HL7 DME Orders Ballot Response

Vassil Peytchev, Epic

The workflow page at http://hl7.org/fhir/us/dme-orders/2020MAY/Workflow.html offers a curious mixture of different workflow communicaiton patterns. It is not clear why this choice was made, as it seems unnecessary, based on one of the core assumptions of the Implementation Guide:

> "There is not an expectation that the data to support medical necessity is complete. The PAO Recipient/Intermediary can optionally fetch additional information from the patient's medical record using FHIR RESTful searches or using the mechanisms defined in the CDex IG."

Choose a workflow communication pattern

It is generally expected that both sides of an exchange for a specific workflow will use the same method of data sharing, e.g. if the order is conveyed via a FHIR message, the response or updates are also conveyed via a message.

The existence of an intermediary should not impose different requirements for the initiator or recipient of the order. The rest of this document provides suggestions on using a RESTful communication pattern (with and without an intermediary), using a FHIR Messaging pattern (with and without an intermediary), and even a theoretical mix of the two.

Each pattern has a set of assumptions that are not necessarily specific to DME orders, and if they need to change based on DME orders requirements, then implementation details will likely also change. In particular, it is not clear what the responsibilities of the Intermediary are supposed to be, so if, for example, they are responsible for gathering the medical necessity information, the patterns may need to change to accommodate that.

FHIR RESTful communication pattern (one possible approach)

Assumptions (without an intermediary)

1. The Supplier has restful query capabilities
2. The Supplier can manage subscriptions on a specific Task resource.
3. The Supplier can authenticate to and is authorized to query the EHR

Flow Diagram (without an intermediary)

The flow diagram has three parts. The first part is the creation of the order.

The second part represents order updates by the supplier, which covers any changes to the status of the order, up to and including the fulfilment of the order.

The third part shows order updates by the EHR (e.g. order change or cancel).

DME Ordering
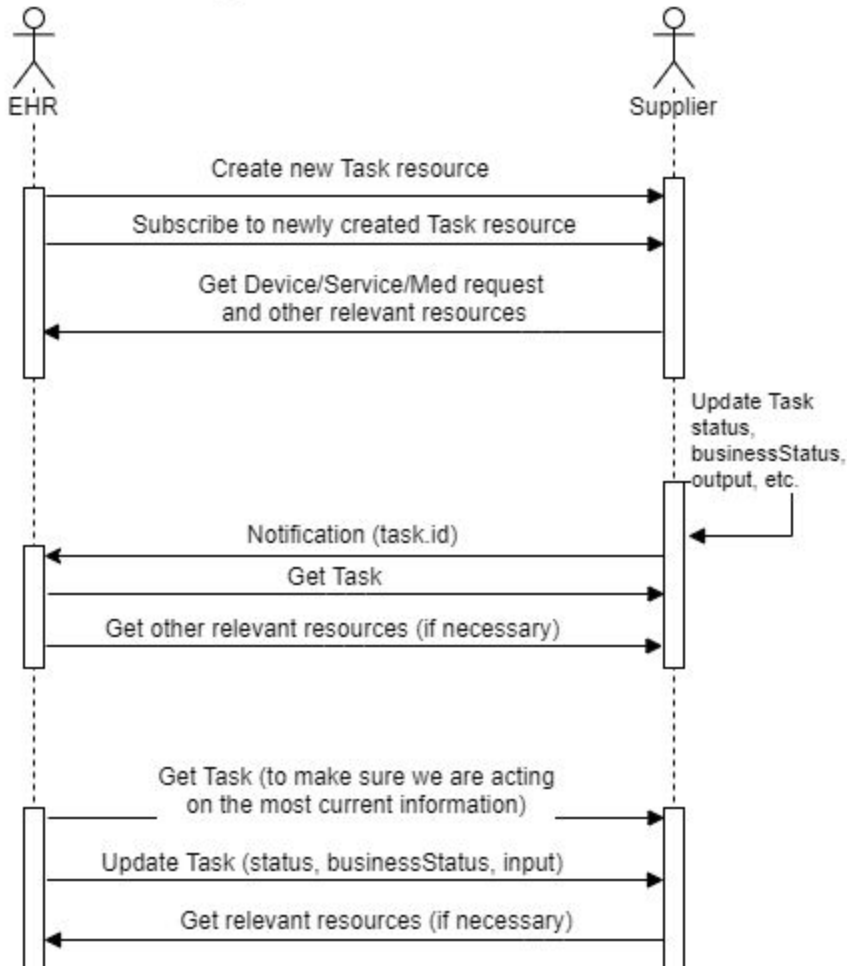
Assumptions (with an intermediary)

1. The Intermediary is the owner and "home" of all Task resources for this workflow.
2. The Intermediary can manage subscriptions on a specific Task resource, and generally on Task resources.
3. The Supplier has restful query capabilities
4. The Supplier has a longstanding subscription for Task resources where they are the Task.performer.
5. The Intermediary has established authorization and authentication relationships with both providers (EHRs) and Suppliers.

6. The intermediary will either use URL-rewriting for all references and links pointing to the EHR or Supplier, or it creates copies of the resources local to the Intermediary to enable RESTful retrieval without direct EHR/Supplier interaction

Flow Diagram (with an intermediary)

The flow diagram has three parts. The first is creation of the order. Note that in this case it is probably useful to have an indication that the Supplier actually is aware of the order, so this extra step is added in the first part.

The second step represents order updates by the supplier, which covers any changes to the status of the order, up to and including the fulfilment of the order.

The third part shows order updates by the EHR (e.g. order change or cancel).

DME Ordering With Intermediary
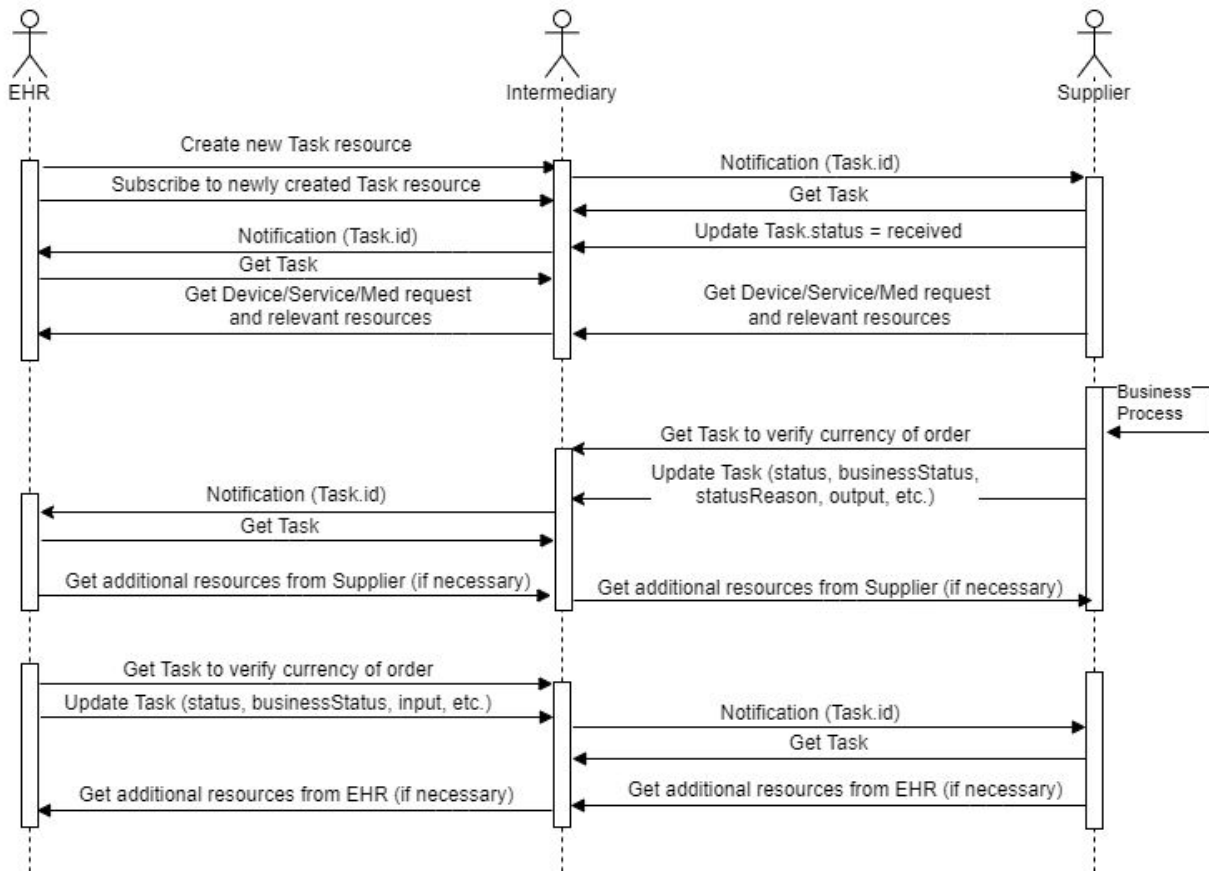
FHIR Messaging communication pattern (one possible approach)

Assumptions (without an intermediary)

1. There is no RESTful access to the information in the EHR, or, all the information needs to be present in a single unit.
2. Any additional information needed by the supplier may be available via RESTful queries from the EHR.
3. The MessageDefinition resource for each message is available as a static artefact to be used as a reference, but it is not present in the message itself.
4. The structure of each message is as follows:
   a. A Bundle where the MessageHeader Resource is the first entry.
   b. MessageHeader.focus references the Task resource in the message Bundle
   c. The resources of all other entries in the Bundle are reachable from the MessageHeader resource (through Task or directly), and any references to resources that are not in the Bundle are logical references (via identifiers, not urls).

Flow Diagram (without an intermediary)

The flow diagram again has three parts. Each message is shown with a synchronous response, that comes from the $process-message endpoint of the receiver.

Note that any messages from the Supplier to the EHR could be presented as an asynchronous message response (similar to the application-level acknowledgements in HL7 v2 messages) using the MessageHeader.response structure, but given that the implementation guide explicitly calls out the use of intermediaries, keeping track of the correct MessageHeader.response.identifier value would be an additional requirement for them that is not necessary - the use of Task allows responses to be linked with the original order without MessageHeader.response.

The first part shows the sending of the order. A successful synchronous response indicates that the order has been received.

The second part represents order updates by the supplier, which covers any changes to the status of the order, up to and including the fulfilment of the order.

The third part shows order updates by the EHR (e.g. order change or cancel).

DME Orders (messaging)
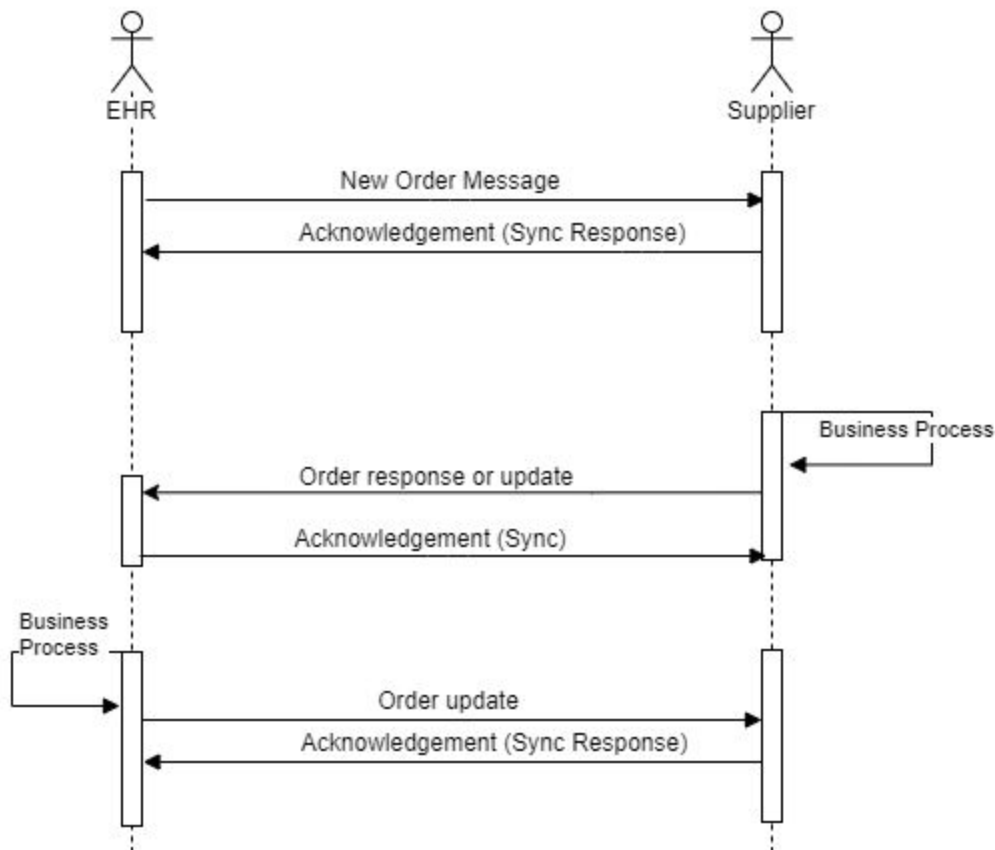
Assumptions (with an intermediary)

1. There is no RESTful access to the information in the EHR, or, all the information needs to be present in a single unit.
2. Any additional information needed by the supplier needs to be exchanged via order update messages, since one of the main purposes of the intermediary is to proxy the access between the EHR and the supplier.
3. MessageHeader.response is not used for the order update messages.
4. The MessageDefinition resource for each message is available as a static artefact to be used as a reference, but it is not present in the message itself.
5. The structure of each message is as follows:
   a. A Bundle where the MessageHeader Resource is the first entry.
   b. MessageHeader.focus references the Task resource in the message Bundle
   c. The resources of all other entries in the Bundle are reachable from the MessageHeader resource (through Task or directly), and any references to resources that are not in the Bundle are logical references (via identifiers, not urls).

Flow Diagram (with an intermediary)

The flow diagram again has three parts. Each message is shown with a synchronous response, that comes from the $process-message endpoint of the receiver.
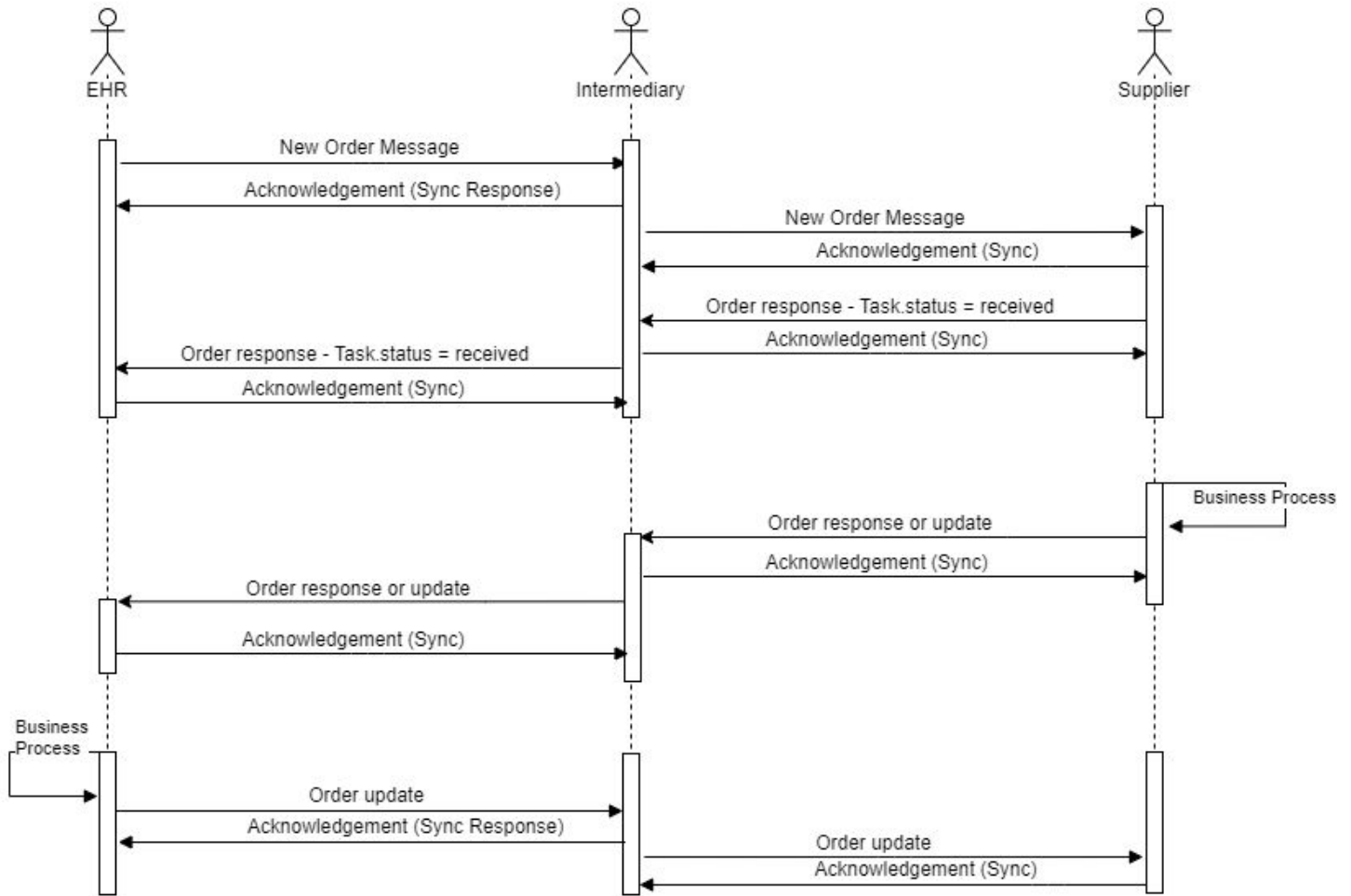
Note that any messages from the Supplier to the EHR could be presented as an asynchronous message response (similar to the application-level acknowledgements in HL7 v2 messages) using the MessageHeader.response structure, but given that the implementation guide explicitly calls out the use of intermediaries, keeping track of the correct MessageHeader.response.identifier value would be an additional requirement for them that is not necessary - the use of Task allows responses to be linked with the original order without MessageHeader.response.

The first part shows the sending of the order. In this case, the diagram shows explicit order status update to "received" as an additional message going from the Supplier to the EHR, in order to ensure similar functionality to the one present in the case without an intermediary.

The second part represents order updates by the supplier, which covers any changes to the status of the order, up to and including the fulfilment of the order.

The third part shows order updates by the EHR (e.g. order change or cancel).

DME Orders (messaging) with Intermediary

EHR   Intermediary   Supplier

- New Order Message (EHR → Intermediary)
- Acknowledgement (Sync Response) (Intermediary → EHR)
- New Order Message (Intermediary → Supplier)
- Acknowledgement (Sync) (Supplier → Intermediary)
- Order response - Task.status = received (Supplier → Intermediary)
- Acknowledgement (Sync) (Intermediary → Supplier)
- Order response - Task.status = received (Intermediary → EHR)
- Acknowledgement (Sync) (EHR → Intermediary)

Business Process (Supplier)

- Order response or update (Supplier → Intermediary)
- Acknowledgement (Sync) (Intermediary → Supplier)
- Order response or update (Intermediary → EHR)
- Acknowledgement (Sync) (EHR → Intermediary)

Business Process (EHR)

- Order update (EHR → Intermediary)
- Acknowledgement (Sync Response) (Intermediary → EHR)
- Order update (Intermediary → Supplier)
- Acknowledgement (Sync) (Supplier → Intermediary)

Mixed communication pattern (one possible approach)

One of the purposes of an intermediary can be to transition between different communication patterns, although this assumes a detailed understanding of the content, and requires the intermediary to manage resources locally. This example shows using the RESTful pattern for the EHR, and messaging with the Supplier.

The first part of creating the order includes the intermediary getting all necessary resources from the EHR to package them as a message which is sent to the Supplier. A successful message to the Supplier is then reflected as a status of "received" in the Task resource, and the EHR is notified of the update.

The second part, updates/responses to the order from the Supplier, starts with a message sent to the intermediary, where all resources are stored and exposed locally, so the EHR can retrieve them after getting the updated Task.

The third part is for order updates from the EHR to the Supplier, and the flow is similar to the new order part.

DME Ordering mixed mode