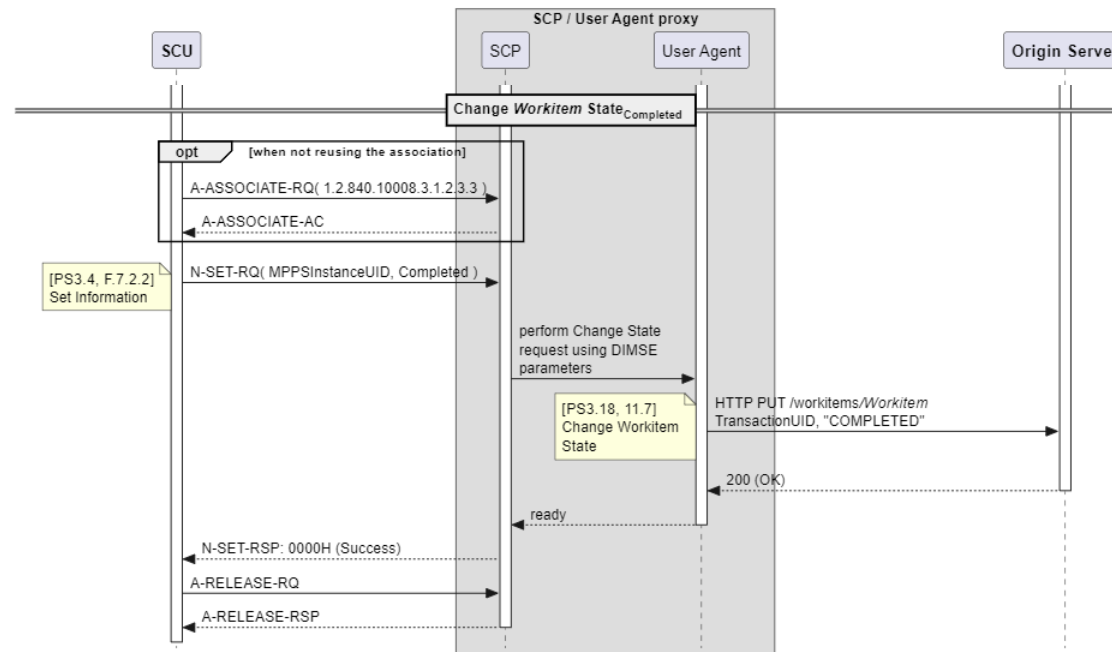


SCP / USER AGENT PROXY – MPPS PROGRESS WITH UPS-RS





SCP / USER AGENT PROXY – MPPS FINALIZATION WITH UPS-RS



NEXT STEPS

- Get a supplement number
- Sketch more scenarios (e.g. reviewing)
- Continue the elaboration of proxies
- Elaborate the 'convert' comments to set clear mappings to datasets / payloads