

the *Availability Digest*

www.availabilitydigest.com
[@availabilitydig](https://twitter.com/availabilitydig)

Future Dates Spell Problems for IT

March 2017

Two major date rollover events are on the horizon for IT systems. They are known as Y2038 and Y2042. Either one of these events could cause applications that use dates beyond the rollover events to crash.¹ And there are many such applications. For instance, life insurance policies and home mortgages can extend decades into the future, well past the rollover dates.



The Y2038 problem is a direct result of the common use of 32-bit date/time fields. The Y2042 problem results from the representation of time in IBM z/OS mainframes.

Though these dates are more than two decades in the future, beware! Time passes quickly, and the longer one waits to correct an application, the more difficult it may be.

Shades of Y2K

It wasn't very long ago that we faced a similar problem. For decades, applications had used a two-digit date field. The year 1983 was simply stored as '83.' This was done because back then memory was very expensive, and any savings in memory were actively sought after.

The two-digit data field worked fine until we approached the year 2000. Were we going to store the year 2003 as '03'? Wouldn't that be treated as 1903?

Massive amounts of effort went into reprogramming applications to move from a two-digit date field to a four-digit date field. I was personally heavily involved in this effort. My company, The Sombers Group, joined a consortium of four other companies to help organizations plow through their applications so that they could become Y2K compliant. The consortium provided a specific project methodology, a complete set of software conversion tools, and experienced resources to fix Year 2000 related problems.

The consortium was called Four2000 and its tools included:

- *Application Repair Methodology for the Year 2000* from Leardata. This tool performed a business and technical impact assessment and prepared a detailed project plan.
- *Q2000* from Questicon helped determine the project's scope and identified every line of code to be fixed.
- *OPTA2000* from TANDsoft was used to identify and test application procedure calls to system time and date functions without having to modify the operating system time and date.

¹ [Y2038/Y2042 Are Business Risks – You Need to Know Today](#), *Softdate*; undated.
[Year 2038 Problem](#), *Wikipedia*.

- *VersaTest* from SoftSell simulated the production environment for integration testing.
- Finally, The Sombers Group provided ongoing application development to enhance applications while maintaining Y2K compliance.

Probably the most expensive task of a Y2K fix was testing it. General industry estimates put the test of costing in the range of 45% to 65% of the total Y2K project cost.² Testing began with the creation of a Master Test Plan. This test plan included several levels of testing – unit testing, system testing, user testing, and performance testing. The testing effort was extensive enough that it required full support from corporate management.

Y2038

The Y2038 Problem

The Y2038 problem is an issue for computing and storage systems in which time values are calculated or stored as 32-bit signed integers. Most 32-bit Unix systems store and manipulate time in this format. Typically, in these systems, time is interpreted as the number of seconds since January 1, 1970.

Using this technique, times cannot be encoded past 03:14:07 UTC on January 19, 2038. Times beyond this time will wrap around and be stored as a negative number. Systems will interpret the wraparound time as December 13, 1901, rather than January 19, 2038.

My Own Experience with Y2038

I myself was a victim of Y2038 years ago. One day as I logged onto my e-mail, I received an insidious error message that said “MAPI Spooler shut down unexpectedly.” Then Microsoft Exchange crashed. No matter what I did, I was dead in the water.

Calls to the Microsoft help desk resulted in the advice to reinstall Exchange. That didn’t work, so I bought Outlook ’97. Same result. I next called Outlook ’97 support. The very knowledgeable technician with whom I spoke said that if the system date were set to later than the Year 2038, the MAPI Spooler would crash. I then realized that my clock battery had died a couple of weeks ago, and I had forgotten to reset my computer’s system date. Sure enough, it was set to the year 2099. Resetting it instantly cured the problem.

The 32-Bit Time Field Problem

The use of 32-bit date/time fields is extensive. There is a major use of embedded systems with such fields. Examples include cell phones and internet routers. These systems rely on storing accurate times and dates. They are increasingly based on Unix-like operating systems that use 32-bit date/time fields. In fact, some Android devices crash and will not restart if their time is changed past 2038.

Programs that run in 32-bit environments but that work with future dates will have to be fixed earlier. For example, a program that works with dates twenty years in the future will have to be fixed no later than 2018.

64-bit systems are generally immune from this problem unless they have 32-bit systems embedded in them. Linux uses a 64-bit time field for 64-bit architectures only. 64-bit time fields introduce a wrap-around date that is twenty times greater than the estimated age of the universe!

² Testing the Year 2000 Fix, *Tandem Connection*; July 1997.

In 2011, an amusing example of the Y2038 problem occurred when the Congressional Budget Office's economic forecasting software was found to be incapable of running economic analyses past the year 2037. Congressman Paul Ryan reported to the media:

"I asked CBO to run the model going out and they told me that their computer simulation crashed in 2037 because CBO can't conceive of any way in which the economy can continue past the year 2037."

COBOL systems of the 1970s to 1990s vintage, of which many are still in operation, will fail on dates after 2038. The MySQL database's built-in timestamp will return 0 after 03:14:07 UTC on January 19, 2038.

There is no universal solution for the Y2038 problem. Any change to the date/time data type will result in code compatibility problems in any application.

But won't all new systems sold in the future have a 64-bit date-time field? Indeed, virtually all new servers, desktop, and laptop computers being sold today have 64-bit hardware and operating systems, as do some high-end cell phones and tablets. However, the hardware and operating systems are only part of the Y2038 issue. One must also consider the application software, peripheral hardware, device drivers, file systems, databases, communication protocols, web servers, and embedded systems. Also, 64-bit systems may be running embedded 32-bit software.³

Furthermore, 32-bit CPUs will continue to proliferate due to factors such as power usage and the higher cost and complexity of 64-bit CPUs.

Testing the Y2038 Fix

Finally, testing the Y2038 fixes made to applications probably will be as extensive as the Y2K testing we described earlier. A Master Test Plan must be developed which will specify the unit and system testing of each application module affected by Y2038. This must be coupled with a deployment plan that sets forth how the newly modified applications will be deployed into service.

As with Y2K, the development of the testing and deployment plans and their execution may well be the major part of the Y2038 project. It is imperative that the Y2038 modifications be thoroughly tested before they are deployed to ensure that all applications will behave properly.

There is one shortcut that can be considered, and that is to exempt non-mission-critical applications from exhaustive testing. Rather, the modified versions can simply be put into production; and any problems that arise can be fixed through the normal bug-fixing procedures.

Y2042

Y2042 is due to a limitation in the representation of time on IBM mainframes running z/OS. In the z/OS operating system, time is represented as a 64-bit integer showing the number of microseconds since January 1, 1900. This integer rolls over on September 17, 2042.

IBM has defined a new 128-bit time format that is available on its new hardware. However, many applications and computer languages continue to rely on the 64-bit format.

Time Testing Utilities

There are utilities available for testing your applications for time-rollover problems. TANDsoft (<http://www.tandsoft.com/files/products/OPTA2000.html>) provides OPTA2000 for clock and time-zone simulation and testing for HPE NonStop systems. Softdate (<http://www.ddvtechnologies.com>) provides time simulation for IBM z/OS, Linux, Unix, and Windows systems.

³ Y2038.com

Summary

The Y2038 and Y2042 date/time rollovers are not that far away. Most of us will still be working two decades from now when these will impact our systems and applications. Now is the time to begin to analyze the impact these rollovers will have on us and to take action to mitigate any serious consequences.