Regis University

ePublications at Regis University

Regis University Student Publications (comprehensive collection)

Regis University Student Publications

Summer 2006

# The Automation of Obtaining Customer Billing Data

Jonathan Seashore
*Regis University*

Follow this and additional works at: https://epublications.regis.edu/theses

Part of the Computer Sciences Commons

## Recommended Citation

Seashore, Jonathan, "The Automation of Obtaining Customer Billing Data" (2006). *Regis University Student Publications (comprehensive collection)*. 329.
https://epublications.regis.edu/theses/329

This Thesis - Open Access is brought to you for free and open access by the Regis University Student Publications at ePublications at Regis University. It has been accepted for inclusion in Regis University Student Publications (comprehensive collection) by an authorized administrator of ePublications at Regis University. For more information, please contact epublications@regis.edu.

**Regis University**
School for Professional Studies Graduate Programs
**Final Project/Thesis**

# Disclaimer

Use of the materials available in the Regis University Thesis Collection ("Collection") is limited and restricted to those users who agree to comply with the following terms of use. Regis University reserves the right to deny access to the Collection to any person who violates these terms of use or who seeks to or does alter, avoid or supersede the functional conditions, restrictions and limitations of the Collection.

The site may be used only for lawful purposes. The user is solely responsible for knowing and adhering to any and all applicable laws, rules, and regulations relating or pertaining to use of the Collection.

All content in this Collection is owned by and subject to the exclusive control of Regis University and the authors of the materials. It is available only for research purposes and may not be used in violation of copyright laws or for unlawful purposes. The materials may not be downloaded in whole or in part without permission of the copyright holder or as otherwise authorized in the "fair use" standards of the U.S. copyright laws and regulations.

## Abstract

XYZ Telecom received bi-monthly customer PDF billing data (identical to the paper customer invoices) from the print vendor Gilmore, Inc via CD media shipped by FedEx. Upon receipt, this data would then be posted to the CD/DVD media server through a manual entry process. The Billing Disputes Team often required immediate access to this data to deal with customer inquiries for telephone dispute calls regarding current or previous billing periods. Through the use of a dedicated VPN connection, VBScript and Windows Server technology an automated solution was put into place which ultimately saved XYZ Telecom approximately 20 hours of IT employee labor per month.

# Table of Contents


PROFESSIONAL PROJECT TITLE PAGE ..... 1

CERTIFICATION OF AUTHORSHIP OF PROFESSIONAL PROJECT WORK ..... 2

AUTHORIZATION TO PUBLISH STUDENT WORK ..... 3

ADVISOR/PROFESSIONAL PROJECT FACULTY APPROVAL FORM ..... 4

PROJECT PAPER REVISION/CHANGE HISTORY TRACKING ..... 5

ABSTRACT ..... 6

TABLE OF FIGURES ..... 10

1.0 CHAPTER ONE – PROJECT INTRODUCTION ..... 11

Statement of business problem ..... 11

1.2 Review of Existing Solution ..... 11

1.3 Statement of project goals ..... 15

1.3.1 Information Technology goals ..... 15

1.3.2 Billing Department goals ..... 15

1.4 Limitations/scope of the project ..... 16

1.5 Summary ..... 17

2.0 CHAPTER TWO – RESEARCH ..... 18

2.1 Introduction ..... 18

2.2 WAN technologies ..... 18

2.2.1 Dial-up ..... 18

2.2.2 ISDN ..... 19

2.2.3 T1 Network Connection ..... 20

2.2.4 Extranet VPN ..... 20

2.3 Data Delivery Methodology ..... 21

2.4 Server Hardware ..... 22

2.5 Enhanced Version of Existing Processes ..... 23

2.6 Build Versus Buy ..... 24

CHAPTER THREE -PROJECT METHODOLOGY (SDLC) ..... 26

3.1 Requirements Phase ..... 29

3.1.1 Requirements Phase Verify ..... 29

3.2 Specification Phase ..... 30

3.3 Specification Phase Verify ..... 30

3.4 Design Phase ..... 31

3.4.1 Network Connection between XYZ Telecom and Gilmore, Inc ..... 31
3.4.2 Server Hardware ..... 33
3.4.3 PDF Compression ..... 34
3.4.4 FTP Download Solution ..... 34
3.4.5 File name format ..... 35
3.4.6 Script Functionality ..... 35
3.4.7 PDF File Count Audit ..... 37
3.4.8 PDF Data Copy ..... 38
3.4.9 Logging ..... 38
3.4.10 PDF Data Security ..... 39
3.4.11 Design Phase Verify ..... 39
3.4.12 Design Phase Documentation ..... 39

3.5 Implementation Phase ..... 40

3.6 Implementation Phase Testing ..... 41

3.7 Integration Phase ..... 41

3.8 Integration Phase Testing ..... 43

3.9 Maintenance Phase ..... 43

3.10 Maintenance Phase Testing ..... 44

CHAPTER FOUR - PROJECT HISTORY ..... 45

4.1 Project Initiation ..... 45

4.2 How the project was managed ..... 46

4.3 How the project ended ..... 46
4.4 Success or failure? ..... 47
4.5 What Project Changes Occurred? ..... 47
4.6 Project Summary ..... 48
CHAPTER FIVE - FUTURE IMPROVEMENTS ..... 50
5.1 Using IPSec to Secure the FTP Protocol ..... 50
5.2 SFTP ..... 51
5.3 GNU PG ..... 51
5.4 Securing the FTP Server Traffic ..... 52
5.5 Add Password To Compressed File ..... 52
5.6 Hardware Encryption Addition ..... 53
CHAPTER SIX - CONCLUSION ..... 54
6.1 Lessons Learned ..... 54
6.2 What Could Have Been Done Differently? ..... 57
6.3 Did the Project Meet Expectations? ..... 57
6.4 Conclusions ..... 58
6.5 Summary ..... 58
BIBLIOGRAPHY ..... 59
APPENDIX A: GLOSSARY ..... 60

# Table of Figures

Figure 1: Original PDF data delivery process ..... 13

Figure 2: XYZ Telecom to Gilmore, Inc. modem connection example ..... 19

Figure 3 - ISDN communication example ..... 20

Figure 4 Leased T1 network connection example ..... 20

Figure 5: Simplified VPN connection example ..... 21

Figure 6 Pull PDF data from Gilmore, Inc. example ..... 22

Figure 7 Gilmore, Inc. data push to XYZ Telecom example ..... 22

Figure 8 A Simple Waterfall Process ..... 26

Figure 9 Waterfall model ..... 28

Figure 10 Gilmore, Inc. and XYZ Telecom's Internet connection ..... 32

Figure 11 – VPN diagram between XYZ Telecom and Gilmore, Inc. ..... 33

Figure 12 XYZ Telecom, Windows Server 2003 ..... 34

Figure 13 Script Checks for Compressed Billing File ..... 36

Figure 14 Script Hourly Check For END.TXT file and Billing Data ..... 37

Figure 15 Gantt Chart of project phases ..... 45

# 1.0 Chapter One – Project Introduction

## 1.1 Statement of business problem

XYZ Telecom received bi-monthly customer PDF billing data (identical to the paper customer invoices) from the print vendor Gilmore, Inc via CD media shipped by FedEx. Upon receipt, this data would then be posted to the CD/DVD media server through a manual entry process. The Billing Disputes Team often required immediate access to this data to deal with customer inquiries for telephone dispute calls regarding current or previous billing periods. The manual data delivery often times resulted in delays of up to a week, causing the billing team to be unable to view customer bills regarding a dispute over the latest billing period. Negative customer feedback to the billing management, due to the delay of PDF billing statement data availability, brought this issue to the forefront.

## 1.2 Review of Existing Solution

The existing solution for transferring the PDF billing data via CD ROM had been in place since 1998, although it had been previously effective, the process needed to be changed. XYZ Telecom's print vendor Gilmore, Inc would send bi-monthly PDF billing data, contained on 4 CD ROM's via FedEx from Ontario, Canada to Denver, Colorado. A designated member of the billing team would receive the package and send it through intra-office mail to a member of the Windows Server Support team so they could begin the data copying process. Just sending it from department to department took up to 24 hours. The process that the Windows Server Support team followed included copying all 4 CD's to a folder on a PC and then placing the

data into the correct, consolidated format. However, in order to load the data onto the CD/DVD media server so it could be read, the consolidated data had to be placed onto a DVD. After creating the DVD of consolidated PDF billing data, the Windows Server Support team member would then physically load and copy the DVD onto the CD/DVD media server. This was a very labor intensive and time-consuming process that would take up to 2 days to occur. Any disturbance in this process due to miscommunication, or a critical member from either team being out on vacation, easily delayed this process up to five business days. Over the past five years, numerous personnel from both teams have changed which ultimately caused some of the original billing CDs to go missing.

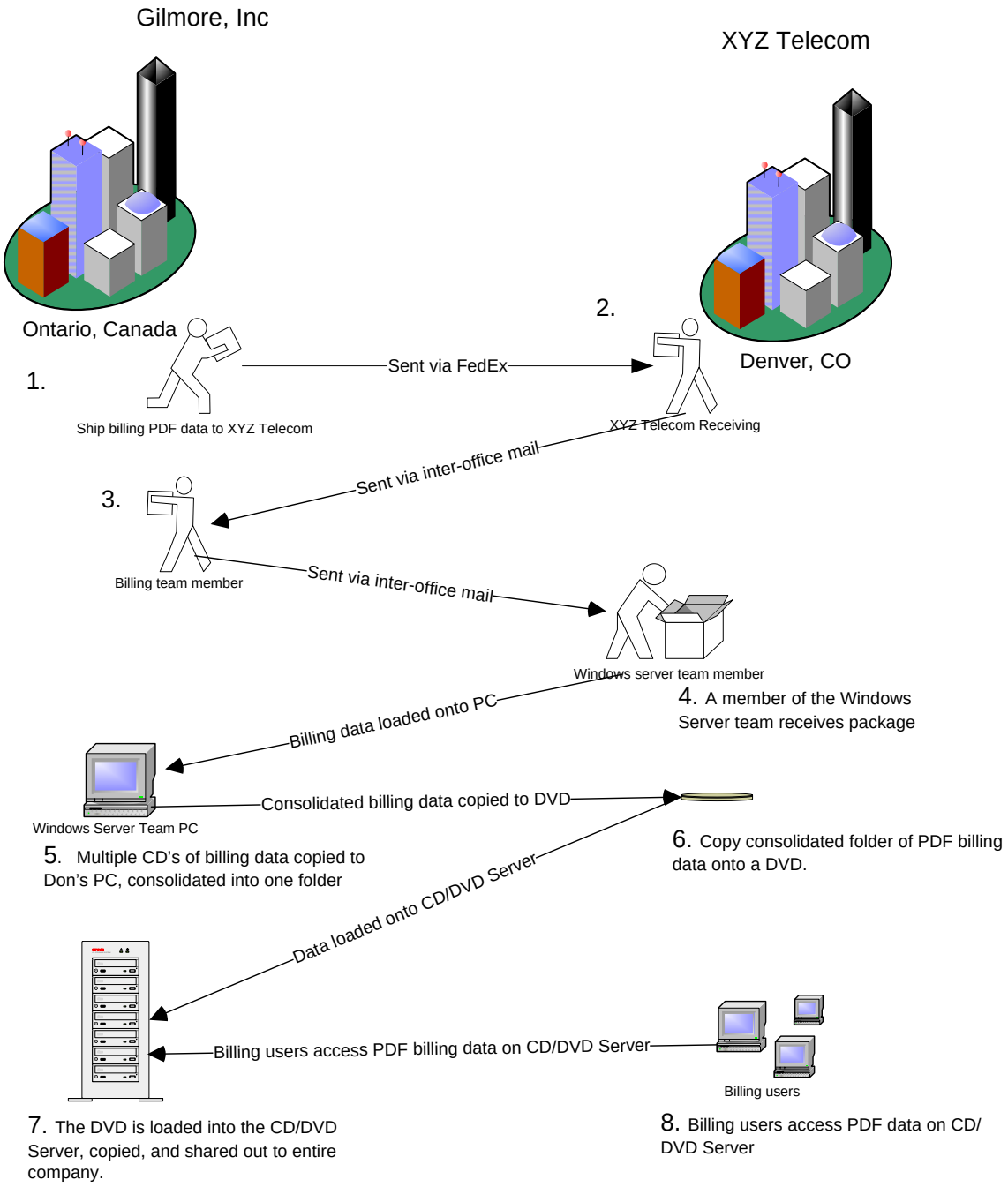


**Figure 1: Original PDF data delivery process**

Upon further investigation of the original delivery process, many design flaws became apparent. First, the existing CD/DVD media server was not covered by any

hardware maintenance contract. The hard drives within the CD/DVD media server were individually locked into the system with the key missing. This server was proprietary in its hardware design and did not support attaching a tape drive to it. Second, no backup of this data had ever occurred before the project began. The preferred method of backup per the manufacturer was to retain the original CD/DVD media. If a manual backup was needed, the server could backup up to writable DVD media, but since the server contained about 125GB of billing PDF data, this was impractical. Repeated attempts to copy large portions of PDF data off the CD/DVD media server failed due to network card errors. Third, the existing CD/DVD media server was only supposed to support 10-15 users for sporadic use. The CD/DVD media server was used by 300 users and frequently exceeded more than 50 concurrent connections. Users regularly complained about performance issues when pulling data off the CD/DVD media server. The IT staff was rebooting this server multiple times per week due to the server being unresponsive to network requests which created many high priority tickets for the Windows Server Support team. Fourth, the PDF billing data was opened to anyone that had access to the internal XYZ Telecom network due to a lack of security. A Microsoft Active Directory with a user account for each employee or contractor was the primary security model. However, the CD/DVD media server did not have the capability to utilize this technology and required individual accounts to be maintained. Fifth, according to government rules and regulations, the PDF billing data is to be retained for seven years. There was serious concern from management regarding the ability to maintain the data without some loss due to the lack of security and no data backup.

Finally, the manual process for posting the billing PDF data internally required approximately 10 hours of XYZ Telecom employee time. This process occurs 24 times per year, creating approximately 240 hours of work to accomplish this manual posting process.

## 1.3 Statement of project goals

Goals from both the IT and billing department were extracted through a series of meetings occurring in the fall of 2004. These meetings occurred after a series of delays that impacted the billing department's ability to get current PDF billing data. Since increasing the level of customer service was becoming a corporate initiative, a project was formed to address this opportunity.

### 1.3.1 Information Technology goals

The Information Technology department sought the following goals:

- A data location with hardware support that could store up to seven years of existing and future PDF data
- To retire the existing CD/DVD media server due its over capacity, lack of hardware support, and inability to perform regular backups
- Reduce manual IT staff involvement with the PDF posting process
- Follow existing processes and procedures for ensuring the data was backed up

### 1.3.2 Billing Department goals

The Billing department sought the following goals:

- A more stable CD/DVD media server, the existing server had stability issues that needed to be addressed
- A process for loading the PDF billing data to the CD/DVD media server in a more timely fashion
- In the event of a hardware failure, the PDF billing data needed to be recoverable within a reasonable amount of time
- Secured billing data so that only the appropriate employees could access the PDF billing data
- Reduction in the amount of time it took for billing PDF data to be posted for the billing staff

## 1.4 **Limitations/scope of the project**

The scope of this project was limited to the items surrounding the past, present and future PDF billing data provided from XYZ Telecom's print vendor, Gilmore, Inc. The project covered the data delivery and storage requirements necessary to refine the process. The items specifically addressed included:

- Network technology between XYZ Telecom and Gilmore, Inc
- Data delivery technology used to transfer the PDF billing files
- PDF billing data storage
- PDF billing data security (file system)
- Ability to recover the PDF billing data

- Server hardware required to hold the existing and future PDF billing data and to facilitate the delivery of it

Further limitations of the project were the time commitments of the project members. Each member had a significant day-to-day workload which was to be balanced along with this project. It was understood that each team member would contribute the necessary time for the project to occur. As workload issues arose, each team member reported their progress or lack of progress with the project manager. The project manager would then work with the appropriate management to ensure that project dates did not slip.

## 1.5 Summary

The billing department required the current PDF billing data to be available in a timely fashion. The previous process for posting PDF billing data was very cumbersome and manual. This process for retrieving the data from Gilmore, Inc needed to be automated in order to reduce the amount of time it took to get the PDF data posted for the billing department.

Both the billing and IT department's goals were assessed to ensure the success of the project. It was decided that several changes needed to be made from the existing architecture to ensure better security and recoverability of existing PDF data. The scope of the project was limited to the billing PDF data transfer, storage, security, recoverability and the server hardware required for this process.

# 2.0 Chapter Two – Research

## 2.1 Introduction

Certain aspects such as WAN technologies and server hardware needed to be researched to determine the best fit for the project. Due to time constraints and the skill sets of some of the project members, not all aspects were fully researched.

## 2.2 WAN technologies

Sending the PDF data electronically seemed to be the ideal solution when it came down to eliminating manual processes. By eliminating the tasks of delivery from Fed Ex, delivery from billing to IT, and the manual loading of data, the process would theoretically be much faster. Determining how to connect XYZ Telecom to Gilmore, Inc. became a topic of research.

### 2.2.1 Dial-up

Dial-up technology used via the PSTN (public switched telephone network) provided a secure method for connecting XYZ Telecom to Gilmore, Inc. This technology has been in existence for many years with a proven track record for use in the industry. Unfortunately, the main drawback to a

dial-up connection was the slow connection speed due the analog technology. Based upon theoretical calculations, downloading a 2GB file over a 56k modem operating at 100% efficiency would take approximately 100 hours. Since the PDF billing data was much larger than 2GB, dial-up technology was not considered a practical option for the project.

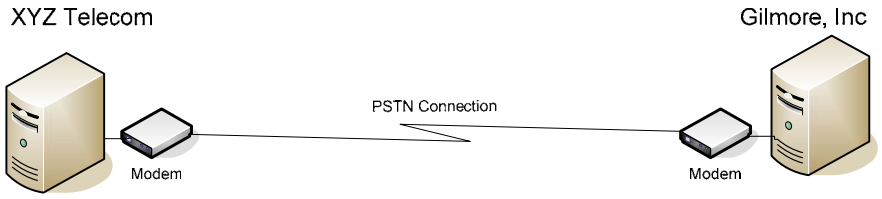


**Figure 2: XYZ Telecom to Gilmore, Inc. modem connection example**

### 2.2.2 ISDN

Similar to dial-up technology, "integrated services digital network" or ISDN, is a digital technology that could facilitate the transfer of data between XYZ Telecom and Gilmore, Inc. An ISDN modem transferring 2GB of data at 100% efficiency would take approximately 45 hours to download, thus, much more efficient than dial-up. An ISDN connection would also be more secure than using an Internet connection due to the network traffic isolation of the leased line. Though ISDN is faster at transferring data than a dial-up connection, it was not fast enough to consider this option when transferring large files.

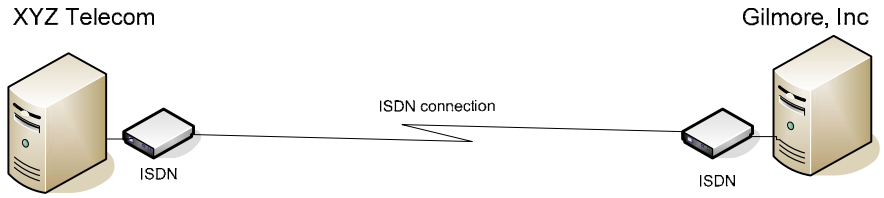


**Figure 3 - ISDN communication example**

### 2.2.3 T1 Network Connection

Similar to an ISDN connection, a direct T1 connection would provide security as well as speeds of 1.544 Mbps per second. This option would also allow the transfer of a 2 GB file in a reasonable amount of time. Downloading a 2 GB file over a dedicated T1 would only take about 3 hours. Unfortunately, leasing a dedicated T1 for this purpose was expensive and beyond what the business was willing to pay.

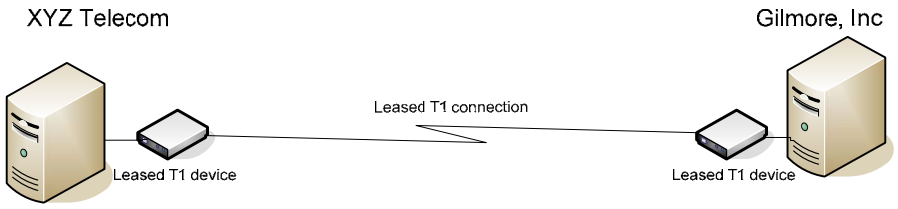


**Figure 4 Leased T1 network connection example**

### 2.2.4 Extranet VPN

Since both XYZ Telecom and Gilmore, Inc. had dedicated Internet connections, the best and most cost-effective solution seemed to be setting up a secure VPN connection over the Internet between the two companies. This involved creating an encrypted IPSec network connection tunnel through

the Internet. XYZ Telecom's network standard was a Cisco 1800 series VPN router to perform such a task. This technology utilized IPSec to encrypt the data traveling through the Internet.

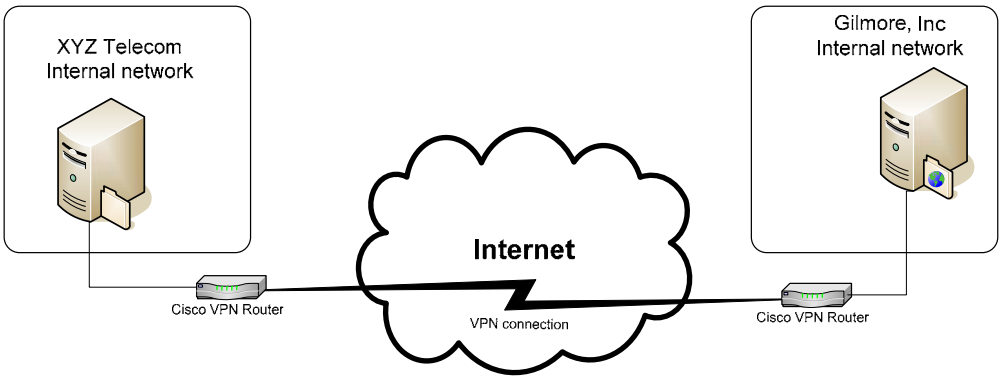


**Figure 5: Simplified VPN connection example**

## 2.3 Data Delivery Methodology

Various data delivery methodologies needed to be considered to ensure an appropriate, supportable method was selected. The project team discussed whether to push or pull the PDF billing data from Gilmore, Inc to XYZ Telecom. Having Gilmore, Inc. post the data to a XYZ Telecom server would minimize the number of tasks that would need to be performed by a XYZ Telecom employee. Gilmore, Inc, however, was not willing to take responsibility for posting the data to a XYZ Telecom asset. If there were server maintenance occurring within XYZ Telecom, the PDF download process could be stopped and restarted as necessary. If Gilmore, Inc. pushed the data to XYZ Telecom when maintenance was occurring, causing the network transfer to fail, XYZ Telecom would have to contact Gilmore, Inc. to restart the data push process. This would ultimately give XYZ Telecom less control over

the billing data retrieval process. It was ultimately determined that XYZ Telecom would have more control if they pulled the data from the Gilmore, Inc. server.

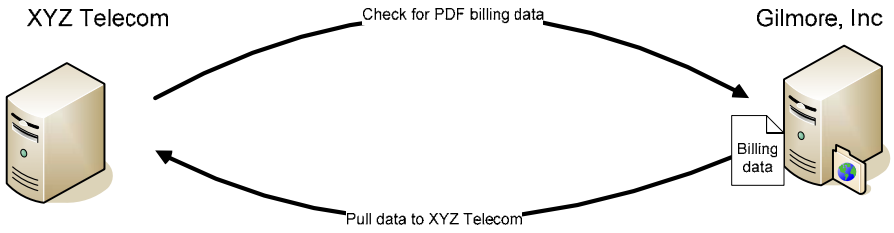


**Figure 6 Pull PDF data from Gilmore, Inc. example**

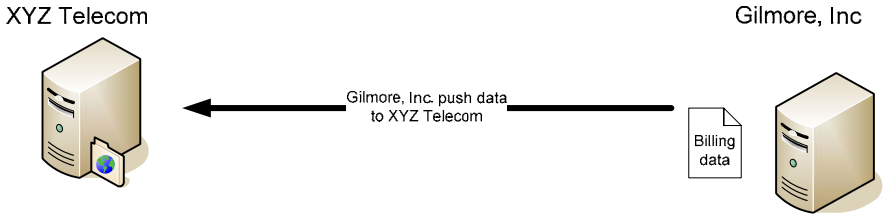


**Figure 7 Gilmore, Inc. data push to XYZ Telecom example**

## 2.4 Server Hardware

It was clear that the existing CD/DVD media server needed to be decommissioned to make way for supported hardware that had the capability to be backed up and was sized accordingly. The reasons for decommissioning the existing CD/DVD media server included:

1. No practical backup solution possible due to the server being proprietary
2. The existing CD/DVD media server had capacity for 10-15 concurrent users, though there were often more than 50 simultaneous connections

3. The existing CD/DVD media server needed to be rebooted approximately 3 times per week due to it being utilized well over capacity
4. Large network copies from the CD/DVD media server would fail, making it difficult to copy to the data to another location for backup
5. Loading the data onto the CD/DVD tower by physical media was cumbersome
6. The “read only” nature of the CD/DVD media server caused the Windows Server Team to re-copy existing CD/DVD media to new media and then transfer the copy to the server so that the appropriate folder structure could be established

Since there were no requirements for CD emulation to occur for the billing department, it was determined that a standard Windows Server would be used. The standard hardware vendor for XYZ Telecom was Hewlett Packard, and a DL380 was purchased with an external drive array with a 500GB capacity which allowed for future growth. This server allowed the existing enterprise tape backup solution to backup the data regularly.

## 2.5 Enhanced Version of Existing Processes

Initial plans of maintaining the existing FedEx data delivery process were considered. By replacing the CD/DVD media server with a regular server, it would be feasible to modify the existing PDF data posting process. By using Windows permissions, it would be possible to delegate the data-posting task to certain

members of the billing team. This would effectively eliminate the Information Technology department's involvement with the bi-monthly PDF data posting process.

Upon further discussion regarding this process, it was determined that the manual posting process was not suitable. The Information Technology department felt that there were too many risks associated with giving members of the billing team access to add and update the billing data. By automating the PDF data transfer process using a network connection between Gilmore, Inc. and XYZ Telecom, the possibility for human error was eliminated.

## 2.6 Build Versus Buy

The decision to build a solution versus purchase a pre-packaged solution was discussed. Many pre-packaged software applications available could automate the file download process. Purchasing pre-packaged software was looked upon favorably because it would leverage existing software, reducing the overall time and cost of the project.

Approximately 7000 individual PDF billing files were delivered per billing period. Rather than ensuring each PDF file was transferred successfully, it was decided the data should be compressed into one file. Upon a successful download, the compressed file needed to be extracted to a specified network location. These multiple factors added more variables that would also need to be automated. Since this process dealt with customer billing data and would need to be thoroughly

logged, it was decided to write the solution using internal XYZ Telecom technical resources from the Windows Server Support team. Writing a custom solution allowed for greater control logging the downloaded events and provided more flexibility for future changes within the XYZ Telecom network.

## 3.0 Chapter Three - Project Methodology (SDLC)

The internal project managers within XYZ Telecom had historically used the "waterfall model" for a standard project software development life cycle or SDLC. A diagram of a simple waterfall process is shown in figure 8.

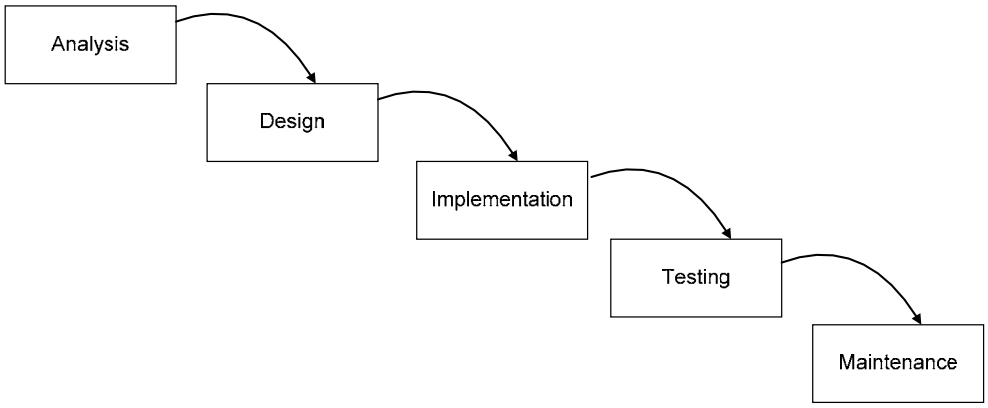


**Figure 8 A Simple Waterfall Process**[1]

According to Kathy Schwalbe, author of "Information Technology Project Management," *"The waterfall life cycle model has well-defined, linear stages of systems development and support. This life cycle assumes that requirements will remain stable after they are defined."*[2]

Once the project requirements were established, they remained fairly static. This ultimately helped the project succeed using the waterfall model software development life cycle. The waterfall model has received criticism for being too

[1] Stevens, Pooley, Using UML – Software Engineering With Objects and Components, updated edition, Addison-Wesley, 2000, pg 47
[2] Kathy Schwalbe, Information Technology Project Management, third edition, Course Technology, 2004, pg 46

rigid of a lifecycle due to the phases being linear. The project team was not severely impacted by this as there were no significant changes to the project.

Stevens and Pooley, authors of Using UML – "Software Engineering with Objects and Components" state that the simple version of the waterfall process was known as the "throw over the wall process"[3] due to its downward flow. Since the diagram implies that a phase is never revisited once it is complete, there is little room for error. To follow this model exactly would require perfections which unfortunately are not always realistic.

The waterfall model encourages a thorough and disciplined method with plenty of testing and documentation. XYZ Telecom did not have an independent software quality assurance (SQA) team specifically for this project, thus the IT members reported any issues found in testing to the project team. With the assistance of the project team, the project manager then assessed the situation, determined the appropriate actions that needed to occur and updated the project schedule accordingly.

[3] Stevens, Pooley, Using UML – Software Engineering With Objects and Components, updated edition, Addison-Wesley, 2000, pg 47

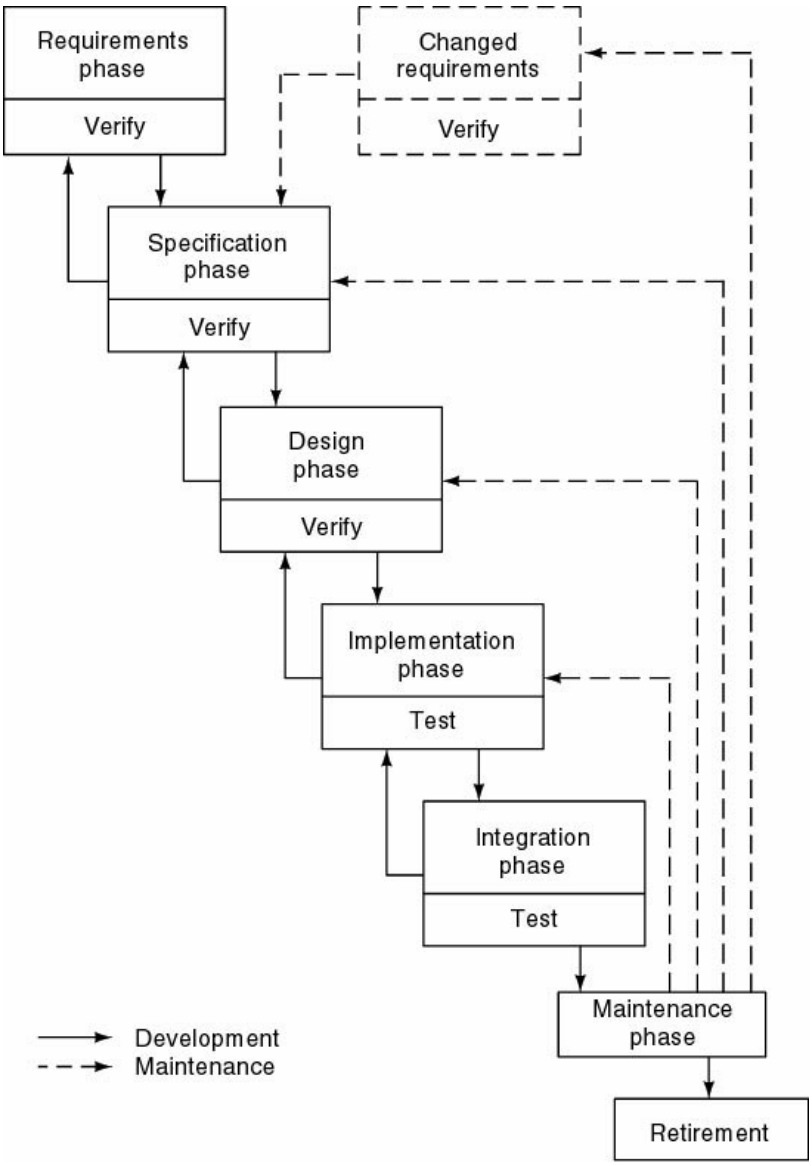


**Figure 9 Waterfall model**[4]

The project team strayed away from the simple waterfall method and adapted to a more advanced and flexible model as shown in figure 9. This approach proved to be more realistic by allowing each phase to be revisited throughout the project.

[4] Stephen R. Schach, Object-Oriented and Classical Software Engineering, fifth edition, 2002, McGraw-Hill, pg 66

Though the project team did not encounter a significant number of changes in the later phases of the project, it was readily known that previous phases could be revisited if necessary. During the final testing phase, it was discovered that items such as email notification and specific logging details discovered in the integration phase were overlooked and thus had to be written into the requirements and specifications phase documentation. This was an excellent example of how using the more flexible model allowed us to make changes late in the project.

## 3.1 Requirements Phase

The requirements phase of the project was based around improving customer service levels and internal cost savings while providing the PDF billing data available in a shorter period of time. The existing manual data delivery process was automated to accommodate this. The following items were core requirements of the project:

- Faster data delivery process to post PDF billing data
- Higher availability of current and existing PDF billing data
- Elimination of human error
- Greater security of PDF billing data

### 3.1.1 Requirements Phase Verify

Upon completion of the requirements phase meetings, a finalized list of requirements (listed in the previous paragraph) was distributed to the project team. Neither the billing personnel, nor the project team had any objections to the

requirements identified. Concerns arose regarding the automation because the Windows Server Team did not have a significant amount of scripting experience. In an effort to alleviate the teams concerns, a member from the Windows Server Team enrolled in a Visual Basic scripting course to help further their scripting skills.

## 3.2 Specification Phase

The specification phase of the project was based upon meeting the requirements that were previously defined in the "requirements phase" of the project. This phase defined the necessary components to the project which included:

- A network connection between XYZ Telecom and Gilmore, Inc. to facilitate faster delivery of data
- Existing PDF billing data needed to be migrated to a solution that had higher availability.
- A compressed data file containing the PDF billing data would utilize the network connection between Gilmore, Inc and XYZ Telecom more efficiently
- File server at Gilmore, Inc. that was accessible to XYZ Telecom
- Script that ran periodically to check for new PDF billing data
- Based upon the current date and time, the script would determine what filename would download.
- Ensure that the correct number of PDF documents arrive
- Extract the PDF billing data to the correct network location
- Detailed logs of the download and automated PDF billing data tasks
- Secure the PDF billing data further within XYZ Telecom

## 3.3 Specification Phase Verify

The specification phase verify process resulted in a project plan document that loosely defined the tasks that needed to occur. The technical aspects were discussed at length to ensure that the proposed solutions were realistic.

## 3.4 Design Phase

The design phase of this project focused on items defined from the specification phase. The defined items were matched with suitable technology that would fulfill each requirement. Technical consideration was taken for the "best practices" of each situation.

### 3.4.1 Network Connection between XYZ Telecom and Gilmore, Inc

A network connection between XYZ Telecom and Gilmore, Inc allowed faster delivery of PDF billing data. The project team decided to use a network connection to transfer the data, which also provided greater control of the data and allowed for further automation. Automating this data delivery process reduced the lag time associated with using a FedEx delivery. Automation also eliminated the need for XYZ Telecom to receive the FedEx package, deliver the package via intra-office mail and physically load the PDF data to the server. The project team decided that a VPN connection between XYZ Telecom and Gilmore, Inc. would be used to save a significant amount of time. This solution was much faster than an ISDN or dial-up option as it would use each company's dedicated Internet connection. Gilmore, Inc. had a T1 connection (1.54 Mbps) to the Internet, while XYZ Telecom had an Internet connection of 100 Mbps, full duplex.

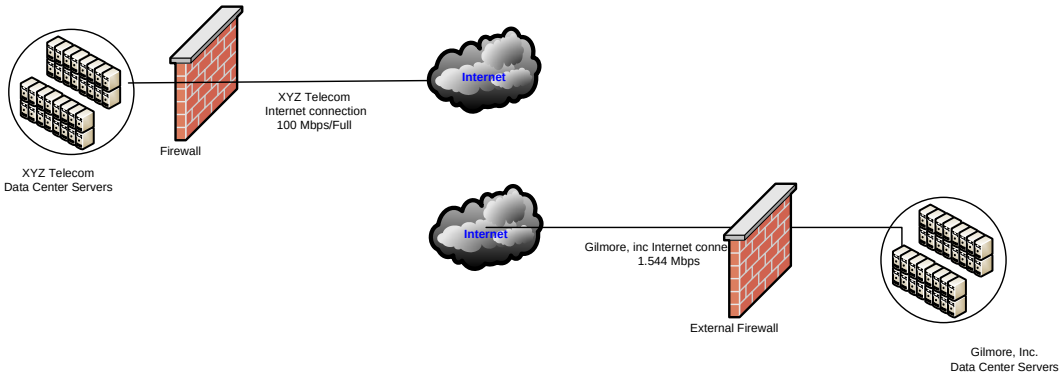


**Figure 10 Gilmore, Inc. and XYZ Telecom's Internet connection**

By using a Cisco VPN solution between the two companies, the compressed PDF data would be encrypted while being transferred across the network. On the XYZ Telecom internal network, the Gilmore, Inc. server would have an accessible IP address for accessing the file server.

Two Cisco 1800 series routers would be used for the VPN connection. These routers used IPSec with 3DES encryption for the VPN traffic between them.[5] Both routers were configured on their external interfaces to only communicate to each other's IP addresses. All traffic from any other IP addresses was dropped and not analyzed.

[5] Cisco Systems, Network Security Features on the Cisco Integrated Services Routers, 2005, Cisco Systems, pg 4

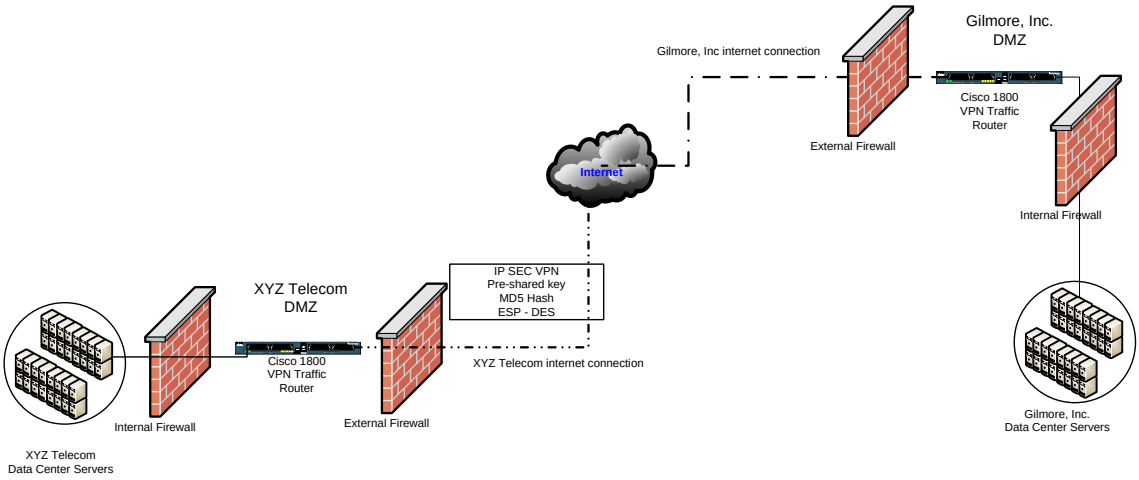


**Figure 11 – VPN diagram between XYZ Telecom and Gilmore, Inc.**

### 3.4.2 Server Hardware

A new Hewlett Packard DL380 G4 server with an external drive array was purchased for this project to hold both the current and existing PDF billing data. The new server contained approximately 700GB of disk space for this data, which allowed for ample storage of the existing PDF billing data with additional growth factored in for the seven-year storage requirement. The existing PDF billing data from the CD/DVD media server was copied to the external drive array on the server running Windows Server 2003. This server allowed the Windows Server Support team to use the standard backup and antivirus software. By moving away from the proprietary, underpowered CD/DVD media server, the server would be backed up nightly using the standard backup system. Since the server ran Windows Server 2003, NTFS security was used to secure the PDF billing data via Windows groups for those that required access. Figure 12 shows a diagram of the Hewlett Packard DL380 with an attached drive array.

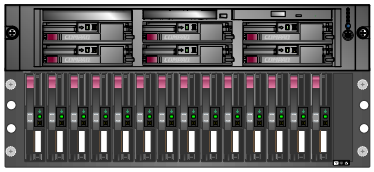

**Figure 12 XYZ Telecom, Windows Server 2003**
**Hewlett Packard DL380 with additional drive array**

### 3.4.3 PDF Compression

Compressing the PDF billing data into one file simplified the logistics of the download from Gilmore, Inc. It was decided that downloading one file versus downloading 7000+ files would be the preferred method. Compressing the PDF billing files into one zip file also increased the efficiency of the data transfer.

### 3.4.4 FTP Download Solution

Once a network connection was configured between XYZ Telecom and Gilmore, Inc, an FTP server was established at Gilmore, Inc. for the PDF billing data. This server was accessible by the XYZ Telecom HP DL380 server through the VPN connection. The PDF data was pulled from XYZ Telecom server which gave greater control as to when PDF billing data was received.

It was decided that Gilmore, Inc would post the compressed PDF billing data on the FTP server at their location. This FTP server was only accessible to XYZ Telecom through the dedicated VPN connection between the two companies. The FTP server was not open to the Internet at Gilmore, Inc. Instead, the server resided on an internal, “non-routable” IP address. All communication between

XYZ Telecom and Gilmore, Inc. was encrypted by the Cisco VPN routers.

### 3.4.5 File name format

The naming convention for the compressed file was MMDDYYYY.zip. Where MM was the month, DD was the day of the month (either 01 or 15 based on the bill run date) and YYYY represented the year. Gilmore placed the current bill run compressed file on their FTP server using the appropriate naming convention. The script determined what the filename would be based upon the current date and time on the server. Since there were two billing periods per month, two different file names were possible each month. Each billing period occurred on the 1st and the 15th of each month. Therefore, if the date was between the 1st and the 14th, it was assumed that the file being downloaded was for the 1st bill run of the month. If the date was between the 15th and the 31st, it was assumed that the file being downloaded was for the 2nd bill run that occurs on the 15th. An example of this for the 2nd billing cycle in April, 2006 was "04152006.zip."

### 3.4.6 Script Functionality

The Windows Server Support group determined that it would be more efficient to pull the PDF data down in a compressed format. Since the data was being pulled from a Gilmore, Inc. server it was necessary to have a script from XYZ Telecom periodically checking to determine if new data was available.

The script would check the Gilmore, Inc server on an hourly basis to determine if the current bill run file was available (as seen below in figure 13).

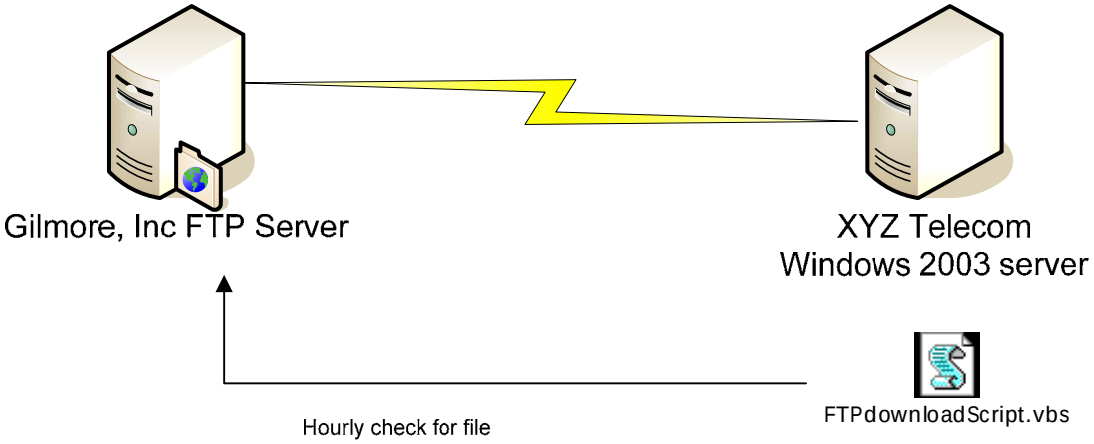


**Figure 13 Script Checks for Compressed Billing File**

Due to the large amount of data (2.5 GB in size), it was important for the script to determine if the compressed file was ready for download. There was potential for the script to attempt to download the current zip file even though it may have not been fully compressed or copied to the Gilmore, Inc server, which would result in download errors. It was agreed that Gilmore, Inc. would place a file named “end.txt” on the file server after the compressed file containing the PDF data was copied to the server to alleviate this problem. The presence of the “end.txt” file would indicate that the compressed file was completed and ready for download. The script periodically checked for the “end.txt” file to determine if there was a new file to download. If the “end.txt” file was detected, the script would download the file to a server within XYZ Telecom. After the zip downloaded completely, the “end.txt” file was then deleted from the Gilmore, Inc. FTP server. Figure 14 shows a diagram of how this process would work.

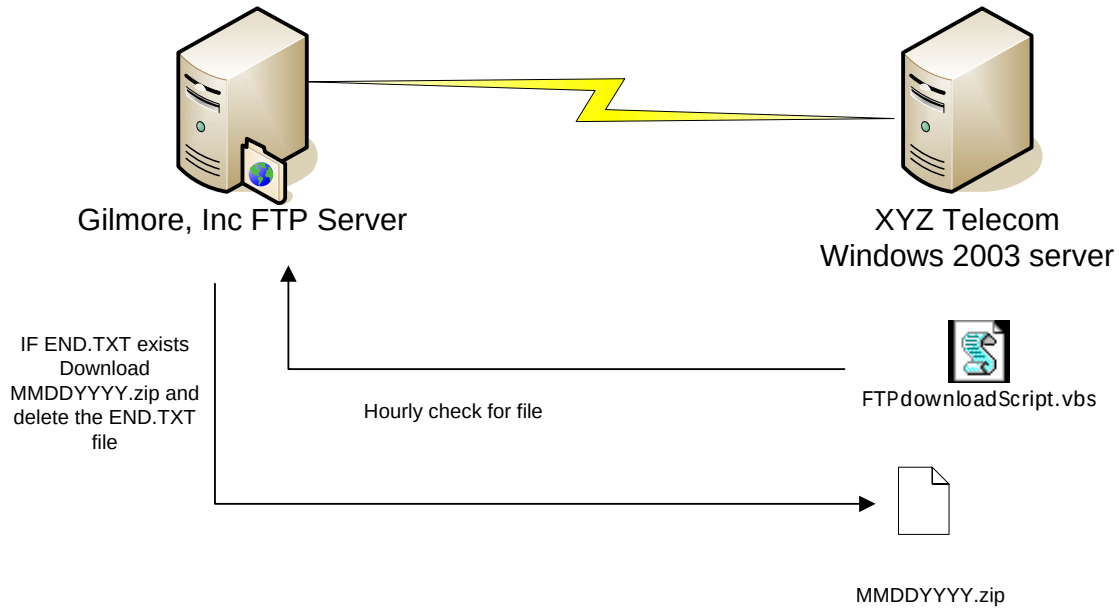


**Figure 14 Script Hourly Check for END.TXT file and Billing Data**

### 3.4.7 PDF File Count Audit

In order to ensure that the proper number of PDF files were transferred from Gilmore, Inc, the project team decided that the zip file should include a file named “audit.txt.” This file contained the total number of PDF files that should be included for the bill run. Upon extracting the zip file to the XYZ Telecom server, the script opened the audit.txt file (also extracted from the compressed file) and read the number on the first line. The script then counted the total number of PDF files extracted for the current bill run and compared this number with the number listed in the audit.txt file. If the two numbers did not match the script failed, if they matched the script would proceed.

### 3.4.8 PDF Data Copy

Once it was determined that the proper number of PDF documents were available, the script verified that the current bill run data did not already exist. If the current bill run data did not exist, the script would create and copy the data into a folder representing the current bill run. The folder format, MMDDYYYY, was similar to the name of the zip file that was downloaded from the Gilmore, Inc. FTP server. Once the copy was complete, the script verified that the data was successfully copied to the new location.

### 3.4.9 Logging

The script logged its activity and progress to a log file as well as to a SQL Server database within XYZ Telecom. The following information was required to be logged for future troubleshooting and historical reference:

- The date/time the script was executed
- “end.txt” status on Gilmore, Inc. FTP server
- “audit.txt” file value
- Actual PDF file totals from extracted zip file
- The billing cycle (i.e. MMDDYYYY)
- The zip file name (i.e. MMDDYYYY.zip)
- Current server name that the script is running from
- Was there an existing billing cycle file with the same name?
- Override value (necessary to determine if the script is run bypassing the calculated billing cycle date.

The log data would be written to an existing SQL Server within XYZ Telecom. This data could be used for additional alerting if it were deemed necessary at a later date. The script also logged the above data locally in a text file. Logging the data to multiple locations allowed for more information to be available in the event of the total server loss.

### 3.4.10 PDF Data Security

The PDF billing data would be secured on a server running Windows Server 2003 using NTFS security. Security groups from the XYZ Telecom Active Directory would be used to secure the PDF billing data. Group owners would be defined for each security group so that the adding and removing of billing users would be delegated to the billing team. The groups created to secure the PDF billing data would be mail enabled so that they would appear in the global address list within Microsoft Outlook. This interface was beneficial because it allowed the group owners to add and remove group members as desired.

### 3.4.11 Design Phase Verify

Upon verification of the design phase, none of the team members discovers any issues that would hinder further progress.

### 3.4.12 Design Phase Documentation

The design phase documentation included a formal network diagram of the proposed configuration between Gilmore, Inc. and XYZ Telecom. This document was distributed to all members of the project team. A written document detailed the desired functionality for the FTP download from the Gilmore, Inc. FTP server.

### 3.5 Implementation Phase

Upon ordering the Cisco 1800 devices for the Gilmore, Inc to XYZ Telecom VPN connection, the project team was informed that the devices were back ordered for three weeks. This did not cause any significant issues as the script being written to transfer the data could be tested against an internal FTP server.

The new Hewlett Packard DL380 server with external drive array arrived and was built and configured per the Windows Server teams specifications. The server had the appropriate antivirus software installed and configured. The Veritas NetBackup client was installed and configured per the Windows Server team standards. The server was added to the daily backup policy to ensure that the data would be recoverable in the event of data loss. The network interface card on the existing CD/DVD media server experienced errors and caused the billing data copies to fail, which lengthened the project over a period of several weeks. Once copied, a member of the Windows Server team verified the total number of files and bytes on the CD/DVD media server matched the total numbers on the Windows 2003 server. As a final check, the billing team verified the data to ensure that none of it was missing.

The FTP download script was written internally and developed against an internal FTP server. Actual billing data was used for testing the scripts functionality per the scripts specifications. Extensive testing was performed on the script functionality to ensure that the script behavior was acceptable once it went into production.

## 3.6 Implementation Phase Testing

The implementation phase testing occurred throughout the phase to continuously refine the scripts functionality. Additional functionality such as writing VBScript log entries to the SQL Server had never been accomplished before by the Windows Server team. Because of this, building the database and refining the logs took longer than anticipated. The Windows Server Support team had to further research and develop these specific skills in scripting.

## 3.7 Integration Phase

The integration phase was a success thanks to all the planning and time that went into the early stages of the project. After the Cisco devices arrived, a member of the network team configured the VPN router for Gilmore, Inc. and shipped it to Ontario to be installed at the Gilmore, Inc site. The VPN connection between XYZ Telecom and Gilmore, Inc was installed and configured per industry standard “best practices.” While a Gilmore, Inc. staff installed and configured the FTP server at their location, a member of the network team from XYZ Telecom configured the routing between the two VPN routers so that data could flow to and from the

Gilmore, Inc. FTP server. The encryption between the VPN routers was set and hardened per the XYZ Telecom network group's specifications. The IP address information was relayed to the Windows Server team so that further testing could be performed against the Gilmore, Inc. FTP server.

It was determined that the PDF download time would take approximately 5-7 hours to complete. Initially, this seemed to be an unacceptable length of time for the billing team members, but they eventually accepted it as it was much faster than the previous FedEx method.

A trial run was performed to determine the readiness of the system. A few problems arose due to the lack of communication among the Gilmore, Inc. staff. First, Gilmore, Inc. did not create the compressed file per the project team's specifications, and second, both the audit.txt file and the compressed file were misnamed. The XYZ Telecom project manager was informed of these failures and set up a series of conference calls to rectify the situation. In order for a successful download utilizing the script, the filename and audit.txt checks had to be bypassed. These modifications were made, and the PDF billing data was available for the internal billing staff using the new process. Since the project was still not fully implemented and the project was overdue, it was rushed into the maintenance phase due to management pressure.

A member of the Windows Server Support team determined that some sort of notification was going to be needed so that both IT and the billing team knew when the automated billing data retrieval process completed. Initially, he added this

functionality for testing, but knew it was just a matter of time before the billing team asked for this. Eventually, the email functionality was implemented into the script.

### 3.8 Integration Phase Testing

A late addition that was caught in the integration phase testing was composed of adding the email notification functionality to the script. At this point, the email notification had not officially been stated as a requirement in the project documentation. Fortunately, this feature was implemented without adding any delay to the project.

### 3.9 Maintenance Phase

The Gilmore, Inc IT staff, however, failed again to follow the naming conventions defined by the project team. At this point, the project team was frustrated at the seemingly incompetent Gilmore, Inc IT staff. After a final conference call, it was discovered that Gilmore, Inc. had not yet fully automated their side of the process, thus explaining the errors and lack of communication. Within 48 hours, the Gilmore, Inc. IT staff had automated the compression and file naming conventions. The process was then repeated to determine if the download would successfully be completed without manual intervention. The download took approximately 7.5 hours and performed as designed. Following the successful PDF bill run, the project team had a brief conference call to relay this information. The automated email notification become more important to the billing staff once the

process was functioning correctly. Several billing staff members outside of the project team requested to be added to the distribution list.

### 3.10 Maintenance Phase Testing

The maintenance phase testing occurred during the first few production runs of the automated download process. The log files were reviewed periodically to determine if any irregular activity occurred. Once the automated script completion email was implemented, nothing else needed to be addressed.

A member of the XYZ Telecom network team reviewed the router logs and verified that encrypted network communication occurred as planned. The data traveling the VPN routers was well within the capacity limits and posed no performance degradation. The script email notification performed as designed and would inform the appropriate users when new PDF billing data was available. This script also notified these same users if there was a problem with the PDF billing data download.

# 4.0 Chapter Four - Project History

The project began formally in early January 2005, lasted 9 months and was completed in late September 2005. Below is a Gantt chart that shows the duration of each phase of the project.

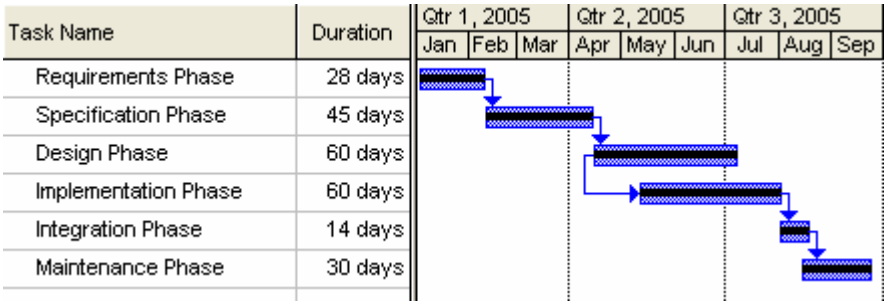


**Figure 15 Gantt Chart of project phases**

## 4.1 Project Initiation

The decision to automate the PDF billing data retrieval process was made in the fall of 2004. This manual data delivery process first began in 1998 when the billing data was published to PDF documents. Those involved with posting the data knew there was a better way to accomplish this process. As the billing data was processed more quickly, the paper bills sent by mail were also arriving faster. When customers called in with billing questions there became a greater need for the PDF billing data to be posted more quickly.

Finding money for business processes that were not technically broken was the main factor for the process to stay manual between 1998 and 2005. As the push to increase customer service levels became more of a company initiative, the existing business processes got additional attention.

## 4.2 How the project was managed

The PDF billing data automation project was managed by a billing manager within XYZ Telecom. This person was very familiar with the project details and did most of the project management tasks during the weekly project meetings. She used Microsoft Project to track the tasks and status of each team member. She used the Waterfall model for the chosen Software Development Life Cycle. Aforementioned, the Waterfall model was a common SDLC used within XYZ Telecom.

The team members that worked in Information Technology were frustrated by the level of detail that was required to be disseminated to the project team. Explaining the details and concepts in such a way that all team members understood how things were going to work took up a majority of most meetings early on. As the project developed into the later stages, the meetings became more brief.

## 4.3 How the project ended

The project ended after several billing period delays due to Gilmore, Inc. not being able to post the correct data to the FTP server in the agreed upon format. The project file specifications were not fully communicated among the Gilmore, Inc members involved with the implementation. One source for the error involved Gilmore, Inc. employees who were not part of the project team being responsible for compressing the PDF documents. Since those people were not familiar with the project, they included the wrong number of PDF billing files, and misnamed the

audit.txt file which caused delay and confusion. The time zone difference seemed to also be a cause of communication delays as problems identified by the XYZ Telecom billing team would be often be noticed after the Gilmore, Inc. office had closed for the day. Once the Gilmore, Inc. staff automated the process on their side to include the audit.txt file and follow the appropriate naming conventions, very few errors occurred. The project failed to run correctly for two full bill runs before Gilmore, Inc. figured out the desired filename formats.

### 4.4 Success or failure?

Ultimately, the project was a huge success as it saved approximately 20 hours of IT employee time per month by eliminating the existing manual process. Since implementing the project, PDF billing data is now available an average of 3-5 days sooner which has led to increased customer satisfaction. This project paved the way for another that leveraged the existing PDF automated process. An online billing project which allowed customers to download their bill directly from a XYZ Telecom website was implemented in the spring of 2006. The effects of these two projects has had a profound impact on the time it takes for the customer to get their billing statements. As a result of this, the billing team has noticed an increase in their customer service satisfaction rating. Not only was the project was a great success for all members involved, but also for all XYZ Telecom customers..

### 4.5 What Project Changes Occurred?

Over the course of the project there were a few items that were either not defined or not fully understood. The main item that was never fully discussed until *after* implementation was an email notification that the process had successfully completed or failed. The script needed to be modified to add this functionality. A distribution list was created for all of those that were interested in the completion of the automated process. It took about 90 days before the distribution list was updated with the appropriate people that required notification.

There were a few minor delays at the beginning of the project while waiting for the back ordered Cisco VPN hardware to arrive at Gilmore, Inc and XYZ Telecom. This did not significantly impact the project dates and ultimately there were no contributing factors that escalated this issue in a bad way.

There was some misunderstanding as to the naming convention of the “end.txt” file. Since most of the project meetings were done via conference call, Jonathan Seashore in the Windows Server Team thought the file was supposed to be named “N.txt.” This was caught during one of the first test runs of the automated system and was easily fixed. The specific name of the file was never written in the project documentation until after this confusion occurred.

## 4.6 Project Summary

In summary, the project was ultimately a great success. Once the Gilmore, Inc. team conformed to the naming specifications the automation worked exactly as designed. As of the time of this writing, there have been 10 successful, consecutive bill runs utilizing the new process. Formal documentation within the Windows Server

team had not been written at the time of implementation.  In order for the project to be a continued success, further knowledge transfer needs to take place among the Windows Server team.

# 5.0 Chapter 5 - Future Improvements

Inherently the most significant item that should be addressed at a later date would be replacing or enhancing the existing Gilmore, Inc. FTP Server with a more secure technology. FTP by design transmits data in “clear text,” meaning the possibility for data to be compromised is much higher than using another method. FTP technology was chosen for this project because of its simplicity and ease of implementation. Since the VPN connection over the internet was secured via IPSec encryption on the VPN routers, this initially mitigated the exposure risk associated with the FTP server. A more realistic threat could be on the Gilmore, Inc. side. Since the PDF data originates and is stored initially at Gilmore, Inc, it is unclear how secure the data really is. The potential for a user within Gilmore, Inc. to download the data from the Gilmore, Inc. FTP server does exist. Since there is no data encryption within Gilmore, Inc., all data would be communicated via “clear text.”

Another improvement that would positively effect this project would be to have Gilmore, Inc upgrade their connection to the Internet. Gilmore, Inc’s T1 connection to the Internet is the bottleneck that causes the PDF download to take 12+ hours to complete. If this connection were upgraded, the download would be much faster.

As time goes on there may become more opportunities to further harden the process for the PDF billing data. The following sections describe additional software or tools that could enhance the existing process to further protect the data.

## 5.1 Using IPSec to Secure the FTP Protocol

By requiring IPSec communication between the XYZ Telecom and Gilmore, Inc. servers, the FTP protocol would become more secure. This further encryption would be performed at the server level, in addition to the IPSec encryption being performed across the VPN routers. In order for this to minimize the risks associated with the FTP server, the Gilmore, Inc. FTP server would have to exclusively communicate using the IPSec protocol. This option could be performed at a later date if it was deemed appropriate.

## 5.2 SFTP

Setting up an SFTP server on the Gilmore, Inc. side would not significantly change the existing functionality. It would, however, further encrypt the data traveling between XYZ Telecom and Gilmore, Inc using an SSL certificate. This would be the easiest way to tighten up the security of the existing FTP network traffic.

.

## 5.3 GNU PG

GNU PG is an open source technology that can be used to encrypt and decrypt data using public key cryptography. This technology is used to keep data private through encryption. A public key is used to encrypt data, and corresponding private key is used to decrypt the data. A public key cannot decrypt the data that it has been encrypted through the same public key. One exception to this rule is if an additional public key is added to the recipient list in which the corresponding private

key is held by the same individual. Then, the encrypted file could be decrypted with the corresponding private key.

This technology could be used to encrypt the compressed file. This would make compromising the data very difficult regardless of who happened to gain access to the file. The GNU PG technology is free to use under the terms of the Free Software Foundation (FSF) General Public License (GPL).[6]

### 5.4 Securing the FTP Server Traffic

Currently, any device on the internal XYZ Telecom network can get to the Gilmore, Inc. FTP server. By changing the XYZ Telecom side Cisco 1800 device to only allow the Windows Server running the script to connect to the Gilmore, Inc FTP server would further strengthen the security from the XYZ Telecom network. This would eliminate the possibility for any computer or machine on the XYZ Telecom internal network from being able to access the FTP server.

### 5.5 Add Password To Compressed File

Adding a password to the compressed file would strengthen the security of the file if it were copied or moved from any server. Combining password security with the other layers of security could ultimately reduce the ability for the data to be compromised.

[6] Free Software Foundation, Inc, GnuPG Frequently Asked Questions, 2003, Free Software Foundation, Inc., pg 1

## 5.6 Hardware Encryption Addition

An AIM, or an advanced encryption module for hardware encryption for the Cisco 1800 VPN routers, could be added at a later date. At the time of implementation, both Cisco 1800's VPN routers used software-based encryption when transferring the data between the two devices. If the network traffic load over the VPN network link became significant, it may be beneficial to purchase an advanced encryption module that could offload the software-based encryption from the router itself. Performing the encryption at the hardware level would offload some of the processing that the router was performing. This would allow for greater utilization of the existing Cisco 1800 series VPN routers. More than likely the growth of billing data will not significantly impact the performance of the router before it becomes EOL or end of life.

# 6.0 Chapter 6 - Conclusion

## 6.1 Lessons Learned

Over the course of the project, the project manager noted several “lessons learned” from various team members on the project. The following were a few of the more important points:

- Communication with critical extended project team members
- Commit to security up front
- Begin the project planning with greater specifics for the end in mind
- Dated project SDLC
- Workload issues for project team members
- Misconceptions about what can and cannot be accomplished via the compressed file transfer

There was a communication breakdown with the extended members of the Gilmore, Inc. project team. The IT project members for XYZ Telecom were not familiar with the person who would specifically be performing the IT tasks on the Gilmore, Inc. side. Since there was no specific Gilmore, Inc. technical contact, Jonathan Seashore worked entirely through the project manager to ensure the Gilmore Inc. IT tasks were performed. Ultimately, Jonathan should have had the Gilmore, Inc. IT team member(s) phone numbers to assist with resolving any IT issues in a quicker fashion. Since he did not have this contact information, he had to relay his IT specific questions or findings through the XYZ Telecom project manager,

who in turn would contact the Gilmore, Inc. project manager, who would then contact the Gilmore, Inc. IT team member to relay the information. Information sharing in this capacity often took several days before questions were answered. When the final production testing took place, once again were there communication issues relaying information between the two IT teams.

Committing to security up front should have been more of a priority. Ideally, the FTP server solution would not have been the preferred method for transferring unencrypted data. This technology was chosen based on some inexperience within the XYZ Telecom IT staff. There were concerns that a higher security solution could potentially become more troublesome to configure, implement and maintain. A higher security solution was never considered because the lack of scripting skills on the Windows Server team had already created a learning challenge. This combination of factors swayed the project team members to go with a technology that they already know, even though it was not inherently secure.

The project team could have focused more on the end product. If the project team would have spent more time visualizing exactly what they wanted for the end product to be, it is possible there would have been less surprises at the end of the project. The email notification feature was never discussed or brought up until the solution was being tested against the production data. Ultimately, this did not turn out to be a significant issue. Another issue that was overlooked by the project team was the download time for the compressed PDF bills. On average, it would take between 8 and 12 hours for the file to transfer to the XYZ Telecom server. This significant delay causes the XYZ Telecom billing staff to wonder when the download

process had begun or if the process was running. Initial plans were only for a final notification message with the success/fail status. Since there was not a beginning notification message sent to the billing users, the Windows Server Team was queried for the status of the download. This required one of the team members to manually check to see if the file was downloading.

The project manager received feedback that the formality of the project increased the amount of time it took to deliver the final product. The XYZ Telecom technical members of the project team understood the technical details that needed to be accomplished. The weekly conference calls to update the project manager seemed redundant and long winded as each member worked independently for a significant portion of the implementation phase. Since the technical project managers did not have significant time to dedicate to this project, the weekly project meetings seemed to provide minimal benefit.

Since the project team members had full schedules before and during the project, progress seemed fairly slow. The project could have been completed in a much more timely manner had more team members had greater availability. Ultimately, the team was able to come up with a realistic timetable for the project and was able to deliver the project fairly close to on schedule.

Upon completion of the project, there were misconceptions among the billing staff users as to the length of time it would take to access the current PDF billing data. Since the transfer of compressed data took between 8 and 12 hours to occur, several billing users were interested in seeing individual PDF documents. These questions occurred quite frequently as the billing staff was typically anxious to see

the billing data as soon as possible. If this requirement was discussed up front, it could have been possible to design the solution to download each file individually or access each file individually from the Gilmore, Inc. server.

## 6.2 What Could Have Been Done Differently?

Greater emphasis could have been put on security up front. Since the project did not have a team member specifically dedicated to information security, the project team was able to take the easy way out in some circumstances. Having a project member looking out specifically for security related issues could have ultimately driven the finished product to a more secure solution.

Greater documentation could have been written for the Windows Server team upon the completion of the project. Very little was written regarding the scripting functionality and expected behavior, and no formal diagrams exist for the Windows Server team to reference. Ultimately, this professional project has, in many ways, defined the data delivery process between XYZ Telecom and Gilmore, Inc for the Windows Server team. An additional diagram with specific IP addresses has been written for the Windows Server team in an effort to transfer the project knowledge to the entire team.

## 6.3 Did the Project Meet Expectations?

The project met expectations and was highly successful. Ultimately, the project team did a great job at identifying the tasks that needed to be automated. The IT staff was successful at automating these identified tasks through scripting,

server and network technology. Through the automation of this former manual transfer of data, over 20 hours per month of labor have been eliminated. By reducing the time it took for the pervious manual process, an increased level of service ultimately resulted to the customers of XYZ Telecom.

## 6.4 Conclusions

The goal of any business should be to leverage technology in such a way that makes the employees more efficient. This is exactly what this project accomplished.

## 6.5 Summary

In summary, XYZ Telecom was able to improve its level of customer service to their customer base by automating an existing manual process. This ultimately saved the company money by reducing reoccurring labor costs and human error elimination. The company's increased customer service ratings translated to less customer turnover.

## Bibliography


Cisco Systems. Network Security Features on the Cisco Integrated Services Routers. 2005 Cisco Systems. Available: http://www.cisco.com/application/pdf/en/us/guest/products/ps5854/c1650/cdccont_0900aecd80169b0a.pdf March 4, 2006.

Cisco Systems. Cisco 1800 Series Integrated Services Routers. 2004 Cisco Systems. Available: http://www.cisco.com/application/pdf/en/us/guest/products/ps5853/c1167/cdccont_0900aecd80181208.pdf. March 3 2006.

Fisher Von Mallard, Michael. Gnu Privacy Guard (GnuPG) Mini Howto (English). 2004. Available: http://www.gnupg.org/gph/en/manual.html March 4, 2006.

Free Software Foundation, Inc. GnuPG Frequently Asked Questions 2002-2004 Free Software Foundation, Inc. Available: http://www.gnupg.org/documentation/faqs.html. March 4, 2006.

Ryan, Mark Dermot., Key certificates and PGP.2005 University of Birmingham. Available: http://www.cs.bham.ac.uk/~mdr/teaching/modules/security/lectures/PGP.html. March 4, 2006.

Schach, Stephen R., Object-Oriented and Classical Software Engineering, fifth edition, 2002, McGraw-Hill, ISBN 0-07-112263-X

Schwalbe, Kathy, Information Technology Project Management, third edition, 2004 Course Technology, ISBN 0-619-15984-7

Stevens, Perdita and Rob Pooley, Using UML, Software Engineering with Objects and Components, 2000, Addison Wesley, ISBN 0-201-64860-1

## Appendix A Glossary

**NTFS -** Acronym for “NT File System” which originated with Microsoft’s Windows NT operating system.

**Active Directory** – A Microsoft technology that utilizes LDAP directory structure to unify users, groups and computer accounts. A common username/password stored in this directory will allow someone to log on to various resources where that particular user has been granted access to.

**AIM** – An advanced integration module which can be added to a Cisco router to offload cycles and increase the performance of the router itself. An AIM module can integrate additional features to the existing router.

**FTP Server** – A network service used for transferring files over the network.

**Global Address List** – A term used for the address book of contact information within Microsoft Exchange. Typically accessed via a Microsoft Outlook client.

**GNU PG** – “GnuPG stands for GNU Privacy Guard and is GNU's tool for secure communication and data storage. It can be used to encrypt data and to create digital signatures. It includes an advanced key management facility and is compliant with the proposed OpenPGP Internet standard as described in RFC 2440. As such, it is aimed to be compatible with PGP from NAI, Inc.”[7]

**ISDN** – Stands for Integrated Services Digital Network. This is typically a leased, dedicated connection that carries analog or digital data through a digital connection.

[7] Free Software Foundation, Inc, GnuPG Frequently Asked Questions, 2003, Free Software Foundation, Inc., pg 1

**Microsoft Outlook** – A program written by Microsoft Corporation specifically for email communications.

**PSTN** – Stands for Public Switched Telephone Network. This is the traditional telephone network for voice traffic.

**Router –** A device used to connect two networks together. This device can route network traffic to various networks based on the traffic destination and the rules that have been defined.

**T1** – A leased, dedicated network connection that can carry up to 1.544 Mbps of data.

**VPN** – A method to communicate over a public network via an encrypted connection. This allows separate parties to communicate securely across a network that would be otherwise unsecured.