10-1999

What is Y2K and how did the University of Nevada system solve it?

Diane R. Dietrich
University of Nevada Las Vegas

What is Y2K and how did the University of Nevada System solve it?

By

Diane R. Dietrich

A professional paper submitted in partial fulfillment of the requirements for the degree of

Master
In
Public Administration

Department of Public Administration
University of Nevada, Las Vegas
October 1999

ABSTRACT

What is the Y2K phenomenon and How Did the University of Nevada System Computer Services solve the Problem a Year Early?

By

Diane R. Dietrich Graduate Student

Dr. William Thompson, Examination Committee Chair
Professor of Public Administration
University of Nevada, Las Vegas

The turn of the century change from 1999 to 2000 will create chaos for the world's computer systems. This paper will discuss the causes of the Y2K problem, the reaction to this problem by the University of Nevada System Computer Services and possible repercussions of this problem for Nevada, the United States and the world.

TABLE OF CONTENTS

ABSTRACT.....ii

TABLE OF CONTENTS.....iii

ACKNOWLEDGEMENTS.....iv

CHAPTER 1 INTRODUCTION.....1

CHAPTER 2 REVIEW OF RELATED LITERATURE.....2

CHAPTER 3 THE Y2K PROBLEM.....7

CHAPTER 4 THE UNIVERSITY OF NEVADA SYSTEM SOLUTION.....14

CHAPTER 5 A SAMPLING OF THE STATUS OF Y2K READINESS.....21

CHAPTER 6 SUMMARY, CONCLUSIONS and RECOMMENDATIONS.....30

BIBLIOGRAPHY.....41

VITA.....43

APPENDIX

Appendix A The Culprit
Appendix B Date Conversion Examples
Appendix C Century Surveillance Examples
Appendix D Version Merger Example
Appendix E Local Modifications
Appendix F HRS System Implementation Plan
Appendix G CUFS/ADVANTAGE Implementation Plan

ACKNOWLEDGEMENTS

I would like to thank Dr. Thompson and Dr. Layne of the UNLV Public Administration Department for their support and encouragement during the preparation of this paper. I would like to also thank my boss, Oscar Lewis, for his understanding and tolerance to allow me the time to complete the paper.

And finally, I would like to thank Dr. Don Zitter, who singlehandedly dragged SCS, kicking and screaming, into the 20th century. Without his leadership, we probably wouldn't be ready for the 21st.

Chapter 1
Introduction

As the last year of the 20th century approaches, the change from 1999 to 2000 will, unless corrected, create computer chaos. This potential digital disaster derives from the inability of computers, which read only the last two years of a date, to differentiate 1900 from 2000. The problem has dominated the technology media in the last year but the problem arose 40 years ago. Organizations around the world are working on the problem. This paper will explore the causes of the Y2K problem.

The University of Nevada System Computer Services has solved the problem and became Y2K compliant in January 1999, a full year ahead of time. How did a small, understaffed and underfunded organization do what the Federal Government could not? How did this computer center accomplish this task with no additional funds nor additional employees while huge corporations with many millions are still struggling? This paper will outline the steps SCS took to get the task done.

The world continues to work to solve this trivial yet far-reaching problem before January 1, 2000. This paper will summarize status of some organizations both within and without the United States of America and provide recommendations on how to eradicate the millennium bug from our lives.

Chapter 2
Review of Related Literature

The media has been filled with articles concerning the Year2000 problem, the Y2K problem or the Millennium Bug. The problem was created when computers were first used for business applications. For years, the technologists and programmers have known and discussed the possible outcomes of the "BUG" but only in the past few years has the public become informed and alarmed by the changing of the year from 1999 to 2000.

On September 6, 1993, Peter de Jager published an article called "DOOMSDAY 2000" in which he wrote "The date change in the year 2000 – an event that may trigger fatal errors in mission-critical systems – is only 2,308 days away. Many Information Systems (IS) people are unprepared or unconcerned....The information systems community is heading toward ...a failure of our standard date format: MM/DD/YY....To save storage ... most IS groups have allocated two digits to the year.... These two-digit dates exist on millions of data files used as input to millions of applications." His article continued, "How do we identify the problem and data and associated calculations? The only choice we have is to examine each line of code and make the necessary changes...One IS person performed an internal survey of his organization and came up with the following results: of 104 systems, 18 would fail in the year 2000. These 18 mission-critical systems were made up of 8,174 programs and data-entry screens as well as some 3,313 databases....this initial survey required 10 weeks of effort." He added, "The cost estimates within this article stated "organizations will each have to spend about 35 to 40 cents per line of

code to convert all their existing systems to accept the change from the year 1999 to 2000. That translates into about $50 million to $100 million for each company....We don't have a choice. We must start addressing the problem today or there won't be enough time to solve it. ...The inability of the industry to even think about such a project is troublesome...this problem is messy, expensive and unromantic. No one wants to go in and tell management they have a multimillion-dollar requirement just to keep the business running and that they really have no options."[1]

Four years later, Newsweek ran an article called "The Day The World Shuts Down" which began "Drink deep from your champagne glasses as the ball drops ... to usher in the year 2000. Whether you imbibe or not, the hangover may begin immediately. The power may go out. Or the credit card ... may no longer be valid. ...or the elevator that took you up to the party ballroom may be stuck on the ground floor." The article continued, "Or the parking garage you drove into earlier in the evening may charge you more than your yearly salary. Or your car might not start. Or the traffic lights may be on the blink...or your phones might not work. The mail may show up but your magazine subscriptions will have stopped...or your insurance policies may have expired...could this really happen? ...according to computer experts, corporate information officers...or anyone who's given the matter a fair hearing, the answer is yes, yes, 2,000 times yes!" The magazine's authors added, "What's the problem? It's called the Millennium Bug. It represents the ultimate indignity: the world laid low by two lousy digits. The trouble is rooted in a seemingly trivial space-saving programming trick – dropping the first two numbers of the date....This digital relic from the days when every byte of computer storage was

precious was supposed to have been long gone by now, but the practice became standard. While any idiot familiar with the situation could figure out that the worlds' computers were on a collision course with the millennium, no one wanted to be the one to bring it up to management....which executive would welcome a message from nerddom that a few million bucks would be required to fix some obscure problems that wouldn't show up for several years?" The article concludes with "The Gartner Group estimates a bill that could go as high as $600 billion...and that doesn't include the litigation that will inevitably follow the system failures."[2]

On December 14, 1998, Peter de Jager wrote an article for his web site called "Failure as Evidence of Effort" in which he stated "Several years ago, even as recent as nine months ago, if asked to describe worst case scenario for the effect of Y2K, it wasn't difficult to paint a gloomy picture. Not all companies had begun their Y2K project. Governments were still debating as to whether or not it was a 'real' problem and many did not know how big the task was and could not, not honestly at least, provide reasonable estimates for when they'd be able to deliver compliant systems". He emphasized "At every step of this project, the technical difficulties we faced paled in comparison to the simple fact that we were not working on the problem. Y2K is not a two digit year problem, Y2K is a problem because we chose not to work on the two digit problem. Lack of effort, not date ambiguity, was always the issue" The author continued, "By highlighting the negative consequences of inaction, we hoped to avoid them. The goal was to get a sleeping giant to awaken....if we had continued to do nothing, we'd have faced several hundred million computer problems ranging from the inconsequential to the extremely catastrophic, all occurring in a very short

period of time....Companies have spent billions of dollars already to fix a problem which the media, five years ago, described as hype and hysteria....1999 has already brought a small crop of Y2K-related problems ...[although] all the reported problems were fixed within a single day". He continued, "the bad news...is that in waking up the giant, we also scared the wits out of the general population. General perception has transformed worst-case scenarios based on lack of effort into ...bizarre stories going around [as] the 'world wide power grid' is at risk...no such thing exists. Another beauty is the one about the single computer...which controls all currency transactions worldwide...which has a Y2K problem and cannot be fixed – again, no such device...Much of this tripe is being served up by folks who have their own agendas of fear and panic. People who for years have either predicted nuclear wars, 100's millions of deaths due to AIDs or the collapse of the banking system...are charlatans of the highest order...they are using Y2K very cleverly and deliberately to create a state of panic". His article went on to say, "this stuff aside, there are legitimate concerns...Many companies and most governments have still not taken this thing seriously...In the midst of all this confusion, companies have taken action. It is possible to prove action has taken place by focusing on failure. Failure which is solid evidence of sincere effort....a few examples of Y2K effort...in 1998 Dublin City, Ireland, they replaced their traffic control system because of Y2K problems and the result was a day of traffic gridlock...SEC charging some companies for not disclosing Y2K risks in their annual reports...banks who are being rapped on the knuckles by the regulators for not doing enough to solve the problem" He added, "The Department of Defense (DOD) was caught lying about systems they claimed

compliant. Consider what went on in the background to catch them in this lie…is proof that not only is work being done to fix our systems but there are people making sure that when you say the work is done, it is actually done…It means we're now paying attention to the problem even if there are some idiots who believe they can lie their way in compliancy. The code knows. Unless fixed, it will fail. No pretense can stop it."[3]

Chapter 3

The Y2K Problem

In February 1999, a special U.S Senate committee, after almost a year of systematic investigation, concluded that "while both government and businesses have worked hard to correct the Y2K problem, their efforts began late, remain insufficient and consequently some incalculable level of economic disruption is inevitable. " The committee's chairmen warn "Make no mistake, this problem will effect us all individually and collectively in very profound ways... I will indeed impact individual businesses and the global economy. In some cases, lives could be at stake."[4]

It has never happened before, and it will probably never happen again. No technical hitch has ever been so pervasive, so expensive or so potentially damaging as the failure of many of the world's computer systems to understand the difference between dates in this century and the next. Ludicrous in its banality but frightening in the unpredictability of its consequences, the "Year 2000 problem" is already the main preoccupation of information-technology departments around the world. As the millennium approaches, it will increasingly preoccupy policy makers and the public too. The cause of the problem is ridiculously simple. In the days when computer memory was scarce, programmers got into the frugal habit of using only two digits to write a date, so 1998 was represented as 98. But, as 99 is a higher number than 00, millions of computers simply cannot place the year 2000. Instead, they may read the first year of the new millennium as the first year of this century, or as a date that does

not exist. Even if they pass that test, they may fail to notice that 2000, unlike most centennial years, is a leap year. (Among the ways devised by Christopher Clavius, a 16th-century Jesuit mathematician, to align human time with astronomical, one was to make every fourth full century year a leap year. As luck would have it, 2000 will be the first year to which this rule applies.) All sorts of functions that depend on dates will therefore go wrong, but in ways that are hard to forecast. Because the millennium-bug problem is so trivial, senior managers have found it hard to take seriously, and politicians have found it even harder.[5]

While the cause of the problem seems simple, the remedy is not. When computers were first used for business applications in the 1950s, the primary goal was to keep the file sizes small. Records within files had to be as concise as possible because memory and storage devices were scarce and expensive.

At this time, the primary input and storage device was the punch card. An example is provided in Appendix A. With a physical limitation of 80 columns per card, if more space was needed multiple cards had to be used. This complicated the programming logic and slowed down the data retrieval processes considerably. As a result, programmers used various techniques to shorten the data. Coded fields were used to shorten 'married', 'divorced', or 'single' to a one-column marital status field of 'M', 'D', or 'S'. Sex types of 'male' or 'female' were shorted to a one-column field of 'M' or 'F'. Sometimes, an alphanumeric scheme was used, allowing 36 possible values to be expressed in a 1-column field (numbers 0 through 9 plus letters A through Z). One of the most effective techniques was to truncate date fields by ignoring the first two digits of the year. Dates were shortened and stored as

YYMMDD, thus saving two digits for every date in the database. Subsequent processing would contain data edits where month (MM) would have to have a valid range of 01 thru 12, day (DD), a valid range of 01 thru 31 and year (YY) a valid range of 00 thru 99. The century digits were hard-coded as a given constant of 19.

As new storage devices like magnetic tape, storage drums and storage disk became more common, files expanding to multiple reels of tape or into drum or disk overflow areas also caused processing complications and increased input-output (I/O) time. To create the most efficient processes possible, system analysts continued to include these data shortening techniques into their system design specifications and programmers applied such techniques in their programs.

An example of these methods was in evidence at the United States Marine Recruit Depot at Parris Island, South Carolina in 1966 where I served as a section supervisor at the depot computer center. These techniques made it possible to collect and store basic data for each USMC recruit on eight punch cards per recruit. Each card had to contain the common key field of USMC identification number in card columns 1 through 7 and the card number in card column 8. This allowed the remaining 72 columns of each card to be used to hold demographic data (USMC address, home address, parents' names, recruit name), medical data (sex, blood type, height, weight) and the numerous test scores and recruit data acquired during the eight-week boot camp. These cards were physically sorted and collated, then fed into the computer to produce testing reports, platoon rosters, paychecks, supply reports and, eventually, upon graduation, orders. In the summers of 1966 and 1967, with 10,000 recruits on base, the computer center was processing 80,000 cards per report.

The base computer consisted of an IBM 1401 computer with 4K of memory, no disk storage and no tape drives, yet was able to not only process recruit system requirements but also depot supply systems, financial accounting reporting and permanent personnel payroll/personnel processing systems.

Information technologists never foresaw that these techniques would be a problem. None of us believed our programming code would last to the turn of the century. Our hope, at that time, was that the programs would be useful for a few years - the idea that they might be around four decades later was inconceivable.

The software problem we "old dinosaurs" created can be explained as a coding problem, a testing problem, and a backlog problem. Hardware chips that are imbedded in our high-tech devices create an additional problem.

The coding problem - Date Size

Because the computer cannot differentiate between 1900 and 2000 with only a two digit year field of 00, all the date fields in every record throughout the organizations' files, must be expanded from six digits (YYMMDD) to eight digits (CCYYMMDD). This means that every record which contains date fields will become larger which means that every file that contains these records becomes larger, which means that every program that reads or writes these larger files, whether it accesses date fields or not, must be changed, recompiled and tested. An example of some of the date changes made is provided in Appendix B.

The coding problem – Calculations

In virtually every program, calculations must be checked and changed so the century is included, otherwise the mathematics of the program will be faulty. For

example, to calculate the age of an employee, the last two digits of the birth year is subtracted from the last two digits of the current year. 99 - 42 gives the correct age of 57 this year, but next year 00 - 42 gives a result of -42. To correct the math for the new century date (i.e. to make the calculation Y2K compliant), the entire four-digit date is required. Thus, to calculate the correct age of 58, subtract 1942 from 2000. To calculate interest, a program would need to determine the difference between today's date and the starting date of a loan. If the start year was 99 and today's date was 00, the result would be negative. Similarly, beginning in 2000, pension benefits would be incorrect since the retirees will be viewed as not yet being born. If a person born in 1930 is calculated using 2-digit years, he/she would be -30 years old (00 - 30).[6]

The coding problem – Date printing

Just as your bank created blank checks with the '19' preprinted in the date field, most dates printed by programs in the past used the same technique, a constant value of '19' and a variable year such as '98' to print the year as '1998'. Every print routine involving dates (paychecks, invoices, loan documents, purchase and shipping documents, etc) will have to be changed to allow the century to also be a variable (i.e. listing of graduate students with projected graduation dates of 1999, 2000, 2001 and 2002). To accomplish this, a variety of coding techniques may be used. Examples of four different formulas to ascertain the correct century are included in Appendix C.

The coding problem – faulty error and end-of-file routines

Another change in the majority of programs will be locating and analyzing programming routines that designated year 00 or 99 as an end-of-file condition or an

error condition. Programmers of old used to move all 0s or 9s to indicate these conditions. In addition, odd numeric configuration dates could present problems. We seem to have survived the infamous "nines problem" dates which could have been interpreted as either nonsense dates or an order to end all processing. April 4, being the 99th day of 1999 and September 9, being the ninth day of the ninth month, were expected to trigger some of these 'nines' problems.[7] Another date with potential problems was 10-01-1999 when the US Federal fiscal year began. We still have to weather the next leap day 02-29-2000 and, of course, the big one, 01-01-2000, when the rollover actually occurs.

The Testing Problem

When the Y2K remedy is finally installed, the most time-consuming part of the process begins - testing all of the changes and fixes. Problems arise as items to be fixed were missed, items were fixed improperly and/or unexpected bugs rise to the surface as systems are tested together. Testing must involve the customer community, as programmers cannot duplicate every condition as well as knowledgeable users. The impact on an organization quickly expands from the information services department to non-IT employees who must share the testing burden.

The Backlog Problem

While the data processing staff is working on the Y2K problem, the normal day-to-day activities must continue. Emergency problems must be dealt with but enhancement requests and new requirements must be postponed until the century date problems are fixed. And when the staff finally comes up for air, this backlog of

requests is waiting. In the case of the UCCSN, HRS installation, there were 188 outstanding requests waiting for attention after the Y2K conversion was completed.

The Imbedded Chip Problem

An additional area of concern involves the imbedded hardware chips residing in devices as different as the automatic coffeepot in your kitchen and nuclear missiles nestled in their silos. Since their reaction to the century change is unknown, each device must be tested for Y2K compliance and replace or repaired if necessary.

Chapter 4

The University of Nevada System Computing Services Solution

Prior to 1990, the UCCSN computing systems were housed on two Control Data Corporation (CDC) computers, located on the university campuses at Las Vegas and Reno. Although providing some minor on-line inquiry processing, the three major application areas of student accounting, payroll/personnel and financial accounting systems. Information was provided on paper with 24-hour turn-around. The programming code was custom-designed, and in most cases, ancient. Reports were delivered to the University of Nevada, Reno (UNR), Truckee-Meadows Community College (TMCC) and Clark County Community College (CCSN) by courier, to Western Nevada Community College (WNCC) in Carson City via a daily armored-car and to Northern Nevada Community College (NNCC) in Elko, by mail. UNLV picked up its output themselves. The applications were centralized with one telecommunications link between the two computers to transfer student information north from the Las Vegas computer and payroll/personnel and financial information south from the Reno computer. In 1981, the disk capacity was so scarce that the source programs were still stored on 80-column cards, with room on disk for only the binary, machine-language code. Student registration was held in ballrooms with punched cards for class capacity (when you were out of cards, the class was full). Input was punched by centrally located keypunch operators from user-batched and courier-delivered documents. Students taking programming classes punched their programs on scarce keypunch machines then waited for hours for the computer to

process their programs, not daring to leave for fear that some other student would steal their program decks. The computer center eventually weaned itself from punched cards but only when they could no longer get parts for their equipment. The rest of the industry had given up cards years before and finally IBM stopped stocking parts for the sorters, collators, gang punchers and interpreters of a bygone age. The computer center had gotten to this backward state due to poor management and lack of funding. Because it was not included in the student head-count funding formula, the increased demand for services was not reflected in an appropriate increase in their budget. As a result, SCS management reacted to the demand for services (and correlating criticism) with a siege mentality. The previous executive director had actually moved his offices and staff completely out of the computer center building, the better to disassociate himself from the fray.

In 1988, a new executive director, Dr. Don Zitter, was hired with the mandate to modernize SCS. Declaring that it was time to join the mainstream of computing, he wrote a white paper to justify moving away from home-grown, batch applications to modern, on-line commercial packages. Instead of selecting hardware, then trying to find packages that would run on the selected platform, Zitter and the user community issued a Request For Proposal (RFP) for a student information system. Proposals were submitted by various software vendors and vendor demonstrations were scheduled throughout the state for the user community. Once the vendor package was selected, then a RFP was issued for an appropriate hardware platform, the main criteria being that it must be able to run the selected vendor package. The hardware platform selected was IBM though not without much conflict as the UNIX

proponents entered the fray. The change also resulted in the academic community being ousted from the SCS computer. Prior to this, the academic community shared the mainframe computer, conducting research and using the computer as a teaching tool. Now all academic work would reside on other hardware platforms and the IBM became a pure administrative computer. With the support of Chancellor Mark Dawson and all seven institution presidents, Zitter convinced the Nevada state legislature to partially fund the project. Because the funding was incomplete, the computing services was forced to purchase used, older equipment but did receive enough funding to procure the new student information system (SIS). Many of the old systems' features were added to the new system, especially in the areas of Financial Aid and Student loans. On January 1990, the IBM equipment was installed at the northern computer center, followed by six months of intensive training, testing and debugging. By August of that year the old student files had been converted and the Student Information System (SIS) was up and running. By March of 1991, the files and programs of the other two main administrative systems, Financial and Payroll/Personnel were converted from CDC format to IBM format and transferred to the IBM in the north and the two Control Data Systems were disconnected.

Part of the package for the SIS system included the right of the University System to purchase the same vendor's financial package for $1.00. Because of the convenience, work began converting the old financial system to the College and University Financial System (CUFS). Although the SIS system was Y2K compliant when purchased, the current CUFS system had not been converted. At this time, very

few vendor packages were Y2K compliant. The CUFS contract required the vendor to provide a Y2K release in the future. CUFS went live July 1992.

While the SIS system eventually was received and psychologically owned by the registrars and admission offices, the CUFS system was not. Some users felt they were forced to accept the new system because of the attractive price, others just didn't like the new system and wanted to stay with the old familiar batch process. Another drawback to CUFS was the size of their user community. The SIS user group was limited to a few student offices at each campus while financial activity occurred in every department throughout the system. The begrudging user acceptance would affect the vendor selection for the new payroll/personnel system.

The search for a vendor package for payroll/personnel began with the statement that the users would entertain any vendor except the one they had for SIS and CUFS. It was to replace a home-grown, customized system that had been written in 1972. After many heated meetings and seemingly endless site visits and vendor demos, a package was selected and the Human Resource System (HRS) conversion began. As with the CUFS project, (and including many of the same users), the vendor package was modified to handle seven institutions (the Desert Research Institute was included in both CUFS and HRS) for personnel activities in addition to business centers north and south for payroll processes. Modifications were needed to deal with one-check production (to handle multiple-campus earnings), grant processing, student employment, college work study, multiple job and account lines per employee and a security system that could differentiate to the employee type and department levels. Much time and energy was consumed getting the seven personnel

offices to agree on procedures and field definitions. The system was tested and given user acceptance in and went live March 1998. The package was not Y2K compliant at the time of installation but the contract stipulated that the vendor would provide releases to accomplish this in the future.

SCS had done an admirable job in converting to a new computer platform and installing not one but three state-wide, on-line application systems in 5 years. The fact that this was on time and under budget was unheard of in the public sector!

As the year 2000 crept closer, SCS began to plan for the Y2K conversion of CUFS and HRS. The advantage of using vendor-written packages, however modified, was now becoming apparent. The vendors would do the research, locate the erroneous dates and routines and send the information in software releases as part of the yearly maintenance fee. As the first Y2K releases were received, the program conversions began, using a software compare program, "Version Merger", purchased just for Y2K. The program compared the old vendor "vanilla" base release against the new, Y2K compliant, vendor "vanilla" base release and the SCS "rocky road with maraschino cherries" heavily customized system. An example of this comparison report is included in Appendix D, where the vendor old base code is identified as "B", the new, Y2K code as "N" and the customized local code as "C". The reports identified the suspect lines of code where the version differed and the conversion teams subsequently researched and changed the code where warranted. The expanded dates, non-variable century dates in print routines and faulty error conditions were identified. Because the vendor releases and the comparison program cut out the search time, SCS programmers could concentrate on changing the code

and reapplying the modifications. This saved roughly one-third of the conversion time.

After the changes had been made and tested by the programmers, the new system was moved into Quality Assurance (QA) where parallel processing was tested with the user community verifying data validity and system compliance. As the essential changes were made, the conversion also provided an opportunity to "piggy-back" local system-wide modifications. SCS took this opportunity to split the CUFS (now called ADVANTAGE) by institution, giving each campus its own region for processing and file storage. Prior to this change, the data was stored alphabetically into north and south regions. This arrangement was fine for the Desert Research Institute which was first alphabetically, but made data retrieval slower for the folks at the end of the file like UNR and Western Nevada Community College (WNCC). HRS also took this opportunity to change Northern Nevada Community College (NNCC) to their new name, Great Basin College (GBC) and correct the truncation of the Judicial College from NCJFC to NCJFCJ. An example of this code is provided in Appendix E. HRS was installed over the Halloween weekend, 1998 and the many steps required are detailed in Appendix F. ADVANTAGE went live over New Years, 1999 and their abbreviated conversion schedule is provided in Appendix G.

The conversion would not have been possible if a director had not had a vision 12 years ago, convinced a chancellor, seven presidents and the Nevada Legislature to support the vision, and acquired a computer that could handle the new technology. Success would not have been possible without a user community, who, while

grudgingly accepting the change, provided the testing, support and acceptance of the new systems.

Chapter 5

A Sampling of the Status of Y2K Readiness

While SCS solved its Y2K problem with an existing staff of 13 programmers, Nabisco, starting in 1997, has spent $36 million, with a project team of 150-200 people. Merrill Lynch, using 1000 people, has spent $425 million to change 170 million lines of code and Union Pacific changed 15,000 COBOL programs, 7,000 Assembler programs and 300,000 FOCUS programs and spent $46 million. The Canadian Imperial Bank of Commerce, who started in 1995, changed 75-100 million lines of code with 400 people at a cost of $132 million dollars![8]

The airlines computerized reservations are performing precisely as planned. United Airlines, Delta and Northwest have tested January 1, 2000 bookings and the systems all passed. The travel industry expects the period around New Year's Eve to be busy, some cruises, particularly to the Caribbean, are already sold out. The travel agents, responding to questions about Y2K-caused catastrophes, have assured the travelers that they don't expect any problems.[9] At the Brainstorm Year2000 National Symposium in San Francisco June 28-30, Walter Taylor, Vice President for Delta Technology, a subsidiary of Delta Airlines, spoke on the state of the airline industry in general and Delta Airlines in particular. He said that the FAA has done a good job fixing problems but a poor job on the public relations side of the problem. Y2K is not a competitive issue within the industry and everyone is sharing information. Last year, Delta carried 105 million passengers in 580 planes. Delta flies over 2800 daily flights, 5000 if Delta's partners are included. Delta has huge dependencies on its supply chain. This includes flight simulators, airport support systems and the flights

crews and support staff that includes 10,000 pilots, 20,000 flight attendants and 20,000 gate personnel. Delta's Y2K philosophy is if you own it, you fix it. If a manager is responsible for a system, he or she has the responsibility to make it Y2K compliant. As of May 1999, 85% of these systems had been fixed, tested and were back in production. The biggest challenge remaining is international airports and air traffic services. The airlines know which airports are of concern and they will institute a no-fly policy unless they are positive that the airspace is safe and Y2K ready. Delta has spent $125 million on Y2K and they have learned some valuable lessons. The Y2K office cannot take responsibility for systems that they do not own. The units themselves are responsible for fixing they problems of the business unit and this stopped the buck-passing. The inventory phase became very instructive when, for the first time, they knew what they really had. Delta implemented a technology review board to review all Information Technology (IT) proposed products and solutions. Now, when anyone wants new software or technology, they have to pass this board first. And lastly, Delta learned that because they scrutinized their business processes so much, they found out what was really driving their business and they could retire a lot of things that weren't helping the bottom line. [10]

Americans who go grocery shopping at the start of the new year aren't likely to find any food shortages due to the Year 2000 computer problem, according to Dan Glickman, U.S. Secretary of Agriculture. The basic food Americans expect to be in their grocery stores will be there. An interruption in the food supply is highly unlikely. Because fewer than three percent of U.S. farmers use automated systems, producers here are less susceptible to problems. One area of concern, however, is the

readiness of U.S. trade partners. The United States imports about 40 percent of its fruit, mostly bananas, and 60 percent of its seafood.[11]

The National Retail Federation (NRF) represents anyone who sells 'stuff'. The NRF launched its Survival 2000 project in 1997 and 150 members are participating in an attempt to create a shared approach to Y2K. Getting disparate retailers to get together is difficult because most of the retailers think they are unique and don't need to partner with others. Early on the NFR realized that it did no good to get retail involved unless manufacturing was along too. They began calling manufacturers and quickly began to focus on Electronic Data Interchange (EDI) which allows automated ordering of merchandise. NFR created a test to see if a 00 would pass through a translator. Over 10,000 companies have taken the test and some of the companies flunked. Outside the United States, all bets are off. One retailer spend the last two years trying to interest people in Spain to the problem and made no progress at all. In addition to the manufacturers, there are shippers, port authorities, customs and the switch to the Euro currency to worry about. There are a lot of unique issues that the retail establishment must deal with. What do you stock up on? Nobody has any kind of an idea of how many batteries people will want to buy before January. The NFR is beginning to find out if industries are going to go on the offensive and recommend buying their products. The bottled water association has decided not to take advantage of Y2K to sell more water, not because they don't want the sales but because of the fear that those big five gallon returnable bottles won't be returned. They are afraid of running out of containers! Retailers, like everyone else, are committed to due diligence but with Y2K, it is difficult to

determine what due diligence is. Nobody has been through Y2K before and there is no precedent.[12]

At a conference in Austin, Texas, hosted by the Texas Year 2000 Office, officials stressed that prison computer systems and embedded computer devices won't open prison doors nor will they issue orders to free everyone. Prisons have contingency plans because something is always going wrong in prisons. This is because there is a group of very dedicated people who are always attempting to break the systems. They are called convicted felons. Prisons are checking for embedded bugs. Federal spokespersons at the conference identified some of the mission critical systems they are fixing. Some of these were the National Crime Information Center (which lists wanted and missing persons and stolen property), The Integrated Automated Fingerprint Identification System (which also contains criminal histories) and the Passenger Accelerated Service System (that tracks frequent travelers entries into the country and also responds to illegal border crossings). The Bureau of Prisons also reported that they had nine mission-critical areas of concern: telephone systems, elevators in high rise facilities, radio systems, perimeter detection systems, programmable controllers for security systems, fire alarm systems, entry management systems and closed circuit television and surveillance systems.[13]

The status of the Federal government's Y2K compliance is a mixed boat. In an open letter to President Clinton, Peter de Jager cited a September 1998 report from U.S. Representative Stephen Horn which stated: The Department of Defense has some 2,965 mission-critical systems which will not be fixed until some time in 2001; The Department of Labor, 61 mission-critical systems, not 100% ready until 2001;

The Department of Health and Human Services, 298 mission-critical systems, not 100% read until 2002; The Department of Energy, 411 mission-critical systems, not 100% ready until 2002; the Department of State, 50 mission-critical systems, not 100% ready until 2027 (not a misprint, the department estimates they will not be able to provide full service for the next 27 years); the Department of Justice, 207 mission-critical systems, not 100% ready until 2030+ (the + sign means they have no idea when they will be ready); the Department of Education, 14 mission-critical systems, not 100% ready until 2030+.[14]

Since then, the Federal government seems to be making slow progress but steady progress, according to a quarterly report issued March 1, 1999 by the House Sub-Committee on Government Management Information and Technology. Subcommittee chairman, Representative Stephen Horn, while praising some agencies for making 'truly remarkable progress', said the Department of Transportation is moving 'at a snails' pace' with only 53% of the departments' systems compliant, and called the progress report by the Department of Defense "alarming". Five agencies, including the Social Security Administration, received grades of 'A'.[15]

The Bureau of Reclamation dams, power plants, water delivery systems and business operations have been inventoried, tested, renovated where necessary, and have contingency plans in place to operate successfully into the year 2000...starting an aggressive program in 1996, Reclamation certified to the Department of Interior that [it] is Y2K ready.[16] The Department of State announced that the process to produce U.S. passports for U.S. citizens is Y2K compliant. The systems used in

producing passports, including all system hardware, software and interfaces, have successfully completed Y2K testing and are ready for year 2000.[17]

The U.S. Postal Service's top technology officer stated that he could not promise that the agency would not encounter Y2K problems in its' computer systems. USPS must still address major issues to complete systems and mail processing equipment corrections and testing, ensure the readiness of hundreds of local facilities and interface with partners in order to be Year2000 ready. He added that the USPS should increase its' focus on contingency planning. Representative Stephen Horn said that among all the federal agencies, it is particularly important that USPS be Y2K compliant because many federal agencies and businesses plan to use hand-delivered mail in case electronic systems fail. USPS will spend a total of $607 million to fix the Y2K problem. It recently reported that 70% of its' 156 mission-critical systems are now compliant.[18] On September 7, 1999, the USPS reported that 99.3% of their business applications were remediated. This includes financial, marketing, mail operation systems. The mail processing equipment was 93% compliant.[19]

A hospital in New Zealand tested 6 patient-controlled intravenous pumps and found 4 were not compliant. Two would have allowed the patient to double-dose if one dose was applied before midnight and one after. The other two did the same, then quit working altogether. The disturbing aspect was a written statement from the manufacturer assuring the hospital that the devices were compliant. The same hospital tested back-up generators under a full load but they quit after three hours.

The generators had an automatic cut-off built into their program. The hospital is now considering back-up generators for their back-up generators.[20]

On Friday, December 11, 1998, 120 countries attended a Y2K coordinators meeting at the United Nations. Assessments of what might happen varied wildly, especially concerning developing countries. These countries don't use as much technology but the technology that is used is often older and relies on software that cannot be updated. In the developed world, the sheer number of computer systems that require checking and correction is the main obstacle.[21] The poor economical condition of some of the countries attending was shown when two-thirds of the attendees had to rely on the World Bank to pay their airfare. Said Howard Rubin, a research fellow in Stamford, Connecticut, "if they can't afford airfare, how are their nations going to fund Y2K?"[22]

On February 3, 1999, the man leading the Russian effort to solve the Y2K problem asked NATO and the U.S. Defense Department for advice and $3 billion dollars in aid. Russia has already agreed to let NATO experts investigate the potential danger to Russian weapons systems. While an errant missile launch brought about by computer clock failure would be unlikely, computer snags could sabotage radar and telecommunication networks that are the backbone of Russia's' system to detect foreign launches. U.S. defense agencies want to place American officers in Russian nuclear control rooms and Russian officers in American control rooms to monitor the century changeover.[23] As of September 1999, the United States and Russia have agreed to jointly staff a Colorado command center that will monitor missile early warning data to avert potential mishaps caused by the Y2K computer

bug. For a few weeks in late December and early January, U.S. and Russian military officials for the first time will sit side by side and review information generated by satellites programmed to detect missile activity around the world. U.S. officials, who have struggled for years to encourage this kind of cooperation, hope the Y2K experience will lead to creation of a permanent joint early warning center in Moscow.[24] Weapons are not the only danger. The nuclear plants may not be able to get accurate temperature information and there could be another Chernobyl. Meanwhile, in China, a survey of that country's' most critical enterprises showed that more than half didn't even know how to detect the computer glitch in their systems.[25]

Air Force General John Gordon, Deputy Director for the Central Intelligence Agency (CIA), told a Senate Armed Services committee that Russia appears particularly vulnerable, raising concerns about the safety of its' missiles, nuclear plants and gas pipelines. A major concern is a mid-winter power outage that could have grave humanitarian consequences for more than one-third of Europe's' natural gas and is run by Soviet-era mainframe computers highly likely to contain Y2K imperfections.[26]

In a report to the Brainstorm Year2000 National Symposium Series that took place in San Francisco in June 28-30, Stephanie Moore, who had spent eight months in Europe this past year, reported on the status of countries concerning their Y2K preparedness. The common denominators in Europe are the general lack of awareness, vulnerabilities in small and medium sized businesses and almost no contingency plans. Many countries are giving out little or no information and often what they do say is 'everything is just fine' variety. She cited Mexico as an example,

where a government representative said in a press release that Mexico was almost finished and everything was being fixed yet only 5% of the people even know what Y2K is. The biggest problem is still a general lack of executive level awareness and sponsorship, similar to the situation in the United States two years ago. Other reasons for such slow progress are the Asian financial crisis, the Japanese banking problems, the Russian economy, and the introduction of the Euro monetary conversion. She produced a list of the best and worst prepared nations, but with the caveat that most of the rating systems used to determine the level of compliance are bogus at best. First on the list of least prepared was Latin America, the Pacific Rim and Eastern Europe which are all still in the awareness and inventory stage, still trying to discover just what and where the problem is. Next was Europe, which is at the impact assessment and repair stage. Following was the US, UK, Canada and Australia which are in the test and implement stage. Few countries are doing contingency planning. In Japan, it is very difficult for the Japanese banks to inquire about the level of compliance of their customers because it would imply they don't trust them. This is also a problem for Japanese businesses who should be checking their supply chains. [27]

Chapter 6

Summary, Conclusions and Recommendations

As Y2K looms closer, what conclusions can we draw from the overload of information available? Will we experience catastrophe or merely inconvenience?

In this paper, I have explained the origin of the Y2K "bug" and how it evolved into such a costly problem to solve. I have described how the University of Nevada System Computing System, through a director's foresight, corrected its administrative software a year ahead of time, within a normal budget and without any additional staff. I have sampled the information available and detailed some of the status of Y2K readiness in organizations both in the United States and throughout the world.

I have no crystal ball to provide a glimpse into the future but I have found some information about what organizations are doing to prepare for the future and some "best guesses" about what the future may hold in store.

Alan Greenspan, Federal Reserve chairman, is concerned that stock piling and inventory hedging against Year 2000 failures may cause more trouble than Y2K itself. He noted that private companies have spent 50 billion dollars on repair efforts and the chance of cascading failures is negligible. He cautioned, however, that the economy could still be effected, depending one how many businesses and consumers alter their normal buying patterns because of unwarranted fears. Should a large number of companies decide to hold even a few extra days of inventories, the increase in production to accommodate such stock building could be quite large.[28]

Whether or not the millennium bug cripples the world's computers, the U.S. economic forecasts raising growth expectations for late 1999 and dimming them for

2000. Economists believe businesses, fearing computer errors may hamper distribution systems, will gather a war chest of inventories and nervous customer purchases of survival items as bottled water, canned food and diesel-powered generators could enhance the effect. Once the next century begins, companies and consumers will find themselves overstocked and will rein in their purchases for several months.[29]

The Federal Reserve Bank plans to have $200 billion available in late December, $50 billion more than usual. The Federal Reserve, the Red Cross and even the President's Y2K Council recommend that people gradually put away about one weeks worth of cash on the unlikely event that a few banks or ATMs may be hit by temporary computer errors or power outages. Experts recommend taking the 'Y2K' money in traveler's checks, which are federally insured and replaceable, if stolen or lost. The American Express Corporation, the largest issuer of traveler's checks, says its systems are Y2K ready.[30]

Banks hope to allay customer fears by touting their progress in solving the Y2K bug. They are concerned that there may be an armored car shortage to transport the extra cash. The contingency is to use a bank vehicle but to use the same person to get the cash as the Federal Reserve will not give cash to just anybody. Banks are also stocking up on paper so they can meet the anticipated rush when everyone asks to have a printed copy of their statements at the end of December.[31]

The nation's financial markets are fully prepared for the year2000 date change. The Securities industry has tested and retested all its computer systems, insuring a trouble-free transition to the 21st Century. Arthur Levitt, chairman of the

Securities and Exchange Commission maintained that "the worst that we have to fear right now is public misperception, public fear'. The SEC has taken enforcement actions again 13 money management firms, 15 securities transfer agents and 37 brokerage firms – all relatively small companies – for allegedly failing to meet deadlines for fully disclosing their computer systems readiness for the Year 2000. The agency has said that it will go to federal court starting December 1 to shut down brokerage firms and transfer agents that aren't ready for the date change. The SEC estimates that only one percent of the 3,900 brokerage firms and 600 security transfer agents aren't ready.[32]

Recognizing that they won't be able to fix all of their 2000 problems in time, many companies have begun to build crisis-management centers from which they will control damage and coordinate the recovery of failed systems during the millennium rollover. As of the end of May, 1999 about 85 percent of Fortune 1000 companies plan to build Y2K command centers for assessing damage caused by the date bug and form managing business recovery efforts. Business people are starting to realize the consequences of their computer systems going down. These command centers will be dealing with customers and suppliers.[33]

The crisis-management centers and other contingency plans will also be the focus of litigation to shield businesses from a potential flood of Y2K computer-related lawsuits. Congress recently passed the Y2K bill which gives businesses 90 days to fix Y2K problems before lawsuits can be filed. It also encourages mediation and seeks to limit frivolous lawsuits by capping punitive damages for small businesses and narrowing the sphere of class action lawsuits.[34]

One of the first victims of the millennium bug was Mark Yarsike, the proud owner of a new, upscale grocery store in Michigan. He believed he was prepared for the throng of customers at the grand opening of his store in 1995. Then the Y2K bug struck his high-tech cash registers, causing them to crash when customers used credit cards with expiration dates of '00' or '01'. The machines mistakenly recognized the cards as nearly a century too old. "we could not process a single credit card or could not take cash or checks'. Yarsike eventually replaced the registers and sued the system manufacturer. He settled his case in late 1998 for $260,000. Since then, lawyers nationwide have filed more than four dozen cases again high-tech companies.[35]

The bill passed by Congress will limit the effect of such lawsuits but will not completely protect companies from liability. According to Ian Hayes of Clarity Consulting, software litigation will be the ambulance chasing of the future. He pointed out that very few lawyers know how to do a technology lawsuit and there is very little case law in this area. This means that 'this is the biggest lawyer technical training program ever devised'. Because there will be so many lawyers not well versed in technology, they will demand and receive jury pools composed entirely of non-technical people. The future will be decided by people who have never touched a computer and can afford to spend months in a courtroom being paid $5 a day. Hayes thinks this is what will come out of those courtrooms:

1. Courts will decide what the standards of software will be. They will have to simple enough to be understood by a jury. This means there will have to be much more documentation and testing.

2. There will be lots of legislation after the fact. Along with the new standards, there will be regulations and rules for testing and probably professional certification and licensing. Along with this will follow professional liability and, eventually, malpractice insurance.

3. There will be a rise in outsourcing, as CIOs send their technology responsibilities and cost off to vendors who can guarantee quality and lawsuit protection.

4. A guarantee on the work – the net effect of lawyers, case law and consumer demand will be a much higher level of software quality. It will change the way software is written and cause the computer business to become a mature industry.

5. This will change the way software is built. Reusability will be more important and there will be more testing.

Hayes went over the list of winners and losers:

The Winners:

1. Lawyers

2. Outsourcers

3. Vendors offering testing and QA control tools

4. The first methodology and tool set that supports the new standards

5. The public in the long run (in the short run, they will be quite unhappy)

The Losers:

1. Computer professionals (short term)

2. Companies with large bases of old software

3. Vendor tools that don't guarantee quality

4. The concept of software programming as an art

(nobody will do it for fun anymore)[36]

Another viewpoint of the future was presented at the Brainstorm Year 2000 National Symposium that took place in Chicago, September 22-24, 1999 by Martyn Emery, founder and CEO of Corporation 2000, Incorporated. He stated that we really do not know what is going to happen with the problems that are expected to arise in a few months. We know that there are countries that have done very little to fix their computers and that some of their basic functions, such as communications and even customs, probably won't work for some period of time. We can make some educated guesses about what will break where, but we don't really know. Martyn focused on the behavioral dimension of Y2K and that the media has not really focused on this aspect, beyond whether people will panic or not and the potential problems from hackers and viruses created just for the millennium change. He mentioned that the rollover will happen during the month of Ramadan, which means that the impact on the countries where Islam is the dominant religion will be different from the rest of the world. Another factor is that January 1 in Islamic countries is a Saturday, which is a normal business day and not a holiday. The Islamic countries will actually be the first in the world to experience Y2K during a normal business day. Martyn sees Y2K less as a survival issue but more as a major competitive issue that will define the major layers in the world economy in the new millennium. With the massive worldwide shift to e-business, if a company or country fails at Y2K, they will also fail to be a player in the e-commerce revolution, knocking themselves out of the running

for the next decade or longer. A company wouldn't last that long, but a whole country could easily lose out to the competitors who finished early, then cleaned up. An example is the country of Mauritius in the Indian Ocean, who sees Y2K as an opportunity to leap frog their businesses into the world economy using e-commerce. They have a government-wide Y2K program that is working very hard on all aspects of the problem. They are advertising themselves as a safe haven for money from Southern Europe or Africa. Mauritius is stockpiling as a contingency and to obtain a competitive advantage. Martyn expects a lot of non-cooperative behavior with Y2K and explained that in times of turmoil, savvy investors can take advantage and make money.[37]

The Federal Bureau of Investigation plans to warn state and local police to be alert for possible attacks at the turn of the millennium by hate or apocalyptic groups or lone wolf members of them. The bureau intends to distribute a 40-page research report, analyzing 'the potential for extremist criminal activity in the United States by individuals or domestic groups who attach special significance to the year 2000'. The significance is based primarily upon apocalyptic religious beliefs or political beliefs concerning the New World Order conspiracy theory. "Such ideologies motivate violent white supremacists who seek to initiate a race war; apocalyptic cults which anticipate a violent Armageddon; radical elements of private citizen militias who fear that the United Nations will initiate an armed takeover of the United States and subsequently establish a One World Government, and other groups of individuals which promote violent millennial agendas" the FBI said.[38]

In the meantime, a Scottsdale cryonics organization, Alcor Life Extension Foundation is making provisions to ensure that 36 people's bodies – or in some cases, only their heads – remain in deep-freeze through the turn of the millennium.[39]

Some people are warning New Year's celebrators that they had better hurry if they want to rent a tuxedo for the celebration.[40] Las Vegas is gearing up for anywhere from 750,000 to 1 million visitors who are expected to celebrate the arrival of the New Year.

Clark County transportation experts say that residents should be more worried about transportation problems stemming from the huge number of people invading Las Vegas for the New Year's party than from the Y2K computer bug. The big concern is that with so many people drinking and having a good time, there is a good chance that traffic lights and signals are going to be vandalized. The county's traffic lights will not be down because of Y2K. The multimillion-dollar high tech computer system that synchronizes traffic signals in the Las Vegas Valley is already Y2K compliant, partly because the county and city are upgrading the system. [41]

Other Las Vegas Valley utilities are also ready to face the Millennium Bug. Nevada Power Company says its 98 percent Y2K ready for critical functions. The company said that it will have 25 percent of its generating capacity in reserve for New Years Eve. Southwest Gas Corporation said all of its 'business-critical systems will run normally on New Year's Day and beyond'. Vendors also promised Southwest Gas they are Y2K compliant. 'Unlike many businesses, the natural gas industry is easily operated manually', said Senior Vice President Dudley Sondeno.

Southwest Gas will be staffing key locations to operate the equipment manually if an unforeseen computer problem develops.

The Las Vegas Valley Water District can operate for limited of periods of time without power. If anything goes wrong with power supplies for the pumps that bring water from Lake Mead, customers will have four to five days of supplies. That's the amount the district will store in reservoirs uphill from customers so gravity can deliver the water, explained Jim Ellsinor, Director of Information Systems. Sprint, the dominant provider of local telephone service in Southern Nevada, has been completely tested and is ready for Y2K. Long distance service is 97 percent tested and will be 100 percent ready by November, 1999. 'Time off and vacation time have been canceled for most people that are in critical positions in case there are problems' said Dawn Merritt, a Y2K specialist at Sprint.

Worriers might note that Las Vegas gets most of its gasoline and some of its jet fuel from two pipelines leading from the Los Angeles harbor. If the pipeline goes down for a day, Las Vegas will never know the difference. If the line goes down for a week, the solution will be to ship fuel by truck, said GATX Terminals spokesman, Aaron Hoffman. Paul Langland, a spokesman for Atlantic Richfield, the dominant gasoline retailer in Las Vegas, expects no problems. Arco stations will build up gasoline supplies during December to make sure no shortages occur.

When the calendar flips to 2000, it's unlikely a Y2K bug will alter operations of video poker or slot machines. 'It won't affect the machines at all – a lot of them don't even have dates and times in them' said Randy Adams, an executive with Anchor Gaming. [42]

At the University of Nevada, Las Vegas, the campus, which is a mere two long blocks from the Las Vegas Strip, will be shut down and surrounded by armed guards throughout the weekend. The system computer center will be powered down after the administrative systems have completed the year-end closing processes. The system programmers will be able to join in the New Year's festivities but all vacation leave has been cancelled from January 3 until any stray Y2K bugs are eradicated. The local hospitals will be on alert, with full staffs as will the police, Nevada Highway Patrol and fire departments, ready to welcome the millions of visitors.

As one of the old "dinosaurs" who helped create this Y2K mess, and as one of the many programmers who howled in the wilderness for these past decades, warning about our two-digit dates, and as one of the folks that were amazed that our old code lived so long, I can only say it has been an interesting ride. I salute all the folks who have been working so long and so hard to fix the bugs. To all the managers who wouldn't listen, I say "I told you so". And for all of us as we enter this new millennium, I recommend that you fill up the car with gas, purchase flashlights and/or batteries (preferably before December 29th), have some cash or travelers checks on hand (just in case your bank is one of the few that didn't get its act together), gather an ample supply of your beverage of choice and keep the phone number for Papa John's Pizza handy. Stay away from the Strip and don't fly to Russia.

HAPPY NEW YEAR!

BIBLIOGRAPHY

[1] DOOMSDAY 2000 by Peter de Jager, ComputerWorld, September 6, 1993, www.year2000.com/archive/cw-article

[2] The Day The World Shuts Down, Newsweek, pp.53-59, June 2. 1997.

[3] Failure as Evidence of Effort. By Peter de Jager, 12/14/1998, www.year2000.com/archive/y2kfailure.html

[4] Rankin, Robert. (1999,February 24). Yearlong inquiry warns of Y2K risks, Las Vegas Review-Journal

[5] A Survey from the Economist, The Millennium Bug, http://www.Year2000.com/archive/Nfeconomist.html

[6]Stern, Nancy & Stern, Robert. (1999) Structured COBOL Programming, (8th Edition, Year2000 update Edition).(Chapter19, pp1-2). New York: John Wiley & Sons, Inc.

[7] Rolling thunder: When the Bug will strike. By Mitch Ratcliffe, 2/6/99, www.msnbc.com/news/227483.asp

[8] (1999, February 18). Computerworld, pp., 24-28.

[9] Travel Industry says systems Pass Test, www.msnbc/com/news/238462.asp

[10] Y2K ANNOUNCE: Brainstorm SF Conf, Part 2, Year2000.com Announcement List, Special Mailing, August 4, 1999, www.year2000.com

[11] Expert: Food system safe from Y2K, www.msnbc.com/news/238750.asp

[12] Y2K ANNOUNCE: Brainstorm SF Conf, Part 5, Year2000.com Announcement list, Special Mailing, September 10, 1999, www.year200.com

[13] Jon Huntress, Year2000.com announcement list, special meeting, January 13, 1999. Prisons and Y2K: Is there a problem? www.year2000.com

[14] Peter de Jager, "Open Letter to President Clinton", November 17, 1998, www.year2000.com/archive/y2Kclinton

[15] Bob Violino, March 1, 1999, Informationweek, Issue:723, www.techweb.com/se/directlink.egi?IWK19990901IS0037

[16] 'Bureau of Reclamation Now Ready for Y2K', Henderson Home News, September 2, 1999.

[17] Successful Y2K Test of U.S. Passport Technology, http://travel.state.gov/ppt_y2k.html

[18] USPS can't promise mail delivery won't encounter Y2K bugs, by Colleen O'Hara, www.cnn.com/TECH/computing/9902/25/usps.idg/index/

[19] Year 2000: Looking to the Future, www.usps.gov/year2000/fix.html

[20] Y2K: A matter of life and death by Paul Brislen, Computerworld, February 1, 1999, pp. 1-4, www.year2000.com.nz/y2kcw31

[21] Year 2000 bug topic of conference. (1998, December 11). Las Vegas Review-Journal.

[22] Sidebar. (1999,January 11-17). Computerworld, p44.

[23] Russia concedes Y2K is a problem. February 3, 1999, www.msnbc.com/news/238057.asp

[24] 'U.S., Russians will collaborate on Y2K', Las Vegas Review-Journal, p. 3A, September 11, 1999

[25] Russia concedes Y2K is a problem. February 3, 1999, www.msnbc.com/news/238057.asp

[26] Many nations face disruptions from Y2K glitch by Jim Abrams, Associated Press, www.bergen.com/morenews/y2k25199902254

[27] Y2k ANNOUNCE: Brainstorm SF Conf. Part 1, Special Mailing, July 21, 1999, www.year2000.com

[28] "Greenspan Warns Against Y2K Panic', Las Vegas Review-Journal, September 19, 1999, p. 7A

[29] 'Y2K Stockpiling May Fuel U.S. GDP', www.foxmarketwire.com/wires/0227

[30] 'Thousands Planning to Stockpile Money', Las Vegas Review-Journal, October 4, 1999, p.1A

[31] Year2000.com Announcement List, May 27, 1999, www.year2000.com

[32] 'Officials Allay Y2K Stock Fears', www.ap.org/Apnews/center_package...AGEID=millenium

[33] 'Companies Begin to Create Y2K Crisis Centers', www.foxnews.com/scitech/background/Y2K/cmp-052999.sml

[34] 'Congress Approves Y2K Bill', Las Vegas Review-Journal, June 30, 1999, p. 2A

[35] 'Millennium Backlash: Companies Brace for Lawsuits Linked to the Y2K Glitch', Las Vegas Review-Journal, February 28, 1999, p. 1K-2K.

[36] Brainstorm Chicago Conference, June 22-25, 1998, 'After 2000 – What Ian Hayes Thinks the Future Will be if we blow it with Y2K' by Jon Huntress, www.year2000.com/archive/huntress/hchicago3.html

[37] Y2K ANNOUNCE: BrainStorm Chicago Conf. Part 1, Year2000.com Announcement list, Special Mailing, October 6, 1999, www.year2000.com

[38] 'FBI warns police to be on alert for fringe millennium violence', Las Vegas Review-Journal, October 21, 1999, p. 9A

[39] 'Group makes Y2K provisions to protect those in crostasis', Las Vegas Review-Journal, October 10, 1999, p. 12B.

[40] 'If you're going to rent or buy a tux for New Year's, hurry', Las Vegas Review-Journal, October 21, 1999, p. 1E

[41] 'Millennium bug doesn't faze officials', Las Vegas Review-Journal, July 8, 1999, p. 7B

[42] 'Companies confident for 2000', Las Vegas Review-Journal, October 24, 1999, pp. 1A, 18-19A.

VITA

Graduate College
University of Nevada, Las Vegas

Diane R. Dietrich Graduate Student

Degrees:

Bachelor of Science, Business Management, 1979
San Jose State University, San Jose, California

Associate of Arts, Chemistry, 1962
Coalinga College, Coalinga, California

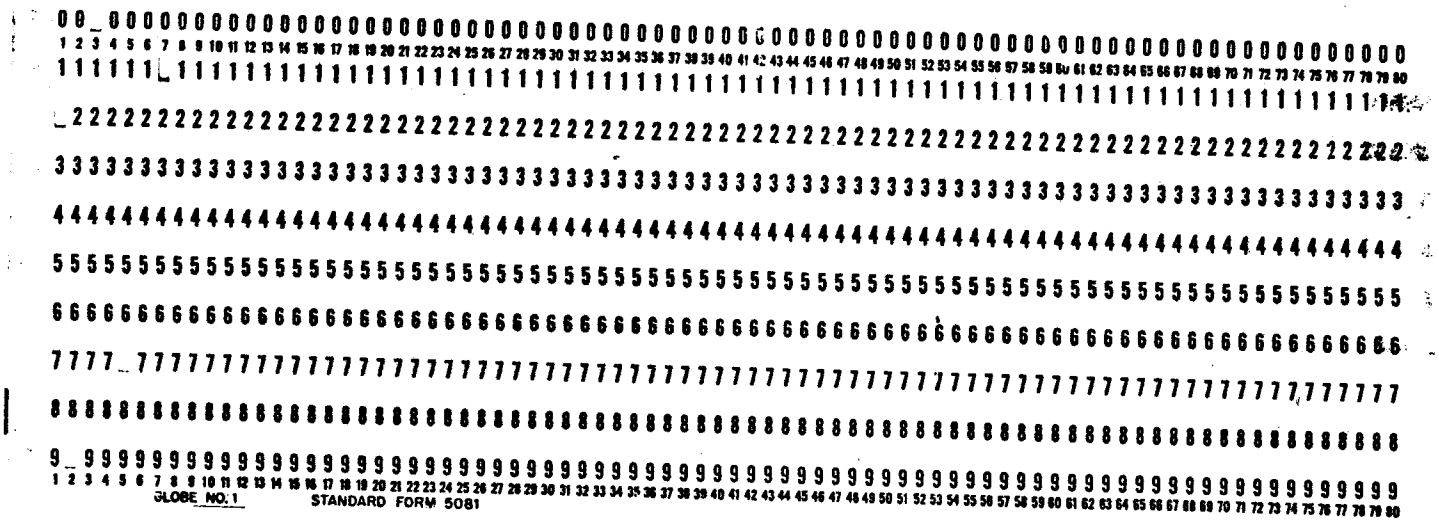
Professional Paper Title:

What is Y2K and how did the University of Nevada System Solve It?

Professional Paper Examination Committee:

Chairperson, Dr. William Thompson, Ph. D.
Committee Member, Dr. Leonard Goodall, Ph. D.
Committee Member, Dr. Karen Layne, Ph. D.
Committee Member, Dr. William Newman, Ph. D.

290-P-1



GLOBE NO. 1 STANDARD FORM 5081

APPENDIX A

```
**********************************************************
* OLD     P E R S O N A L   D A T A   S E G M E N T   # 1 *
**********************************************************
    10  XDBS-STATUS-EFF-DATE.
        15  XDBS-STATUS-YR          PIC XX.
        15  XDBS-STATUS-MO          PIC XX.
        15  XDBS-STATUS-DA          PIC XX.

    10  XDBS-EMPLOYMT-DATE.
        15  XDBS-EMPLOYMT-YR        PIC XX.
        15  XDBS-EMPLOYMT-MO        PIC XX.
        15  XDBS-EMPLOYMT-DA        PIC XX.

    10  XDBS-NEXT-SAL-REVIEW-DT.
        15  XDBS-NXT-SAL-REV-YR     PIC XX.
        15  XDBS-NXT-SAL-REV-MO     PIC XX.
        15  XDBS-NXT-SAL-REV-DA     PIC XX.

    10  XDBS-RETRMT-PLAN-DATE.
        15  XDBS-RETRMT-PL-YR       PIC XX.
        15  XDBS-RETRMT-PL-MO       PIC XX.
        15  XDBS-RETRMT-PL-DA       PIC XX.

    10  XDBS-HEALTH-PLAN-DATE.
        15  XDBS-HEALTH-PL-YR       PIC XX.
        15  XDBS-HEALTH-PL-MO       PIC XX.
        15  XDBS-HEALTH-PL-DA       PIC XX.

    10  XDBS-ORIG-HIRE-DATE.
        15  XDBS-ORIG-HIRE-YR       PIC XX.
        15  XDBS-ORIG-HIRE-MO       PIC XX.
        15  XDBS-ORIG-HIRE-DA       PIC XX.

    10  XDBS-LAST-PERS-ACTN-DT.
        15  XDBS-LAST-PERS-ACTN-YR  PIC XX.
        15  XDBS-LAST-PERS-ACTN-MO  PIC XX.
        15  XDBS-LAST-PERS-ACTN-DA  PIC XX.
```

```
**********************************************************
* Y2K     P E R S O N A L   D A T A   S E G M E N T   # 1 *
**********************************************************
    10  XDBS-STATUS-EFF-DATE.
        15  XDBS-STATUS-CC          PIC 9(02).
        15  XDBS-STATUS-YR          PIC 9(02).
        15  XDBS-STATUS-MO          PIC 9(02).
        15  XDBS-STATUS-DA          PIC 9(02).

    10  XDBS-EMPLOYMT-DATE.
        15  XDBS-EMPLOYMT-CC        PIC 9(02).
        15  XDBS-EMPLOYMT-YR        PIC 9(02).
        15  XDBS-EMPLOYMT-MO        PIC 9(02).
        15  XDBS-EMPLOYMT-DA        PIC 9(02).

    10  XDBS-NEXT-SAL-REVIEW-DT.
        15  XDBS-NXT-SAL-REV-CC     PIC 9(02).
        15  XDBS-NXT-SAL-REV-YR     PIC 9(02).
        15  XDBS-NXT-SAL-REV-MO     PIC 9(02).
        15  XDBS-NXT-SAL-REV-DA     PIC 9(02).

    10  XDBS-RETRMT-PLAN-DATE.
        15  XDBS-RETRMT-PL-CC       PIC 9(02).
        15  XDBS-RETRMT-PL-YR       PIC 9(02).
        15  XDBS-RETRMT-PL-MO       PIC 9(02).
        15  XDBS-RETRMT-PL-DA       PIC 9(02).

    10  XDBS-HEALTH-PLAN-DATE.
        15  XDBS-HEALTH-PL-CC       PIC 9(02).
        15  XDBS-HEALTH-PL-YR       PIC 9(02).
        15  XDBS-HEALTH-PL-MO       PIC 9(02).
        15  XDBS-HEALTH-PL-DA       PIC 9(02).

    10  XDBS-ORIG-HIRE-DATE.
        15  XDBS-ORIG-HIRE-CC       PIC 9(02).
        15  XDBS-ORIG-HIRE-YR       PIC 9(02).
        15  XDBS-ORIG-HIRE-MO       PIC 9(02).
        15  XDBS-ORIG-HIRE-DA       PIC 9(02).

    10  XDBS-LAST-PERS-ACTN-DT.
        15  XDBS-LAST-PERS-ACTN-CC  PIC 9(02).
        15  XDBS-LAST-PERS-ACTN-YR  PIC 9(02).
        15  XDBS-LAST-PERS-ACTN-MO  PIC 9(02).
        15  XDBS-LAST-PERS-ACTN-DA  PIC 9(02).
```

APPENDIX C

```
000074          MOVE Q000-TRAN-DATE TO XMDY-YYMMDD.
000075          MOVE CORR XMDY-YYMMDD TO XMDY-CCYYMMDD.
000076          IF XMDY-YY OF XMDY-CCYYMMDD > '94'
000077             MOVE XMDY-CENTURY-19 TO XMDY-CC OF XMDY-CCYYMMDD
000078          ELSE
000079             MOVE XMDY-CENTURY-20 TO XMDY-CC OF XMDY-CCYYMMDD
000080          END-IF.
000013     *
000014          MOVE CORR XMDY-YYMM              TO XMDY-CCYYMMDD.
000015          IF XMDY-YY OF XMDY-YYMM < 50
000016            MOVE XMDY-CENTURY-20           TO XMDY-CC OF XMDY-CCYYMMDD
000017          ELSE
000018            MOVE XMDY-CENTURY-19           TO XMDY-CC OF XMDY-CCYYMMDD
000019          END-IF.
000015          MOVE CORR XMDY-YYMMDD        TO XMDY-STD-INPUT.
000016          MOVE XMDY-STD-INPUT          TO XTAT-PAY-PER-DATE.
000017          IF  XMDY-DATE-IS-VALID
000018          AND XTAT-PAY-PER-DATE NOT = ZEROS
000019                              AND NOT = ALL '9'
000020              MOVE CORR XMDY-YYMMDD    TO XMDY-CCYYMMDD
000021              IF  XMDY-YY OF XMDY-CCYYMMDD > '90'
000022                  MOVE XMDY-CENTURY-19 TO XMDY-CC OF XMDY-CCYYMMDD
000023              ELSE
0[illegible]24                  MOVE XMDY-CENTURY-20 TO XMDY-CC OF XMDY-CCYYMMDD.
0[illegible]09     *
000210          IF XMDY-YR OF XMDY-CCYYMMDD > '1899' AND < '1950'
000211              MOVE '20' TO XMDY-CC OF XMDY-CCYYMMDD.
000212     *
000213          MOVE XMDY-CCYYMMDD
000214            TO XDBS-APPT-END-DATE              OF R95-20-SEGMENT.
000215     *
000216          MOVE XDBS-APPT-ENTRY-DATE          OF R94-20-SEGMENT
000217            TO XMDY-YYMMDD.
000218          PERFORM ADD-CENTURY-YYMMDD-9200.
000219          PERFORM CHECK-FOR-VALID-DATE-9300.
```

```
VPSPRINT 6.1.018   THURSDAY JANUARY 8,1998      10:32:33 UADDRD1  UNS1        **
VPSPRINT 'UADDRD1.FROG4.DATA(*)' U9000 CLASS(X) COP(1)

PSVM      + UADDRD1      &              BBASE    SCLM       96/06/2715:07AADHRSS.PROD.SRCVEN.OLD                CPOTXCMF    CCUST
   N  *    COPYID=CPOTXCMF WRITTEN 11/01/96 RELEASE 09.05.00  HRMS.
  C   000100*    COPYID=CPOTXCMF WRITTEN 11/15/96 RELEASE 09.04.05  HRMS.
 B    *    COPYID=CPOTXCMF WRITTEN 11/01/93 RELEASE 09.03.00  HRMS.
 BCN  *01  COMM-TAB.
 BCN  03 CURRENT-REC-KEY.
 BCN  05 CURR-KEY15.
 BCN  07 CURRENT-ENT-ID-NUM.
 BCN  09 CURRENT-XORG-KEY  PICTURE X(18).
 BCN  09 CURRENT-ID-NUM    PICTURE X(9).
   N  07 CURRENT-EFF-DATE     PICTURE 9(8).
 BC   000800           07 CURRENT-EFF-DATE     PICTURE 9(6).
   N  03 NEXT-EFF-DATE                PICTURE 9(8).
 BC   000900      03 NEXT-EFF-DATE                PICTURE 9(6).
 BCN  03 ACTIONS.
 BCN  05 PERSONNEL-ACTIONS        PICTURE 9 OCCURS 16.
 BCN  03 EDITS.
 BCN  05 CONSISTENCY-EDITS        PICTURE 9 OCCURS 60.
 BCN  03 SEGMENT-STATUS-ARRAY.
 B N
   N  05 SEGMENT-STATUS-TABLE    PICTURE 9 OCCURS 49.
 B    05 SEGMENT-STATUS-TABLE    PICTURE 9 OCCURS 53.
 B N
  C   001500* JULY 1994, C. COHEN, GTN MOD., CHANGE 8687
  C   001600*          05 SEGMENT-STATUS-TABLE   PICTURE 9 OCCURS 53.  VANILLA
  C   *          05 SEGMENT-STATUS-TABLE   PICTURE 9 OCCURS 68.  SCS MOD.
  C   05 SEGMENT-STATUS-TABLE   PICTURE 9 OCCURS 70.
  C   001700*
 BCN  03 IMAGE-TABLE.
   N  05 BEFORE-IMAGE-SEGMENT    OCCURS 49 TIMES.
  C   001500* JULY 1994, C. COHEN, GTN MOD., CHANGE 8687
  C   001900*          05 BEFORE-IMAGE-SEGMENT    OCCURS 53 TIMES.   VANILLA
  C   *          05 BEFORE-IMAGE-SEGMENT    OCCURS 68 TIMES.   SCS MOD.
  C   05 BEFORE-IMAGE-SEGMENT    OCCURS 70 TIMES.
 B    05 BEFORE-IMAGE-SEGMENT    OCCURS 53 TIMES.
 BCN   10 BEFORE-IMAGE-ELMNT   PICTURE 9 OCCURS 100 TIMES.
 B N  05 BEFORE-IMAGE-GTN-ARRAY  OCCURS 200 TIMES.
  C   001500* JULY 1994, C. COHEN, GTN MOD., CHANGE 8687
  C   002100*          05 BEFORE-IMAGE-GTN-ARRAY  OCCURS 200 TIMES.
  C   *          05 BEFORE-IMAGE-GTN-ARRAY  OCCURS 300 TIMES.
  C   001500* SEPT 1994, UADRAL1, WO 443, GTN 300 TO 500 MODS  NEXT LINE***
  C   05 BEFORE-IMAGE-GTN-ARRAY  OCCURS 500 TIMES.
 BCN   10 BEFORE-IMAGE-GTN     PICTURE 9  OCCURS 8 TIMES.
 BCN  03 COMM-BATCH-NO                PICTURE X(3).
 BCN  03 COMM-ORIGIN                  PICTURE X(3).
 BCN  03 COMM-ERROR-CODE              PICTURE 999.
 BCN  03 DED-ARRAY-SW                 PICTURE 9.
 BCN  03 BEFORE-IMAGE-INDX            PICTURE S9(3) COMP-3.
 BCN  *
 BCN  ************************************************************************
 BCN  *    THIS CONTROL BLOCK IS THE MAJOR COMMUNICATIONS TABLE              *
 BCN  *    BETWEEN SYSTEM ROUTINES AND USER SUBPROGRAMS/SYSTEM               *
 BCN  *    SUBPROGRAMS.  IT IS USED BY PP0800XX, PP1200XX, USER08,           *
 BCN  *    AND USER12.                                                       *
 BCN  *    CURR-KEY15 IS THE KEY OF THE RECORD PRESENTLY BEING               *
 BCN  *    PROCESSED. NEXT-EFF-DATE IS THE EFFECTIVE DATE WHICH WILL         *
 BCN  *    BE PROCESSED NEXT FOR THE SAME EMPLOYEE.  (IF THE CURRENT         *
 BCN  *    EFFECTIVE DATE IS THE LAST FOR AN EMPLOYEE, THEN                  *
 BCN  *    NEXT-EFF-DATE WILL EQUAL ALL "9".)  PERSONNEL-ACTIONS IS AN       *
```

1/8/98

APPENDIX E

```
000594        *
000595        *    HERE CHANGE NNCC TO GBC AND NCJFC TO NCJFCJ IN NEW RECORD
000596        *
000597             IF XPYH-ORG-ID OF XNEW-PAY-AUDIT-REC = 'NNCC'
000598                MOVE 'GBC' TO XPYH-ORG-ID OF XNEW-PAY-AUDIT-REC.
000599             IF XPYH-ORG-ID OF XNEW-PAY-AUDIT-REC = 'NCJFC'
000600                MOVE 'NCJFCJ' TO XPYH-ORG-ID OF XNEW-PAY-AUDIT-REC.
000601             IF XPYH-PRIOR-XORG-ID OF XNEW-PAY-AUDIT-REC = 'NNCC'
000602                MOVE 'GBC' TO XPYH-PRIOR-XORG-ID OF XNEW-PAY-AUDIT-REC.
000603             IF XPYH-PRIOR-XORG-ID OF XNEW-PAY-AUDIT-REC = 'NCJFC'
000604                MOVE 'NCJFCJ' TO XPYH-PRIOR-XORG-ID OF XNEW-PAY-AUDIT-REC.
000605             IF XPYH-PAY-ORG OF XNEW-PAY-AUDIT-REC = 'NNCC'
000606                MOVE 'GBC' TO XPYH-PAY-ORG OF XNEW-PAY-AUDIT-REC.
000607             IF XPYH-PAY-ORG OF XNEW-PAY-AUDIT-REC = 'NCJFC'
000608                MOVE 'NCJFCJ' TO XPYH-PAY-ORG OF XNEW-PAY-AUDIT-REC.
000609        *
000610         EXIT-PROGRAM-0100.
000611        *
000612             EXIT PROGRAM.
```

APPENDIX F

HRS 9.5 Release
Implementation Plan

Week of October 19th – October 27th

1. Create missing copybooks from vendor library
2. Promoting vendor batch programs
   2.1. Compile vendor batch programs
   2.2. Promoting vendor batch programs to QA
   2.3. Promoting vendor batch programs to production
3. Promoting vendor online programs
   3.1. Compile vendor online programs
   3.2. Promoting vendor online programs to QA
   3.3. Promoting vendor online programs to production
4. Promoting UNS batch programs
   4.1. Compile UNS batch programs
   4.2. Promoting UNS batch programs to QA
   4.3. Promoting UNS batch programs to production
5. Promoting UNS online programs
   5.1. Compile UNS online programs
   5.2. Promoting UNS online programs to QA
   5.3. Promoting UNS online programs to production
6. Promote UNS JCL to production
7. Copying southern utility PDS
   7.1. Copying southern utility (CNTL) to production
   7.2. Change all qualifiers from 'ZADHRMQ' to 'ZADHRMP' (CNTL) in production
   7.3. Copying southern utility (PARMLIB) to production
   7.4. Change all qualifiers from 'ZADHRMQ' to 'ZADHRMP' (PARMLIB) in production
   7.5. Copying southern batch PARMLIB to production
   7.6. Change all qualifiers from 'ZADHRMQ' to 'ZADHRMP' (batch PARMLIB) in production
8. Copying northern utility PDS
   8.1. Copying northern utility (CNTL) to production
   8.2. Change all qualifiers from 'YADHRMQ' to 'YADHRMP' (CNTL) in production
   8.3. Copying northern utility (PARMLIB) to production
   8.4. Change all qualifiers from 'YADHRMQ' to 'YADHRMP' (PARMLIB) in production
   8.5. Copying northern batch PARMLIB to production
   8.6. Change all qualifiers from 'YADHRMQ' to 'YADHRMP' (batch PARMLIB) in production
9. Promoting OLAG
   9.1. Promote vendor OLAG to production
   9.2. Promote UNS southern OLAG to production
   9.3. Promote UNS northern OLAG to production

10.3. Promote UNS northern OLAG to production

11. Review conversion programs (converting from production to QA)
    - 11.1. Control file conversion
        - 11.1.1. CTLGBC (change system date in JCL)
        - 11.1.2. CVTCDBS (south only)
        - 11.1.3. CVTCDBN (north only)
    - 11.2. CVTENC
    - 11.3. CVTEXPH
    - 11.4. CVTOBJ
    - 11.5. CVTSUS
    - 11.6. EDB file conversion
        - 11.6.1. EDBGBC
        - 11.6.2. CVTEDB
        - 11.6.3. EDBRSTR
        - 11.6.4. NOLIM
    - 11.7. Payroll History conversion
        - 11.7.1. PYHGBC
        - 11.7.2. CVTPYHLD
    - 11.8. Employment History conversion
        - 11.8.1. HDBBKUP
        - 11.8.2. HDBGBC
        - 11.8.3. HDBRSTR
    - 11.9. Convert applicant tracking (?)
    - 11.10. PARGBC
    - 11.11. PAYGBC
    - 11.12. PCFGBC
    - 11.13. THFGBC
    - 11.14. SDBGBC
    - 11.15. Creating job to create all temporary files with correct record lengths

October 28th - Morning

12. Convert payroll history
13. Convert merge par
14. Convert PAYSUB file
15. Convert object file
16. Convert expense history

October 28th - Evening

17. Run Daily programs in 9.3
18. Run payroll edit in 9.3
19. Run monthly reports in 9.3
20. Convert employee database
21. Convert suspense file
22. Convert control file
23. Convert security file
24. Repro account line history to QA
25. Convert employment history

October 29th

26. Convert PCF file
27. Convert THF file
28. Convert encumbrance file
29. Convert applicant tracking
30. Run employee balance program in production
31. Run employee balance program in QA
32. Run change attributes for VSAM files in QA
33. Bring QA CICS up for users for inquiry
34. Change CICS regions to point to new LOADLIBS and file definitions
35. Change VSAM attributes – (remove IMBED and ERASE)
36. Back up files using SNAP (9.3 – production)
37. Run job to create all temporary files
38. Copy CICS screens from QA to production
39. Copy PYH file from QA to production
40. Copy CXE file from QA to production
41. Copy merge par file from QA to production
42. Copy PAYSUB file from QA to production
43. Copy PCF file from QA to production
44. Copy THF files from QA to production
45. Copy security file from QA to production
46. Copy encumbrance file from QA to production
47. Copy object file from QA to production
48. Copy expense history from QA to production
49. Copy employee database from QA to production
50. Copy NXE file from QA to production
51. Copy suspense file from QA to production
52. Copy control file from QA to production
53. Copy BASEFIL from QA to production
54. Copy employment history from QA to production
55. Copy applicant tracking from QA to production
56. Run change attributes for VSAM files in production

57. Build chart of accounts using new programs
58. Run CFMPP13D
59. Copy job submittal from QA to production
60. Build JCL file for job submittal
61. Build JOB file for job submittal
62. Promote FOCUS MFDs from QA to production
63. Promote FOCUS EXECs from QA to production
64. Change share EXECs prefix from 'J' to 'H' in production
65. Copy help files from QA to production
66. Verify CICS regions
67. Change VSAM definitions based on optimize program
68. Celebrate!!

APPENDIX G

CUFS Conversion Schedule

Schedule of Events for December 31, 1998 and January 1-4, 1999

December 31

6:30 AM	Bring CUFS production up to update the DATE table with current date. Once the files have opened and the region is ready for on-line activity, close CUFS production files leaving the region, A06P, active.
7:00 AM	CUFS analysts will begin submission of batch jobs to load and process payroll charges, encumbrances and reversals. This process is expected to take 30 minutes to 2 hours. On successful completion of these jobs and file backups, application files will be opened for on-line activity.
7:30-9:00 AM	Bring the production applications on-line.
5:00 PM	CUFS production files will be closed. Nightly processing will begin. Processing will include all normal nightly jobs as well as month-end processing for fixed assets, indirect cost and closing of period 0698. Month-end reports for accounting period 0698 should be scheduled by each institution to run this night.
January 1	
4:00 AM	Bring the CUFS production region, A06P, up. Open application files for on-line processing and clean-up.
10:00 AM	Close CUFS production region Run backups and nightly cycle to clear processed documents. Run A123/B123. A223/B223, and A614 reports.
January 1 thru 3	With completion of CUFS production batch jobs, begin submission of conversion utilities for CUFS to ADVANTAGE conversion. As conversion and load utilities are completed for each institution, run A123/B123, A223/B223 and A614 reports for comparison to CUFS reports.
January 4	IF all is well, bring ADVANTAGE production on-line. System administrators must verify entries in the SOPT/SOP2, ESOP/ESO2, TSPC AND SPEC tables for Fiscal 99 before any on-line activity takes place.