An Enterprise Architecture Approach to Application Selection and Infrastructure Improvement at a Small Enterprise

Ronald King

Grand Valley State University

Follow this and additional works at: http://scholarworks.gvsu.edu/cistechlib

Recommended Citation

http://scholarworks.gvsu.edu/cistechlib/196
An Enterprise Architecture Approach to Application Selection and Infrastructure Improvement at a Small Enterprise

By
Ronald King
December, 2014
An Enterprise Architecture Approach to Application Selection and Infrastructure Improvement at a Small Enterprise

By
Ronald King

A project submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Information Systems

at
Grand Valley State University
December, 2014
# Table of Contents

Abstract .................................................................................................................................................. 4
Introduction ........................................................................................................................................ 4

Methods/Procedures ......................................................................................................................... 5
  Architecture Development Method ................................................................................................. 7
  Architecture Vision .......................................................................................................................... 8
  Business Architecture ..................................................................................................................... 8
  Information and Technology Architecture ....................................................................................... 9
  Opportunities and Solutions ........................................................................................................... 11
  Migration Planning .......................................................................................................................... 12
  Implementation Governance ........................................................................................................... 13
  Change Management ....................................................................................................................... 13

Results .................................................................................................................................................. 13

Conclusion/Discussion ......................................................................................................................... 17

Bibliography ....................................................................................................................................... 19

Appendix A: AR System Requirements ............................................................................................... 20

Appendix B: AR Process ...................................................................................................................... 22

Appendix C: Migration Plan .................................................................................................................. 22
Abstract
Dominion Systems’ current account receivables (AR) system is out dated and unable to be maintained. The process that has resulted from the evolution of the AR system has left the company working for the application rather than the application working for the company. To address the AR process issues and help decide the next stage of their AR application, Dominion is going to introduce an Enterprise Architecture (EA). By using The Open Group Architecture Framework to guide them through this transition, Dominion has not only created a new application but has put in place a foundation for the documentation of all of their processes and infrastructure. This paper describes the start of Dominion’s EA journey that led to the development of their new AR system.

Introduction
Dominion Systems, Inc. has is a Grand Rapids based software company that provides cloud-based payroll, labor, and human resource services to other businesses. Dominion has been in business for forty years and has seen many applications and products come and go. Currently, their accounts receivable system is a home-grown custom application that has been in service for the last twenty years. Just as the business has evolved, the company has decided it’s now time for their accounts receivable system to do the same.

Dominion has been SOC Type II compliant for a number of years, and has been ISO certified in the past. They have many documented processes that are scrutinized by the various auditors, but this only covers a portion of the operations regarding their product and service offerings. As a result, this has left some other internal processes neglected. In addition, another issue is these process documents offer the information in ways that may be difficult for employees to learn from.

An Enterprise Architecture (EA) takes these artifacts plus any other valuable artifacts and organizes them in ways that they’re easily accessible throughout the entire organization. EA as practice has been around for twenty-five years. (Sessions, 2007) The goal of an EA is to produce a framework that will assist in creating a current, midrange, and future point of reference for changes within the organization. (Technology Training Limited) It helps a company break down individual department silos, and promote a more singular solid unit across the entire organization.

This can be a major benefit to an organization, but there is a challenge to incorporating this type of system. The initial cost to the company of documenting and cataloging all their processes can be too great. Even after the system has been started, the contents will need be maintained and updated as items change over time. This creates more overhead on future projects. To get the highest return on investment organizations need to focus on smaller changes that deal with current proposed changes. (Bloomberg, 2014) This allows the company to remain flexible, while optimizing the benefits in the shortest amount time, instead of constantly striving for completion.
To evaluate the benefits of having an EA, Dominion is going to start with the documentation of their AR system project. They will need to decide which Enterprise Architecture methodology will provide the most value to the company. Then Dominion can use their chosen EA methodology to assist in determining if creating their own AR application will provide more benefits to Dominion than purchasing an “off the shelf” application.

Methods/Procedures

There are many different standards in the Enterprise Architecture world. The four biggest methodologies in practice today are the Zachman Framework, The Open Group Architecture Framework (TOGAF), Federal Enterprise Architecture, and the Gartner Methodology. (Sessions, 2007) Each of these approaches has their own set of advantages and disadvantages.

The Zachman Framework focuses on creating artifacts for the different players within the organization. (Zachman, 2008) Each person in the organization has a different perspective on each step of the process. The framework will create artifacts focusing on function, data, time, people, networking and motivation, Each of these areas focused on from the view point of the following people; owner, planner, designer, builder, subcontractor and the business as a whole. (Zachman, 2008) This leads to an understanding that everyone is heard and solutions can be developed with everyone’s voice in mind.

Though everyone’s perspective is accounted for, this framework doesn’t help in creating the architecture. It simply provides a way to organize but not how to implement the changing architecture.

Figure 1:  

ENTREPRISE ARCHITECTURE - A FRAMEWORK™

The Federal Enterprise Architecture (FEA) is the EA developed and used by the United States Government. Its goal is to unite all of its agencies by creating a common language around a set of core areas. (The Common Approach to Federal Enterprise Architecture, 2012) By having all of the agencies using the same terminology promotes cross communication allowing agencies to reuse and repurpose pieces of technology. This framework is too robust for the needs of Dominion. The framework has been designed for use by large, multi-agency organizations such as a state government. Dominion is a small company without the need for divisional boundaries. They do not have multiple divisions creating and purchasing their own applications. Dominion needs a more scalable, and simple EA framework.

Figure 2:

Gartner developed their own approach to EA for their customers. Gartner bases their EA around three groups; owners, specialists, and implementers. When these three groups of people are on the same page it increases business value along with profits. (Sessions, 2007) The main problem with the Gartner methodology is that the majority of documents outlining the methodology are only available in the Gartner Library, which is only accessed by Gartner customers beyond a paywall. Also, Gartner only defines their success by how profitable the artifact or technology is to the company. With Dominion trying to evaluate the usefulness of having an EA, purchasing Gartner’s services are not in their budget at this time.

This leaves TOGAF as the methodology Dominion has chosen at this time. It provides the most flexibility to the company and allows for, even encourages a starting scope limited to the needs of the company. In the Dominion case, the scope of the EA effort will focus on the need for a new AR system.
There is a plethora of supporting resources available that are cost effective and it is a highly business driven option. The Open Group also provides assistance in creating the artifacts needed for this framework. These factors have made TOGAF a much better fit for Dominion.

TOGAF breaks down the architecture documents into four types of architecture. Artifacts can be classified as business, application, data, or technical architecture. (Session) The business architecture outlines the business processes, procedures and strategies it uses to conduct its operations. The application architecture breaks down what applications the organization utilizes and how they communicate with one another. The data architecture houses the documentation on where the company’s data is stored, how it’s accessed, and who is authorized to view it. Finally, the technical architecture stores information about the infrastructure, hardware and software used to make these applications run. (Session)

**Architecture Development Method**

The main focus of TOGAF is the Architecture Development Method referred to as ADM. (Session, 2007) ADM will help guide you through each of the architectures and provide the steps to developing a solid foundation, allowing the business to make informed decisions to meet its goals. ADM in its most complete form can be broken down into nine phases. (Session) The phases of the ADM are shown in Figure 3.

![Figure 3](Session, 2007)

The key benefit of Dominion using the ADM process is that it is easily customizable. These phases can be removed, reordered, and/or have any additional phases the company feels necessary to be introduced into the process. Each company has a unique culture and history so it is important the methodology adapts to its environment and works for the company, not against it. For this first attempt at creating these architecture documents Dominion is going to follow Session’s phases (Figure 3) starting with the vision.
**Architecture Vision**

The first phase of this methodology is the Architecture Vision. This whole process starts with a request for change. This request can be generated from any area within the business. Dominion management has previously indicated that their older applications need to be retired, and replaced to stay competitive in the industry. As Dominion continues to grow their AR application is struggling to meet the demand of their operations, and is easily the next candidate to be upgraded.

**Business Architecture**

Dominion put together a project team consisting of the CFO, office manager, and the employee responsible for entering payments into the system. They quickly discovered there were no documents outlining Dominion’s current AR process, and it heavily relied on knowledge transfer from employee to employee. Since this is the beginning of the EA effort there is a need to create these artifacts and add them to a catalog. That way they can be referenced and maintained for the future. More importantly, these documents are needed to properly define the requirements for the replacement AR system. In the early meetings the CFO described invoice-based accounting and how Dominion has applied these principals. He walked through the different reports that can be produced from the AR system and how they use them to handle bookkeeping. From this discussion it was stressed that each payment must apply to an invoice, they walked through the aging reports, deposit reports, how the bank account is reconciled, and how these transactions affect the general ledger. A complete list of business requirements gathered in this meeting is listed in Appendix A.

Over the next couple of mornings employees were shadowed as they completed their daily AR tasks. After reviewing the employees’ steps a workflow diagram of their daily activities was created. In addition to the diagram, limitations caused by the current platform were documented so they could be addressed in any proposed solution. It was also noted that there is a set of instructions the employees’ use that contain steps on how certain items need to be entered into the system not discussed in any of the previous conversations. These instructions also highlight some common issues that the employees previously encountered and their solution to the problem. These sorts of things can be devastating to a company, after any system or personnel change over these instructions can be lost if not properly maintained. This could potentially lead to major problems in the AR process if this were to occur. Having an EA in place will reduce this risk and provide a way to manage these types of items. The work flow diagram documenting the processes is attached as Appendix B.

Once Dominion defined what the business needs from a new AR system, their meetings started to focus on alternative applications to satisfy the company’s needs. The team created a set of criteria and graded some off the shelf, cloud-based products that met Dominion’s criteria. The list of products was
narrowed down to one application, but before a decision was made they wanted to compare it to a Dominion built solution which is discussed later.

**Information and Technology Architecture**

With the business requirements documented, it was now time to turn towards the next ADM phases: documenting the Information and Technology Architecture. For this project Dominion decided to combine these two phases into one. At larger companies these maybe split among different groups and leadership within the IT organization, each of which has their own schedules. Dominion’s small size does not allow for such a splitting of responsibilities.

The first item to tackle as part of ADM’s Technology Architecture was to document Dominion’s current infrastructure. To properly recommend a solution, details about each of the servers and their interconnections needs to gathered so an accurate solution can be presented to the company. This task proved to be more difficult than it should have been. Historically, all of Dominion’s IT needs were performed by developers, forcing them to be jacks of all trades. This practice led Dominion to focus on its products they offered and left some internal IT affairs neglected. As Dominion grew, this model changed approximately three years ago to have the developers develop applications, and added network engineers to handle managing and configuring the hardware and software infrastructure.

Dominion upgraded their infrastructure (networks, servers, storage devices) quickly over the last few years. There were changes being made and no one took the time to keep the documentation of these systems up to date. So, when turning to the network staff for help documenting the EA, most of the artifacts proved to be out of date and portrayed an inaccurate depiction of Dominion’s infrastructure. They also had very limited documentation on systems in existence prior to three years ago.

After getting the server information, meetings were held with each department head to make sure there was an accurate record of each application the department used. Using this list, a network diagram of the servers, and the functions each server performs was created (Figure 4). The diagram also describes each of the applications that Dominion uses, on which server the application resides, or if it is a cloud-based product utilized by Dominion.
While creating these EA artifacts, they found that the current AR system was programmed using Visual Fox Pro. The program and data are hosted on Dominion’s file server where it can be accessed by any Dominion employee. The majority of employees only have access to its reporting features, while a few employees have the rights to add payments, credits, and manual invoices. This application has invoices imported into it from Dominion Source, Dominion’s cloud based payroll and human resource product. While a separate import from Dominion’s Client Maintenance application maintains the client information accessed by the AR application.

Client Maintenance is another older Dominion Application. It was used as the central client repository when Dominion had many desktop applications. As clients were added onto one of the payroll systems, information was transferred into this application and then distributed to Dominion’s internal applications. By doing this EA approach Dominion was able to determine there is only one other application besides the AR application that still uses Client Maintenance, the Funds Distribution System.
This provides another benefit having an Enterprise Architecture in place. It has identified inefficiencies that are forcing employees to export client information from Dominion Source into Client Maintenance just so it can be transferred into FDS and AR. This entire Client Maintenance application can be phased out by creating the client export directly from Dominion or by upgrading their FDS application into a future project.

**Opportunities and Solutions**

If Dominion decides to incorporate the off the shelf solution much of the IT process will be minimized. The application will be hosted in the cloud, so there is no hardware burden on the company. The major obstacle of this purchase would be data integrity. Dominion Source will need to be the master database for the new system. All changes will need to be made within the product and then exported and imported into the proposed AR system. This product does not offer an API so Dominion will need to create an import into the new system for client information, and invoices from Dominion Source making it not a real time based solution.

If Dominion chooses the path of recreating an updated solution to their AR program more decisions have to be made. With this choice they need to address programming, reports, a location to run the application, data storage, and any connection that will need to be created between machines for data movement. The EA effort described above provides the information needed to make those decisions. The logical choice for the application is to run it on the environment that houses the internal version of Dominion’s product. All of Dominion’s employees have a user set up on the server that can easily be set up to control access to the various functions within the AR application. This will lead to starting EA security artifacts for their internal system.

Choosing to create the application in house will make data issues easier to deal with. A connection controlled by the internal application could be created to access the billing and client information stored in the client environment. This will save the need for any importing of data, and will provide for real time AR processing. New database structures will need to be made for the AR payment data within the current enterprise data model. Reports will need to be created to provide the information in formats that are familiar to the users of the system. All of this can be done in a manner consistent with the current architecture and tailored to Dominion’s needs, avoiding the inconvenience and risk associated with a commercial AR application.

After IT compared the custom application and the off the shelf application the results were supplied back to the project group and a meeting was called with the project team and the CEO to supply them with a recommendation. First, they reviewed the notes and documents taken from shadowing people as they entered in payments into to the current AR system. This topic was quite eye opening for the group. They hadn’t realized how much of a manual and time consuming process this is for the employees. This became the focal point of the remainder of the meeting as the team went over the two proposed solutions.
The next item on the agenda was to evaluate the proposed solutions. First, they discussed the “off the shelf” application. They walked through a trial of the application while comparing its features to items in the AR process. They talked over the items Dominion would have change in their process, and what it would cost them as company to deploy this solution. They took into account the cost of all the user licenses that would be needed, the labor to get the imports up and running and any other changes for service items. This demonstrates another of the benefits an EA repository provides: a single point of accessing the architecture documentation needed to perform such an analysis in one sitting.

The cost of this solution came in over the budget Dominion was initially planning on spending, especially after factoring all of the reoccurring monthly costs of the product. From here they brainstormed on what things could be changed internally to make this more appealing. An idea was proposed to add the majority of the reporting to the customer system, and make it accessible to their employees that way. This would eliminate all but four of the licenses Dominion would be required to purchase for the commercial application, but add an increase in the amount of labor to get the systems up and running smoothly.

Next the team dove into the evaluation of a Dominion-built application. This option took a little more time to get up and running, but also provided the company with the most flexibility for process automation. This option also had the highest short term cost, but once that initial investment was made, there would be limited additional costs incurred only when the business decided to make changes to the AR application. The plan for this custom built application is to remake the existing feature set as a base line. Then prioritize and add in additional features as a long-term implementation project.

It seemed that coming into the meeting most individuals were thinking the off the shelf application was going to be the clear-cut favorite. That idea was a mere memory by the end of the meeting. The “off the shelf” application while providing more features, does not fit into the current EA and will not help to automate and improve Dominion’s AR process. Dominion would spend more effort accommodating the system rather than making the process as automated and efficient as possible. The proper choice turned out be an internally developed AR application that fits into the Dominion EA, sets Dominion up for the future, and provides a real time Accounts Receivable solution.

**Migration Planning**

Now that Dominion has chosen a replacement for their AR system. A migration plan needs to be drawn up and added into the EA so that the company can have a smooth transition into the new system. The plan will cover when and how the code will be moved. It will include a strategy for validation that the system is working, and producing the same output as the old system. Refer to Appendix B for the full Migration plan for the activation of the new AR System.
**Implementation Governance**

With Dominion choosing to program their own solution, the TOGAF ADM implementation governance phase for the AR project is going to be subject to Dominion’s development process. Each item will be programmed by a developer then passed to the Quality Assurance (QA) department. When everything is confirmed to be working as requested it is passed and ready to be deployed.

**Change Management**

The last phase of our ADM cycle is change management. Dominion already has a change management system in place called FogBugz. A project named AR has been added into that system so each request can be categorized and grouped together. Also being added into system is a new type of change, process improvement. This type of change would be used for changes in process. If process improvement requires a system change, a separate case can be create as a sub case to accompany this request. Currently any user is allowed to request a change. These changes must then be approved by their manager. All changes requiring a programming change or EA change are then prioritized and scheduled to a milestone by an approval committee.

Walking through the Architecture Development Method helped Dominion make the decision on what route to take in upgrading this application, but this is only the first step of a TOGAF Enterprise Architecture. Creating these artifacts won’t solve anything if they’re not accessed and organized in a beneficial way to the organization. The TOGAF framework uses what it calls the “Enterprise Continuum” to combat this problem. (TheOpenGroup) This continuum includes a central repository that stores all of the business and IT documents providing the foundation for future endeavors. (TheOpenGroup) As the repository is updated records of the changes will need to be kept for reference of historical changes.

**Results**

To start the Dominion’s Enterprise Continuum I’m going to track these documents using an Excel Workbook. Document names, the dates that each were last modified, and links to each of the corresponding artifacts will be maintained to provide a central location to find information about the business. Different worksheets in the workbook will be created for each process and application, grouping these related items together.

Already through this project there are many candidates to be added into our repository. There is a work flow for the AR process, a network diagram of our IT infrastructure, a listing of the current applications Dominion uses, and a foundation with the Architecture Development Method to base future changes on. Dominion has documents for their change management process and new code deployment process that also need to be cataloged into the new Enterprise Architecture.
Creating the new AR application will allow Dominion to fully document this section of the architecture. To create the new application, Dominion will have a work breakdown structure, data models to support the application, and detailed help text to guide employees through the new system. Keeping these artifacts in the EA will provide Dominion with the necessary tools to make informed decisions about any further changes to their AR architecture.

The entities needed for the application consist of deposits, payments, credits, invoices, manual invoices, invoice detail, and billing items. Dominion Source already automatically creates invoices each time a client processes their payroll. This will serve as a model for the manual invoices. Each invoice contains a list of billing items and an amount associated with each one. Some invoices or pieces of invoices are written off for various reasons, and in these cases a credit needs to be applied to the invoice. The majority of invoices have a payment applied to them. Each of the payments needs to belong to a deposit. Later these deposits get reconciled with Dominion’s bank records. Figure 5 depicts the relationships between these items in an Entity Relationship Diagram.

**Figure 5:**

From the project requirements and the ERD, the work breakdown structure shown in Figure 6 was created with the tasks needed to complete the project. Tasks are broken down per screen and function into manageable pieces. They are then entered into Dominion’s change management system FogBugz for tracking. This system is used by Dominion’s developers to track the stages of their tasks from development, into testing, and it continues until the case has been verified working in a production environment. The parent case for this project has been entered into the architecture so that these individual steps can be referred to in the future.
The AR functionality was incorporated into the administrator section of the application. The final product consists of four new screens added into the system. The first screen in figure 7 allows the user to open and close deposits. As deposits are opened they’re added to the list of open deposits. This list provides a way to add payments into the deposit, view the list of payments assigned to that deposit, and close the deposit when it’s ready to be posted.

When the user chooses to add a payment to an open deposit they’re navigated to the payment screen shown in Figure 8. The user is first prompted to enter the client code that the payment is for and click the “Get Invoices” button. This event populates the screen with all of the open invoices belonging to that client. The user then will need to enter the payment amount and check number into the provided fields. If this payment fulfills or exceeds the cost of an invoice the user checks the “Is Paid” check box marking the invoice as closed. When the payment each payment is saved and the deposit total in Figure 7 is updated to reflect the new total.
To apply a credit to an open invoice the user must navigate to the page shown in Figure 7. There is a gray button in the lower left corner to “Apply Credit”. Once the button is clicked it will navigate you to Figure 8, but this time no deposit is selected to apply the payment to. Any payment recorded with this method is saved into the system as a credit.

If the user navigates to the “Manual Invoice” menu item there are presented with Figure 9. This screen allows the user to create an invoice and apply it to the chosen client. A list of all the available billing items is provided to the user. For every item they want to add an amount will need to be entered and “Add Item” clicked. A list is then populated below showing the user all of the items they have selected. When finished and user selects “Create Invoice” it is then available to have payments recorded against it, and appears in the payment list shown in Figure 8.

**Figure 8:**

**Figure 9:**
All of the reporting items can be found under the “AR Reporting” menu item. Each report is added into the dropdown list at the top of the page. After the user selects which report they want to run the screen adds rows to the table prompting the user to supply the parameters their chosen report requires. After values are supplied the user selects “Create Report” and depending of if the user is just viewing or saving the file, the report is downloaded or displayed to the user. Figure 10 demonstrates the AR Reporting page.

**Figure 10:**

![AR Reports](image)

This application satisfies Dominion’s immediate needs for their AR Application, but is getting paving the way for automation. One example being is that in the future this system will be able to record payments automatically for clients either paying their invoice electronically or by a Dominion printed check. This new offering already saves time for their AR process. It allows more than one user can be in the system at a time entering payments, removes the need for employees to export and import invoices, and provides a significant gain in report performance.

**Conclusion/Discussion**

Creating an Enterprise Architecture is not an easy undertaking. It will take discipline and dedication to expand and maintain the Enterprise Continuum, but the benefit a company can gain from deploying it can be fruitful. In Dominion’s case they have already started to see gains in other areas besides just accounts receivables.

During the span of the AR upgrade Dominion has also begun the process of changing hosting providers for their production and disaster recovery sites. The network diagram cataloged with this architecture provided the network team with the machines and the services that they need to have installed at the new location. Having these structures organized and available reduce risk. Lowering the likelihood of items being overlooked, and the work can be estimated more accurately than if he EA did not exist.
The start of the EA also has led to the organization of documents between the installations and client services department. While interviewing the departments for their applications it was found that each department has separate folders and documents for many of their processes. Often these items were hard to find and some of the documents that were supposed to describe the same process were different. When mentioning to the manager an Enterprise Architecture has been started and can be used to help this cross departmental separation, they started to add in documents per process. This gets each department looking in one area for the information and removes the problems associated with the department silo effect. This can lead to process improvement by unifying each of their separate interpretations and lead to a more overall productive opportunity for Dominion to provide their employees with consistent information regardless of their department.

To fully implement an Enterprise Architecture will take some time. It will require not only procedural changes but cultural changes too. As more items get added, a critical point will arise where the methodology can be perceived as a burden rather than a benefit. For the EA to survive and thrive it will need to become the norm for the organization. It takes a constant effort to maintain changes and continue to document items into the architecture. Viewing the EA as a unit to drive business solutions well help the employees and the executives pushing forward in the EA. (Bloomberg, 2014) The development world has been embracing the Agile development model, and applying this type of methodology to EA helps to breakdown the EA artifacts and the processes they govern more valuable to the company. (Bloomberg, 2014)

Employees already work together for the common goal of the organization, so having architecture in place to provide them with the information they need from a unified source will promote common knowledge. For now Dominion will continue to explore and add to their architecture. It will be a gradual process to get a complete Enterprise Architecture, but as the company continues to grow and change items will be added into the continuum. Eventually, another more sophisticated mechanism for the document management will arise. When that time comes Dominion will have an architecture that works with them to determine when the change will occur and assist in making the decision.
Bibliography


Appendix A: AR System Requirements

1. Invoices
   a. Track which invoices have been paid and which are outstanding.
   b. Invoices are automatically generated when a client processes payroll.
      i. Invoices totaling 0$ or less can be marked paid by default.
   c. An invoice can be manually generated by authorized users.

2. Payments
   a. Every payment entered must correspond to an invoice.
   b. Payments can be made by check or direct deposit bank transaction.
   c. Clients sending a check for multiple invoices must be entered for each invoice.
   d. The amount of the payment entered cannot exceed the invoice total.

3. Deposits
   a. Deposits are to be made for each bank transaction.
   b. Every payment must be assigned to a deposit.
   c. Deposits need to have a running total of each payment applied to it.
   d. All checks are to be entered as one weekly deposit.
   e. Each direct deposit transaction will be entered as a separate deposit.

4. Credits
   a. A credit can be applied to any voice.
   b. Can be partial or full amount of an invoice.
   c. If full amount can make invoice as paid.
   d. Credits do not get included in any deposit figures.

5. Security
   a. Only system admins can have access to the AR functionality.
   b. 3 levels of access for system admins
      i. Read Only - Reporting (any system admin)
      ii. Partial – Create payments and deposits in addition to above
      iii. Full – Can create credits and manual invoices in addition to above

6. Reports
   a. Reports can be run by day or a date range.
   b. Aging Report
      i. Shows invoices outstanding invoices over the last 90 days starting from the date selected
ii. Also shows payments that were over invoice total.
c. Deposit Ledger
   i. Shows all deposits per day in the date range and their total.
d. Payment Ledger
   i. Create by client or client group
   ii. Shows payments collected per grouping during the date range.
e. Billing Item Total
   i. Show all line items or an individual item.
   ii. Display total collected under each billing line item during the date range.
f. Credit Listing
   i. Lists all credits added over a date range.
Appendix B: AR Process
Appendix C: Migration Plan

1. Deployment process will be governed by Dominion Release process.
   a. Tuesday new code changes will be merged and deployed to staging server
   b. Code will be tested by developer and QA
   c. Thursday after 6:30 pm, code will be deployed to production server and tested.

2. Apply config changes to the Dominion Server to access client and billing information.

3. Starting on a Friday after Deposits have been posted.
   a. Mark all invoices as paid through the preceding Thursday.
   b. Run aging report from Fox Pro system to ID clients with outstanding invoices.
   c. Update outstanding invoices to show they are not paid.

4. Starting the following Monday.
   a. Run systems in parallel for at least one week.
   b. Have employee responsible for entering payments alternate with backup entering payments in each system Monday – Thursday.
   c. Friday enter any remaining payments into each system and post deposits.
   d. Each day compare results of payment and deposit logs verifying new system is working correctly.
   e. Compare aging report to ensure the amount of outstanding invoices is the same across both systems
   f. Repeat additional weeks as required.

5. Any changes after deployment will be addressed though standard change management process governed through FogBugz.

6. Archive data from legacy AR application for historical reference.