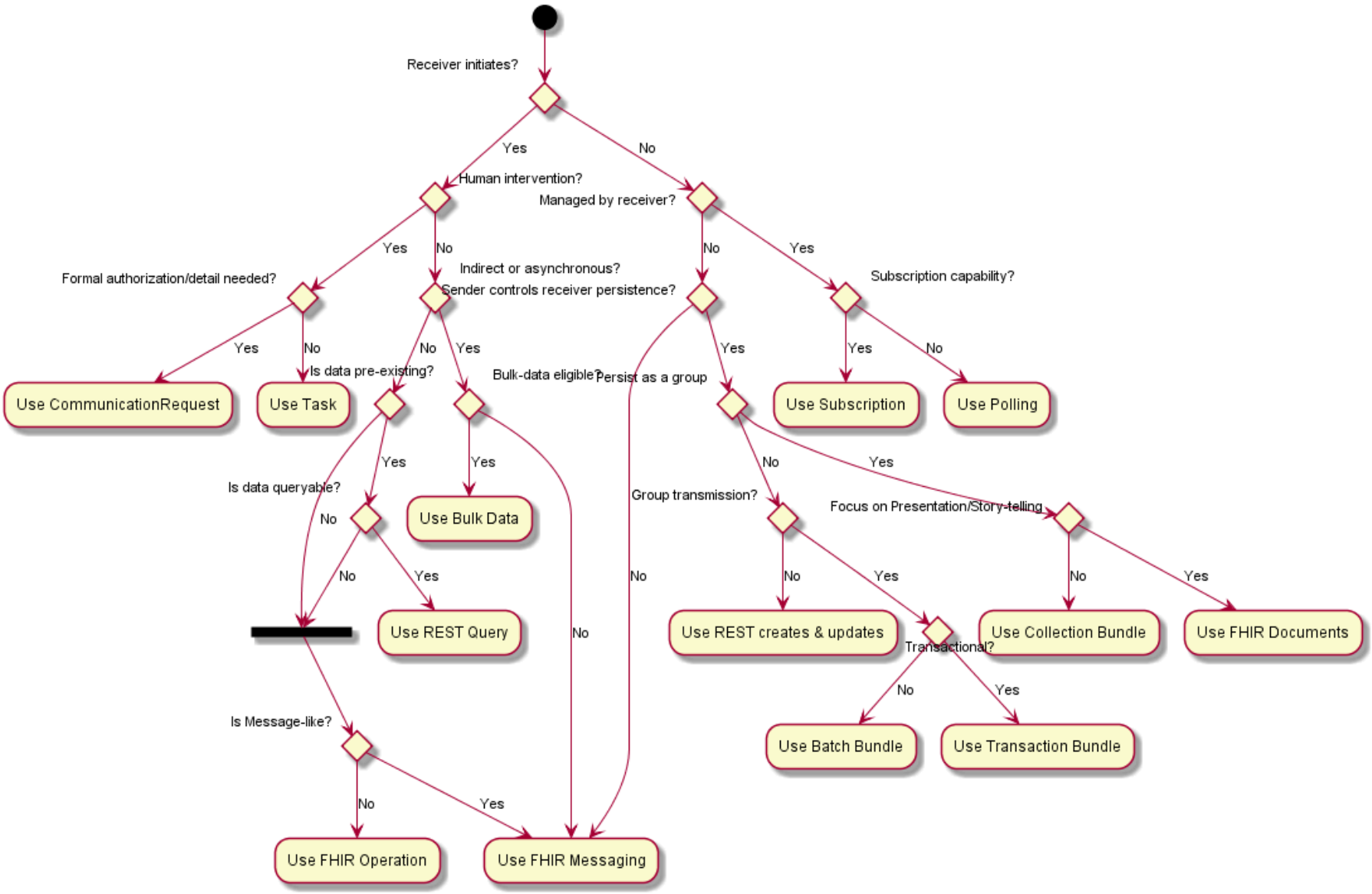
This page is now superseded by content here: https://build.fhir.org/ig/HL7/davinci-ehrx/exchanging

In general, the preferred mechanism for all information exchange is REST because it requires the least negotiation and configuration to enable new use-cases. However, in some cases, REST won't meet the use-case requirements. This flow-chart describes the decision tree in determining the most appropriate exchange mechanism for a given use-case. It is based on the discussion found in the Guidance for data sharing Google document distributed earlier.

The decision tree is explained in more detail in the text below.



- Put loop to top from Task & CommunicationRequest (and reference to Workflow). Change REST Query to REST search.
- Messaging and operations are functionally equivalent. I.e. Both mechanisms can be used to accomplish identical functions. However, there are small differences that may make one more appealing in different architectural circumstances:
    - Messaging provides explicit support for routing (you can identify a target recipient independent of who you initially transmit to)
    - Operations are slightly lighter weight (no need for MessageHeader)

    - Messaging allows multiple operations to be communicated with a single endpoint, so if there is a cost to establishing distinct endpoints, messaging might have an advantage.
    - FHIR Messaging may integrate more easily with back-end systems that are messaging-based
    - Some regulatory environments may mandate an architectural approach
    - It's possible to replicate a messaging approach using a custom operation. This is discouraged as it essentially creates a 'custom' mechanism to do something where there's already a standard operation ($process-message)
    - Asynchronous delivery of operation results requires polling, where as messaging doesn't necessarily
    - There may be standard operations defined (e.g. Bulk Data with $everything), in which case they should be used preferentially to messaging.
- Drop the separate call-out for bulk data
- Make clear that 'messaging' and FHIR documents are discussed in a general (transport-independent manner). Colors?
- Call out the fact that "human managed" might not always involve a human (e.g. AI) but is only used when there's the *potential* need for human intervention.

Detail

1. Who initiates?
   What causes information to flow? Is it a request by the data consumer or an event that occurs within the system of the data producer (e.g. data updated, user indicates need to share?)
   Notes:
   - sender-initiated data flows can include expectation for behavior on the part of the receiver, while receiver-initiated flows generally don't impose any such expectations.
   - Some recipients may be reluctant/unable to initiate data transfer for legal/policy reasons - i.e. it's deemed acceptable to be pushed data, but not to pull it. Typically this stems from right of access/permissions concerns where only the sender is deemed to have authority as to what information should be shared and when

   Receiver

   a. Human intervention by data source?
      Will humans need to be in the loop to either compile/create the data or to authorize/filter the data to be disclosed? This can include the creation of documents.

      Yes

      i. Formal authorization/detail needed?
         Is there a need for a formal authorization (i.e. a signed 'order') requesting the sharing of the data? Is a detailed description of the data to be retrieved (medium, subject, time-range, etc.) needed, or is it sufficient to simply specify a code, reference or attachment that indicates the data to be conveyed?

Yes - CommunicationRequest

Post a CommunicationRequest indicating what information needs to flow - and to whom. Fulfillment of this request can be initiated using any of the mechanisms defined in the FHIR workflow execution pattern pages.

No - Task

Post a Task to somewhere it will be seen by the information sender (could be the information sender's system or a system it monitors by polling or subscription). The Task code would indicate the need to transmit information, the Task input would indicate the information to be transmitted and the Task.for would indicate who is to receive the information

No

ii. Indirect or asynchronous?

Do the sender and receiver have the ability to establish a direct network connection (possibly through a proxy), or is there a need for routing? Is there an expectation that the response will be 'immediate' (milliseconds to a few seconds), or will there be a significant lag (minutes to days) before the request is responded to?

Yes

1. Bulk-data eligible?

   Bulk data gives a mechanism to do some sorts of queries in an asynchronous manner (though it still requires a direct connection). See here for details about what sorts of queries bulk data will support. If bulk data meets the need for asynchronous, it should be used

   Yes - Use bulk data

   No - Use FHIR Messaging

No

2. Is data pre-existing?

   Is the data pre-existing or is there a need to 'create' the relevant record(s) prior to returning them? This might be done in an automated fashion (e.g. calculating an average) or manual fashion (e.g. assembling, scanning and organizing information into a document)

   Yes

   a. Is data queryable?

      This is driven by several factors

      - Is the data of interest something that already exists as a persisted, retrievable record (as opposed to Is the information sought something that must be calculated/generated)
      - Is the set of data reasonably assembled and efficiently retrieved using the query mechanisms available (standard or custom search parameters, _include and _revinclude, _query, graphQL)?
        - Notions of what constitutes 'reasonable' and 'efficient' are subjective and will vary by context.
      - Is the data sender capable of executing a query?
        - Many legacy interfaces (including data access control mechanisms) are not designed to support real-time query.

- Querying data requires an infrastructure whereby data recipients know the endpoint locations to access data and those endpoints are capable of authenticating and determining the authorization of the requesting users/systems
- As a result, additional investment to replace or enhance existing interfaces may be needed to support query

  Lack of capability creates the question whether to invest in the capability or choose an alternate exchange mechanism. This typically comes down to a 'pay now or pay later' scenario. Setting up query mechanisms is more costly than establishing a single point-to-point interface. However, once built, query mechanisms can be used for multiple purposes, while other mechanisms tend to be limited to a single use-case - meaning substantial incremental costs for each new use-case.

Yes

Use REST Query

No

Continue as if 'No' for pre-existing

No

b. Message-like?

Does the data passed back consist purely of a Bundle of inter-related resources with no organization beyond their relationships and a single starting point or collection of starting points or is it organized into a set of named elements, some of which may have individual data type values rather than being full resources?

Simple Bundle

Use messages

Complex structure/individual values

Use an operation

Sender

b. Managed by receiver?

Can the receiver configure/manage what information is to be transmitted by the sender without humans on the sender side needing to take any action (e.g. changing configuration files, writing code)?

Yes

i. Subscription applicable?

Does the sender support a Subscription endpoint and can (and will) they expose a topic that covers the information needed by the receiver? Finally, is the security model for data access appropriate with a subscription?

- Subscriber might be eligible at subscription time, but not at the time of the event
- Recipient might not be same as individual receiving subscription
- Security model might be optimized differently for subscriptions than queries (and thus not appropriate for all types of requests)

- **Yes**

  Subscription

- **No**

  Polling by query at regular intervals

No

ii. Sender chooses record operation behavior?

Does the receiver expose RESTful endpoints for relevant resources and does the sender make specific determinations about what resources will be created, updated and/or deleted on the receiver's system?

Yes

1. Persist as a group

   Is the intention that the shared data be stored as a frozen 'collection' or is each resource to be managed independently

   Yes

   a. Focus on Presentation/Story-telling?

      Is the intention to have tight control over how data is displayed (including order of presentation) to ensure consistent understanding of context (i.e. for human to human communication)?

      Yes

      FHIR Document

      No

      Collection Bundle

   No

   b. Group transmission?

      Are multiple resources transmitted at once for efficiency/dependency reasons or is it sufficient to send each independently?

      Yes

      i. Transactional?

         Must all of the data operations transmitted by the sender be processed as a single unit of work?

         Yes

         Transaction Bundle

         No

         Batch Bundle

      No

      Individual REST creates and updates

No

Messaging