

Y2k millennium bug essay



**ASSIGN
BUSTER**

The Millennial sun will first rise over human civilization in the independent republic of Kiribati, a group of some thirty low lying coral islands in the Pacific Ocean that straddle the equator and the International Date Line, halfway between Hawaii and Australia. This long awaited sunrise marks the dawn of the year 2000, and quite possibly, the onset of unheralded disruptions in life as we know it in many parts of the globe. Kiribatis 81, 000 Micronesians may observe nothing different about this dawn; they only received TV in 1989. However, for those who live in a world that relies on satellites, air, rail and ground transportation, manufacturing plants, electricity, heat, telephones, or TV, when the calendar clicks from 99 to 00, we will experience a true millennial shift. As the sun moves westward on January 1, 2000, as the date shifts silently within millions of computerized systems, we will begin to experience our computer-dependent world in an entirely new way. We will finally see the extent of the networked and interdependent processes we have created. At the stroke of midnight, the new millennium heralds the greatest challenge to modern society that we have yet to face as a planetary community.

I am describing the year 2000 problem, known as Y2K (K signifying 1000.) Nicknamed at first " The Millennial Bug," increasing sensitivity to the magnitude of the impending crisis has escalated it to " The Millennial Bomb." The problem begins as a simple technical error. Large mainframe computers more than ten years old were not programmed to handle a four digit year. Sitting here now, on the threshold of the year 2000, it seems incomprehensible that computer programmers and microchip designers didn't plan for it. But when these billions of lines of computer code were

being written, computer memory was very expensive. Remember when a computer only had 16 kilobytes of RAM? To save storage space, most programmers allocated only two digits to a year. 1993 is 93 in data files, 1917 is 17. These two-digit dates exist on millions of files used as input to millions of applications. Programmers did whatever was required to get a product up and working; no one even thought about standards.

The same thing happened in the production of microchips as recently as three years ago. Microprocessors and other integrated circuits are often just sophisticated calculators that count and do math. They count many things: fractions of seconds, days, inches, pounds, degrees, lumens, etc. Many chips that had a time function designed into them were only structured for this century. And when the date goes from '99 to '00 both they and the legacy software that has not been fixed will think it is still the 20th century — not 2000, but 1900. Y2K Date calculations affect far more millions of systems than those that deal with inventories, interest rates, or insurance policies. Every major aspect of our modern infrastructure has systems and equipment that rely on such calculations to perform their functions. We are dependent on computerized systems that contain date functions to effectively manage defense, transportation, power generation, manufacturing, telecommunications, finance, government, education, healthcare, and more. The list is longer, but the picture is pretty clear. We have created a world whose efficient functioning in all but the poorest and remotest areas is dependent on computers. It doesn't matter whether you personally use a computer, or that most people around the world don't even have telephones. The world's economic and political infrastructures rely on computers. And not

isolated computers. We have created dense networks of reliance around the globe. We are networked together for economic and political purposes. Whatever happens in one part of the network has an impact on other parts of the network. We have created not only a computer-dependent society, but, also an interdependent planet.

We already have had frequent experiences with how fragile these systems are, how failure's cascade through a networked system. While each of these systems relies on millions of lines of code that detail the required processing, they handle their routines in serial fashion. Any next step depends on the preceding step. This serial nature makes systems, no matter their size, vulnerable to even the slightest problem anywhere in the system. In 1990, ATTs long distance system experienced repeated failures. At that time, it took two million lines of computer code to keep the system operational. But just three lines of faulty code brought down these millions of lines of code. (6)And these systems are lean; redundancies are eliminated in the name of efficiency. This leanness also makes the system highly vulnerable. In May of this year, 90% of all pagers in the U. S. crashed for a day or longer because of the failure of one satellite. Late in 1997, the Internet could not deliver email to the appropriate addresses because bad information from there one and only central source corrupted their servers. (6)I would now like to describe in greater detail, the extent of Y2K. As a global network of interrelated consequences, it begins at the center with the technical problem, legacy computer codes and embedded microchips.

For the last thirty years thousands of programmers have been writing billions of lines of software code for the computers on which the world's economy

and society now depend. Y2K reporter Ed Meagher describes “ old, undocumented code written in over 2500 different computer languages and executed on thousands of different hardware platforms being controlled by hundreds of different operating systems . . . that generate further complexity in the form of billions of six character date fields stored in millions of databases that are used in calculations.(1) The Gartner Group, a computer-industry research group, estimates that globally, 180 billion lines of software code will have to be screened.(3) Peter de Jager notes that it is not unusual for a company to have more than 100, 000, 000 lines of code—the IRS, for instance, has at least eighty million lines. The Social Security Administration began working on its thirty million lines of code in 1991. After five years of work, in June, 1996, four hundred programmers had fixed only six million lines. The IRS has 88, 000 programs on 80 mainframe computers to debug. By the end of last year they had cleaned up 2, 000 programs. (6) Capers Jones, head of Software Productivity Research, a firm that tracks programmer productivity, estimates that finding, fixing and testing all Y2K-affected software would require over 700, 000 person-years.(5) Also at the center of this technical time bomb are the embedded microprocessors. There are somewhat over a billion of these hardware chips located in systems worldwide. They sustain the world’s manufacturing and engineering base. They exist in traffic lights, elevators, water, gas, and electricity control systems. They’re in medical equipment and military and navigation systems. America’s air traffic control system is dependent upon them. They’re located in the track beds of railroad systems and in the satellites that circle the earth. Global telecommunications are heavily dependent on them. Modern

cars contain about two dozen microprocessors. The average American comes in contact with seventy microprocessors before noon every day. Many of these chips aren't date sensitive, but a great number are, and engineers looking at long ago installed systems don't know for sure which is which. To complicate things further, not all chips behave the same. "Recent tests have shown that two chips of the same model installed in two different computers but performing the same function are not equally sensitive to the year-end problem. One shuts down and the other doesn't." (6) That is why some companies are junking their computer systems and spending millions, even hundreds of millions, to replace everything. It at least ensures that their internal systems work.

The global economy is dependent upon computers both directly and indirectly. Whether its your PC at home, the workstation on a local area network, or the GPS or mobile telephone that you carry, all are integral parts of larger networks where computers are directly connected together. Failure in a single component can crash the whole system; that system could be an automobile, a train, an aircraft, an electric power plant, a bank, a government agency, a stock exchange, an international telephone system, the air traffic control system. If every possible date-sensitive hardware and software bug hasn't been fixed in a larger system, just one programming glitch or one isolated chip potentially can bring down the whole thing. Modern business is completely reliant on networks. Companies have vendors, suppliers, customers, outsourcers (all, of course, managed by computerized data bases.) For Y2K, these highly networked ways of doing business creates a terrifying scenario. The networks mean that no one

system can protect itself from Y2K failures by just attending to its own internal systems. General Motors, which has been working with extraordinary focus and diligence to bring their manufacturing plants up to Year 2000 compliance, (based on their assessment that they were facing catastrophe,) has 100, 000 suppliers worldwide. Bringing their internal systems into compliance seems nearly impossible, but what then do they do, with all those vendors who supply parts? GM experiences production stoppages whenever one key supplier goes on strike. What is the potential number of delays and shutdowns possible among 100, 000 suppliers? (7)“ The oil and gas industry is highly automated and the task to remediate all critical systems is enormous,” said committee Chairman Robert Bennett, R-Utah. “ It appears they started too late.” (8)Congressional auditors agree that the FAA has made great progress recently, but they still question whether the agency will really be ready. That many systems in that many locations, extremely difficult to do. If FAA can pull it off, great. We hope they can. Im not sure they can with the thoroughness of testing well be looking for, said Joel Willemsen of the General Accounting Office. (9)The global air traffic control system will roll over to 2000 all at once, at midnight, Greenwich Mean Time, on Dec. 31, or at six oclock in the evening in New York and three in the afternoon in San Francisco. Thats the time to watch the skies, and the airports, for disaster and long lines. (10) Our assessments suggest that the global community is likely to experience varying degrees of Y2K-related failures in every sector, in every region and at every economic level. The global picture that is slowly emerging is cause for concern, said Jacquelyn Williams-Bridgers, the State Departments Inspector General. (11)Excel part of the Microsoft Office Suite’s

(95, 97 and 2000) found in over 90% of the worlds computers. Is found to be not able to fully be capable of dealing with the year 2000 change over.

Most of the (analysis) tools on the market today do not look for the date function everywhere it can possibly exist within an Excel workbook, Falcon said. Its notuncommon for users to create user-defined names that represent formulas and functions. ... The tools dont find the date function within the user-defined names. (12)Horizon tested the spreadsheet using Viasoft Inc. s OnMark 2000 Assess versions 3. 0 and 4. 0, Symantec Corp. s Norton 2000, 2000Tools Group Inc. s DateSpy Professional, Greenwich Mean Time-UTAs Check 2000 PC Deluxe, ClickNet Software Corp. s ClickNet, and Advanced System Technologies Ltd. s Datefind-db. Eachcompliance-checking tool failed to find the error, although they flagged several other bugs in the software, according to Falcon. (12)The only analysis tool that identifies the error, according to Falcon, is IST Development Inc. s Year 2000 Analysis Suite. (12) This is a very specific example of a formula in a spreadsheet, said Dan Rickard, technical support manager for OnMark, a division of Viasoft. Out of 100 million files, only a minute number might have this problem. To put that into perspective, Rickard said a typical Fortune 500 company has about 100 million files. (12)Think of what happens if the following areas go down and stay down for months or even years: banks, railroads, public utilities, telephone lines, military communications, and financial markets. What about Social Security, and Medicare? If Social Security and Medicare go down, it will affect millions of people. Yet both programs are at risk. (13)Experts say only 8 percent of all date-related millennium bug errors will hit on Jan. 1, 2000. Each of the following dates (Table 1) and (Table 2) marks the beginning of an

important fiscal year for government. On April 1, Canada, Japan and New York State begin their Fiscal Year 2000. Forty-six states begin Fiscal Year 2000 on July 1. The U. S. government starts its Fiscal Year 2000 on Oct. 1. For all intents and purposes, these dates are the real beginning of 2000 for government benefits and programs. And, because government is the largest consumer of virtually every product and service on earth, it is a critical date for suppliers and companies that depend on payments from government. If errors occur in government computers, interfering with the payment of Social Security, Medicare, Veterans or other benefits, a large and very influential segment of the population will immediately be in an uproar.

(10) To assess others readiness, U. S. diplomats used a standard survey to collect information on host countries Y2K programs, vulnerability to short-term economic and social turmoil, reliance on technology in key infrastructure sectors and the status of Y2K corrective efforts. Overall about half of the 161 countries assessed by U. S. officials were reported to be at medium to high risk of Y2K-related failures in their telecommunications, energy and/or transportation sectors. (11) But the relatively low level of computerization in key sectors of the developing world may reduce the risk of prolonged infrastructure failures, Williams-Bridgers said. (11) Table One shows the dates that could cause problems with software or hardware.

Table Two shows additional dates for agencies that manage banking information. The first Day/Date/Year event — Friday, January 1, 1999 — has passed without major incident in Oregon. (FYI: In a recent teleconference of US Y2K coordinators, it was reported that only two states suffered major “hiccups” on 1/1/99 and these were quickly fixed.) Any software that looks or

schedules things in yearly increments should now be projecting for the year 2000 and not 1900.

The final two dates of the first table are for the century rollover. This Day/Date/Year problem is often misunderstood. Software that plans ahead or calculates dates into the next year needs to be able to distinguish 1900 from 2000 and beyond. Further, it is complicated by the fact that at this turn of century, there is also a leap year. Only those centuries divisible by 400 experience a leap year day, February 29. There are reports of some software forgetting that March 1 follows February 29. The software has created a February 30, 31, and so on. This type of software needs to have a new date projection calculator made so that " 00" becomes 2000 and not 1900. Even if the software assumes this change, the display needs to be able to change from showing the " 19" for the century to " 20" for the century at the correct time. The other option is to use four digits for the year. With this, all the size of the files and displays must be changed to show four digits for the year.

The next trouble area has to do with what was taught as a standard software practice for decades: the use of 9999 as an " end-of-file" or " end-of-record" place holder. Any software that looks ahead to September 9, 1999 and reads the record of 9/9/99 (or, as stored in some software, 9999) as " end-of-file" will have problems. This software must be restructured.

Friday, January 1, 1999 First annual plan to look ahead past the rollover date.

Thursday, July 1, 1999 Quarterly plan that includes 9/9/99. Fiscal Year start date.

Wednesday, Sept. 1, 1999 Monthly plan that includes 9/9/99.

Thursday, Sept. 9, 1999 Ninth day of ninth month of 1999.

Friday, Oct. 1, 1999 First quarterly plan to look ahead past rollover date.

Wednesday, Dec. 1, 1999 First monthly plan to look ahead past rollover date.

Friday, Dec. 31, 1999 Rollover date.

Saturday, Jan. 1, 2000 Rollover date. Ensure 01/01/00 is a Saturday.

*Monday, Jan. 3, 2000 First business day of 2000. Thursday, Jan 6, 2000 First possible weekday mistaken for a weekend day.

Tuesday, Feb. 29, 2000 Leap Year. 2100, 2200, and 2300 are not leap years.

Wednesday, Mar. 1, 2000 Leap Year rollover for the month of March.

Saturday, Apr. 1, 2000 Possible false change to Daylight Savings Time (DST).

Sunday, Apr. 2, 2000 Actual change to DST.

Monday, Apr. 3, 2000 First business day after quarter ends Mar. 31, 2000.

Friday, Apr. 14, 2000 Last business day for US 1999 tax transactions.

Saturday, Apr. 15, 2000 1999 tax filing deadline for US.

Sunday, Apr. 16, 2000 First day after 1999 tax filing deadline.

Monday, Apr. 17, 2000 First business day after 1999 tax filing deadline.

Saturday, Oct. 28, 2000 Possible false change back to standard time.

Sunday, Oct. 29, 2000 Actual change back to standard time.

Sunday, Dec. 31, 2000 Century rollover. Last day of 20th century.

Monday, Jan. 1, 2001 Century rollover. First day of 21st century.

* Hardware and embedded chips rollover on this date. If the device's data display reads " 01-01-00" and knows the day of the week is Saturday, you do not have a problem.

Friday, Jan. 7, 2000 First weekly payday.

Friday, Jan. 14, 2000 First semi-monthly payday.

Monday, Jan. 31, 2000 First Monthly payday. First Monthly close.

Friday, Mar. 31, 2000 First Quarter close.

Friday, Dec. 29, 2000 First Yearly close.

Sunday, June 30, 2002 Last day European national currencies are acceptable.

Monday, July 1, 2002 First day of Euro-only transactions in the EMU.

Table Two above shows dates that are critical to accounting or banking software. If a computer assumes " 00" is 1900 instead of the year 2000, the day of the week will be off for such things as weekly, semi-monthly and monthly paydays, etc. The last two days of this table are reminder dates for organizations such as banks that may be trading in various European currencies. It could be very costly if traders miscalculate the last day of that

type of trading. (14)Bibliography: 1). The Complexity Factor" by Ed Meagher, (09. 03. 99), 05: 19 p. m.

[www. year2000. com/archive/complexity. html](http://www.year2000.com/archive/complexity.html)2). Y2K could disrupt Medicare benefits, (08. 23. 99), 01: 26 a. m.

GAO report finds Medicare systems severely behindASSOCIATED PRESS[http://www. msnbc. com/news/200532. asp](http://www.msnbc.com/news/200532.asp)3)." Industry Wakes Up to the Year 2000 Menace," FORTUNE, April 27, 19984). The Washington Post, "If Computer Geeks Desert, IRS Codes Will Be ciphers," December 24, 19975). Business Week, March 2, 19986).(08. 28. 99), 09: 56 p. m.

[http://www. wbn. com/y2ktimebomb/index. htm](http://www.wbn.com/y2ktimebomb/index.htm)7). See dates (08-10-99) (06-30-98)(09. 05. 99), 11: 09 p. m.

[http://www. Yardeni. com/y2kreporter. html](http://www.Yardeni.com/y2kreporter.html)8). U. S. Senate Panel Worried On Oil Sector Y2K Readiness. (08. 17. 99), 07: 19 p. m.

(05. 24. 99), Folsom, Robert, Dow Jones Newswires.

[http://www. deja. com/getdoc. xp? AN= 4816044869](http://www.deja.com/getdoc.xp?AN=4816044869)). Fear of flying increases as Y2K nears. (08. 17. 99), 08: 40 p. m.

By Bob Hager, NBC News correspondent[http://www. msnbc. com/news/250247. asp](http://www.msnbc.com/news/250247.asp)10). Rolling thunder: Fits and starts: When bug will strike. (08. 28. 99), 08: 59 a. m.

Mitch Ratcliffe, ZDNN[http://www. msnbc. com/news/MILLBUGBIGPIC_Front. asp](http://www.msnbc.com/news/MILLBUGBIGPIC_Front.asp)11). Y2K glitch likely to disrupt trade. (08. 25. 99), 03: 34 a. m.

<https://assignbuster.com/y2k-millennium-bug-essay/>

REUTERS<http://www.msnbc.com/news/292367.asp>http://www.msnbc.com/news/TECHY2K_Front.asp(12). Excel may not calculate Y2K, (08. 24. 99), 3: 20 a. m.

Grant Du Bois, PC Week, ZDNNhttp://www.msnbc.com/news/TECHY2K_Front.asp(13). The Year the Earth Stands Still, (09. 06. 1999), 2: 19 p. m.

<http://garynorth.com/14>). Dangerous Date Categories, (08. 23. 99), 02: 23 a. m.

Statewide Year 2000 Project, (05. 10. 99)<http://y2k.das.state.or.us/danger.htm>