

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 (or in some systems deleted)

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 (unless the resource was deleted)

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: <ul style="list-style-type: none">• 200 OK - If the merge request doesn't expect any issues (although warning may be present) for a preview, or was completed without issues if not a preview• 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 be corrected• 422 Unprocessable Entity - Business rules prevent this merge from completing The Parameters resource will include: <ul style="list-style-type: none">• 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)

If a client needs the server to create a new patient merged from the 2 patient resources, the client should create a new patient record and then call the merge operation to merge each old patient 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 cleared/absent.
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. The source Patient record will be marked as inactive, and add the link property to the target patient (except where systems delete the record)
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.

Note: During the pre-merge validation stage, a system may perform other internal checks/business rules.

Merging Identifiers

If the result patient resource 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 resource is **not** provided (only the identifier/reference to select it), 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. *(Review if the marking of old still makes sense here for ALL provided identifiers that get copied across when the result patient isn't there)*

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.

Note: 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.

Review Note: Are there other implications of these reference updates that should be identified relating to versions - (such as where a version specific reference was included)

Note: While this processing is occurring, if a client requests clinical data for either source or target patient, an OperationOutcome with an informative message MAY be included in the resulting bundle indicating this processing is ongoing.

Post Merge Expectations

Once the patient resources have been merged:

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

- 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 (Note: some systems may have cleared all the other properties making this a stub resource)
- 404 not found (when the merge system deleted the resource)

Note: If the system "knows" that the resource was there it would be preferable to return a stub patient 202 as described above.

Note: Security implications such as those from SMART tokens could restrict access here.

When performing a 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 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 no patient resource (case where the old patient was deleted)
- **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**

Accessing the Patient \$everything operation on the source patient resource (now marked as inactive) will return an OperationOutcome and http status of 400 Bad Request. The error message should inform that the patient has been merged and should follow the Patient link to access the \$everything content.

Searching content (e.g. Observations) based on patient ID:

- Observation?patient=Patient/pat01 would return a 200 Ok Bundle with no results (as all have been moved to Patient/pat02), an OperationOutcome may be included indicating that the patient was merged into patient xxx
- Observation?patient=Patient/pat02 would return all the data that is referencing pat02, and all the data that was referencing pat01 (which was updated by the merge operation to reference pat02)

Use Case - Polling using search on Observations to get updates:

- Observation?patient=Patient/pat02 (initial client call at start of year, gets all patient obs)
- Observation?patient=Patient/pat02&_lastupdated=ge2019-03 (client calls at start of march to detect any new patient obs)
- (Patient pat01 is merged into pat02 during April)
- Observation?patient=Patient/pat02&_lastupdated=ge2019-06 (client calls at start of june to detect any new patient obs, but misses all the pat01 observations prior to June)
- Client needs to check for a provenance record of the merge having taken place to determine that they need to refresh the local content to see the older data

In this case need to detect the patient merge, and perform a fresh retrieval of all content, there is no way for the server to return any error codes in this use case, and may also need to consider notification mechanisms too.

Creating/Updating content (e.g. Observations) that reference the old Patient ID: (feedback on this required)

- 422 Unprocessable Entity with an OperationOutcome indicating that the patient referenced was merged into patient xxx (this is also the existing behavior if the patient resource was deleted)
- ~~412 Pre-condition failed with an OperationOutcome indicating that the patient referenced was merged into patient xxx (if the client provided an #Match header)~~
- ~~400 Bad Request with an OperationOutcome indicating that the patient reference did not resolve~~
- **201 Created if the service is able to automatically process the request and reallocate, this could occur during the merge data processing stage, otherwise the above code should be returned**

Merge Notification Mechanisms

The indication that a merge has been completed can be notified through several ways:

- Using FHIR Messaging to invoke the same operation
- 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 (requires the system to have both patient resources)**
- **Client data refresh notification (to be defined, could be triggered by merge, security changes, system migrations, consent changes, etc...)**
- Using Subscriptions to detect the merge operation has occurred (on Provenance and/or Patient)
- Polling the Merge Provenance resource (or the Patient resource for the relevant link change)
- (other non standard notification channels)

These notifications can be sent to other downstream systems, partners, or other applications (including EMPs). An EMPI could expose the merge operation, and therefore be a notification sender.

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

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

Note: When using the identifier parameters (rather than id) you should be using the same assigner (which in the example above would be the PAS/ADT or clinical system), this may be configured in the sending notification system, such as an EMPI based on local business rules.

Impact on Subscriptions

Subscriptions on merges are most likely to be used by applications connecting directly to the system. Many use cases could consider using FHIR Messaging (or other messaging e.g. v2 messages) to communicate the merge occurred.

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
 - Provenance(s) as an event
 - operation itself as an event (the Task resource, although that may not exist, so just a pre-defined topic)
- Will all the data that is patched over to the target patient ID be notified
 - Systems may not notify that the content was changed, and rely on the merge notification to advise if required
 - *Also note the Client data refresh notification discussion above*

Additional work needs to be done on this as the Subscription v2 resource matures, we should check with FHIR Infrastructure and other implementers working on the new content ([Michael Donnelly](#), [Jenni Syed](#))

Mapping HL7v2 Merge to FHIR

There are several flavours of merging in V2 and [Irma Jongeneel/Alex de Leon](#) are going to provide the rough details so that we can explain how to convert into FHIR.

HL7v2 Merges, Move and Linking

Below is a summary of the Merge, Move and Link operations. They are included here as the v2 concepts of Merge, Move and Link may differ (or not) based on the establishment of the Patient, Encounter and Account as separate FHIR resources. The Merge, Move and Link operations have 3 levels: Patient Identifier, Patient Account, and Patient Visit.

Definitions: Merge, move, and change identifier events

The term "identifier" is used throughout this section. An identifier is associated with a set (or sets) of data. For example, an identifier (PID-3 - Patient Identifier List) may be a medical record number which has associated with it account numbers (PID-18 - Patient Account Number). Account number (PID-18 - Patient Account Number) is a type of identifier which may have associated with it visit numbers (PV1-19 - Visit Number).

This section addresses the events that occur usually for the purposes of correcting errors in person, patient, account, or visit identifiers. The types of errors that occur typically fall into three categories:

- **Duplicate identifier created**
The registrar fails to identify an existing person, patient, account, or visit and creates a new, "duplicate" record instead of using the existing record. A "merge" operation is used to fix this type of error.
- **Incorrect identifier selected**
The registrar mistakenly selects the wrong person, patient, or account and creates or attaches a patient, account, or visit underneath the incorrect person, patient, or account. A "move" operation is used to fix this type of error.
- **Incorrect identifier assigned**
The registrar accidentally types in the wrong new identifier for a person, patient, account, or visit. This type of mistake usually occurs when identifiers are manually assigned (not system generated). A "change identifier" operation is used to fix this type of error.

Note that we are addressing only scenarios 1 and 2 as most identifiers are assigned by the related systems, today.

Patient record links

Linking two or more patients does not require the actual merging of patient information; following a link trigger event, sets of affected patient data records should remain distinct. However, because of differences in database architectures, there may be system-dependent limitations or restrictions regarding the linking of one or more patients that must be negotiated.

There are multiple approaches for implementing MPis. It is useful for the purpose of MPI mediation to support two types of linkage. Explicit linkage requires a message declaring a link has been made between multiple identifiers. Implicit linkage is performed when a receiving system infers the linkage from the presence of multiple identifiers present in *PID-3-patient identifier list*.

In an MPI setting, the A24-link patient information message is preferred for transmitting an explicit link of identifiers whether they are in the same or different assigning authorities. The A37 unlink patient information message is preferred for transmitting the explicit unlinking of identifiers.

Implicit linkage of identifiers, sometimes called passive linking, has been implemented using various messages. An acknowledged method is inclusion of multiple identifiers in *PID-3-patient identifier list*, which the receiving system implicitly links. An MPI or application that makes such an implicit linkage can generate an A24 - link patient information message to explicitly notify another system of this action.

Merge Events

ADT/ACK - Merge Patient - Patient Identifier List (Event A40) **(This is the primary event to be considered associated with FHIR Patient merge)**

A merge has been done at the patient identifier list level. That is, two PID-3 - Patient Identifier List identifiers have been merged into one.

An A40 event is used to signal a merge of records for a patient that was incorrectly filed under two different identifiers. The "incorrect source identifier" identified in the MRG segment (MRG-1 - Prior Patient Identifier List) is to be merged with the required "correct target identifier" of the same "identifier type code" component identified in the PID segment (PID-3 - Patient Identifier List). The "incorrect source identifier" would then logically never be referenced in future transactions. It is noted that some systems may still physically keep this "incorrect identifier" for audit trail purposes or other reasons associated with database index implementation requirements.

(TODO: INSERT MAPPING here)

ADT/ACK - Merge Account - Patient Account Number (Event A41) - *For information only, not in scope of FHIR for now*

A merge has been done at the account identifier level. That is, two PID-18 - Patient Account Number identifiers have been merged into one.

An A41 event is used to signal a merge of records for an account that was incorrectly filed under two different account numbers. The "incorrect source patient account number" identified in the MRG segment (MRG-3 - Prior Patient Account Number) is to be merged with the "correct target patient account number" identified in the PID segment (PID-18 - Patient Account Number). The "incorrect source patient account number" would then logically never be referenced in future transactions. It is noted that some systems may still physically keep this "incorrect identifier" for audit trail purposes or other reasons associated with database index implementation requirements.

Implementer Note: This is not merging the Patient, but merging the account, but is the same concept, should we also be including this concept as another potential operation?

ADT/ACK - Merge Visit - Visit Number (Event A42) - *For information only, not in scope of FHIR for now*

A merge has been done at the visit identifier level. That is, two PV1-19 - Visit Number identifiers have been merged into one.

An A42 event is used to signal a merge of records for a visit that was incorrectly filed under two different visit numbers. The "incorrect source visit number" identified in the MRG segment (MRG-5 - Prior Visit Number) is to be merged with the required "correct target visit number" identified in the PV1 segment (PV1-19 - Visit Number). The "incorrect source visit number" would then logically never be referenced in future transactions. It is noted that some systems may still physically keep this "incorrect identifier" for audit trail purposes or other reasons associated with database index implementation requirements.

Implementer Note: Would be interesting to determine if these are used in production (A41 and A42)

Move Events

ADT/ACK - Move Patient Information - Patient Identifier List (Event A43)

A move has been done at the patient identifier list level. Identifier to be moved in the PID-3 - Patient Identifier List and MRG-1 - Prior Patient Identifier List will have the same value. The "from" (incorrect source patient ID) and "to" (correct target patient ID) identifiers have different values. See A43 examples in section 5. The identifiers involved in identifying the patient to be moved (MRG-1 - Prior Patient Identifier List) may or may not have accounts, which may or may not have visits. In any case, all subordinate data sets associated with the identifier in MRG-1 - Prior Patient Identifier List are moved along with the identifier, from the "incorrect source patient ID" to the "correct target patient ID."

ADT/ACK - Move Account Information - Patient Account Number (Event A44) - *For information only, not in scope of FHIR for now*

A move has been done at the account identifier level. That is, a PID-18 - Patient Account Number associated with one PID-3 - Patient Identifier List has been moved to another patient identifier list.

An A44 event is used to signal a move of records identified by the MRG-3 - Prior Patient Account Number from the "incorrect source patient identifier list" identified in the MRG segment (MRG-1 - Prior Patient Identifier List) to the "correct target patient identifier list" identified in the PID segment (PID-3 - Patient Identifier List).

ADT/ACK - Move Visit Information - Visit Number (Event A45) - *For information only, not in scope of FHIR for now*

A move has been done at the visit identifier level. That is, a PV1-19 - Visit Number or PV1-50 - Alternate Visit ID associated with one account identifier (PID-18 - Patient Account Number) has been moved to another account identifier.

An A45 event is used to signal a move of records identified by the MRG-5 - Prior Visit Number or the MRG-6 - Prior Alternate Visit ID from the "incorrect source account identifier" identified in the MRG segment (MRG-3 - Prior Patient Account Number) to the "correct target account identifier" identified in the PID segment (PID-18 - Patient Account Number).

Event A47? (this is similar to A43), Epic implements this and not A43

Link Events

ADT/ACK - link patient information (event A24) - *For information only, not in scope of FHIR for now*

The A24 event is used when the first PID segment needs to be linked to the second PID segment and when both patient identifiers identify the same patient. Linking two or more patients does not require the actual merging of patient information; following a link event, the affected patient data records should remain distinct. For example, this event could be used in a hospital network environment in which there are multiple campuses and in which records need to be linked. For example, hospital A, hospital B, and hospital C would each keep their own records on a patient, but an A24 link event would be sent to a corporate-wide MPI to enable the coupling of ID information with the corporate ID number. It is used for corporate data repositories, etc. This event is not meant to link mothers and babies since a field exists (*PID-21-mother's identifier*) for that purpose. See Section 3.5.3, "Patient record links," for a discussion of issues related to implementing patient link messages and MPI issues.

This event can also be used to link two patient identifiers when a patient changes from inpatient to outpatient, or vice versa. This event can also be used to link two visits of the same patient.

The fields included when this message is sent should be the fields pertinent to communicate this event. When other important fields change, it is recommended that the A08 (update patient information) event be used in addition.

ADT/ACK - unlink patient information (event A37) - *For information only, not in scope of FHIR for now*

The A37 event unlinks two PID segments previously linked with an A24 (link patient information) event.

Safety Checklist

todo

Other Notes

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

this case is only in the notification handling, not the merge request.

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

In HL7v2, the trigger message only reports the results of the operation, and has PID and merged from PID

The HL7v3 in particular looks like they link content, and send a duplicates resolved message interaction.

Q & A

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))

Answer: As this operation is not idempotent it is only defined for POST.

Question: How should privacy related meta tags be handled?

Answer: 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 remaining.

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

Answer: Yes, the task resource can be used to track the asynchronous portions.

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)?

Answer: (As detailed in the merging identifiers section above - **there are still questions on this to check if adequate detail**) 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?

Answer: This would return an error as noted.

Patient A merged into Patient B (success)

Then try to merge Patient B into Patient A?

using MRN identifiers - same resource is detected - return error "err: Same resource"

using the patient IDs - the source A has been deleted, but the B resource will have the link ID, and a provenance - return error "err: Target patient already merged"

using the patient IDs - the source A marked as merged into B and has links between each other, and a provenance - return error "err: Target patient already merged"

Try to merge Patient A into Patient B again?

this will always return the error "err: Same resource"

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

Answer: Yes, as noted above.

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

Answer: After some discussion, the Provenance is the preferred resource to track this content, as will likely remain with the clinical data retention, AuditEvents may be cleansed periodically. AuditEvents are not always available/exposed to all users.

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.

Answer: yes this will be documented, but not initially. This was not successful in v2.

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.

Answer: This should be up to the server's discretion as to if the processing is completed synchronously (as there was very little content to be merged) or if it requires significant time, and return a Task that the client can follow to get completion. The Fhir Async processing system doesn't really apply, as we do not expect there to be content to download later, just the indication that it is complete.

To differentiate, the status 200 indicates all is done (essentially a synchronous result) or 202 that the merge processing is continuing in the background, and you need to monitor the Task for completion (returned in the parameters result).

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.

Answer: There is a preview flag to perform a test run of the merge to detect any issues, and optionally return the scale in an informative message.

e.g. Issue.diagnostics "Merge would update: 10 years of content or 120 resources"

This informative note is really just to assist someone performing the merge to detect that the the correct merge direction is being done, and not merge the 10 year old record into the 1 day old record, but do the other way around. A server may return an error if it detects this case too.

It is at a servers discretion if this is included in the output, and could be of any metric - explicit counts, estimates, age of record (the time to calculate these may influence what metric is decided to be used by a server)

Question: Should we be able to merge 3 patients into 1?

Answer: No, in order to keep things simple for all cases, merging will be from 1 record into another, which can be repeated for multiple resources.

Patient A B, C B at the same time would be possible, however A B, B C would not be possible, and cause a conflict. These would be separate calls, and not part of the same request.

This concurrency would be an internal business rule, and an existing error condition "target already merged" could be used to indicate this case.

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.

Answer: A server MAY provide an additional outcome in any search bundle with the informative information.

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?

Answer: There are 2 options here, delete the patient completely, or mark it as inactive with the link to the active patient (and optionally clear all other data values to ensure that it is not accidentally used - and is quite obvious it's not for use)

Question: Should a search on 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?

Answer: If the resource was physically deleted, then it won't be returned. If the data was cleared from the source patient to prevent accidental use, then it won't be returned, Otherwise the inactive flag should be checked.

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.

Answer: This is not currently planned to be addressed by FHIR with a specific operation, and may be done through regular FHIR restful updates at this stage.

----- up to here -----

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

Answer: Suggest expiring the token, but will refer to the broader group on Zulip.

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.

Question: (During join PC/PA session) [Amit Popat](#) questions the optional nature of the source-patient and target-patient resource references, and encourages always requiring the actual resource.id be specified. The reason for the additional identifiers is to ensure that the correct resource is selected, as with v2 we included additional demographic data to ensure a safe match.

Question: Post merge when committing (PUT/POST) a resource (e.g. Observation) that references the old inactive patient, should it succeed, re-target the resource, or return a failure?

Question: Post merge when searching for an observation using the old patient ID, should it be found? - I think not, but possibly return an op outcome, but this needs discussion

Question: Need to have some mechanism for notifying that a client needs to "refresh" its view of a patient record, as something has significantly changed, e.g. Merge occurred, security permissions changed, policy updated, historic data imported, backup restored etc. End result is that the client needs to discard its local cache, and re-read it all.

Question: Can custom business rules impact a merge request?

Answer: Yes, various systems can have their own internal rules that they may check to ensure that 2 records can be merged - e.g. Checking that both patient resources being merged don't have conflicting clinical content, such as different blood types.

Question: Should resources linked through logical identifiers be impacted by the merge update referencing data stage?

Question: Should the server processing the merge be able to create a new patient resource regardless of the request?

Question: Would the API pattern of returning a Questionnaire for additional information be appropriate to for this operation if additional information was desired to complete the merge?

Question: What happens if the server doesn't expose Provenance resources?

Question: Should AuditEvent/Provenance resources be updated?

Question: Should the Provenance (the simple one) be returned by the Operation in the parameters resource?

Question: Should the more detailed Provenance resource be referenced by the Task on completion?

Question: What should occur when an EMPI tries to merge 2 patients by identifier only, and no patient resource with the target identifier exists? The current definition indicates that an error would be returned.

Answer: As things are currently designed, the call would fail, and expect that the caller would then simply update the patient resource to include the new identifier, and mark the old one as "old", which requires no further processing of data, as none exists.

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>
            </identifier>
        </Patient>
    </resource>
</parameter>

```

```

    <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.	422 Unprocessable Entity
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	Bad Request
err: Source Patient not found	The source patient was not found based on the provided parameters	422 Unprocessable Entity
err: Target Patient not found	The target patient was not found based on the provided parameters	422 Unprocessable Entity
err: Target/Source not duplicates	Attempt to merge 2 records that are known to not be duplicates of each other. <i>(Previous manual marking of the resources was done, and will need to be removed before retrying)</i>	422 Unprocessable Entity
err: Target patient already merged	The Target patient resource was previously merged into another patient record, and is not a suitable target for merging.	422 Unprocessable Entity
err: Target patient inactive	The Target patient resource is marked as inactive Note: Further feedback on this case?	422 Unprocessable Entity

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	-
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 Note: This would likely be returned/evaluated during the preview stage/mode if implemented.	-
info: Patient merge in progress	Note: This is applicable to other search operations on resources referencing these patient resource(s), and not specifically the merge operation itself The patient record referenced by these records is currently being merged to/from another patient resource. Data may be incomplete, or inconsistent. <i>(this MAY be returned during a clinical data search using the patient ID as a search parameter)</i>	-

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..4	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>

```