

# Evolution Of The Millennium Bug In Accounting Based Systems

Ronald C. Clute, (E-mail: cluter@mscd.edu), Metropolitan State College of Denver  
George B. Garman, (E-mail: garmang@mscd.edu), Metropolitan State College of Denver

## Abstract

*The year 2000 problem has serious consequences for accountants today and will continue to impact accounting based systems for several years to come. Year 2000 problems will potentially disrupt calculations of all kinds. The problem is not limited to just software, but extends to computer hardware and, in fact, any device that uses a microprocessor with date fields such as computerized cash registers and network routers. Accountants need to protect themselves and their clients from year 2000 related systems crashes. This article provides an overview of what the year 2000 problem is and provides some recommendations for dealing with it. In general, all businesses (clients, CPA firms, and software vendors) should get a written statement confirming year 2000 compliance from the maker of any device they use, sell, or support that uses a microprocessor with date fields.*

## Introduction

In the early years of computer development no one looked far enough into the future to worry about the change from the 20th to the 21st century. Beginning in the 1960's when computer resources were in short supply, programmers and system architects saved space by truncating the first two digits of the century field, which they presumed to be 19 for the foreseeable future. For example, the year 1968 was recorded as 68. The two digit savings may not seem significant today, but it was then. Many software companies continued this practice well into the 1990's. The year 2000 problem is that computer software using only two digits will record the year 2000 as the year 00, which may cause problems with date dependent calculations. To make matters worse, this practice was imbedded in hardware via BIOS

---

*Readers with comments or questions are encouraged to contact the authors via e-mail.*

software and computer chips that will not recognize the new century.

Can computer technology fail in such a major way? The answer is absolutely yes. Lou Marcoccio testified before congress that the Gartner Group predicts that 30% to 50% of all companies and government agencies worldwide will experience at least one mission critical system failure (Marcoccio, page 4). Just this year upwards of 90% of all pagers in the US were silenced when a communications satellite failed (Scott, page 17). Computers do what they are programmed to do. If the programming is not well thought out, the results may be in error.

## Computer Simulation of year 2000 problems

As a part of testing for year 2000 compliance, many companies are simulating turn of the century transactions and calculations. Ac-

ording to one study (Scott, page 16), the results indicate that about a third of the computers work just fine, a third shut down entirely, and a third work but yield incorrect answers. The problem isn't so much with computers that shut down entirely because those should be caught before the year 2000 arrives. The main problem is with computers that work and give answers that are believable, but inaccurate. These are the problems that will be difficult to detect. A pension payoff that is off by a few hundred dollars or a mortgage payment that is off by a few dollars will be difficult to catch. A computer generated letter indicating a pension payment of zero because a retired employee has not been born will result in a quick response from the employee.

### **Year 2000 Problems Are Here Already**

We do not have to wait for the year 2000 to see year 2000 problems nor will they end with the year 2000. Nearly all depreciation schedules in existence today (1998) end in 2000 or later. They have already encountered the year 2000 issue. Many inventory pull dates and most hard copy file destroy dates also extend into the year 2000. Year 2000 problems will not end with the new millennium. The problems and lawsuits resulting from non-year 2000 compliant hardware and software will continue for many years.

### **Software Considerations**

Common year 2000 problems found in software are arithmetic operations, character arrays, screen output, and the use of flags. Many programs use the final two digits of years to determine time differences in arithmetic operations. Hence, 1991 is three years before 1994 ( $94 - 91 = 3$ ). These programs cannot determine the time difference between 1991 and 2000 ( $00 - 91$ ). Some programs use character arrays internally to store the last two digits of a year. Such formats don't allow for the expansion of dates to four digits to properly handle the year 2000 and beyond. Some programs have hard coded the 19

to appear on the screen when it receives a screen print command. Internally, only the last two digits are maintained although all four digits appear on the screen. Many programs use dates as a flag to indicate a special condition. For example, some programs use 01/01/01 as a date initializer. Other programs use 12/31/99 to indicate the final or last possible date. Many DOS clocks maintain the current date by counting the number of days since January 1, 1980.

### *Accounting Software*

Some accounting software companies provide a written guarantee that their software is year 2000 compliant for a specific computing environment. The trick here is in the fine print. For example, can this year 2000 compliant software correctly handle and interpret erroneous data received from non-year 2000 compliant vendor, customer, or cash register? The answer in most cases is no. This is a problem for most accounting software because of the extensive use of electronic commerce in businesses. Many businesses send and receive orders to and from around the world on a daily basis. Who can feel certain that are all in compliance? To be year 2000 compliant, your software and the software used by all you correspond with electronically must be year 2000 compliant.

Even the AICPA's own accounting software, ATB, is not scheduled to be year 2000 compliant until early 1999 (Scott, page 17). The authors have heard, but not confirmed, that the AICPA may not update their software at all because of the cost. In many cases, it is less costly to simply develop a new accounting software package than to re-code an old, out of date package. In the opinion of the authors, most DOS based accounting software packages are probably not worth fixing. Even if a new year 2000 compliant version of an accounting software package is developed, who is going to pay for the new version? How long can a business wait to get the new version? The ideal time to convert or upgrade to a new accounting software package is

at the end of the year. Less acceptable but still workable is a quarter-end conversion. Month-end (unless it falls on a weekend) and weekend conversions should not be attempted. The question is how many year-ends or quarter-ends are left? Another issue is how long was the new version in beta testing? The authors firmly believe that the first version of any software will have at least some bugs in it. There will surely be many problems with these last minute releases of accounting software.

One of the problems accounting software companies have is that many of the packages available today were not developed from the ground up using a single language. Many accounting packages were developed and evolved over the years using several languages that have been pieced and patched together. No one knows for sure what lurks in those old lines of code. Did the programmer realize that the year 2000 was an exception to the exception for the general rule for leap years? The exception to the exception has only occurred four other times in recorded history. The general rule and the exceptions are discussed below.

#### *Computer Network Software*

Computer networks are also a consideration. One of the problems with computer networks is that they are everywhere. Even small businesses with four or five employees often have a computer network. Computer networks are difficult to test for year 2000 compliance. A good recommendation here is to try to get the computer network software company to provide a written guarantee that their software is year 2000 compliant in your computing environment. Computer network hardware considerations are discussed elsewhere in this article.

#### *Other Software*

Other software that may have a year 2000 problem includes any software that uses date fields. Accounting is the obvious place to

examine first because it is mission critical software and makes extensive use of dates. Other less obvious software includes computerized cash registers, bar-coding software, web pages, database software, and spreadsheet software. Even further afield, but no less important to a business, are telephone systems, heat and air conditioning systems, security systems, automated factory equipment, elevators, and automobiles.

#### **Hardware Considerations**

Nearly all of the popular news media attention has been given to software applications and how they deal with year 2000 issues. However, hardware problems with the year 2000 are just as important. No one knows exactly how many non-year compliant computer chips are in use. In 1995, 3.5 billion computer chips were sold, and the number increased to 7 billion in 1997. A computer consulting firm, Greenwich Mean Time, estimates that 93 percent of all PC's shipped before 1997 have BIOS software installed that either does not roll over to the year 2000 or does not realize that the year 2000 is an exception to the exception to the leap year rule (Baklarz, page 45). The failure rate declines to 47 percent for all computers sold in 1997. No data are available for 1998.

The general rule is that most 486 based computers and many Pentium based computers running slower than 166 MHz are not year 2000 compliant because of the BIOS software they use (Scott, page 18). These older computers will work just fine for non-date sensitive applications. However, most accounting software is date sensitive. Businesses should contact the maker of their computers and related hardware and request a written guarantee concerning year 2000 compliance.

In a PC three components come into play with regards to the date. The first is the complementary metal oxide semiconductor (CMOS) real time clock, which maintains a clock and 100 year calendar by using a battery.

It stores seconds, minutes, hours, days of the month, and years each in two digit values. The second is basic input output system (BIOS) software, which maintains among other things the century part of the year (e.g., 19, 20). The third is in the disk operating system (DOS and windows). When DOS (or windows) boots up it initializes its current date by reading the CMOS date and converting it to days since January 1, 1980. This is the date an application or program receives when a date is requested.

The year 2000 problem is not restricted to microcomputers, minicomputers, and mainframes. Networks using a component called routers connect most organizations. These routers contain chips that must also be year 2000 compliant.

#### **Other Considerations**

As if year 2000 problems were not enough to worry about, the year 2000 is also a leap year. In other words, February 2000 will have 29 days. Century years are never leap years unless they are divisible by 400, which is true of the year 2000. This is an exception to the general rule that leap years occur every four years except in years divisible by 100. Software, hardware, networks, and other related hardware and software should be tested to see if 29 days are allowed in February 2000.

#### **Legal Considerations**

Where there are problems of this magnitude, lawyers and lawsuits will surely follow. Already there have been lawsuits against Macola, SBT, and Intuit accounting software companies. This is surely the tip of the iceberg, as most year 2000 problems will not surface until the year 2000. Fortunately, year 2000 problems should be covered under accountant's errors and omissions insurance. No doubt accountants will be sued over year 2000 problems because of their perceived deep pockets and insurance to cover settlement losses.

The best defense for accountants is to prevent the lawsuit in the first place with good planning. Accountants should have clients list all software and hardware that may have a year 2000 problem. For each item a statement should be obtained from the software or manufacturing company indicating year 2000 compliance. Non-year 2000 compliant software and hardware should be replaced. Accountants should not wait to test for year 2000 compliance or to install new software, as there will certainly be a shortage of qualified personnel to fix problems in 1999. Auditor's engagement letters should incorporate year 2000 concerns. Clients should be encouraged to back up all mission critical applications such as accounting software immediately before January 1, 2000.

#### **Conclusions**

The year 2000 problem is a serious problem for accountants today and will continue to be a problem for several years to come. Year 2000 problems will potentially disrupt calculations of all kinds. The problem is not limited to just software, but extends to computer hardware and, in fact, any device that uses a microprocessor with date fields such as computerized cash registers and routers. Accountants need to protect themselves and their clients from year 2000 related problems. This article provided an overview of what the year 2000 problem is and gives some recommendations for dealing with it. In general, all businesses (clients, CPA firms, and software vendors) should get a written statement confirming year 2000 compliance from the maker of any device they use, sell, or support that uses a microprocessor with date fields. The final recommendation is don't fly anywhere on 01/01/00. Pun intended! 📖

#### **References**

1. Baklarz, Ronald R., "The Enemy Within - Y2K Issues Below the Application Level" *IS Audit & Control Journal*, Volume III, 1998, pp. 45-48

2. Boef, Anthonie, "The Year 2000 Issue in Perspective" *IS Audit & Control Journal*, Volume II, 1997, pp. 12-14.
3. Chaabouni, M., "A Framework For Testing Year 2000 Application Conversions" <http://www.year2000.com/archive/chaabouni.html>, 7 pages.
4. Coopers & Lybrand, LLP, "Assessing the Impact of Year 2000" <http://www.us.coopers.com/cas/yr2000.html>, 2 pages.
5. Courtney, Harley M. and Daniel C. Benco, "Can Your Software Make It Into Year 2000?" *Journal of Accountancy*, December 1997, Vol. 184, No. 6, pp. 35-40.
6. Gerner, Michael, "Why Has The Year 2000 Problem Happened?" <http://www.io.com/~deweerd/year2000/reasons.html>, 6 pages.
7. Harding, Wayne E., "Spreadsheets Face The Millennium" *Journal of Accountancy*, December 1997, Vol. 184, No. 6, pp. 41-42.
8. Harding, Wayne E., "Year 2000: Opportunity and Exposure for the Accountant" [http://www.gps.com/Waynes\\_Web/Perspectives/Year2000.htm](http://www.gps.com/Waynes_Web/Perspectives/Year2000.htm), 2 pages.
9. INPUT, "Vendor Liability and the Y2000 Crisis" <http://www.input.com>, 10 pages.
10. Jones, Capers, "The Global Economic Impact of the Year 2000 Software Problem" [http://www.spr.com/html/year\\_2000problem.htm](http://www.spr.com/html/year_2000problem.htm), 69 pages.
11. Kanter, Howard A., "The Year 2000 Problem: Challenges for IS Audit" *IS Audit & Control Journal*, Volume II, 1997, pp. 48-55.
12. Marcoccio, Lou, "Year 2000 Global State of Readiness and Risks to the General Business Community", Expert testimony provided to the U. S. Senate Special Committee on the year 2000 technology problem, October 7, 1998, Washington D. C., [HTTP://gartner12.gartnerweb.com/public/static/aboutgg/pressrel/testimony1098.html](http://gartner12.gartnerweb.com/public/static/aboutgg/pressrel/testimony1098.html), 16 pages.
13. McKell, Lynn J. and Marshall Romney, "Risks and Liabilities" *Journal of Accountancy*, December 1997, Vol. 184, No. 6, pp. 43-46.
14. Parker, Robert G., *An Executives Guide to the Year 2000*, Information Systems Audit And Control Association, 1998, 64 pages.
15. Reid, Warren S., "2001: A Legal Odyssey" <http://www.year2000.com/archive/legal.html>, 7 pages.
16. Scott, Robert W., "The Year 2000 Bomb" *Accounting Technology*, July 1998, pp. 16-24.
17. Smith, David M., "The Year 2000: More Than We Planned" *IS Audit & Control Journal*, Volume IV, 1997, pp. 53-55.
18. Zarowin, Stanley, "The Millennium Muddle" *Journal of Accountancy*, December 1997, Vol. 184, No. 6, pp. 33-35.

