

Other bugs out there too!

Y2K: Don't Let It Bug You More Than It Has To

by Wolf H. Koch, PhD

THE QUESTION OF WHAT WE ARE DOING ABOUT THE Y2K issue has already become one of the most debated topics in the media and at cocktail parties. Unfortunately, much of the available information deals with media sensationalism and is of little value in business planning for the new millennium.

The "birth" of the Y2K bug

During the 1960s, in the early days of computers, memory allocation for programs was at a premium. At the time, much was accomplished with few computing resources. For example, in 1968 my research team had 20KB (not MB) of memory allocated for developing distillation process models on a mainframe computer. (There are 1,000 kilobytes in a megabyte.) The *real* heroes were those who could condense their code the most.

COBOL, which was the standard business programming language, started using a two-digit code for the year from its beginning. Programmers fully realized that this could become a problem in the future, when computers saw "00" for 2000 and would not know how to distinguish between 1900 and 2000. But the thinking was that *those problems were going to be 40 years away and since everyone was aware of the problems, there would be sufficient time to correct them.*

Over the next 30 years, much programming was accomplished by revising and building onto the old codes—something the old programmers had never envisioned. During the last few years many of them have come out of retirement and helped review an estimated 80 billion lines of computer code potentially at risk. Today, estimates of what will happen when computer clocks turn to "00" vary from a nuisance to Armageddon.

Other bugs

Here's a little publicized fact: other programming bugs with potentially adverse consequences are out there and may provide some insight to what we may expect at year end. *Wired*, a publication covering technology issues, listed three other bugs—two of which will occur around mid-year and may become a harbinger of what is to come on January 1st:

- GPS satellites measure time in weeks, for a period of 1,024 weeks and reset to 0 after the 1,023rd week. For the oldest satellites placed in orbit on January 5, 1980, the first GPS Rollover will occur on August 22, 1999. Much of the world's e-commerce uses GPS data, and the rollover has not ever been tested on live transactions. It could cause a 20-year mistake in financial transactions!

- COBOL programs use "9999" as an indicator for *End-of-File*. On September 9th of this year, some programs may interpret the date as the end-of-file command and stop executing.

- Most networked computers run in Unix. The Unix clock marks the time in seconds, beginning with January 1, 1970. Unix End-of-File markers are designated by "999,999,999." On *September 8, 2001*, Unix time will pass the end-of-file equivalent and may provide for some unexpected results.

The source of the Y2K problem

The "Y2K problem" has to do much less with the actual date rollover than with how that rollover manifests itself within our business structure and in everyday life. Today's world is totally interdependent through interconnected networks.

All financial institutions—the phone and electric power infrastructures as well as the transportation sector—depend totally on interconnected networks. Since we are not totally aware of the compounding dependencies within the existing structures, it is continued on page 2

extremely difficult to predict the ultimate effects of even insignificant isolated failures.

While interdependent networks present the first problem, another problem concerns *embedded chips*, which have far-reaching impact on our everyday life. There are very few things in our society today that are not operated through embedded chips: appliances, power tools, automobiles, airplanes and all the switches and nodes that regulate traffic along the networks. In fact, billions of embedded chips are installed with a great uncertainty as to exactly where they are today and what programming logic they contain.

Locating these chips only in machines decreed to be "mission-critical" will be a daunting, if not impossible task.

While network interdependencies and embedded chips represent the major potential problem, their interactions may cause an additional far-reaching dilemma. A system or machine coming to a halt on January 1st may not pose a significant setback if it can be reset and started.

If, however, the system keeps on operating and now starts to corrupt data, the effects may be much more significant and may be undetected for days—causing havoc somewhere else along a network. In most applications, however, problems will manifest themselves primarily as nuisances. For example,

clocks in VCRs and other appliances may not allow programming past this year (or may need to be programmed for a year in the distant past).

What to do

One important thing distinguishes the Y2K problem from all other disasters: we know exactly when it will occur. While most of us obviously have little impact on solving the problem at anything beyond our own sphere of influence, some preparation is warranted to ensure that our businesses and lives suffer the least consequences:

1. Review contracts with system integrators, network specialists and software vendors to ensure that their products and services are or will be Y2K-compliant.
2. Ensure that suppliers are compliant and can deliver product or stockpile sufficient product to cover business disruptions. Develop contingency plans with key suppliers.
3. Consider interdependencies with suppliers and service providers. Exercise care in providing guarantees against business disruptions to customers.
4. Assess vulnerability in your own manufacturing operation by analyzing work and information flow. Inspect all equipment, tools and instrumentation to inventory embedded sys-

tems, and test or check with vendors to ensure compliance.

5. Make redundant backups of all critical information on December 31 and keep hard copies of important records. Corrupting the only copy of backup data in a corrupt system would be a serious setback.

6. Finally, every media story I have read has suggested keeping extra cash on hand.

While I have tried to provide a brief overview of the issues, more detailed information is, of course, beyond the scope of this column. Many publications have devoted much space to the Y2K problem. Noteworthy is the January issue of *Business 2.0*, which presented an international overview and listed the following web site portals for additional information:

www.y2klinks.com
www.everything2000.com ☰

Wolf Koch is President and founder of *Technology Resources International, Inc. in Batavia, Illinois. He provides services in technology planning, product development and testing and litigation support. He is an expert in retail service station technology, product distribution and natural gas vehicles and associated fueling systems.*