

RLS PUBLISH BUSINESS RULES AND DESIGN GUIDELINES PROPOSAL

This document outlines the current business rules and design guidelines for the publish operation.

Development team would like to proposed the following and find out if this shall meet all current business requirements.

The PUBLISH operation shall support the new, update, and logical delete publish types. Thru the publish operation data partners shall be able to publish, update and L. delete records.

Action Type	Description
NEW	To publish a new record on RLS index
UPDATE	To update existing patient information or record description information
LOGICAL DELETE	To delete existing patient information or record description on RLS index. This shall be a logical delete, since no data shall be physically deleted from database.

The following design constraints and business rules were identified by Development team.

1. AHCCCS HIE will not support publishing of **NEW** patient information with no related record descriptors.
2. AHCCCS HIE shall not support “**BATCH PUBLISH.**” This means that every record published in RLS will need to be published one transaction at a time (one message at a time). There will be no single message with several records to be published.
3. AHCCCS HIE shall save all transactional information on audit logs to keep a history of all records being updated. Transactional information refers to all the data within the publish response/request message.
4. AHCCCS HIE will rely solely on the transactional audit log to trace any changes made to a record thru the publish operation.
5. AHCCCS HIE will not create a back up copy every time the data is being updated thru the publish operation, since it shall rely on the transactional audit log.
6. Two levels of validation shall be performed, one at the schema level and another at the data element level to make sure mandatory data fields are present within the published message. Mandatory data fields shall be validated per message specification.
7. Publish Results response will consist either SUCCESS or FAILURE, no more information will be given to data partners.
8. Reason for failure must be returned within the Publish Response Message.
 - o Reason for failure may include

- Invalid Format
- Required Field Missing
- Patient Information Missing

Publish Message Data Elements

The following is a list of data elements that can be part of the publish message. If any of the mandatory data elements are missing the publish operation will fail and notification will be sent to the data provider source system through a publish response message.

As shown below the publish message consists of 4 data sets, which include patient information, data provider information, and record information.

The system requires certain data elements related to data provider information and record location information to be Mandatory.

Data Element	Mandatory or Optional
Patient Information	
AHCCCS ID	O
MRN	M
INSURANCE ID	O
LAST 4 OF SSN	O
MULTIPLE BIRTH	O
ADDRESS1	O
ADDRESS2	O
CITY	O
STATE	O
ZIP CODE	M
FIRST NAME	M
MI	O
LAST NAME	M
GENDER	M
DOB	M
Data Provider Information	
ORGANIZATION ID	M
DP ADDRESS	O
DP CITY	O
DPSTATE	O
DP ZIP CODE	O
ORGANIZATION ID	M
DP NAME	M
CONSENT	O
PUBLISH ACTION TYPE	M
Record Information	

RECORD ID	M
RECORD TYPE	M
DESCRIPTOR	M
DATE OF SERVICE	M
Record Location Information	
SOURCE SYSTEM ID	M
RECORD SOURCE NAME	M
GATEWAY ID	M

USE CASES

The following use cases for publish are proposed for implementation.

All publish messages must contain all mandatory elements for data provider information and record location information. Publish operation will failed if this information is not available.

USE CASE	ACTION	Patient Information	Record Information	Data Provider Information	Record Location Information
1	NEW	X	X	X	X
2	UPDATE	X	X	X	X
3	UPDATE	X		X	X
4	DELETE	X	X	X	X
5	DELETE	X		X	X

NOTE: Data Partners will not be able to update patient record information and publish new record information at the same time. Two messages are required for this.

Preconditions to all the use cases are:

RLS received Publish NEW message

RLS validates message, if message is invalid it breaks and returns publish FAILED notification, otherwise it will continue to publish the information.

RLS has already logged Publish message in audit trail

USE CASE 1:

RLS searches to see if patient already exists in RLS index

If patient exists, then NEW record descriptor information shall be created and linked to this patient.

Otherwise, a NEW patient record and NEW record descriptor record shall be created on RLS and linked together.

RLS will return publish SUCCEFUL to data partners.

USE CASE 2:

RLS searches to see if patient AND patient descriptor information already exists in RLS index.

If they both exist, then patient information and descriptor information will be overwritten by what is received in the publish message.

Otherwise, RLS will return publish FAILED to data partners.

Existing data for which we did not receive updated information for will be deleted. (e.g if FN was not received on the publish update, and we already have that value stored for the patient, then it will be deleted by the update)

USE CASE 3:

RLS searches to see if patient already exists in RLS index.

If patient exists, then patient information will be overwritten by what is received in the publish message.

Otherwise, RLS will return published FAILED to data partners.

Existing data for which we did not receive updated information for will be deleted. (e.g if FN was not received on the publish update, and we already have an existing value stored for the patient, then it will be deleted by the update)

USE CASE 4:

RLS searches to see if patient and record descriptor information already exists in RLS index

If they both exist, then ONLY record descriptor information will be deleted.

Otherwise, RLS will return published FAILED to data partners.

USE CASE 5:

RLS searches to see if patient exists in RLS index.

If patient exists, then BOTH patient information and all record descriptor will be deleted.

Otherwise, RLS will return published FAILED to data partners.