Merge Operation

WORK IN PROGRESS

Operation POST **`[base]/Patient/$merge`**

*(this does **not** perform on the actual resource instance e.g. [base]/Patient/01/$merge)*

The target is the remaining patient resource (survivor), the source will be marked as inactive

The source-patient resource will be updated to add a new link reference to the target-patient resource (link-type=replaced-by), and also update the status to inactive

The target-patient resource will be updated to add a new link reference to the source-patient resource (link-type=replaces) - *and must be included in the result-patient parameter if used (if the source patient is deleted, then this link is not required)*

Parameters

Parameter Name	Card	Type	Description
source-patient	0..1	ResourceReference	A direct resource reference to the **source** patient resource (this may include an identifier)
source-patient-identifier	0..*	Identifier	The Identifier(s) of the **source** patient resource; all of these identifiers MUST be present in the located resource (in addition to the one included in the resource reference above - if included) (The purpose of this property is to ensure that the correct source patient resource is selected)
target-patient	0..1	ResourceReference	A direct resource reference to the **target** patient resource This is the surviving patient resource, the target for the merge
target-patient-identifier	0..*	Identifier	The Identifier(s) of the **target** patient resource; all of these identifiers MUST be present in the located resource (The purpose of this property is to ensure that the correct source patient resource is selected)
result-patient	0..1	Resource (Patient)	The details of the Patient resource that is expected to be updated to complete with and must have the same patient.id and provided identifiers included. This resource MUST have the link property included referencing the source patient resource It will be used to perform an update on the target patient resource The absence of this parameter the servers should copy all identifiers from the source patient into the target patient, and include the link property (as shown in the example below) This is often used when properties from the source patient are desired to be included in the target resource The receiving system may also apply other internal business rules onto the merge which may make the resource different from what is provided here.
preview	0..1	boolean	If this is set to true then the merge will not be actually performed; an OperationOutcome will be returned in the Parameters response that will indicate that no merge has occurred and may include other diagnostic info if desired, such as the scale of the merge. e.g. Issue.details.text "Preview only Patient merge - no issues detected" e.g. Issue.diagnostics "Merge would update: 10 years of content or 120 resources" The resulting target patient resource will also be returned in the result
RETURN	1..1	Parameters	The status of the response will be one of: • 200 OK - If the merge request doesn't expect any issues (although warning may be present) or was completed without issues • 202 Accepted - The merge request has been accepted and does not expect any issues and will continue processing the merge in the background, and you can monitor the Task for completion • 400 Bad Request - There are errors in the input parameters that need to corrected • 422 Unprocessable Entity - Business rules prevent this merge from completing The Parameters resource will include: • The Input parameters to the operation • An OperationOutcome containing errors, warnings, and information messages • The resulting merged Patient resource (or a patient reference if the patient is not committed) • Optionally a Task resource to track any additional processing that was required

There must be at least 1 source patient/patient-identifier parameter and at least 1 target patient/patient-identifier parameter

The result-patient.id must be the same as the target patient reference (if the patient reference is provided as an input parameter)

In the case where a new patient is desired to be merged from the 2 patient resources, then a new patient resource should be created before attempting to merge, and then call the merge operation for each resource into the newly created patient resource.

A server May decide to delete the source record, but this is not defined by the standard merge operation, and if this occurs then the target patient's link property will remain unchanged.

Merge Processing

The merge operation will have multiple stages, and some of these may take additional time for processing and thus be done asynchronously

Stage	Description
Preview Merge	(Optional) This is a call to the operation (with preview=true) that simply checks for potential errors and warnings, without committing any changes. This may not be able to capture all possible causes of errors that could be encountered during the processing of the data patching. The returned Patient resource is a preview only and has not been committed. Hence the version number and last_modified date would be different.
Initiate Merge	This stage processes the input parameters checking for errors/warnings and begins the changes to the patient resources. If the system is able to complete the processing of all reference data to the target patient, then it may be complete and no task is required. Otherwise a task for tracking would be created and monitor the progress of the merge.
Data Processing	The rest operation may have returned, and processing is ongoing to patch any other resource that references the source patient to reference the target patient. This may take a considerable period of time in some systems where the volume of records being updated is large.
Completed (or failed)	All data processing is complete, and the Task is marked as completed (maybe with errors)

During the Data Processing stage the patient resource and resources referencing the source patient may be indeterminate until the merge processing operation completes.

Note: Some servers may also update the inactive source patient resource to remove most of the data to make it more clear that the resource should not be used, and the replaced-by link is the key information. Even to the extent of clearing the name and contact details etc.

Merging Identifiers

If the result patient is provided in the parameters to the operation, then it is assumed that the caller has correctly included all the required identifiers desired to be in the target patient (though must include the identifiers specified in the input parameters).

If the result patient is not provided, then the values provided in the request parameters (source-patient.identifier and all source-patient-identifiers) will be copied into the target resource and marked as old.

The server may also migrate other identifiers (and properties) at its discretion, and choose to mark these as old or not.

Note: If an identifier value is masked, then the server will migrate the identifier value correctly, regardless of the masking.

Updating Data that References the source Patient

The merge data processing SHALL update all references that refer to the source patient to reference the target patient.

While updating resources that reference the source patient, ensure that the target patient link value isn't also accidentally updated. (don't make it point at itself)

A Provenance resource SHOULD be created that references the source and target patient resources (or just the target-resource if the source was deleted, or the source patient's old identifier) indicating that the merge has occurred. (example below)

A Provenance resource MAY be created to link all of the resources that referenced the source patient that could then provide information to a potential un-merge operation.

Some resources that have been updated as a result of the merge, such as AuditEvent, Provenance, may have been digitally signed, and this change would invalidate the signature. There may be other reasons impacting the updates that should be considered, further feedback on this specific use case is required.

Post Merge Expectations

Once the patient resources have been merged:

A GET on the old Patient resource ID return: e.g. GET [base]/Patient/pat01

- 200 OK and returns the old Patient which is now marked as inactive, and has the link (replaced-by) populated with the new Patient ID
- 404 not found (when the merge system deleted the resource)

Should as SEARCH by the old Patient Resource ID return: e.g. GET [base]/Patient?_id=pat01 (often used as a substitute for direct GET when doing _include for the managing org/general practitioner)

- 200 Ok Bundle with no patient resource (case where the old patient was deleted)
- 200 Ok Bundle with the inactive patient which is marked as inactive and has the link (replaced-by) populated in it (that you'll need to follow to get any further data)
- 200 Ok Bundle with the target patient resource (with the link to the one from the search) and not include the old patient resource
- 200 Ok Bundle with both the target and old patient resources

How would interactions on $everything or the patient compartment on the old patient resource be effected by the merge? It could redirect itself?

How will this effect the $match operation? should it also return the inactive resources? If that operation definition does not return inactive resources, then no problem here.

Impact on Subscriptions

There is a whole discussion on the impact on subscriptions, so won't try and replicate all of that, but instead summarize the issues

- What can be used as triggers for the subscription
  - Patient update with new link values
  - operation itself as an event
  - Provenance(s) as an event
  - The Task doing the Merge operation
- Will all the data that is patched over to the target patient ID be notified

Notification Mechanisms

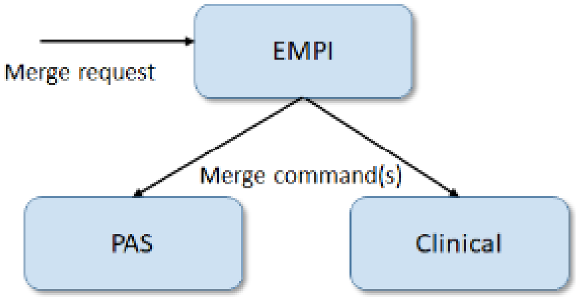
How does the EMPI (or other receiving system) notify other dependent systems that the merge has occurred/completed?

This could be through several ways. It could be through:

- An integration engine sending HL7v2 A40 merge message *(A18 may also be applicable in backward compatibility modes)*
- Directly calling the $merge operation on the dependent systems
- Using FHIR Messaging to invoke the same operation
- Using Subscriptions to detect the merge operation has occurred (active discussions in subscriptions on this)

Other Notes

The context of the merge operation could be either within an EHR (or similar system), or within an EMPI service.



The system that receives the initial $merge operation and processes this and returns the result.

Other downstream systems may be updated at a later time. This update request could also use the same $merge operation on the downstream system, or could use other integration engine approaches, such as sending v2 messages. The downstream merge would know which 2 records were in the downstream system, and thus what data should be updated.

Consideration should be taken to ensure that the correct data is acted on.

The downstream systems may not have all identifiers that the EMPI has, the EMPI may be configured to know what "types" of identifiers should be propagated to which systems.

When using the identifier parameters (rather than id) you should be using the same assigner (which in the example above would be the PAS or clinical system)

Merging different records in downstream systems, joining records in the EMPI

In the case where there aren't 2 records in the clinical system, then a simple update could be used.

Further consideration on the identifier assigning authority needs to be done.

In HL7 v2, the trigger message only reports the results of the operation, and has PID and merged from PID

Question: Should support for a simple query parameter form of the operation be permitted? (using the form used in search url|value for identifiers, and the resource references will just include the reference value (no identifiers)

Cooper Thompson 9/15/2019 3:51 PM - this wouldn't be idempotent, so I would think this really should be a POST, even if it *could* be a GET.

Question: How should privacy related meta tags be handled? When reconciling 2 patient resources to be merged, any security or privacy related tags should be included in the resulting target resource with the most sensitive values surviving.

Question: Should a **Task** resource be included to track things that might take longer to process? Yes.

Question: Should we be explicit about how the identifiers from the source resource should be copied into the target resource (and if they should be marked with use=old), the system will not be expected to make these decisions in the operation, the caller of the operation will make those decisions, which are defined in the result-patient resource. For a case where this was an old MRN, we recommend that these also remain in the target resource so that if content comes in (such as a lab result) with the old MRN on it, then it can be attached to the appropriate.

Question: What should happen if the 2 patient resources are already merged, an error code?

Question: Should the merge operation update/correct/change all the resources that reference the patient to link to the resulting patient resource. E.g. Any Observations referencing Patient/01 will be changed to Patient/02

Question: Provenance/AuditEvent actions - any security issues here?

After some discussion, the Provenance is the preferred resource to track this content, as will likely remain with the clinical data retention, however the AuditEvents may be cleansed periodically.

Question: Will the spec also describe an $unmerge style operation, or will that be essentially a manual activity on the API using the Provenance /AuditEvents to determine context.

Question: Should the updating process be able to be performed synchronously, or async? - Prefer to support Async given the potential duration the processing could take for some long term records

Note: The preferAsync header is intended for large data results, not long data processing results.

Question: Should there be a mode parameter to get an estimate on what would be impacted by the merge - an aggregate on how many resources on each side etc.

Question: Should we be able to merge 3 patients into 1? Suggest keep it simple at 1 into 1 for now.

Question: During the processing of data for a merge operation (which can take take some time), any query for a resource that references either source or target patient could be returning partial content, and operation outcome could be returned in the search bundle indicating this - it's really a warning or information case. Resources like medications or allergies could be significant in this situation.

Question: What should happen to the source patient post merge? Should it be deleted, marked as inactive, hidden from searches? What impact on subscriptions and lists?

Question: Should there be a pre-merge or dry run merge operation to check the scale of the operation, this could be useful to check the scale. If the results were to be reported, it can be an estimate, and doesn't really need to be broken down into resource types, its just to help advise age of the record. This **preview** parameter satisfies this need.

Question: Should a search no the old MRN identifier return both the old patient and target patient resources, or just the target patient resource? How do we filter out the old resource? Rely on the inactive flag?

Question: Should the merge itself mark the source as inactive? Suggest that this is a good idea.

Question: Should other resources, such as Patient Accounts, be merged too? Or is that a supplementary exercise? Could they be tagged with Task resources for review?

Other considerations could be claims to be withdrawn, communications in process to be cancelled.

Question: Should anything happen with SMART App Launch tokens that have the Patient ID in them? If they were revoked, that would be the cleanest, as apps need to handle this and would just challenge for a new token, which would get the new value

Question: If the 2 patient resources had links to each other, should those be removed?

Question: How would a merge be reflected in subscriptions?

Question: Should the merge operation also be reflected in the Person Resource - expect that would just update as per any other resource type with a reference to the patient resource.

Question: How can we merge multiple patients into 1 record? This is typically done using multiple calls to the operation. e.g. merge: AC, merge: BC

Question: How do Patient (or other users) accounts with access be changed (these aren't fhir resources)

Question: What happens in a Rollback scenario? When can that occur?

Question: Should searches on other resources include an OperationOutcome in the bundle that indicates that there is a processing action in progress that could effect the results.

Example $merge operation parameters resource

```
POST: [base]/Patient/$merge
<Parameters xmlns="http://hl7.org/fhir">
  <parameter>
    <name value="source-patient" />
    <valueReference>
      <reference value="Patient/01" />
      <identifier>
        <use value="official" />
        <type>
          <coding>
            <system value="http://www.hospital-a/localid" />
            <code value="HospitalA" />
            <display value="Hospital A" />
          </coding>
          <text value="Hospital A" />
        </type>
        <system value="urn:oid:2.16.840.1.113883.3.72.5.9.1" />
        <value value="1000000001" />
        <assigner>
          <display value="Hospital A" />
        </assigner>
      </identifier>
    </valueReference>
  </parameter>
  <parameter>
    <name value="patient" />
    <valueReference>
      <reference value="Patient/02" />
      <identifier>
        <use value="official" />
        <type>
          <coding>
            <system value="http://www.hospital-a/localid" />
            <code value="HospitalA" />
            <display value="Hospital A" />
          </coding>
          <text value="Hospital A" />
        </type>
        <system value="urn:oid:2.16.840.1.113883.3.72.5.9.1" />
        <value value="1000000002" />
        <assigner>
          <display value="Hospital A" />
        </assigner>
      </identifier>
    </valueReference>
  </parameter>
  <parameter>
    <name value="patient-identifier" />
    <valueIdentifier>
      <system value="http://example.org/SSN" />
      <value value="804234513" />
    </valueIdentifier>
  </parameter>
  <parameter>
    <name value="result-patient" />
    <resource>
      <Patient xmlns="http://hl7.org/fhir">
        <id value="02" />
        <identifier>
          <use value="official" />
          <type>
```

```
            <coding>
              <system value="http://www.hospital-a/localid" />
              <code value="MRN" />
            </coding>
            <text value="Hospital A MRN" />
          </type>
          <system value="http://www.hospital-a/localid" />
          <value value="1000000002" />
          <assigner>
            <display value="Hospital A" />
          </assigner>
        </identifier>
        <identifier>
          <use value="old" />
          <type>
            <coding>
              <system value="http://www.hospital-a/localid" />
              <code value="MRN" />
            </coding>
            <text value="Hospital A MRN" />
          </type>
          <system value="http://www.hospital-a/localid" />
          <value value="1000000001" />
          <assigner>
            <display value="Hospital A" />
          </assigner>
        </identifier>
        <identifier>
          <system value="http://example.org/SSN" />
          <value value="804234513" />
        </identifier>
        <name>
          <family value="LINCOLN" />
          <given value="MARY" />
        </name>
        <name>
          <use value="old" />
          <family value="WASHINGTON" />
          <given value="MARY" />
        </name>
        <gender value="female" />
        <birthDate value="1954-07-04" />
        <link>
          <other>
            <reference value="Patient/01" />
            <display value="Mary Lincoln" />
          </other>
          <type value="replaces" />
        </link>
      </Patient>
    </resource>
  </parameter>
</Parameters>
```

Reporting Errors/Outcomes

Any errors will be reported with an OperationOutcome resource and could include

Issue	Description	Http Status
err: Same resource	The Source and Target matching resulted in the same FHIR Patient resource, likely already merged.	Bad Request
err: Missing Source Parameters	There are no source patient parameters, please include either a source-patient, source-patient-identifier parameter (or both)	Bad Request
err: Missing Target Parameters	There are no target patient parameters, please include either a target-patient, target-patient-identifier parameter (or both)	Bad Request

err: Target Patient Id mismatch	The target patient id does not match the patient id in the result-patient resource **(should we permit this to create a new resource?)**	Bad Request
err: Source Patient not found	The source patient was not found based on the provided parameters	Bad Request
err: Target Patient not found	The target patient was not found based on the provided parameters	Bad Request
err: Target/Source not duplicates	Attempt to merge 2 records that are known to not be duplicates of each other. *(Previous manual marking of the resources was done, and will need to be removed before retrying)*	422 Unprocessable Entity
err: Target patient already merged	The Target patient resource was previously merged into another patient record	422 Unprocessable Entity
err: Target patient inactive	The Target patient resource is marked as inactive	Bad Request
info: Target Patient updated	Additional notes on what happened to the target patient resource on update, such as if fields weren't updated as requested due to internal business rules etc	422 Unprocessable Entity
info: Update summary	Other notes that are included reporting on what changed, such as how many resources were/may be effected	
warn: Recommend reverse merge	The source resource is much larger than the target resource (in terms of resources that reference it) and recommend that the merge occur in the other direction	

FHIR Messaging

The FHIR Request Message should be a Bundle with:

Resource		Description
MessageHeader	1..1	The Messaging header The focus of the message will be the Parameters resource
Parameters	1..1	The same Parameters object that would be passed to the $merge operation
~~Patient (result)~~	~~1..1~~	~~The resulting Patient resource that we would be expecting to be completed~~ ~~The Patient.id of this resource may be absent, as with the messaging, it may not be known a message generation~~ Brian: This is not required, as it is already in the parameters resource, however should note that this is required in the parameters object when used in messaging
Patient (source)	0..1	Source Patient resource (may not be complete, but should have enough to be able to identify the source record) This is the details of the patient resource that will be marked as inactive after the merge
Patient (target)	0..1	Target Patient resource (may not be complete, but should have enough to be able to identify the target record) This is the details patient resource that will remain active after the merge operation is complete

Note: If the target patient resource has an id, then the resulting patient.id value must be either blank, or the same resource Id.

Operationally, on the receiving system, the end result will be that the id on the patient result and target patient are the same.

The FHIR Response Message should be a bundle with:

Resource		Description
MessageHeader	1..1	The Messaging header The focus of the message will be the Parameters resource
Parameters	1..1	The parameters resource that was included in the request
OperationOutcome	1..1	The results of the merge operation

Patient	0..1	The resulting patient resource from the merge operation (required when the result was a successful operation)
AuditEvent	0..1	An operation event that includes the full details of the operation, including references to all the resources that were updated as a result of the merge

Example Provenance

```
<Provenance xmlns="http://hl7.org/fhir">
    <id value="add4712f870b484dada83e80a249d7fb" />
    <meta>
        <versionId value="2" />
        <lastUpdated value="2019-09-15T17:39:26.3561523-04:00" />
    </meta>
    <text>
        <status value="generated" />
        <div xmlns="http://www.w3.org/1999/xhtml">
          <span style="color: gray;">target:</span> Patient/pat2/_history/41<br /><span style="color: gray;"
>target:</span> Patient/pat1/_history/63<br /><span style="color: gray;">activity:</span> Merge Record
Lifecycle Event<br /><hr /><span style="color: gray;">who:</span> Fixmeup, Steve Dr
        </div>
      </text>
  <target>
        <reference value="Patient/pat2/_history/41" />
    </target>
    <target>
        <reference value="Patient/pat1/_history/63" />
    </target>
    <occurredPeriod>
        <start value="2019-09-15T17:38:56.3087526-04:00" />
        <end value="2019-09-15T17:39:26.3544498-04:00" />
    </occurredPeriod>
    <recorded value="2019-09-15T17:38:56.3087526-04:00" />
    <reason>
        <coding>
            <system value="http://terminology.hl7.org/CodeSystem/v3-ActReason" />
            <code value="PATADMIN" />
        </coding>
        <text value="patient administration" />
    </reason>
    <activity>
        <coding>
            <system value="http://terminology.hl7.org/CodeSystem/iso-21089-lifecycle" />
            <code value="merge" />
        </coding>
        <text value="Merge Record Lifecycle Event" />
    </activity>
    <agent>
        <type>
            <coding>
                <system value="http://terminology.hl7.org/CodeSystem/provenance-participant-type" />
                <code value="performer" />
            </coding>
            <text value="Performer" />
        </type>
        <who>
            <identifier>
                <value value="UID123234" />
            </identifier>
            <display value="Fixmeup, Steve Dr" />
        </who>
    </agent>
</Provenance>
```