Sabbatical Report for Fall 2012

Ron Little

Introduction

I spent my sabbatical working with Lost Creek Consulting of Springfield, Oregon. They provide two types of services for healthcare organizations.

1. Migrate patient data from one type of electronic health record (EHR) system to another
2. Develop the interface between two health care organizations to transfer patient data

David Erickson, the owner and CEO of Lost Creek Consulting, has served on the advisory committee for the CIT department at Lane Community College. He was very helpful regarding my sabbatical project by providing me these resources:

- a cubicle with desk
- access to their network system and servers
- the ability to observe the workflow of the employees performing EHR data migrations

For more information regarding the organization see www.lostcreekconsulting.com.

Project Objectives

Lost Creek Consulting is a healthcare information technology (IT) organization that services healthcare organizations such as hospitals and clinics that provide patient care. Therefore, even though they don't provide patient care directly themselves, Lost Creek was a good fit for my four objectives which are listed below.

Four Objectives:

1. Develop significant expertise in a commercial electronic health records (EHR) system such as Centricity, EpicCare, NextGen or Practice Fusion.
2. Evaluate and analyze at least two significant current healthcare workflow processes. This would be accomplished by shadowing one or more physicians in order to observe and document the processes.
3. Determine improvements to the processes to integrate into the EHR processes.
4. Configure the EHR to support the process modifications and improvements.

Project Outcomes and How the Objectives Were Accomplished

I spent significant time studying and using Centricity and other EHR systems. In addition I was trained on the software tools that Lost Creek has developed that help them accomplish their work. I documented a number of their current processes by observing their healthcare IT professionals perform their job responsibilities. These current processes were then analyzed for improvements and recommendations were made to alter process steps to streamline the processes in order to improve efficiency. Finally in some instances I assisted the project team in implementing the modified process. Processes were documented utilizing activity diagrams, process descriptions and workflow process maps (also called workflow charts).

Concerning Objective #1 Above

The exposure I obtained to Centricity, EpicCare and other EHR applications expanded my knowledge of health IT systems and has already had a positive influence on curriculum in courses I teach. For example winter term 2013 in HI 208 and CS 275 I added student exercises relating to Centricity and other EHRs. In addition I will be modifying lecture content and activities in a number of other courses I teach and/or lead. These modifications will be a result of the knowledge I gained from this experience.

Concerning Objective #2 Above

The primary two healthcare IT workflow processes performed by Lost Creek that I chose to analyze were:

1. Export patient data from the EHR currently utilized by the clinic or hospital. The organization terms this the extraction or export process.
2. Verify data accuracy and import patient data into the new EHR chosen by the clinic or hospital. The organization terms this the conversion or import process.

This objective was accomplished by shadowing Lost Creek's health IT professionals rather than shadowing physicians. I observed each the three operation's analysts that performed the tasks of the extractions and conversions. I also observed the project manager who manages more than 20 extraction and conversion projects at any given time. These projects involved exporting from or importing into Centricity, EpicCare, Amazing Charts, GreenWay, Aprima, and other EHR systems. The exposure to Lost Creek's processes and these EHR applications and their databases has expanded my knowledge of manipulating large data sets. This will have a positive impact on the courses that are part of Lane's Database Specialist Certificate since I teach these courses and coordinate the certificate.

Concerning Objective #3 Above

While observing the analysts and project manager I documented the current processes and analyzed these processes to determine recommended improvements to the processes related to EHR data migrations. The recommendations were submitted to the operations manager and CEO of in the form of reports. This not only resulted in gaining more experience with process analysis

and process improvement, my knowledge of project management (which I also teach) was enhanced due to my ability to apply that knowledge to their project management mechanisms.

Concerning Objective #4 Above

I assisted with some of the modifications to the processes relating to the EHR data migration process. This typically involved reconfiguration or redesign of some of the process steps and how some of the EHR related software tools were utilized. The hands on experiences I had regarding these processes have been quite beneficial to my understanding of EHR features and functions.

Additional Comments

In addition to the two primary processes several other related processes were analyzed for improvements. They were:

- Project planning process for extraction and conversion projects
- Change control process for extraction and conversion projects
- Risk assessment process for extraction and conversion projects

Just as with the two primary processes, reports were provided to Lost Creek identifying recommendations for improvements based on the analysis. In some cases I provided assistance to implement those improvements.

I have provided the following documents along with this sabbatical report as samples of the reports developed for Lost Creek.

- Extraction and Conversion Process Descrip Planning Phase
- Extraction and Conversion Process Descrip Remaining Phases
- Analysis of Change Control Procedure
- Analysis of Risk Assessment Process

I am very grateful to David and his employees. They provided a work setting that improved my knowledge of health IT systems and operations, grew my understanding of the advances in healthcare related technology and increased my enthusiasm to teach health informatics and database related topics.

Analysis of Change Control Procedure

Ron Little

Introduction

This analysis is based on the Change Control Procedure used by LCC as of 11/16/2012. This procedure is described in the document "Change Control Procedure as of 11-16-2012". You should refer to it before reading this analysis.

Current Change Control Procedure

LCC's current Change Control Procedure is contains specifics steps to generate a change request. It takes into account the importance of obtaining client buy-off when extra charges need to be assessed due to the client taking the project over budget or when the client requests tasks that are out of scope. However there are several concerns regarding the current Change Control Procedure that may exacerbate problems with projects. Those identified here are:

- Lack of searchable issue and change logs
- Lack of guidelines to determine what issues should initiate a change request
- Scarcity of change requests when the delay is the client's responsibility

Each of these concerns is described in more detail below.

<u>Lack of searchable issue and change logs</u>

The change information is stored only in a Word document that serves as the Project Change Request. This makes it difficult to search for past issues that may be similar. In addition, it limits the information that can be logged regarding the issue that instigated the change request. In many cases it would be important to log information about why the issue was or was not escalated to a change request. For this reason it would be beneficial to create and maintain an issue log for projects. The issue log would track all issues whether or not they became change requests.

A change log could also be created and maintained to easily track what issues become change requests, how they altered the cost and schedule, when a change request was carried out and whether the change request was completed within its estimated cost and duration.

Since issues aren't logged, it is difficult to identify which issues that didn't result in a change order still delayed the project. Thus the reason(s) that a project is late may not be documented. In turn, this may cause clients to believe that unmet due dates and go-live dates are more often LCC's fault than they actually are, harming LCC's reputation.

Lack of guidelines to initiate change requests

There are no formal guidelines to determine what issues should initiate a change request documented. Therefore one analyst may follow a different pattern of initiating requests than another. This is likely to result in inconsistencies regarding the types of issues that are escalated to change requests. For example an analyst may verbally commit to the client to perform work outside of the scope and perform the work without sufficient analysis of how it may delay the project. Without formal guidelines the analyst may feel permitted to determine whether the extra work should be performed by his/her own analysis. This allows for issues to be easily misanalysed and may result in underestimating cost and time requirement increases on a greater number of projects than if formal guidelines were in place.

Scarcity of change requests

Although there are many reasons projects are late, LCC is not submitting change requests on many of them that could be deemed the client's fault. Whether the client or LCC is at fault, the issue can be recorded in an issue log. After that if it is not LCC's fault, some evaluation could be done to determine if a change request should be submitted to the client. Due to LCC's desire to accommodate the client we often don't submit charges on extra work performed for the client.

The decision to accommodate the client should be built into the change request process. If the determination is to not generate a change request for an issue created by the client, the reasons for doing so would be logged in the issue log for future reference and analysis. If it is LCC's fault some analysis could be done to determine how to reduce the chances of the issue and/or its impact in the future. The results of the analysis could be recorded in the issue log so that our solutions as well as our problems are tracked.

There are a number of reasons why projects are delayed that may be out of our control. Some of the reasons LCC could consider originating change requests for include but are not limited to:

- Client delayed responding or performing their tasks
- There were unforeseen issues with the performance of the client's hardware, software or network
- Client misunderstood our deliverables and pushed for us to do more and we accepted without a change order
- Client wanted something outside of the original scope and we agreed to perform it without a change order
- Third parties change procedures and processes without involving LCC and our client; leaving the client with no recourse but to hope we'll be accommodating

The concerns discussed above and the inconsistency of the process underscore that the current Change Control Procedure could be improved. A possible alternative Change Control Procedure

is presented next. The following is meant to be a procedure to critique and modify by the LCC team if it is determined that the problems specified above should be addressed. For a comparison refer to the document "Change Control Procedure as of 11-16-2012" which is the current procedure I suggest modifying.

Modified Change Control Procedure

I recommend that the following Change Control Procedure be evaluated and considered for adoption by LCC. I believe it constitutes an improvement over the current procedure. An evaluation of this suggested procedure by the LCC team may produce changes to it that result in even more improvement.

1. Any and all problems or issues identified by the client or an LCC employee are recorded in the Issue Log. Some of the Issue Log fields are filled in immediately and others are filled in later. The Issue Log design and its fields could be determined by the LCC team. To provide for a place to start the discussion of its design, I've provided some suggestions here. The log could be an Excel spreadsheet or part of a knowledge database. Information recorded could include:
    - Description of the issue
    - Date issue identified
    - Who identified the issue
    - Description of the cause of the issue
    - Who caused the issue (Client or LCC)
    - Whether or not the issue was escalated to a change request
    - Date decision to escalate or not was made
    - Description of how the issue will influence the project
    - Amount of estimated additional time the issue will require
    - Specifics used to determine if the issue should escalate to a change request
    - Whether or not the issue was corrected
    - Date it was corrected
    - Information on how it was corrected if it was
    - Amount of time the issue actually required

2. All LCC project stakeholders are informed of the issue.

3. The project manager and analyst(s) determine if the issue could increase project time.

4. If it is determined that the issue may increase project time, an analyst performs an evaluation to identify any extra time required for it. In some cases programmer support will be necessary. In this case the analyst meets with the programmer to obtain the estimated time for any programming which is added to any extra time that may be required by the analyst(s).

5. The project manager and analyst(s) determine if the issue should be escalated to a change request based on a set of guidelines. These guidelines would be created by the LCC team as part of adopting a new Change Control Procedure. Some of the guidelines could be
    - If the issue is not LCC's fault and could increase project time
    - If a request by the client is outside the scope of the project (Even if it may not push out due dates or increase client charges, it could be processed as a change request so that it is readily apparent the extra LCC does to satisfy the client)
    - If the issue is LCC's fault and will cause a project delay, this should be communicated to the client. If the issue is not escalated to a change request, a report including new project schedule could be generated and sent to the client.

6. The appropriate information is added to the Issue Log.

7. If the decision is to escalate the issue to a change request, the change request is logged in the Change Log. The design and fields of this log would be determined by the LCC team. It could be an Excel spreadsheet or part of the same knowledge database as the Issue Log. The fields could be the same as those in the current Project Change Request plus a few addition fields such as the date signed by the client and the date signed by LCC.

8. The Project Change Request is generated as a report from Excel or the database system.

9. If due dates will be altered the project manager modifies the project schedule.

10. The Project Change Request and modified project schedule are reviewed by the operations manager and then David. If there are no modifications required, the project schedule and change request are made available to the client for signature.

11. Once the signed change request is returned, the operations manager, project manager or analyst sign for LCC.

12. A copy signed by the client and an LCC employee is sent to the client.

13. If there are new tasks required by the change request, they are entered into ActiveCollab.

14. Once any tasks have been performed to address the issue or change request, the appropriated information is entered in the Issue Log.

15. All LCC stakeholders periodically review the Issue Log and Change Log.

16. The Issue Log and Change Log are periodically analyzed to evaluate repeating issues, unresolved issues, whether the issue required more time than projected, what issues cause project delays, projects that are completed even later than the altered go-live, etc. Excel or database report(s) could be created to assist with this.

Analysis of Risk Assessment Process (11/19/12)

Ron Little

Introduction

This document is an analysis of LCC's risk assessment process and contains suggestions concerning the implementation of a formalized process. The purpose of risk assessment is to ask the question "What could go wrong with the project?" A risk mitigation plan that specifies a possible contingency for each risk can be defined in advance. This attempt to look forward helps prepare for many of the issues that could disrupt a project.

Risk Assessment Process

Although LCC does not typically use the term risk assessment, there is a risk analysis that occurs in the planning phase of the project. When the project is evaluated to determine if it is feasible, many of the risks that relate to issues that have been encountered in previous projects are considered. However LCC could benefit by defining a more formalized process that plans beforehand how to handle risks that transpire.

A high level set of steps to consider are listed here. Details of performing these steps are subsequently discussed.

1. Identify potential risks
2. Estimate likelihood of the risk occurring
3. Estimate potential impact on the project if the risk occurs
4. Develop a risk mitigation plan that specifies possible contingencies
5. Report/track the risks that occur by recording them in a risk log

Details of the Process

The first step is to identify as many of the potential risks as possible. One way to accomplish this is to engage in a group brainstorming session that attempts to list anything that could go wrong. The list of risks should include those related to:

- human resources
  - is there redundancy of job duties (more than one C# programmer)
  - skill level of people
  - availability (what other projects are they working on; could they leave)

- technology resources
    - hardware
    - software
    - network infrastructure
- other resources, such as
    - product availability from vendors
    - client's availability to meet

Common Risks

The types of risks that are usually considered fall under a number of categories. Here are some common causes of missed project deadlines.

- Unrealistic timeline expectations
- Loss of personnel (analyst, programmer, project manager, etc.)
- Client doesn't perform their function in a timely fashion
- Communication issues (more on this later)
- Personnel issues
    - Temporary loss (PM, programmer or analyst is out sick)
    - Permanent loss (PM, programmer or analyst takes another job)
- Scope changes
    - Usually increase the project time and costs
- Slow hardware/software processing (image or document conversion)
- Vendor issues
    - Vendor cannot deliver needed product such as a new computer quickly enough
    - Could view LCC programmers as a "vendor" providing software tools. What if they take longer to program a solution than anticipated?
- Legal issues
    - Client threatening or initiating a lawsuit could delay or put on hold a project
- Technology unavailable
- Project complexity (project team may not be experienced in the current type of project)
- Team skills (project team members are still inexperienced in usual projects)
- Quality of outcome (Does it meet standards?)

Risk Matrix

During the planning phase of a project, a risk matrix can be used to list and describe the risks as well as specify the preferred contingency in the event the risk occurs. The risk matrix should include:

- Risk ID or name
- Risk description
- Risk impact if it occurs (can rate low, medium, high or 1 – 10)
- Risk probability (can rate low, medium, high or 1 – 10)
- Risk priority (multiply risk impact by risk probability)
- Risk contingency to mitigate the risk if it occurs

Risk Evaluation

After identifying the risks, evaluate the risk based on the probability that it will occur and the potential loss (impact) that may result. In addition a risk priority value and contingency is entered into the risk matrix. The following describes the information that would be entered into a risk matrix like the example shown below.

- The risk name and description fields of the matrix come from the step of identifying the risks.
- Rate the impact on the project using a scale of 1 to 10 or a scale of low, medium, or high
- Rate the probability the risk will occur using a scale of 1 to 10 or a scale of low, medium, or high
- Obtain a risk priority value (an advantage of using a scale of 1 – 10 for impact and probability is that one can multiply the probability times the impact to obtain this value)
- Specify a contingency to mitigate each risk
    - By the time the project starts, it should have been decided which contingency to use if the risk occurs
    - Usually the risk table would only include one contingency per risk
    - In the example below, two are listed under some risks to show that part of the risk analysis is to select one best solution or course of action before the risk ever occurs
- Many contingencies will increase the project cost or the project time so another column specifying the dollar value of the estimated cost increase could be added, if it can be accurately estimated
- What and when to communicate to the client could also have been predetermined so another column could specify that information

Example Risk Matrix

Risk Name	**Description**	**Probability** Scale of: 1-10	**Impact** Scale of: 1-10	**Risk Priority (Probability X Impact)**	**Contingency to mitigate the risk**	**Estimated Cost Increase**	**What & when to tell client**
Temporary loss of analyst	Analyst cannot come to work for a short time	7	6	42	-delay the project -or re-delegate job duties		
Permanent loss of analyst	Analyst completely unavailable	3	10	30	-delay the project and re-delegate job duties to another analyst -or delay the project & hire another analyst		
Failure of analyst's	Hard drive or operating	5	8	40	-virtual machines		

computer	system failure				available on a server		
Client system inaccessible	Analyst cannot access due to client system failure or client delay	8	7	56	-if possible perform part of the data manipulation or analysis locally		
Temporary loss of project manager	Is ill, on vacation or has a family emergency	7	7	49	-have an analyst trained to step in		
Others…							

Logging Risks

The final step of the process is to log the risks that occur. The purpose is to record enough information to be able to track and evaluate the actual project impact and the effectiveness of the contingency plan. One approach to track the risks that occur during the project is to utilize a Risk Log with fields like those listed in the Example Risk Log below.

- Risk ID
- Risk description
- Trigger – what triggered the risk occurring
- Risk contingency that was actually performed. Rarely this may be different from the planned contingency in the Risk Matrix
- Description of the actual project impact. This may be different from the projected project impact in the Risk Matrix.
- Actual cost of the risk. This may be different from the estimated cost in the Risk Matrix.
- Risk monitor (usually the project manager, but there may be exceptions)

Example Risk Log

Risk Name	**Description**	**Trigger**	**Contingency or Mitigation Process**	**Impact Description**	**Actual Cost**	**Monitor**
Temporary loss of analyst	Analyst cannot come to work for a short time	Analyst is ill	-delayed the project	Late on the project by 3 days		Project Manager
Permanent loss of analyst	Analyst completely unavailable	Found another job	-delayed the project	Project 4 weeks late		Project Manager
Failure of analyst's computer	Hard drive or operating system failure	Hard drive crash	-virtual machines available on a server	Analyst system back up same day no delay		Project Manager
Others…						

Another approach would be to use an Issue Log to log the risks. See the document titled "Analysis of Change Control Procedure". Some fields, like those above, could be added to the Issue Log in this document to specify the Risk ID, its trigger and its actual impact on the project.

Extraction and Conversion Process Description (Development, Test and Execution Phases)

Ron Little

As of 12/11/2012

Introduction

The purpose of an extraction and conversion project is to export patient data from the client's old Electronic Medical Record (EMR) and import it into a new EMR. The old EMR is considered the source EMR and the new is considered the target EMR. Since typically the manufacturer of the source and target EMRs are different the fields in the two databases don't align. For example the fields used for patient IDs may not be named the same, may not be the same data type or may have different size limitations. Some fields may not have a similar field in the new database. Issues like these create the need to properly convert the data in order to import it into the new database.

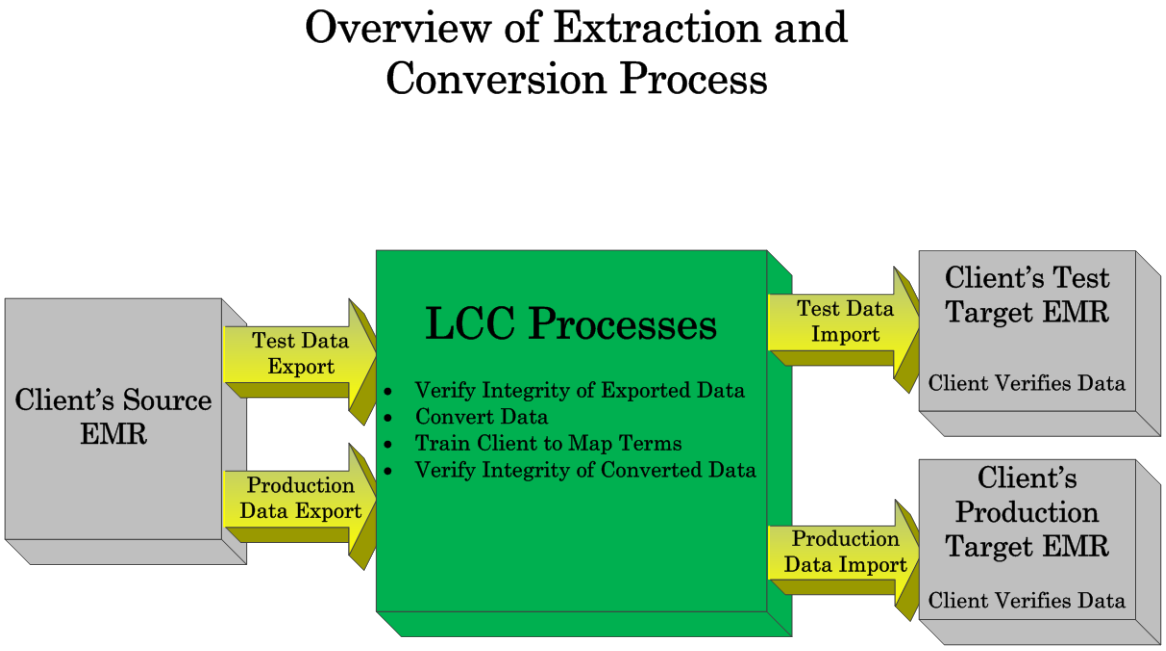
The process described here assumes that LCC is performing both the extraction and the conversion. There may be differences in the process if LCC performs only the extraction or only the conversion. This document picks up with the development phase which follows the planning phase. The planning phase of this process is described in the document "Extraction Conversion Process Descrip Planning Phase".

The process is described from a high level view first. The flow of data within the process is covered next. Finally the detailed steps that need to be performed are specified.

Process Overview

A set of test data is first exported from the source EMR. An LCC analyst analyzes the data, converts it, trains the client to use LCC software tools and analyzes the data again after converting it. Once the data is imported into the target EMR configured for testing, the client verifies the integrity of the test data. Once the process is proven to properly import the test data, an export of production data from the source EMR is performed and many of the same conversion steps are performed. This production data is imported into the target EMR that will be the client's production health records system. A high level view of this process is shown in figure 1 below.

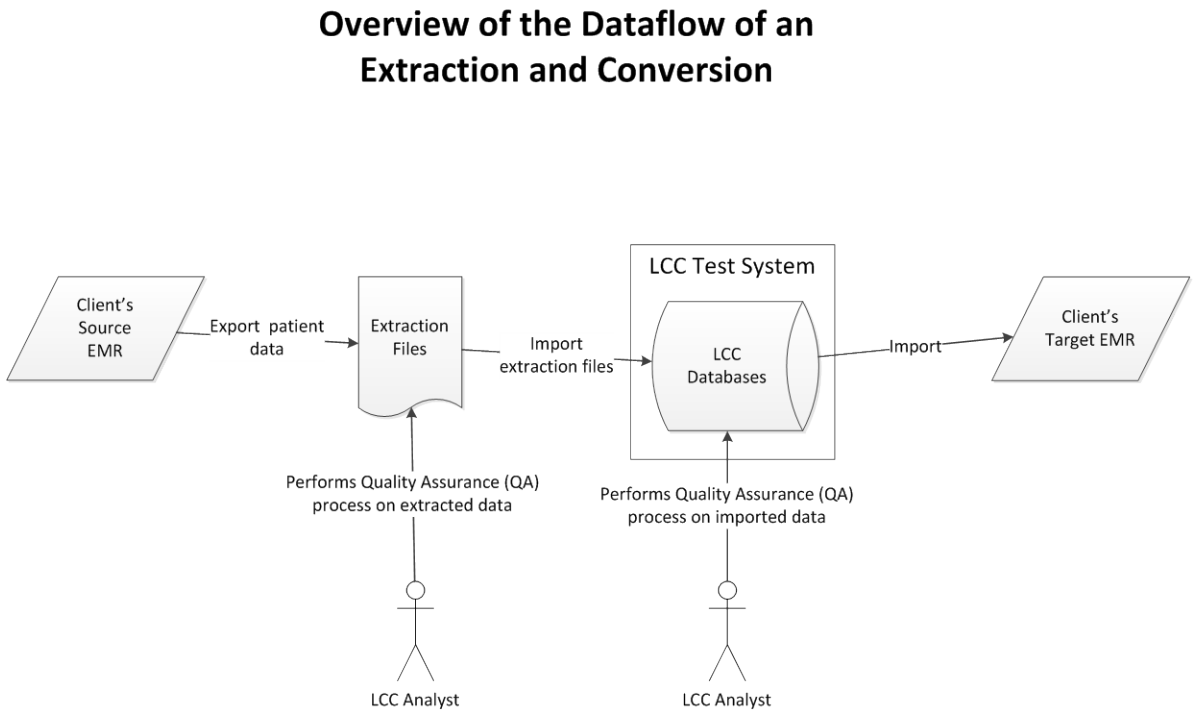
Figure 1: High Level Process View



Data Flow in the Process

A high level view of the flow of the data is shown in the figure below. The exported data is first stored in a set of extraction text files. These files are imported into an LCC test system to allow greater data visibility and better analysis.

Figure 2: High Level Dataflow View



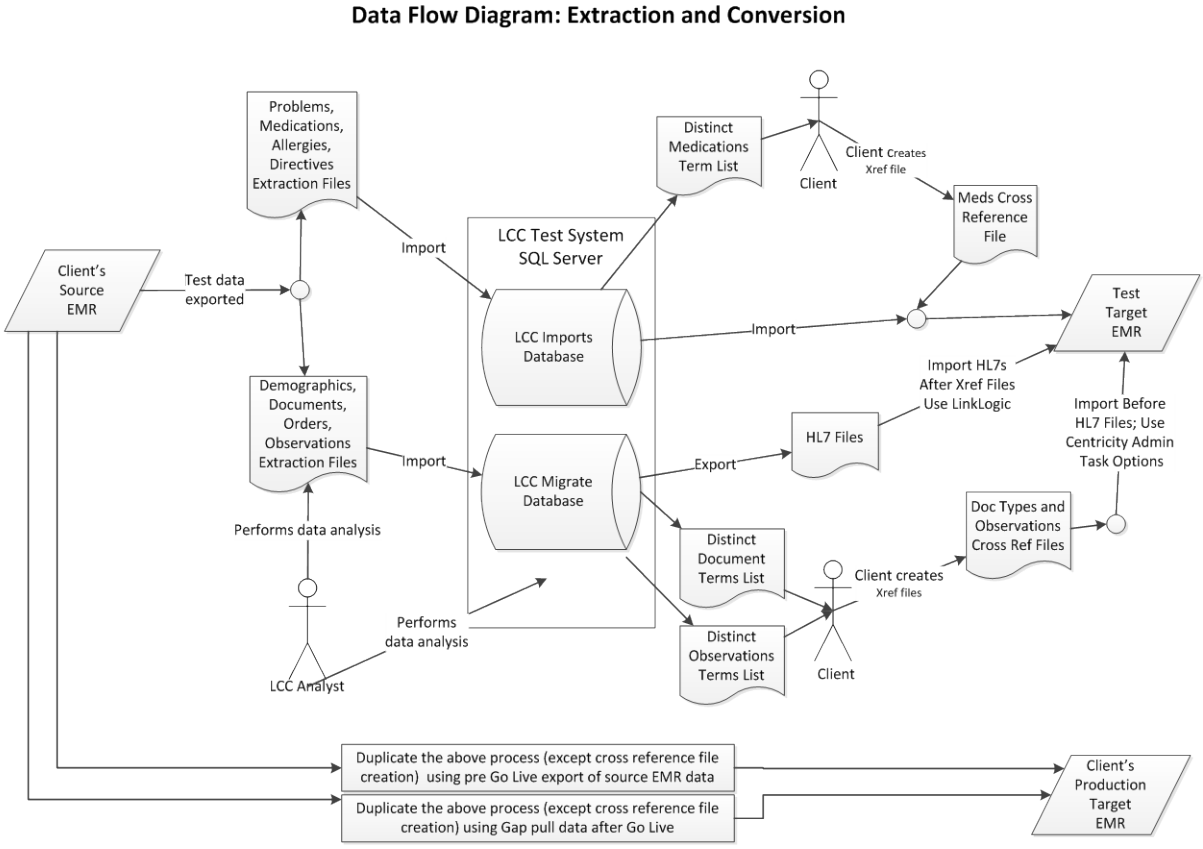
A more detailed view of the data flow is presented in the next figure. The specific types of extractions files are shown. The roles of the LCC databases are represented. The LCC Imports database is used to help convert the problems, medications, allergies and directives. The LCC Migrate database helps with demographics, documents, orders and observations. The Migrate database creates HL7 files containing this information. Medications, document types and observations must be properly mapped to be imported into the target EMR. In addition to the inconsistencies in how the data is stored in the two EMRs, there can be variations in the methods medical providers use to record this kind of patient data. LCC software tools and text files are used to aid in automating this cross reference mapping of data into the target EMR.

As the following table shows, there are actually three extractions from the source EMR. The client verifies the integrity of the data after each extraction is imported into the target EMR. After the test and production extractions a third extraction referred to as the "gap data" is performed. The gap data extraction consists of the data that was added to the source EMR in the time period that passed between the production extraction and when the production EMR goes live. The go live date is the day the new EMR replaces the old EMR.

	Extraction Step	Percentage EMR Data	OK to Continue
1st Extraction	Test data extraction	Perhaps 80+% of final data	Client signs off
2nd Extraction	Production - Test + Gap since 1st	Perhaps 95+% of final data	Client signs off
3rd Extraction	Gap extraction after go live	100 % of EMR data	Client signs off

If the target EMR is Centricity, LinkLogic is used to import the HL7 files.

Figure 3: Detailed Dataflow View



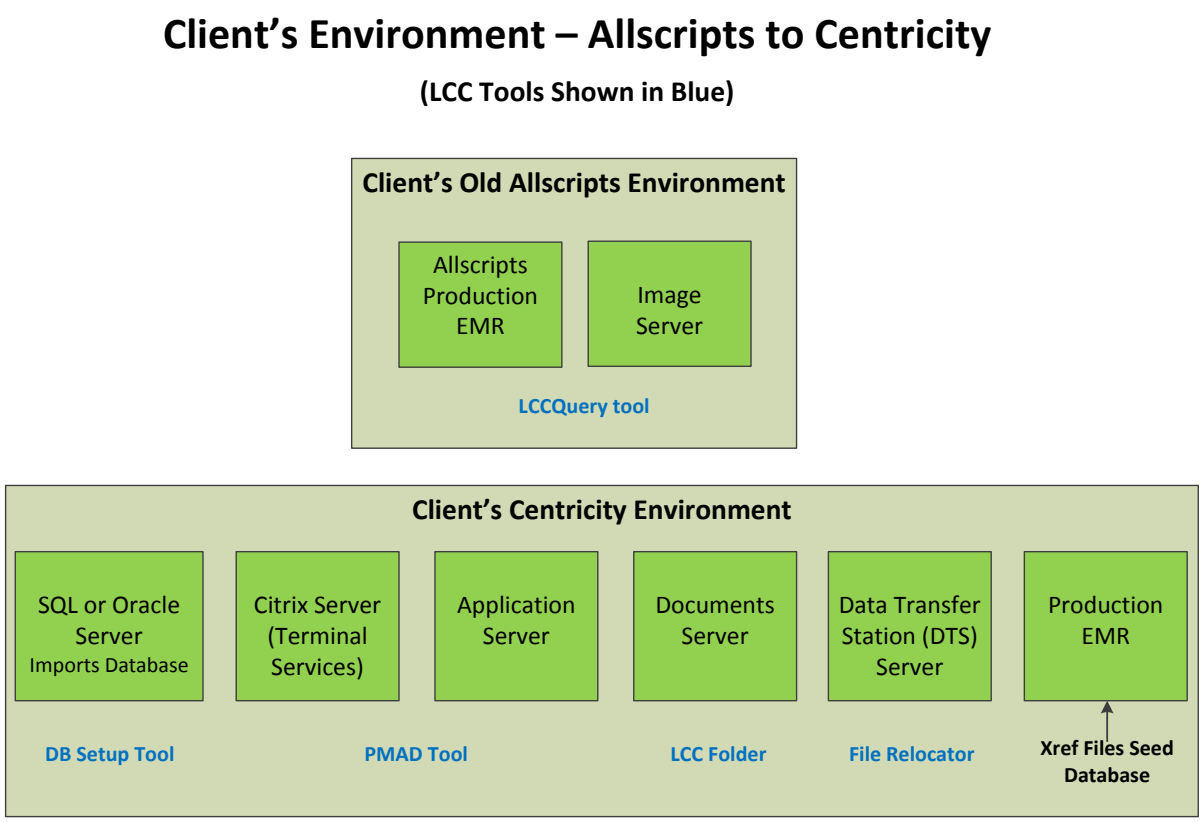
A version of the above diagram that includes what tools are used and when they are used in the process can be found in the appendix of this document.

Detailed Process

The details of an extraction and conversion project are listed below as a step-by-step process. There is also a process map of this same level of detail (done as a swimlane diagram) available in a Microsoft Visio file that mirrors these steps. The process map is not shown here since it is too large to fit into a Word document. Although other EMRs can be the target EMR, this description is specific to projects where the target EMR is Centricity. The steps will vary for other target EMRs.

Understanding the high level view of the client's server environment helps to comprehend the steps listed below. Figure 4 shows what the server environment might be, if the client's data is being migrated from Allscripts to Centricity. LCC tools that are installed in the client's environment are shown in blue.

Figure 4: Client's Server Environment



Process steps:

1. The **Planning Phase** is completed first. For the steps in the planning phase see the document "Extraction Conversion Process Descrip Planning Phase".

2. **Development Phase:** Project Manager provides the client with the LCC Access Requirements document which includes a request the client create the test environment (if not previously done in the planning phase).

3. The client provides remote access to LCC and creates an EMR test environment. Note: The EMR test environment needs to be a full copy of the production environment.

4. If LCC is performing the extraction, client provides access to the source EMR database or a copy of it. If LCC is not performing the extraction skip to step 8.

5. Create a sample set of extractions files by exporting from the client's source EMR database. These are based on the client proposal and may include the problems, medications, allergies,

directives, demographics, documents, orders and observations extraction files. Currently the LCC analyst uses LCC QuickQuery or SQL.

6. Create set of extraction files with all needed data.

7. Client verifies and approves extraction files with all data. (Extraction may be done by LCC or third party.)

8. **Test Phase begins:** The LCC tools are installed in the test environment. These tools include LCC QuickQuery, DBSetup, Cross Reference Editors and PMAD tool. For information on using these tools see the LCC "Tools and Procedures Manual" and "The Client's Guide to Using LCC Tools".

9. The LCC Imports, LCC Config and LCC Migrate databases are created using the DBSetup tool.

10. Load the extraction files into the LCC databases and provide file counts for each element to the client. Load problems, medications, allergies, and directives files into the LCC Imports database (using LCC Imports Loader). Load demographics, documents, orders and observations files into the LCC Migrate database (using LCC Migrate Loader).

11. Convert images using LCC Image to PDF tool if required. Multiple images may be merged into a single document with multiple pages.

12. Install cross references editors for labs and medications on client's machine.

13. Distinct terms lists are created for medications, observations and document types for client mapping. These lists are created by the analyst using data in the Imports and Migrate databases and provided to the client.

14. LCC analyst trains a client representative that has clinical experience to use the cross reference editors.

15. The client uses the cross reference editors and terms lists to create cross reference files.

16. The analyst obtains the completed cross reference files and verifies the mapping is complete.

17. Analyst installs PMAD tool in client test environment.

18. Using HL7 file Writer the analyst creates demographics HL7 file (exporting the data from the LCC Migrate database).

19. Analyst loads demographics per client's specifications into the client's test EMR.

20. The analyst uses PMAD tool to import updated cross reference files into client test imports database.

21. Analyst trains the client to use the PMAD tool and PMAD Encounter form which is used to import problems, medications, allergies and directives into client's target system.

22. Analyst sets up Centricity relationship (MSH 4,5).

23. Analyst imports observation and document XRef files into the test target EMR using the Centricity Admin Task Options.

24. Using the HL7 File Writer tool and LCC Migrate database, the analyst generates HL7 files for documents, orders, observations and images.

25. These HL7 files are imported by the analyst into Centricity using LinkLogic after the cross reference files are imported into the Task Manager.

26. Train the client's LinkLogic manager on loading and reviewing import.

27. The client reviews the data in the test target EMR. If there are problems the analyst corrects them and submits them for client review again, until the client approves the test data concluding the Test Phase.

28. **Execution Phase begins:** The analyst installs and configures the File Relocator and PMAD tools in the client's production environment (see figure 4 above).

29. If LCC performs the extraction, the analyst uses LCC QuickQuery or SQL to export production data.

30. Analyst converts images that are not yet loaded.

31. Analyst uses Imports Loader to move problems, medications, allergies and directives extraction files into the LCC Imports database.

32. Use Migrate Loader to move demographics, documents, orders and observations extraction files into the LCC Migrate database.

33. Install PMAD tool in client production environment. If the client has implemented a Citrix server it is installed there. If not it is installed on all the clients that contain the EMR.

34. Import Xref files to client production imports database.

35. Using HL7 file Writer the analyst creates demographics HL7 file (exporting the data from the LCC Migrate database).

36. Analyst loads the demographics into the client production EMR.

37. Analyst trains the client to use the PMAD tool and PMAD Encounter form if training not completed in test phase.

38. Analyst imports observation and document XRef files into the production (target) EMR using the Centricity Admin Task Options.

39. Using the HL7 File Writer tool and LCC Migrate database, the analyst generates HL7 files for documents, orders, observations and images.

40. These HL7 files are imported by the analyst into Centricity using LinkLogic after the cross reference files are imported into the Task Manager.

41. **Go Live:** Client verifies will go live with production system.

42. **Gap data sub-process begins:** Use LCC QuickQuery or SQL to export gap data, if LCC performs the extraction. The gap data is the EMR data the client added since the last extraction.

43. Analyst converts images that are not yet loaded.

44. Use Imports Loader to move problems, medications, allergies and directives extraction files into the LCC Imports database.

45. Use Migrate Loader to move demographics, documents, orders and observations extraction files into the LCC Migrate database.

46. Update client's production Imports database.

47. Using HL7 file Writer, the analyst creates demographics HL7 file (exporting the data from the LCC Migrate database).

48. Analyst loads the demographics into the client production EMR.

49. Import observation and document XRef files into the production (target) EMR using the Centricity Admin Task Options (if additions to the XRef files have occurred).

50. Using the HL7 File Writer tool and LCC Migrate database, the analyst generates HL7 files for documents, orders, observations and images.

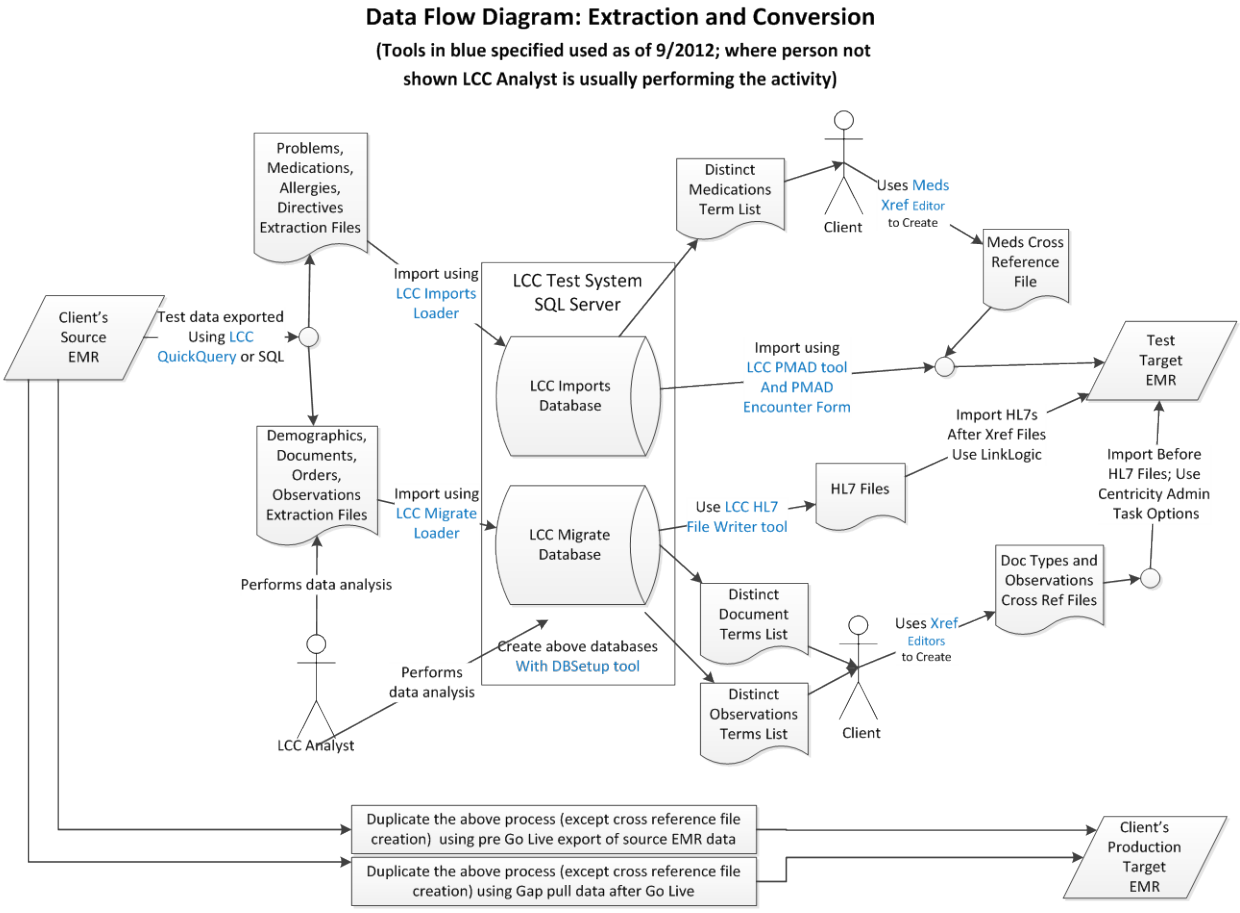
51. The analyst imports final HL7 files of gap data into production EMR (Centricity) using LinkLogic.

52. Client accepts project and sign off that it is completed.

Appendix

The tools specified in the diagram below were those being used in September 2012. LCC's tools to accomplish this process may change over time.

Figure 5: Detailed Dataflow View with Tools



Analysis of Change Control Procedure

Ron Little

Introduction

This analysis is based on the Change Control Procedure used by LCC as of 11/16/2012. This procedure is described in the document "Change Control Procedure as of 11-16-2012". You should refer to it before reading this analysis.

Current Change Control Procedure

LCC's current Change Control Procedure is contains specifics steps to generate a change request. It takes into account the importance of obtaining client buy-off when extra charges need to be assessed due to the client taking the project over budget or when the client requests tasks that are out of scope. However there are several concerns regarding the current Change Control Procedure that may exacerbate problems with projects. Those identified here are:

- Lack of searchable issue and change logs
- Lack of guidelines to determine what issues should initiate a change request
- Scarcity of change requests when the delay is the client's responsibility

Each of these concerns is described in more detail below.

<u>Lack of searchable issue and change logs</u>

The change information is stored only in a Word document that serves as the Project Change Request. This makes it difficult to search for past issues that may be similar. In addition, it limits the information that can be logged regarding the issue that instigated the change request. In many cases it would be important to log information about why the issue was or was not escalated to a change request. For this reason it would be beneficial to create and maintain an issue log for projects. The issue log would track all issues whether or not they became change requests.

A change log could also be created and maintained to easily track what issues become change requests, how they altered the cost and schedule, when a change request was carried out and whether the change request was completed within its estimated cost and duration.

Since issues aren't logged, it is difficult to identify which issues that didn't result in a change order still delayed the project. Thus the reason(s) that a project is late may not be documented. In turn, this may cause clients to believe that unmet due dates and go-live dates are more often LCC's fault than they actually are, harming LCC's reputation.

Lack of guidelines to initiate change requests

There are no formal guidelines to determine what issues should initiate a change request documented. Therefore one analyst may follow a different pattern of initiating requests than another. This is likely to result in inconsistencies regarding the types of issues that are escalated to change requests. For example an analyst may verbally commit to the client to perform work outside of the scope and perform the work without sufficient analysis of how it may delay the project. Without formal guidelines the analyst may feel permitted to determine whether the extra work should be performed by his/her own analysis. This allows for issues to be easily misanalysed and may result in underestimating cost and time requirement increases on a greater number of projects than if formal guidelines were in place.

Scarcity of change requests

Although there are many reasons projects are late, LCC is not submitting change requests on many of them that could be deemed the client's fault. Whether the client or LCC is at fault, the issue can be recorded in an issue log. After that if it is not LCC's fault, some evaluation could be done to determine if a change request should be submitted to the client. Due to LCC's desire to accommodate the client we often don't submit charges on extra work performed for the client.

The decision to accommodate the client should be built into the change request process. If the determination is to not generate a change request for an issue created by the client, the reasons for doing so would be logged in the issue log for future reference and analysis. If it is LCC's fault some analysis could be done to determine how to reduce the chances of the issue and/or its impact in the future. The results of the analysis could be recorded in the issue log so that our solutions as well as our problems are tracked.

There are a number of reasons why projects are delayed that may be out of our control. Some of the reasons LCC could consider originating change requests for include but are not limited to:

- Client delayed responding or performing their tasks
- There were unforeseen issues with the performance of the client's hardware, software or network
- Client misunderstood our deliverables and pushed for us to do more and we accepted without a change order
- Client wanted something outside of the original scope and we agreed to perform it without a change order
- Third parties change procedures and processes without involving LCC and our client; leaving the client with no recourse but to hope we'll be accommodating

The concerns discussed above and the inconsistency of the process underscore that the current Change Control Procedure could be improved. A possible alternative Change Control Procedure

is presented next. The following is meant to be a procedure to critique and modify by the LCC team if it is determined that the problems specified above should be addressed. For a comparison refer to the document "Change Control Procedure as of 11-16-2012" which is the current procedure I suggest modifying.

Modified Change Control Procedure

I recommend that the following Change Control Procedure be evaluated and considered for adoption by LCC. I believe it constitutes an improvement over the current procedure. An evaluation of this suggested procedure by the LCC team may produce changes to it that result in even more improvement.

1. Any and all problems or issues identified by the client or an LCC employee are recorded in the Issue Log. Some of the Issue Log fields are filled in immediately and others are filled in later. The Issue Log design and its fields could be determined by the LCC team. To provide for a place to start the discussion of its design, I've provided some suggestions here. The log could be an Excel spreadsheet or part of a knowledge database. Information recorded could include:
    - Description of the issue
    - Date issue identified
    - Who identified the issue
    - Description of the cause of the issue
    - Who caused the issue (Client or LCC)
    - Whether or not the issue was escalated to a change request
    - Date decision to escalate or not was made
    - Description of how the issue will influence the project
    - Amount of estimated additional time the issue will require
    - Specifics used to determine if the issue should escalate to a change request
    - Whether or not the issue was corrected
    - Date it was corrected
    - Information on how it was corrected if it was
    - Amount of time the issue actually required

2. All LCC project stakeholders are informed of the issue.

3. The project manager and analyst(s) determine if the issue could increase project time.

4. If it is determined that the issue may increase project time, an analyst performs an evaluation to identify any extra time required for it. In some cases programmer support will be necessary. In this case the analyst meets with the programmer to obtain the estimated time for any programming which is added to any extra time that may be required by the analyst(s).

5. The project manager and analyst(s) determine if the issue should be escalated to a change request based on a set of guidelines. These guidelines would be created by the LCC team as part of adopting a new Change Control Procedure. Some of the guidelines could be
    - If the issue is not LCC's fault and could increase project time
    - If a request by the client is outside the scope of the project (Even if it may not push out due dates or increase client charges, it could be processed as a change request so that it is readily apparent the extra LCC does to satisfy the client)
    - If the issue is LCC's fault and will cause a project delay, this should be communicated to the client. If the issue is not escalated to a change request, a report including new project schedule could be generated and sent to the client.

6. The appropriate information is added to the Issue Log.

7. If the decision is to escalate the issue to a change request, the change request is logged in the Change Log. The design and fields of this log would be determined by the LCC team. It could be an Excel spreadsheet or part of the same knowledge database as the Issue Log. The fields could be the same as those in the current Project Change Request plus a few addition fields such as the date signed by the client and the date signed by LCC.

8. The Project Change Request is generated as a report from Excel or the database system.

9. If due dates will be altered the project manager modifies the project schedule.

10. The Project Change Request and modified project schedule are reviewed by the operations manager and then David. If there are no modifications required, the project schedule and change request are made available to the client for signature.

11. Once the signed change request is returned, the operations manager, project manager or analyst sign for LCC.

12. A copy signed by the client and an LCC employee is sent to the client.

13. If there are new tasks required by the change request, they are entered into ActiveCollab.

14. Once any tasks have been performed to address the issue or change request, the appropriated information is entered in the Issue Log.

15. All LCC stakeholders periodically review the Issue Log and Change Log.

16. The Issue Log and Change Log are periodically analyzed to evaluate repeating issues, unresolved issues, whether the issue required more time than projected, what issues cause project delays, projects that are completed even later than the altered go-live, etc. Excel or database report(s) could be created to assist with this.