AHCCCS Search Engine

Conceptual Design

Author Anthony Christianson
Author Position
Date 11/28/07

Version: 1.0

11/28/2007

Revision & Sign-off Sheet

Change Record

Date	Author	Version	Change Reference
12/4/07	Anthony Christianson		

Reviewers

Name	Version Approved	Position	Date

Distribution

Name	Position

Document Properties

Item	Details
Document Title	Conceptual Design
Author	
Creation Date	
Last Updated	

AHCCCS

Table of Contents

AHCCCS

Conceptual Design Summary

The purpose of the AHCCCS Search Engine is to consolidate a patient's demographic data into a patient index to minimize the number of items returned during a patient lookup.

The design of the RLS mandates that each patient *record* is a single entry. Therefore, a single patient could feasibly have one or more entries in the RLS system. The AHCCCS Search Engine will store each patient as a single entry, eliminating duplicate patient entries. As a result, the number of entries in the engine will be significantly less than the amount of entries in the RLS system. The smaller size of the engine will result in faster searches.

In addition, the AHCCCS Search Engine will store possible aliases of a patient. If two patient records in the RLS system are for the same patient, but there is minor discrepancy of the patient demographic data, depending on business rules, it will be possible for the engine to identify these records as belonging to the same patient and merge the second set of demographics in with the patient's current demographics.

Business Situation

When publishing a patient, the accuracy of the published data can be questionable. Whether it is a spelling variation of a name or inaccurate data entry, the potential for a patient to have records in the RLS under varied data is to be expected. The AHCCCS Search Engine will be populated with information that consolidates a patient's information into fewer, more accurate items to increase the intelligence and performance of a search.

Overview of Conceptual Solution Alternatives

The alternative to not creating an AHCCCS Search Engine is to not create an engine and allow searching the RLS system directly. This would pass the patient search criteria directly to the RLS. This would require the querying entity to filter out duplicates or possibly display hundreds of matches for the same patient or display no matches for the patient due to inaccurate search criteria.

Advantage of not creating the AHCCCS Search Engine

- Patient data stored in a single place eliminating redundant data
- Reduce development time for designing, creating and testing the patient index

Disadvantage of not creating the AHCCCS Search Engine

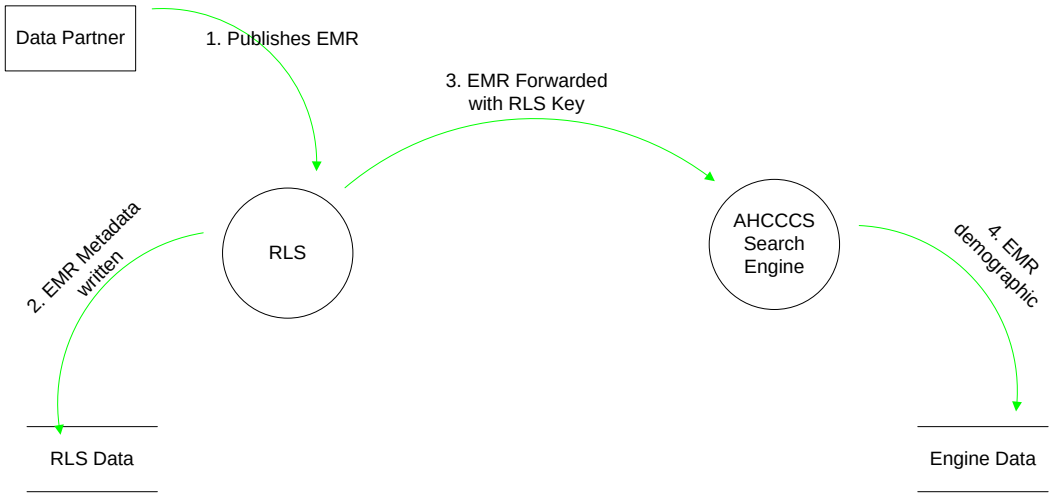
- Patient search performance
- Patient search inaccuracy

AHCCCS

Solution Architecture

The AHCCCS Search Engine will be a separate component within the RLS system. The AHCCCS Search Engine will be utilized when called directly from the Patient Directory service or when the RLS identifies that an incoming request should use the component. The AHCCCS Search data store will contain the consolidated patient information in a search optimized format with enough information to accurately identify a patient. The patient index will be populated in real-time when new patient records are published to the RLS system.

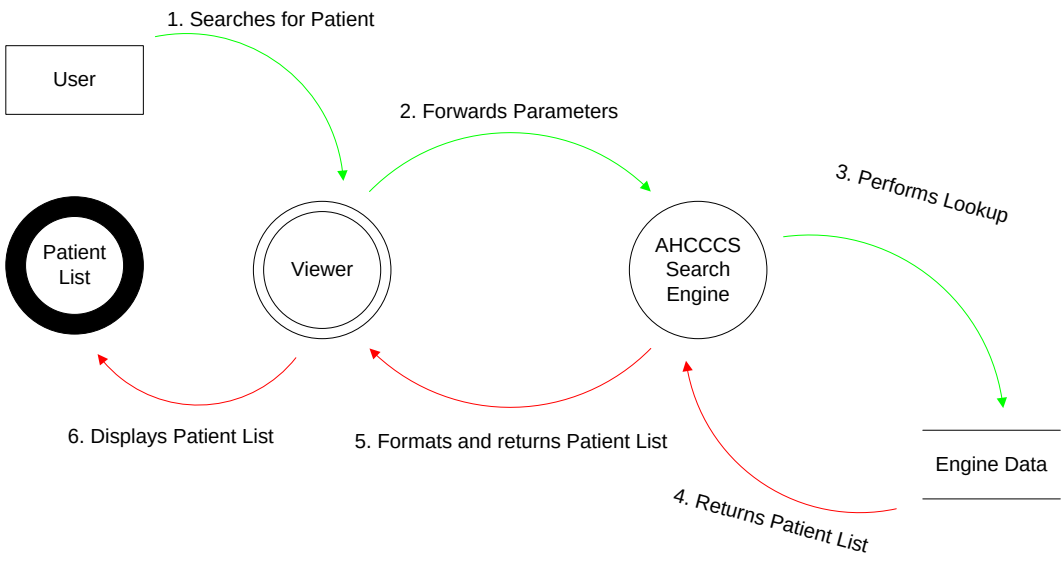
Patient Record Publish



1. Data Partner publishes an EMR to the RLS system
2. RLS insert/updates the EMR's metadata in the RLS
3. RLS forwards the EMR to the AHCCCS Search Engine with the RLS EMR key
4. Patient Index writes the EMR demographic data to the engine data store including the RLS patient key for cross-referencing

AHCCCS

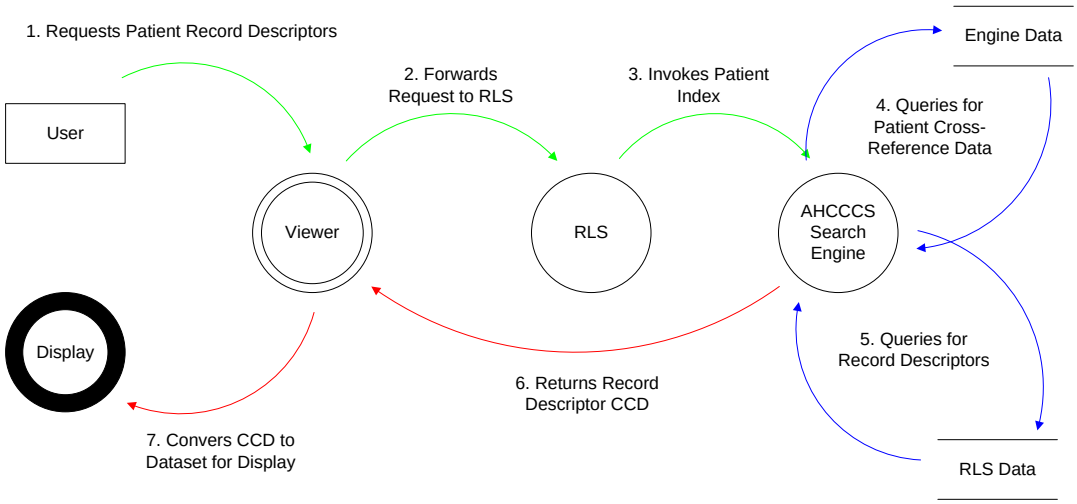
Patient Search



1. User searches for Patient information
2. Viewer validates search parameters and forwards parameters to Patient Index
3. AHCCCS Search Engine performs lookup using parameters
4. AHCCCS Search Engine receives patient list that match given parameters
5. AHCCCS Search Engine compile list returns to Viewer
6. Viewer displays patient list

AHCCCS

Patient Record Search



1. User requests a patients record descriptors
2. Viewer creates message with AHCCCS Search index and AHCCCS Search code system
3. RLS identifies the search code system and forwards parameters to AHCCCS Search Engine
4. AHCCCS Search Engine queries engine data for RLS cross-reference keys
5. AHCCCS Search Engine queries RLS for records descriptors.
6. AHCCCS Search Engine formats RLS results into CCD
7. Viewer converts CCD into dataset for display