

Y2038/Y2042 Are Business Risks - What You Need to Know Today

Introduction

Date and time logic is pervasive in most software applications, yet this logic is inherently complex and prone to errors due to conditions such as leap years, time zones, and rollovers. Two major rollover events will occur in the years 2038 and 2042. Commonly referred to as Y2038 and Y2042, these events which are 22 and 26 years from now may seem far off, but time passes quickly. In fact, they will occur within most of our lifetimes. Many applications such as life insurances policies, home loans, and government retirement and social insurance programs already go well beyond 2038/2042 and are alive today with many millions of clients. As a business leader, it is important to be aware of these events because they are a serious risk to your business. They can be addressed by planning ahead, as explained later in this document.

The Y2038 and Y2042 events will significantly impact many business systems and products, and pose very real risks and dire consequences if not addressed, potentially causing many companies to go out of business.

In order to assure business continuity beyond Y2038 and Y2042, there is really no getting around the need to identify and address the issues caused by these events. Addressing some issues may be postponed for a while, but over time, the extent of the issues will only grow, becoming increasingly difficult, expensive, and time-consuming to address. Simple actions taken now can help achieve and maintain compliance. Identifying and correcting any issues early will provide peace of mind to you and your stakeholders, and a competitive advantage over companies who postpone taking action.

Background

Y2038 is due to a limitation in the representation of time as a 32-bit signed integer. On most 32-bit operating systems, time is usually represented by the number of seconds since Jan 1, 1970, which will rollover on Jan 19, 2038. Beyond that date, the behavior of affected applications is undefined. Applications may crash entirely or display incorrect dates. Most modern servers have 64-bit CPUs and operating systems, but this doesn't mean Y2038 is resolved just by using them. This is because the 32-bit (and 64-bit) applications and their associated databases running on 64-bit systems may still have Y2038/Y2042 issues. There may also be Y2038 issues in storage systems and interfaces between systems. Very many products also contain embedded 32-bit CPUs which run 32-bit operating systems and are susceptible to Y2038.

Y2042 is due to a limitation in the representation of time on IBM mainframes running z/OS. Time in z/OS is represented by a 64-bit integer with the number of microseconds since Jan 1, 1900, which will rollover on Sept 17, 2042. IBM has defined a 128-bit time format available on newer hardware, but many applications, supporting software, and computer languages continue to rely on the 64-bit time format.

You may be wondering how Y2038/Y2042 compare to Y2K. Y2K was due to the rollover event from 1999 to 2000. Addressing Y2K was typically a matter of modifying two-digit year fields like '99', to four-digit fields like '1999' so the roll over into 2000 would be correct, whereas '99' would otherwise have become '00', a Y2K bug. Y2038 and Y2042 may similarly manifest as dates and times being displayed incorrectly, but that is only a small part of Y2038/Y2042. These are much more complex issues since they are associated with how time itself is represented within computers, applications, etc.

Additional background information on Y2038/Y2042 can be found on [Wikipedia](#) and [y2038.com](#).

Impact to Industry

Y2038 will be the single biggest computer date-related issue of the 21st century. The global impact to address Y2038 and Y2042 will be staggering, amounting to multiple \$trillion USD by some estimates.

Virtually all industries will be impacted due to the pervasiveness of digital technology in business and modern society. Some industries are already susceptible to date/time issues due to future-dated calculations in many applications and systems. Other industries have products with expected lifetimes of 20+ years. Many government agencies, particularly regulatory agencies, should be considering Y2038/Y2042 because of their obligation to public safety and social stability. Below are some industries, product types, and government agencies in which the impact of Y2038/Y2042 should already be under consideration:

Insurance Providers - Medium- to long-term contracts and policies, such as Annuity, Health, Life, Occupational, and Invalidity

Banks / Credit Unions - 30-, 50-, 100-year loans and leases, ATMs

Other Financials - Pensions, superannuations, stock markets, brokerages

Electric Power / Utilities – coal, natural gas, nuclear, hydro, solar, wind

Transportation - Of people, animals, and goods/services by air, rail, road, water, cable, pipeline, space

Infrastructure - Satellites, network routers, points of sale, stop lights, elevators, thermostats, IoT devices

Medical Devices / Instruments - Pacemakers, life-support systems, X-ray systems

Government Agencies - US examples are NAIC, FDIC, SSA, DOE, PUC, FAA, DOT

Recommended Actions

Here are some relatively simple actions you can take **NOW** to address the risks from Y2038 and Y2042:

- **Designate a Compliance Officer** to serve as the company focal point for Y2038/Y2042 compliance. This person will help guide the company to compliance and be responsible for maintaining documentation on compliance status for business systems, products, and projects.
- **Implement corporate policies for compliance.** For example, one highly recommended policy is that all new projects be designed, tested, and maintained for Y2038/Y2042 compliance. The policy should require that compliance exceptions be formally documented, approved, tracked, and resolved with due diligence.
- **Assess your company's exposure to Y2038/Y2042.** The discovery process should include all digital systems used or produced by your company, from the largest business system to the smallest embedded system, and interfaces between them. It should include all databases and storage systems, third party and internally developed systems, application software, development tools, and custom network protocols.
- **Use your Quality Assurance processes now.** By future time travel with date and time simulation (DTS) testing of your applications, you can verify correctness for any future date, be it end of month, end of quarter, end of year, and beyond 2038/2042. When future date and time bugs are found now in your own applications, you can plan to address them. With bugs in third-party products, your compliance officer should contact those vendors and request that they accordingly update their products to be compliant.
- **Budget for Y2038/Y2042 now.** The above actions cannot occur without a financial commitment from your company's management team. Postponing these actions does have a cost, likely much higher than you expect. The cost is not just money... it is also time (since issues will take longer to fix later) and employee satisfaction (since your best developers generally want to work on new things, not on date issues in old systems). It is much easier and less expensive in the long run to find and fix the issues as early as possible. Budgeting now will save money, both over the long run, and for your date and time QA issues of today.

Recommended Products

The Softdate Date & Time Simulation Data Protection Suite from DDV Technologies

For more information, contact: Softdate@ddvtechnologies.com

The Softdate Suite has a choice of options to enable any organization to perform future and past time travel date and time simulation testing in a very agile manner at any stage in their enterprise app developments:

- Softdate/z supports all apps and systems on z/OS mainframes such as DB2, CICS, IMS, all languages including *exclusive* support for CICS MRO, full Parallel Sysplex, and much more, plus the *unique* options: Java & WebSphere Application Server (WAS) for z, Software AG's Natural & ADABAS for z and Beyond-2042 support *.
- Softdate/z exclusively consumes only 0.01% or less background CPU overhead in comparison to the 2% to 5% of old MVS-based competitors, saving many MSUs/MIPS. At a valid average estimate of USD \$4,500 per MIP per year, Softdate/z always immediately and substantially reduces mainframe costs.
- Softdate/x supports all variations of Linux and Unix in all IBM, Oracle, SAP, WAS, and other database and app environments.
- Softdate/w supports all supported versions of Microsoft Windows in all IBM, Oracle, SAP, WAS, and other database and app environments.
- The Softdate Broadcaster option, run from a Windows workstation, broadcasts virtual clock settings across any number of Windows, Linux and Unix systems servers that are participating in a date and time (DTS) test scenario. It allows simple and automatic coordination of your virtual clock settings across any number of Windows and Linux/Unix systems running Softdate **.
- The SETS (Softdate Enterprise Time Synchronization) option, run from a Windows workstation, enables extended cross-enterprise date and time testing. SETS synchronises virtual clock settings across any number of IBM z/OS, Windows, Linux, and Unix systems servers participating in a DTS test scenario. It provides simple and automatic synchronization of your virtual clock settings across an unlimited number of enterprise-wide interconnected systems running Softdate.
- In summary, Softdate provides everything that is needed for 2038/2042 compliance for apps executing across the Linux, Unix, Windows and z/OS world, well beyond 2038 and 2042 today.

* The Softdate/z Beyond-2042 option allows beyond 2042 future date and time testing of z/OS apps now to ensure all app logic is correct even though the IBM z/OS hardware, operating system software, databases, and programming language environments cannot.

** Softdate Broadcaster can be upgraded to SETS when Softdate/z for z/OS is acquired.

Recommended Services

Y2038.com offers a full suite of services for Y2038, Y2042, and other computer date issues, including general consulting/outsourcing, assessment, planning, mitigation, testing/certification, training, legal, and media.

For more information, contact: services@y2038.com

- **General Consulting** - Y2038.com provides general consulting and supplemental/outsourced on-site or off-site tech staff. We have broad-based expertise in many technologies and can help as little or as much as you require, from analyzing a single system to your entire enterprise organization.
- **Assessment** - Y2038.com helps assess your company's exposure to Y2038/Y2042. We provide hands-on assistance in discovering which systems have issues, and estimates on level of effort to address.
- **Planning** - Once systems and issues are discovered, you will need to formulate a plan for addressing those issues. Y2038.com can help with the planning phase for addressing Y2038/Y2042 issues.
- **Mitigation** - Y2038.com can directly help address specific Y2038/Y2042 issues. We work directly with you and your employees to review and modify code as needed to address the issues.
- **Testing and Certification** - Y2038.com provides independent testing and certification for Y2038/Y2042 compliance. We test beyond the rollover events and provide test reports along with certifications. Industry-wide Y2038 certification does not yet exist. Until then, we offer our own certification.
- **Training** - Y2038.com provides Y2038-related classes and seminars. We offer on-site or web-based presentations and/or training classes for audiences from Boards of Directors to development teams.
- **Legal** - Y2038.com provides domain expertise for computer date-related legal matters. As domain experts, we can provide facts regarding these issues. We can also help with forensic investigations on cases associated with Y2038/Y2042 issues.
- **Media** - Y2038.com provides Y2038/Y2042 information to media outlets. If you are creating public media related to Y2038 or other computer date issues, please feel free to contact us for domain expertise or to confidentially pre-screen or proof-read media.