2013 Solution Proposal

for

From

((CENTRIC))

www.centricconsulting.com

Team Centric

Mark Stetzer

Praneet Loke

Don Dillard

Table of Contents

EXECUTIVE SUMMARY

Centric Consulting is proud to present our solution proposal for this year's Hoosier Healthcare Innovation Challenge. We have chosen Challenge #2 – Healthcare Exchange Data Management.

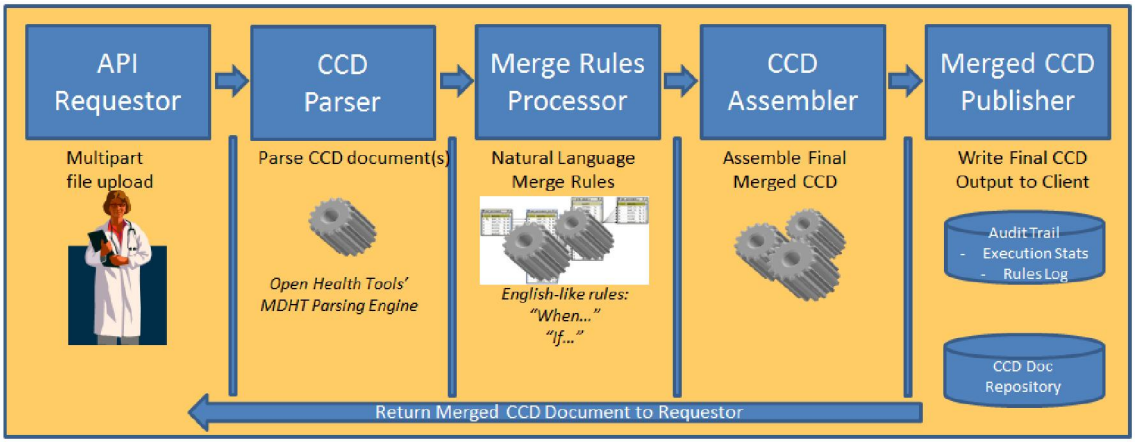
We believe this solution provides the foundation for a useful tool whose concepts rely upon and combine existing technologies in an innovative way. These technologies will provide an avenue for non-technical people to support CCD Merging.

Our solution overcomes the technical burdens of defining mapping and merging programming through the user of a natural language rules engine. In doing so, defining the CCD Merge can be defined, specified, reviewed, and executed from a business person's perspective instead of relying upon programmers and IT departments to build complex, static programs and mapping rules that must be maintained by expensive resources.

Process Overview

1. The CCD Merge API will accept any number of CCD documents in a single merge request, and is built on simple HTTP operations
2. CCD documents are uploaded to the API in their XML format, where they are then parsed into an object model defined by Open Health Tools' MDHT (Model Driven Health Tools) open source product
3. The model objects are then run through a series of configurable rules in a rules engine - the rules themselves are written in natural language instead of code for easier comprehension by semi- and non-technical personnel
4. The output of the rules engine run is a single CCD document
5. The last operations happen in parallel: returning the single merged CCD document to the API requester, AND writing an audit trail of all the documents to a persistent store (database) - this ensures maximum throughput for requests

The rules engine is driven by an open source Drools back end and managed by a simple natural language rules. The solution proposal for the challenge is limited to only one version of the CCD. However, beyond this challenge, this solution could be expanded to support multiple versions of the CCD as well as other standardized healthcare dataset definitions.



CCD Merge Process / Information Flow

WORKING MODEL

API Requestor

Centric's solution will utilize a multipart file upload to receive CCD documents from a specified location and transmit them to the merge process. The multipart file upload feature allows for the transmission of multiple CCD documents in a single transmission.

CCD File Parser

Centric's solution will utilize an already existing parser to process incoming CCD documents. The tool is called the MDHT Parsing Engine, and it is provided by Open Health Tools. This is an open source document parsing engine and will input a CCD XML document and present the data in that document to our solution's processing engines.

Merge Rules Processor

Centric's solution will utilize a natural rules processing engine to accomplish the actual merging of datasets. The natural rules engine is an open source tool calls Drools. The tool is written in Java and receives natural langue inputs for processing. Drools is already recognized in the industry as an effective natural rules process. The use of this engine will more than likely make the Centric solution unique and will also provide for flexibility and maintainability over time, since merge rules will not be stored in programming language and require developers to maintain.

Rules will be defined in a simple interface as a kind of "fill in the blank" exercise, where a series of natural language statements are provided ahead of time with placeholders for decision points, and business users just have to fill in those placeholder values. An example of a rule is as follows:

> **"When** only one document has a *Problems* section with an observation of *Osteoporosis Other*
> **When** the above observation has a status of *completed*
> **Then** add this observation"

CCD Assembler

Once each of the CCD documents have been parsed and merged through the rules engine, the CCD Assembler will prepare the output document, identify any errors encountered in the process, save the merged CCD document in the repository, and save any audit trail information into the audit trail repository.

Merged CCD Publisher

The final step of the process returns the merged CCD document to the requestor. The audit trail information will be available for query purposes to identify which rules were triggered during the merge process as well as the reasons for any errors that were encountered.

Solution Execution / Operations Environment

Solution can be hosted on Linux, Windows, MacOSX, etc. at Amazon or in a sponsor's environment. The solution will be written in Java, Scala, and Javascript, and will require server resources with the Java Virtual Machine (JVM) to run. The merge API can scale indefinitely; the only limiting factor will be the persistent storage layer that keeps an audit trail of all merge transactions.

Rules Capture Process

To start rules will likely be specified in text files, but the Drools Guvnor interface can be used after initial deployment to manage the rules at a higher level.

Centric will publish an initial set of rules that shows all the options the sponsor has to work with, and then allow the sponsor to specify the remainder of the rules as necessary and manage the rules going forward.

Centric's solution will provide a framework for error handling as part of the deliverable.

CCD Merge Process Notes

For this solution, the merge process is triggered manually. In later iterations, the process can be automatically triggered by the arrival of source documents or a timed business event.

Due to the natural language processor, the merge process will not hard-code the mapping rules. The natural language instructions will determine how the merge will execute individual field mappings and decide upon the actions to be taken when conflicts arise. These rules can be specified at the field level. Actions may include, but not limited to:

- Newest takes precedence
- Oldest takes precedence
- Specific dataset takes precedence over another
- Report the conflict and allow the user to resolve manually

Data Model

The Centric solution will result in two distinct persistent data repositories:

Repository	Description
CCD Document Repository **	This repository will contain the merged CCD documents in their final state. This repository is made up of two tables: transactions and transaction_documents.
CCD Process Audit Trail **	This repository will contain information regarding how and when the CCD merge process executed. It will retain when the process was triggered and any exception conditions encountered during the process. In addition, it will reflect which business rules were triggered during the CCD merge process. This repository is in the transaction_rules table.

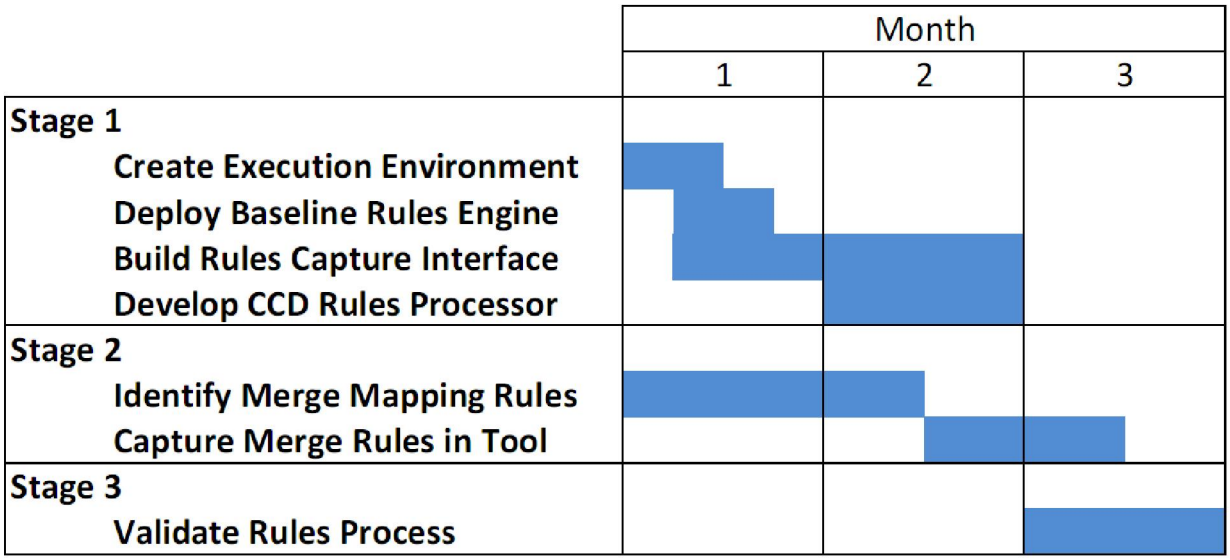
*** The data model for these two repositories is contained in the Appendix of this document.*

PROJECT AND DEVELOPMENT PLANS

The project to implement this solution will take place over a period of xxx months. The development effort will take place in three primary stages:

- Stage 1 will focus on creating the execution and rules engine execution environment. This is the technical portion of the development that will combine the open source DROOLS engine with the features to capture natural language rules, process the rules against the source datasets, and producing the single merged CCD.
- Stage 2 will focus on capturing the rules necessary to process the CCD merge. These natural language rules will be entered into the solution built in Stage 1.
- Stage 3 will focus on validating the technical solution and preparing it for deployment.

The following is an overall depiction of the project and development plan:



The project will utilize a project team of two resources. One resource will provide the primary programing and development tasks, while the second resource will provide the business analysis and rules specifications to accomplish the CCD Merge. Finally, in the later stages of the project, the developer resource will deploy the solution, while the business analyst resource will validate the solution.

IMPLEMENTATION AND INTEGRATION PLANS

Team Centric will deploy this solution onto a public domain location for download by any healthcare consumer organization needing the service. Therefore, this solution can be used by all. This solution will leverage the core rules engine and natural language rules for processing a single version of the CCD.

The expanded versions of this solution will expand the business rules to allow for multiple versions of the CCD as well as other dataset types.

This solution can be deployed for any consumer. The Amazon AWS platform will provide sufficient environmental capacity for a consumer of this service. For simplification and the protection of data, each consumer should be deployed in their own execution instance.

Later versions of this product can expand upon this solution to allow for multiple versions of the CCD and managed through the natural language rules specified for the merge.

Later versions of this product can also expand to support additional dataset types through the expansion of the base natural language rules and providing access to additional source and target data environments.

This solution is only limited by the number of rules that can be captured, the accessibility of source datasets, and the security requirements of the consumer.

APPENDIX

CCD Merge Data Model

Table: transactions

Column	Description
Id	unique auto-assigned incrementing number representing every individual merge transaction
created_at	timestamp with time zone indicating when the transaction was recorded

Table: transaction_documents

Column	Description
id	unique auto-assigned incrementing number representing every individual document included in a merge request
transaction_id	reference to parent transaction
created_at	timestamp with time zone indicating when the document record was recorded
contents	compressed representation of the original document submitted to the API

Table: transaction_rules

Column	Description
id	unique auto-assigned incrementing number representing every individual rule fired during a merge request
transaction_id	reference to parent transaction
created_at	timestamp with time zone indicating when the rule record was recorded
contents	text of the rule that fired during this merge transaction