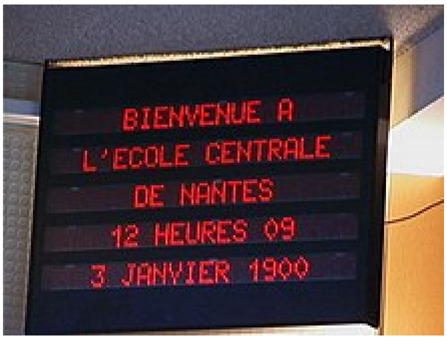
Year 2000 problem



An electronic sign at École centrale de Nantes incorrectly displaying the year 1900 on 3 January 2000

The **Year 2000 problem**, also known as the **Y2K problem**, the **Millennium bug**, **Y2K bug**, the **Y2K glitch**, or **Y2K**, refers to events related to the formatting and storage of calendar data for dates in and after the year 2000. Problems were anticipated, and arose, because many programs represented four-digit years with only the final two digits – making the year 2000 indistinguishable from 1900, and two-digit years from '01 through '31 also being mistaken for days, and '01–'12 mistaken for months in varying date formats. The assumption of a twentieth-century date in such programs could cause various errors, such as the incorrect display of dates and the inaccurate ordering of automated dated records or real-time events.

In 1997, the British Standards Institute (BSI) developed the DISC PD2000-1 technical standard defining "Year 2000 Conformity requirements" as four rules:[1]

1. No valid date will cause any interruption in operations.[1]
2. Date-based functionality must behave consistently for dates prior to, during and after year 2000.[1]
3. In all interfaces and in all storage, the century must be unambiguous, either specified, or calculable by algorithm.[1]
4. Year 2000 must be recognised as a leap year.[1]

It identifies two problems that might exist in written date formats and many computer programs. First, the practice of representing the year with two digits became problematic with logical error(s) arising upon "rollover" from xx99 to xx00. This had caused some date-related processing to operate incorrectly for dates and times on and after 1 January 2000, and on other critical dates which were billed "event horizons". Without corrective action, long-working systems would break down when the "... 97, 98, 99, 00 ..." ascending numbering assumption suddenly became invalid.

Secondly, some programmers had misunderstood the Gregorian calendar rule that states years that are exactly divisible by 100 are not leap years, assuming that the year 2000 would not be a leap year. While this is true, there is an exception that states years divisible by 400 are leap years – thus making 2000 a leap year.

A third problem is much more common: the two-digit year format increases the errors in interpreting any particular date when various order or formats of YY/MM/DD are not clearly specified: as the first 31 years of a century can be mistaken for days, and any of the first 12 years can be mistaken for months. For instance, when exactly are the following examples: 02/03/05, 21/12/22, 07-12-18, 10-11-30?

Correcting all of this, however, was not the largest part of the problem. By 1997, AT&T had estimated that "60% of the time and money needed for its total compliance efforts" would be devoted to testing the source code changes made to address the issue.[2]

Companies and organisations in some countries, but not all, checked, fixed, and upgraded their computer systems to address the anticipated problem.[3] Very few computer failures were reported when the clocks rolled over into 2000.[4]

Contents

Background

Y2K is a numeronym and was the common abbreviation for the year 2000 software problem. The abbreviation combines the letter *Y* for "year", the number 2 and a capitalized version of *k* for the SI unit prefix kilo meaning 1000; hence, *2K* signifies 2000. It was also named the "Millennium Bug" because it was associated with the popular (rather than literal) rollover of the millennium, even though most of the problems could have occurred at the end of *any* century.

Computerworld's 1993 three-page "Doomsday 2000" article by Peter de Jager was called "the information-age equivalent of the midnight ride of Paul Revere" by *The New York Times*.[5][6][7]

The Year 2000 problem was the subject of the early book *Computers in Crisis* by Jerome and Marilyn Murray (Petrocelli, 1984; reissued by McGraw-Hill under the title *The Year 2000 Computing Crisis* in 1996). The first recorded mention of the Year 2000 Problem on a Usenet newsgroup occurred on 18 January 1985 by poster Spencer Bolles.[8]

The acronym Y2K has been attributed to Massachusetts programmer David Eddy[9] in an e-mail sent on 12 June 1995. He later said, "People were calling it CDC (Century Date Change), FADL (Faulty Date Logic). There were other contenders. Y2K just came off my fingertips."[10]

The problem started because on both mainframe computers and later personal computers, storage was expensive, from as low as $10 per kilobyte, to in many cases as much as or even more than US$100 per kilobyte.[11] It was therefore very important for programmers to minimize usage. Since programs could simply prefix "19" to the year of a date, most programs internally used, or stored on disc or tape, data files where the date format was six digits, in the form DDMMYY, DD as two digits for the day, MM as two digits for the month, and YY as two digits for the year. Also used was the shorter YYDDD where DDD was the day number within the year. As space on disc and tape was also expensive, these also saved money by reducing the size of stored data files and databases.[12]

Some programs, when facing two-digit years, could not distinguish between 2000 and 1900. Dire warnings at times were in the mode of:

> The Y2K problem is the electronic equivalent of the El Niño and there will be nasty surprises around the globe. — *John Hamre, United States Deputy Secretary of Defense*[13]

Options on the De Jager Year 2000 Index, "the first index enabling investors to manage risk associated with the ... computer problem linked to the year 2000" began trading mid-March 1997.[14]

Special committees were set up by governments to monitor remedial work and contingency planning, particularly by crucial infrastructures such as telecommunications, utilities and the like, to ensure that the most critical services had fixed their own problems and were prepared for problems with others. While some commentators and experts argued that the coverage of the problem largely amounted to scaremongering,[15] it was only the safe passing of the main event itself, 1 January 2000, that fully quelled public fears.

Some experts who argued that scaremongering was occurring, such as Ross Anderson, professor of security engineering at the University of Cambridge Computer Laboratory, have since claimed that despite sending out hundreds of press releases about research results suggesting that the problem was not likely to be as big as some had suggested, they were largely ignored by the media.[15] In a similar vein, the Microsoft Press book *Running Office 2000 Professional*, published in May 1999, accurately predicted that most personal computer hardware and software would be unaffected by the year 2000 problem.[16] Authors Michael Halvorson and Michael Young characterized most of the worries as popular hysteria, an opinion echoed by Microsoft Corp.[17]

Programming problem

The practice of using two-digit dates for convenience predates computers, but was never a problem until stored dates were used in calculations.

The need for bit conservation

> "I'm one of the culprits who created this problem. I used to write those programs back in the 1960s and 1970s, and was proud of the fact that I was able to squeeze a few elements of space out of my program by not having to put a 19 before the year. Back then, it was very important. We used to spend a lot of time running through various mathematical exercises before we started to write our programs so that they could be very clearly delimited with respect to space and the use of capacity. It never entered our minds that those programs would have lasted for more than a few years. As a consequence, they are very poorly documented. If I were to go back and look at some of the programs I wrote 30 years ago, I would have one terribly difficult time working my way through step-by-step."
>
> —Alan Greenspan, 1998[18]

In the first half of the 20th century, well before the computer era, years were handwritten, in full without abbreviation or an apostrophe. In fact, cursive forms of the year may have actually looked like this: "....in the year of our Lord, Nineteen Hundred Seventeen." After 1931, it became increasingly common to abbreviate the 4-digit year with the two-digit contraction, including an apostrophe, such as '42, short for 1942. After 1931, it was impossible to mistake a two-digit year for a day of the month. This was the very beginning of the Y2K problem. With this practice, continued into the next century, two-digit year formats can easily be mistaken for a day or month, for the first 31 years of a century.

Business data processing was done using unit record equipment and punched cards, most commonly the 80-column variety employed by IBM, which dominated the industry. Many tricks were used to squeeze needed data into fixed-field 80-character records. Saving two digits for every date field was significant in this effort.

In the 1960s, computer memory and mass storage were scarce and expensive. Early core memory cost one dollar per bit. Popular commercial computers, such as the IBM 1401, shipped with as little as 2 kilobytes of memory.[19] Programs often mimicked card processing techniques. Commercial programming languages of the time, such as COBOL and RPG, processed numbers in their character representations. Over time the punched cards were converted to magnetic tape and then disc files, but the structure of the data usually changed very little. Data was still input using punched cards until the mid-1970s. Machine architectures, programming languages and application designs were evolving rapidly. Neither managers nor programmers of that time expected their programs to remain in use for many decades. The realisation that databases were a new type of program with different characteristics had not yet come.

Early attention

There were exceptions, of course. The first person known to publicly address this issue was Bob Bemer, who had noticed it in 1958 as a result of work on genealogical software. He spent the next twenty years trying to make programmers, IBM, the government of the United States and the ISO aware of the problem, with little result. This included the recommendation that the COBOL PICTURE clause should be used to specify four digit years for dates.[20]

In the 1980s the brokerage industry began to address this issue, mostly because of bonds with maturity dates beyond the year 2000. By 1987 the New York Stock Exchange had reportedly spent over $20 million, including "a team of 100" programmers on Y2K.[21]

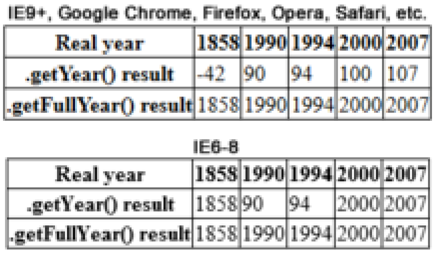
Despite magazine articles on the subject from 1970 onward, the majority of programmers and managers only started recognising Y2K as a looming problem in the mid-1990s, but even then, inertia and complacency caused it to be mostly unresolved until the last few years of the decade. In 1989, Erik Naggum was instrumental in ensuring that internet mail used four digit representations of years by including a strong recommendation to this effect in the internet host requirements document RFC 1123 (https://tools.ietf.org/html/rfc1123).[22] On April Fools' Day of 1998 some companies set their mainframe computer dates to 2001, so that "the wrong date will be perceived as good fun instead of bad computing" while having a full day of testing.[23]

While using 3 digit years and 3 digit dates within that year was used by some, others chose to use the number of days since a fixed date, such as 1 January 1900.[24] Inaction was not an option, and risked major failure. Embedded systems with similar date logic were expected to malfunction and cause utilities and other crucial infrastructure to fail.

Saving space on stored dates persisted into the Unix era, with most systems representing dates to a single 32-bit word, typically representing dates as elapsed seconds from some fixed date, which causes the similar Y2K38 problem.

Resulting bugs from date programming

Storage of a combined date and time within a fixed binary field is often considered a solution, but the possibility for software to misinterpret dates remains because such date and time representations must be relative to some known origin. Rollover of such systems is still a problem but can happen at varying dates and can fail in various ways. For example:



IE9+, Google Chrome, Firefox, Opera, Safari, etc.

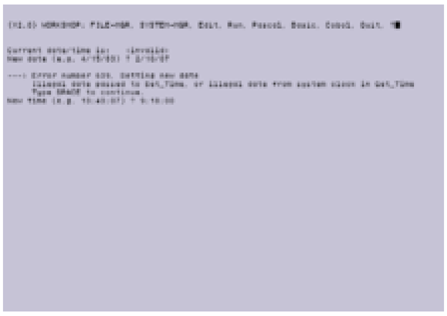
Real year	1858	1990	1994	2000	2007
.getYear() result	-42	90	94	100	107
.getFullYear() result	1858	1990	1994	2000	2007

IE6-8

Real year	1858	1990	1994	2000	2007
.getYear() result	1858	90	94	2000	2007
.getFullYear() result	1858	1990	1994	2000	2007

Webpage screenshots showing the JavaScript .getYear() method problem, which depicts the Year 2000 problem

- An upscale grocer's 1997 credit-card caused crash of their 10 cash registers, repeatedly, due to year 2000 expiration dates, was the source of the first Y2K-related lawsuit.[25]
- The Microsoft Excel spreadsheet program had a very elementary Y2K problem: Excel (in both Windows and Mac versions, when they are set to start at 1900) incorrectly set the year 1900 as a leap year for compatibility with Lotus 1-2-3.[26] In addition, the years 2100, 2200, and so on, were regarded as leap years. This bug was fixed in later versions, but since the epoch of the Excel timestamp was set to the meaningless date of 0 January 1900 in previous versions, the year 1900 is still regarded as a leap year to maintain backward compatibility.
- In the C programming language, the standard library function to extract the year from a timestamp returns the year minus 1900. Many programs using functions from C, such as Perl and Java, two programming languages widely used in web development, incorrectly treated this value as



```
{V1.0} WORKSHOP: FILE-MGR, SYSTEM-MGR, Edit, Run, Pascal, Basic, Cobol, Quit, ?

Current date/time is:   <invalid>
New date (e.g. 4/15/83) ? 2/15/07

----> Error number 635. Setting new date
      Illegal date passed to Set_Time, or illegal date from system clock in Get_Time
      Type SPACE to continue.
New time (e.g. 10:45:07) ? 9:10:00
```

An Apple Lisa does not accept the date

the last two digits of the year. On the web this was usually a harmless presentation bug, but it did cause many dynamically generated web pages to display 1 January 2000 as "1/1/19100", "1/1/100", or other variants, depending on the display format.

- JavaScript was changed due to concerns over the Y2K bug, and the return value for years changed and thus differed between versions from sometimes being a four digit representation and sometimes a two-digit representation forcing programmers to rewrite already working code to make sure web pages worked for all versions.[27][28]
- Older applications written for the commonly used UNIX Source Code Control System failed to handle years that began with the digit "2".
- In the Windows 3.x file manager, dates displayed as 1/1/19:0 for 1/1/2000 (because the colon is the character after "9" in the ASCII character set). An update was available.
- Some software, such as Math Blaster Episode I: In Search of Spot which only treats years as two-digit values instead of four, will give a given year as "1900", "1901", and so on, depending on the last two digits of the present year.

Date bugs similar to Y2K

4 January 1975

This date overflowed the 12-bit field that had been used in the Decsystem 10 operating systems. There were numerous problems and crashes related to this bug while an alternative format was developed.[29]

9 September 1999

Even before 1 January 2000 arrived, there were also some worries about 9 September 1999 (albeit less than those generated by Y2K). Because this date could also be written in the numeric format 9/9/99, it could have conflicted with the date value `9999`, frequently used to specify an unknown date. It was thus possible that database programs might act on the records containing unknown dates on that day. Data entry operators commonly entered 9999 into required fields for an unknown future date, (e.g., a termination date for cable television or telephone service), in order to process computer forms using CICS software.[30] Somewhat similar to this is the end-of-file code `9999`, used in older programming languages. While fears arose that some programs might unexpectedly terminate on that date, the bug was more likely to confuse computer operators than machines.

Leap years

Normally, a year is a leap year if it is evenly divisible by four. A year divisible by 100, however, is not a leap year in the Gregorian calendar unless it is also divisible by 400. For example, 1600 was a leap year, but 1700, 1800 and 1900 were not. Some programs may have relied on the oversimplified rule that *a year divisible by four is a leap year*. This method works fine for the year 2000 (because it is a leap year), and will not become a problem until 2100, when older legacy programs will likely have long since been replaced. Other programs contained incorrect leap year logic, assuming for instance that no year divisible by 100 could be a leap year. An assessment of this *leap year problem* including a number of real life code fragments appeared in 1998.[31] For information on why century years are treated differently, see Gregorian calendar.

Year 2010 problem

Some systems had problems once the year rolled over to 2010. This was dubbed by some in the media as the "Y2K+10" or "Y2.01K" problem.[32]

The main source of problems was confusion between hexadecimal number encoding and binary-coded decimal encodings of numbers. Both hexadecimal and BCD encode the numbers 0–9 as 0x0–0x9. But BCD encodes the number 10 as 0x10, whereas hexadecimal encodes the number 10 as 0x0A; 0x10 interpreted as a hexadecimal encoding represents the number 16.

For example, because the SMS protocol uses BCD for dates, some mobile phone software incorrectly reported dates of SMSes as 2016 instead of 2010. Windows Mobile is the first software reported to have been affected by this glitch; in some cases WM6 changes the date of any incoming SMS message sent after 1 January 2010 from the year 2010 to 2016.[33][34]

Other systems affected include EFTPOS terminals,[35] and the PlayStation 3 (except the Slim model).[36]

The most important occurrences of such a glitch were in Germany, where upwards of 20 million bank cards became unusable, and with Citibank Belgium, whose digipass customer identification chips failed.[37]

Year 2038 problem

The original Unix time datatype (`time_t`) stores a date and time as a signed long integer (on 32-bit systems a 32-bit integer) representing the number of seconds since 1 January 1970. During and after 2038, this number will exceed $2^{31} - 1$, the largest number representable by a signed long integer on 32-bit systems, causing the Year 2038 problem (also known as the Unix Millennium bug or Y2K38). As a long integer in 64-bit systems uses 64 bits, the problem does not realistically exist on 64-bit systems that use the LP64 model. The Y2K38 problem has been dealt with on Linux systems with XFS filesystem by rolling out the Linux kernel version 5.10 on 14 December 2020. However, the fix for the timestamps would work only till the year 2486.

Programming solutions

Several very different approaches were used to solve the Year 2000 problem in legacy systems. Five of them follow:

Date expansion

Two-digit years were expanded to include the century (becoming four-digit years) in programs, files, and databases. This was considered the "purest" solution, resulting in unambiguous dates that are permanent and easy to maintain. However, this method was costly, requiring massive testing and conversion efforts, and usually affecting entire systems.

Date windowing

Two-digit years were retained, and programs determined the century value only when needed for particular functions, such as date comparisons and calculations. (The century "window" refers to the 100-year period to which a date belongs.) This technique, which required installing small patches of code into programs, was simpler to test and implement than date expansion, thus much less costly. While not a permanent solution, windowing fixes were usually designed to work for many decades. This was thought acceptable, as older legacy systems tend to eventually get replaced by newer technology.[38]

Date compression

Dates can be compressed into binary 14-bit numbers. This allows retention of data structure alignment, using an integer value for years. Such a scheme is capable of representing 16384 different years; the exact scheme varies by the selection of epoch.

Date re-partitioning

In legacy databases whose size could not be economically changed, six-digit year/month/day codes were converted to three-digit years (with 1999 represented as 099 and 2001 represented as 101, etc.) and three-digit days (ordinal date in year). Only input and output instructions for the date fields had to be modified, but most other date operations and whole record operations required no change. This delays the eventual roll-over problem to the end of the year 2899.

Software kits

Software kits, such as those listed in CNN.com's *Top 10 Y2K fixes for your PC*:[39] ("most ... free") which was topped by the $50 *Millennium Bug Kit*.[40][41]

Bridge programs

Date servers[42][43] where Call statements are used to access, add or update date fields.[44]

Documented errors

Before 2000

- On 1 January 1999, taxi meters in Singapore stopped working, while in Sweden, incorrect taxi fares were given.[45]
- On 28 December 1999, 10,000 card swipe machines issued by HSBC and manufactured by Racal stopped processing credit and debit card transactions.[15] The stores relied on paper transactions until the machines started working again on 1 January.[46]

On 1 January 2000

When 1 January 2000 arrived, there were problems generally regarded as minor.[47] Consequences did not always result exactly at midnight. Some programs were not active at that moment and problems would only show up when they were invoked. Not all problems recorded were directly linked to Y2K programming in a causality; minor technological glitches occur on a regular basis.

Reported problems include:

- In Australia, bus ticket validation machines in two states failed to operate.[47]
- In Ishikawa, Japan, radiation-monitoring equipment failed at midnight; however, officials stated there was no risk to the public.[48]
- In Onagawa, Japan, an alarm sounded at a nuclear power plant at two minutes after midnight.[48]
- In Japan, at two minutes past midnight, the telecommunications carrier Osaka Media Port found errors in the date management part of the company's network. The problem was fixed by 02:43 and no services were disrupted.[49]
- In Japan, NTT Mobile Communications Network (NTT Docomo), Japan's largest cellular operator, reported that some models of mobile telephones were deleting new messages received, rather than the older messages, as the memory filled up.[49]
- In France, the national weather forecasting service, Météo-France, said a Y2K bug made the date on a webpage show a map with Saturday's weather forecast as "01/01/19100".[47]
- In Sheffield, United Kingdom, incorrect risk assessments for Down syndrome were sent to 154 pregnant women and two abortions were carried out as a direct result of a Y2K bug causing

miscalculation of the mothers' age. Four babies with Down syndrome were also born to mothers who had been told they were in the low-risk group.[50]
- In the United States, the US Naval Observatory, which runs the master clock that keeps the country's official time, gave the date on its website as 1 Jan 19100.[51]
- In the United States, as a direct result of the Y2K glitch, at midnight computers at a ground control station ceased processing information from an unspecified number of spy satellites. The military implemented a contingency plan by 03:00 AM, and restored all normal functionality in approximately two days.[52]
- In the United States, 150 Delaware Lottery racino slot machines stopped working.[47]

After January 2000

On 1 March 2000

Problems were reported on 1 March 2000, which followed Y2K's first Leap Year Day,[53] but these were mostly minor.[54]

- In Japan, around five percent of post office cash dispensers failed to work.
- In the same country, data from weather bureau computers was corrupted.
- In Bulgaria, police documents were issued with expiration dates of 29 February 2005 and 29 February 2010 (which are not leap years) and the system defaulted to 1900.[55]
- In the United States, the Coast Guard's message processing system was affected.
- At Reagan National Airport, check-in lines lengthened after baggage handling programs were affected.
- At Offutt Air Force Base south of Omaha, Nebraska, records of aircraft maintenance parts could not be accessed.

On 31 December 2000 or 1 January 2001

Some software did not correctly recognise 2000 as a leap year, and so worked on the basis of the year having 365 days. On the last day of 2000 (day 366) these systems exhibited various errors. These were generally minor.

- In Norway, some trains were delayed until their clocks were put back by a month.[56]
- A "Y2K-like bug" affected a New York City government wireless system that was down for 10 days because of a GPS system's date-rollover problem.[57][58] Infrastructure affected included "traffic lights, license-plate readers used by cops and other key functions."

Since then

In addition to the Year 2000 problem, various issues have occurred due to errors involving overflows. A specific issue with time formatting caused the destruction of the NASA *Deep Impact* spacecraft. It tracked time in one-tenth of a second increments, and so was lost on 11 August 2013 at 00:38:49 when its internal clock reached exactly 2^{32} tenths of a seconds since 1 January 2000, the time value being stored in a signed 32-bit integer which overflowed on the date and at the time in question.

Some software used a process called date windowing to fix the issue by interpreting years 00-19 as 2000-2019 and 20-99 as 1920-1999. As a result, a new wave of glitches started appearing in 2020, including parking meters in New York City refusing to accept credit cards, issues with Novitus point of sale units, and some utility companies printing bills listing the year 1920.[59]

Government responses

Bulgaria

Although only two digits are allocated for the birth year in the Bulgarian national identification number, the year 1900 problem and subsequently the Y2K problem were addressed by the use of unused values above 12 in the month range. For all persons born before 1900, the month is stored as the calendar month plus 20, and for all persons born after 1999, the month is stored as the calendar month plus 40.[60]

Canada

Canadian Prime Minister Jean Chrétien's most important cabinet ministers were ordered to remain in Ottawa, Canada's capitol, and gathered at 24 Sussex Drive to watch the clock.[5] 13,000 Canadian troops were also put on standby.[5]

Netherlands

The Dutch Government promoted Y2K Information Sharing and Analysis Centers (ISACs) to share readiness between industries, without threat of antitrust violations or liability based on information shared.

Norway and Finland

Norway and Finland changed their national identification number, to indicate the century in which a person was born. In both countries, the birth year was historically indicated by two digits only. This numbering system had already given rise to a similar problem, the "Year 1900 problem", which arose due to problems distinguishing between people born in the 20th and 19th centuries. Y2K fears drew attention to an older issue, while prompting a solution to a new problem. In Finland, the problem was solved by replacing the hyphen ("-") in the number with the letter "A" for people born in the 21st century (for people born before 1900, the sign was already "+").[61] In Norway, the range of the individual numbers following the birth date was altered from 0–499 to 500–999.

Romania

Romania also changed its national identification number in response to the Y2K problem, due to the birth year being represented by only two digits. Before 2000, the first digit, which shows the person's gender, was 1 for males and 2 for females. Starting from 1 January 2000, the Romanian national identification number starts with 5 for males and 6 for females.

Uganda

The Ugandan government responded to the Y2K threat by setting up a Y2K Task Force.[62] In August 1999 an independent international assessment by the World Bank International Y2k Cooperation Centre found that Uganda's website was in the top category as "highly informative". This put Uganda in the "top 20" out of 107 national governments, and on a par with the United States, United Kingdom, Canada, Australia and Japan, and ahead of Germany, Italy, Austria, Switzerland which were rated as only "somewhat informative". The report said that "Countries which disclose more Y2k information will be more likely to maintain public confidence in their own countries and in the international markets."[63]

United Kingdom

The British government made regular assessments of the progress made by different sectors of business towards becoming Y2K-compliant and there was wide reporting of sectors which were laggards. Companies and institutions were classified according to a traffic light scheme ranging from green "no problems" to red "grave doubts whether the work can be finished in time". Many organisations finished far ahead of the deadline.

United States

In 1998, the United States government responded to the Y2K threat by passing the Year 2000 Information and Readiness Disclosure Act, by working with private sector counterparts in order to ensure readiness, and by creating internal continuity of operations plans in the event of problems and set limits to certain potential liabilities of companies with respect to disclosures about their Year 2000 program.[64][65] The effort was coordinated out of the White House by the President's Council on Year 2000 Conversion, headed by John Koskinen.[66][67] The White House effort was conducted in co-ordination with the then-independent Federal Emergency Management Agency (FEMA), and an interim Critical Infrastructure Protection Group, then in the Department of Justice, now in Homeland Security.

The US government followed a three-part approach to the problem: (1) outreach and advocacy, (2) monitoring and assessment, and (3) contingency planning and regulation.[68]

A feature of US government outreach was Y2K websites including Y2K.GOV, many of which have become inaccessible in the years since 2000. Some of these websites have been archived by the National Archives and Records Administration or the Wayback Machine.[69][70]

The logo created by The President's Council on the Year 2000 Conversion, for use on Y2K.gov

Each federal agency had its own Y2K task force which worked with its private sector counterparts; the FCC had the FCC Year 2000 Task Force.[68][71]

Most industries had contingency plans that relied upon the internet for backup communications. However, as no federal agency had clear authority with regard to the internet at this time (it had passed from the US Department of Defense to the US National Science Foundation and then to the US Department of Commerce), no agency was assessing the readiness of the internet itself. Therefore, on 30 July 1999, the White House held the White House Internet Y2K Roundtable.[72]

The U.S. government also established the Center for Year 2000 Strategic Stability as a joint operation with the Russian Federation. It was a liaison operation designed to mitigate the possibility of false positive readings in each nation's nuclear attack early warning systems.[73]

International cooperation

The International Y2K Cooperation Center (IY2KCC) was established at the behest of national Y2K coordinators from over 120 countries when they met at the First Global Meeting of National Y2K Coordinators at the United Nations in December 1988. IY2KCC established an office in Washington, D.C. in March 1999. Funding was provided by the World Bank, and Bruce W. McConnell was appointed as director.

Juno Internet Service Provider CD labeling Y2K-compliance

IY2KCC's mission was to "promote increased strategic cooperation and action among governments, peoples, and the private sector to minimize adverse Y2K effects on the global society and economy." Activities of IY2KCC were conducted in six areas:

- National Readiness: Promoting Y2K programs worldwide
- Regional Cooperation: Promoting and supporting co-ordination within defined geographic areas
- Sector Cooperation: Promoting and supporting co-ordination within and across defined economic sectors
- Continuity and Response Cooperation: Promoting and supporting co-ordination to ensure essential services and provisions for emergency response
- Information Cooperation: Promoting and supporting international information sharing and publicity
- Facilitation and Assistance: Organizing global meetings of Y2K coordinators and to identify resources

IY2KCC closed down in March 2000.[74]

Private sector response

- The United States established the *Year 2000 Information and Readiness Disclosure Act*, which limited the liability of businesses who had properly disclosed their Y2K readiness.
- Insurance companies sold insurance policies covering failure of businesses due to Y2K problems.
- Attorneys organised and mobilised for Y2K class action lawsuits (which were not pursued).[75]
- Survivalist-related businesses (gun dealers, surplus and sporting goods) anticipated increased business in the final months of 1999 in an event known as the **Y2K scare**.[76]
- The Long Now Foundation, which (in their words) "seeks to promote 'slower/better' thinking and to foster creativity in the framework of the next 10,000 years", has a policy of anticipating the Year 10,000 problem by writing all years with five digits. For example, they list "01996" as their year of founding.
- While there was no one comprehensive internet Y2K effort, multiple internet trade associations and organisations banded together to form the Internet Year 2000 Campaign.[77] This effort partnered with the White House's Internet Y2K Roundtable.

The Y2K issue was a major topic of discussion in the late 1990s and as such showed up in most popular media. A number of "Y2K disaster" books were published such as *Deadline Y2K* by Mark Joseph. Movies such as *Y2K: Year to Kill* capitalised on the currency of Y2K, as did numerous TV shows, comic strips, and computer games.

Fringe group responses

A variety of fringe groups and individuals such as those within some fundamentalist religious organizations, survivalists, cults, anti-social movements, self-sufficiency enthusiasts, communes and those attracted to conspiracy theories, embraced Y2K as a tool to engender fear and provide a form of evidence for their respective theories. End-of-the-world scenarios and apocalyptic themes were common in their communication.

Interest in the survivalist movement peaked in 1999 in its second wave for that decade, triggered by Y2K fears. In the time before extensive efforts were made to rewrite computer programming codes to mitigate the possible impacts, some writers such as Gary North, Ed Yourdon, James Howard Kunstler,[78] and Ed Yardeni anticipated widespread power outages, food and gasoline shortages, and other emergencies. North and others raised the alarm because they thought Y2K code fixes were not being made quickly enough. While a range of authors responded to this wave of concern, two of the most survival-focused texts to emerge were *Boston on Y2K* (1998) by Kenneth W. Royce, and Mike Oehler's *The Hippy Survival Guide to Y2K*.

Y2K was also exploited by some fundamentalist and charismatic Christian leaders throughout the Western world, particularly in North America and Australia.[79] Their promotion of the perceived risks of Y2K was combined with end times thinking and apocalyptic prophecies in an attempt to influence followers.[79] The *New York Times* reported in late 1999, "The Rev. Jerry Falwell suggested that Y2K would be the confirmation of Christian prophecy — God's instrument to shake this nation, to humble this nation. The Y2K crisis might incite a worldwide revival that would lead to the rapture of the church. Along with many survivalists, Mr. Falwell advised stocking up on food and guns".[80] Adherents in these movements were encouraged to engage in food hoarding, take lessons in self-sufficiency, and the more extreme elements planned for a total collapse of modern society. The *Chicago Tribune* reported that some large fundamentalist churches, motivated by Y2K, were the sites for flea market-like sales of paraphernalia designed to help people survive a social order crisis ranging from gold coins to wood-burning stoves.[81] Betsy Hart, writing for the *Deseret News*, reported that a lot of the more extreme evangelicals used Y2K to promote a political agenda in which downfall of the government was a desired outcome in order to usher in Christ's reign. She also noted that, "the cold truth is that preaching chaos is profitable and calm doesn't sell many tapes or books"[82] These types of fears and conspiracies were described dramatically by New Zealand-based Christian prophetic author and preacher Barry Smith in his publication, "I Spy with my Little Eye", where he dedicated a whole chapter to Y2K.[83] Some expected, at times through so-called prophecies, that Y2K would be the beginning of a worldwide Christian revival.[84]

It became clear in the aftermath that leaders of these fringe groups had cleverly used fears of apocalyptic outcomes to manipulate followers into dramatic scenes of mass repentance or renewed commitment to their groups, additional giving of funds and more overt commitment to their respective organizations or churches. The *Baltimore Sun* noted this in their article, "Apocalypse Now — Y2K spurs fears", where they reported the increased call for repentance in the populace in order to avoid God's wrath.[85] Christian leader, Col Stringer, in his commentary has published, "Fear-creating writers sold over 45 million books citing every conceivable catastrophe from civil war, planes dropping from the sky to the end of the civilised world as we know it. Reputable preachers were advocating food storage and a "head for the caves" mentality. No banks failed, no planes crashed, no wars or civil war started. And yet not one of these prophets of doom has ever apologised for their scare-mongering tactics."[84] Some prominent North American Christian ministries and leaders generated huge personal and corporate profits through sales of Y2K preparation kits, generators, survival guides, published prophecies and a wide range of other associated merchandise. Christian journalist, Rob Boston, has documented this[79] in his article "False Prophets, Real Profits — Religious Right Leaders' Wild Predictions of Y2K Disaster Didn't Come True, But They Made Money Anyway".

Cost

The total cost of the work done in preparation for Y2K[86] is estimated at[87] over US$300 billion ($445 billion as of January 2018, once inflation is taken into account).[88][89] IDC calculated that the US spent an estimated $134 billion ($199 billion) preparing for Y2K, and another $13 billion ($19 billion) fixing problems in 2000 and 2001. Worldwide, $308 billion ($457 billion) was estimated to have been spent on Y2K remediation.[90]

Organisation of the remedial work

The remedial work was driven by customer demand for solutions.[86] Software suppliers, mindful of their potential legal liability,[75] responded with remedial effort. Software subcontractors were required to certify that their software components were free of date-related problems, which drove further work down the supply chain.

By 1999, many corporations required their suppliers to certify,[89] often on their own variation of a basic form, that all of their software is Y2K compliant. Some merely accepted remedial updates and then signed. Many businesses or even whole countries who spent little effort themselves nonetheless suffered only minor problems.

There are two ways to view the events of 2000 from the perspective of its aftermath:

Supporting view

This view holds that the vast majority of problems had been fixed correctly, and the money was well spent. The situation was essentially one of preemptive alarm. Those who hold this view claim that the lack of problems at the date change reflects the completeness of the project, and that many computer applications would not have continued to function into the 21st century without correction or remediation.

- Expected problems that were not seen by small businesses and small organisations were in fact prevented by Y2K fixes embedded in routine updates to operating system and utility software[91] that were applied several years before 31 December 1999.
- The extent to which larger industry and government fixes averted issues that would have more significant impacts had they not been fixed, were typically not disclosed or widely reported.[92]
- It has also been suggested that on 11 September 2001, infrastructure in New York City (including subways, phone service, and financial transactions) was able to continue operation because of the redundant networks established in the event of Y2K bug impact[93] and the contingency plans devised by companies.[94] The terrorist attacks and the following prolonged blackout to lower Manhattan had minimal effect on global banking systems.[95] Backup systems were activated at various locations around the region, many of which had been established to deal with a possible complete failure of networks in Manhattan's Financial District on 31 December 1999.[96]

Opposing view

The contrary view asserts that there were no, or very few, critical problems to begin with. This view also asserts that there would have been only a few minor mistakes and that a "fix on failure" approach would have been the most efficient and cost-effective way to solve these problems as they occurred.

- Countries such as South Korea and Italy invested little to nothing in Y2K remediation,[97] yet had the same negligible Y2K problems as countries that spent enormous sums of money.[98]

- The lack of Y2K-related problems in schools, many of which undertook little or no remediation effort. By 1 September 1999, only 28% of US schools had achieved compliance for mission critical systems, and a government report predicted that "Y2K failures could very well plague the computers used by schools to manage payrolls, student records, online curricula, and building safety systems".[99]
- The lack of Y2K-related problems in an estimated 1.5 million small businesses that undertook no remediation effort. On 3 January 2000 (the first weekday of the year), the Small Business Administration received an estimated 40 calls from businesses with computer issues, similar to the average. None of the problems were critical.[100]
- The absence of Y2K-related problems occurring before 1 January 2000, even though the 2000 financial year commenced in 1999 in many jurisdictions, and a wide range of forward-looking calculations involved dates in 2000 and later years. Estimates undertaken in the leadup to 2000 suggested that around 25% of all problems should have occurred before 2000.[101] Critics of large-scale remediation argued during 1999 that the absence of significant reported problems in non-compliant small firms was evidence that there had been, and would be, no serious problems needing to be fixed in *any* firm, and that the scale of the problem had therefore been severely overestimated.[102] However, this can be countered with the observation that large companies had significant problems requiring action, that Y2K programmers were fully aware of the variable timescale, and that they were working to a series of earlier target dates, rather than a single fixed target of 31 December 1999.[92]

Counterpoint

A counterpoint to most of the opposing view points above is that Y2K software updates were often sent to all users regardless of their active participation in the remediation.[103][104]

See also

- IPv4 address exhaustion, problems caused by the limited allocation size for numeric internet addresses
- ISO 8601, an international standard for representing dates and times, which mandates the use of (at least) four digits for the year
- Perpetual calendar, a calendar valid for many years, including before and after 2000
- YEAR2000, a configuration setting supported by some versions of DR-DOS to overcome Year 2000 BIOS bugs
- 512k day: an event in 2014, involving a software limitation in network routers.
- *Y2K*, a 1999 American made-for-television science fiction-thriller film directed by Dick Lowry
- "Life's a Glitch, Then You Die" is a "Treehouse of Horror segment" from *The Simpsons* eleventh season. The segment sees Homer forget to make his company's computers Y2K-compliant and this caused a virus to be unleashed upon the world

References

1. BSI Standard (http://www.empress.com/whatsnew/Y2K/bdd200.htm) on year 2000.
2. Bruce Caldwell; Andy Patrizio (1997-09-15). "Testing for 2000". *InformationWeek*. pp. 42–56 – via ProQuest ABI/INFORM Collection.
3. "Leap Day Tuesday Last Y2K Worry" (https://www.wired.com/2000/02/leap-day-tuesday-last-y2k-worry). *Wired*. 2000-02-25. Retrieved 2016-10-16.

4. Carrington, Damian (2000-01-04). "Was Y2K bug a boost?" (https://web.archive.org/web/20040422221434/http://news.bbc.co.uk/2/hi/science/nature/590932.stm). *BBC News*. Archived from the original (http://news.bbc.co.uk/1/hi/sci/tech/590932.stm) on 2004-04-22. Retrieved 2009-09-19.
5. Eric Andrew-Gee (2019-12-28). "Y2K: The strange, true history of how Canada prepared for an apocalypse that never happened, but changed us all" (https://www.theglobeandmail.com/canada/article-y2k-20th-anniversary-how-canada-prepared). *The Globe and Mail*.
6. Cory Johnson (1999-12-29). "Y2K Crier's Crisis" (https://www.thestreet.com/opinion/y2k-criers-crisis-839189). *TheStreet*.
7. Barnaby J. Feder (1998-10-11). "The Town Crier for the Year 2000" (https://archive.nytimes.com/www.nytimes.com/library/tech/98/10/biztech/articles/11prof.html#1). *The New York Times*.
8. Bolles, Spencer (1985-01-19). "Computer bugs in the year 2000" (https://groups.google.com/forum/#!topic/net.bugs/ZGlqGwNaq3I). Newsgroup: net.bugs (news:net.bugs). Usenet: 820@reed.UUCP (news:820@reed.UUCP). Retrieved 2019-08-15.
9. American RadioWorks (http://americanradioworks.publicradio.org/index.html) Y2K Notebook Problems (http://americanradioworks.publicradio.org/features/y2k/notebook.html) – *The Surprising Legacy of Y2K (http://americanradioworks.publicradio.org/features/y2k/a2.html)*. Retrieved 22 April 2007.
10. Rose, Ted (1999-12-22). "Who invented Y2K and why did it become so universally popular?" (http://articles.baltimoresun.com/1999-12-22/news/9912220295_1_y2k-term-masters). Retrieved 2017-02-11.
11. A web search on images for "computer memory ads 1975" returns advertisements showing pricing for 8K of memory at $990 and 64K of memory at $1495.
12. Kappelman, Leon; Scott, Phil (1996-11-25). "Accrued Savings of the Year 2000 Computer Date Problem" (http://www.comlinks.com/mag/accr.htm). *Computerworld*.
13. Looking at the Y2K bug (http://www.cnn.com/TECH/specials/y2k/), portal on CNN.com Archived (https://web.archive.org/web/20060207191845/http://www.cnn.com/TECH/specials/y2k/) 7 February 2006 at the Wayback Machine
14. Piskora, Beth (1997-03-01). "The Dow decimal system". *The New York Post*. p. 26.
15. Presenter: Stephen Fry (2009-10-03). "In the beginning was the nerd" (https://www.bbc.co.uk/iplayer/episode/b00mz53r/Archive_on_4_In_the_Beginning_Was_the_Nerd/). *Archive on 4*. BBC Radio 4.
16. Halvorson, Michael (1999). *Running Microsoft Office 2000*. Young, Michael J. Redmond, Wash.: Microsoft Press. ISBN 1-57231-936-4. OCLC 40174922 (https://www.worldcat.org/oclc/40174922).
17. Halvorson, Michael; Young, Michael (1999). *Running Microsoft Office 2000 Professional*. Redmond, WA: Microsoft Press. pp. xxxix. ISBN 1572319364. "As you learn about the year 2000 problem, and prepare for its consequences, there are a number of points we'd like you to consider. First, despite dire predictions, there is probably no good reason to prepare for the new millennium by holing yourself up in a mine shaft with sizable stocks of water, grain, barter goods, and ammunition. The year 2000 will not disable most computer systems, and if your personal computer was manufactured after 1996, it's likely that your hardware and systems software will require little updating or customizing."
18. Testimony by Alan Greenspan, ex-Chairman of the Federal Reserve before the Senate Banking Committee, 25 February 1998, ISBN 978-0-16-057997-4
19. The name IBM 1401 reflected the smallest amount of memory: 1,400 characters. "IBM 1401 Reference manual" (https://web.archive.org/web/20100809033646/http://www.bitsavers.org/pdf/ibm/140x/A24-1403-5_1401_Reference_Apr62.pdf) (PDF). Archived from the original (http://www.bitsavers.org/pdf/ibm/140x/A24-1403-5_1401_Reference_Apr62.pdf) (PDF) on 2010-08-09.
20. "Key computer coding creator dies" (https://www.washingtonpost.com/wp-dyn/articles/A4138-2004Jun24.html). *The Washington Post*. 2004-06-25. Retrieved 2011-09-25.

21. Andrew-Gee, Eric (2019-12-28). "Y2K: The strange, true history of how Canada prepared for an apocalypse that never happened, but changed us all" (https://www.theglobeandmail.com/canada/article-y2k-20th-anniversary-how-canada-prepared). *The Globe and Mail*.
22. Braden, Robert, ed. (October 1989). "Requirements for Internet Hosts -- Application and Support" (http://tools.ietf.org/html/rfc1123). Internet Engineering Task Force. Retrieved 2016-10-16.
23. D. Kolstedt (1997-11-15). "Helpful Year 2000 hint". *CIO magazine*. p. 12.
24. "Thinking Ahead". *InformationWeek*. 1996-10-28. p. 8. "extends .. the 23rd century"
25. Patrizio, Andy (1997-09-15). "Visa Debits The Vendors". *InformationWeek*. p. 50.
26. "Microsoft Knowledge Base article 214326" (http://support.microsoft.com/kb/214326). Microsoft Support. 2015-12-17. Retrieved 2016-10-16.
27. "JavaScript Reference Javascript 1.2" (http://docs.sun.com/source/816-6410-10/date.htm#1194138). Sun Microsystems. Retrieved 2009-06-07.
28. "JavaScript Reference Javascript 1.3" (http://docs.sun.com/source/816-6408-10/date.htm#1194138). Sun. Retrieved 2009-06-07.
29. "The Risks Digest Volume 4: Issue 45" (http://catless.ncl.ac.uk/Risks/4.45.html). *The Risks Digest*.
30. Stockton, J.R., "Critical and Significant Dates (http://www.merlyn.demon.co.uk/critdate.htm)" *Merlyn*.
31. A. van Deursen, "The Leap Year Problem (http://www.st.ewi.tudelft.nl/~arie/papers/klint.pdf)" *The Year/2000 Journal* 2(4):65–70, July/August 1998.
32. "Bank of Queensland hit by "Y2.01k" glitch" (http://www.crn.com.au/News/163864,bank-of-queensland-hit-by-y201k-glitch.aspx). CRN. 2010-01-04. Retrieved 2016-10-16.
33. "Windows Mobile glitch dates 2010 texts 2016" (http://news.cnet.com/8301-13860_3-10425455-56.html?tag=newsLatestHeadlinesArea.0#addcomm). 2010-01-05.
34. "Windows Mobile phones suffer Y2K+10 bug" (https://web.archive.org/web/20131023061135/http://www.techradar.com/news/world-of-tech/windows-mobile-phones-suffer-y2k-10-bug-661062). 2010-01-04. Archived from the original (http://www.techradar.com/news/world-of-tech/windows-mobile-phones-suffer-y2k-10-bug-661062) on 2013-10-23. Retrieved 2010-01-04.
35. "Bank of Queensland vs Y2K – an update" (http://www.itwire.com/content/view/30308/53/). 2010-01-04.
36. "Error: 8001050F Takes Down PlayStation Network" (https://gizmodo.com/error-8001050f-takes-down-playstation-network-5482365). *Gizmodo*.
37. "2010 Bug in Germany" (http://www.rtlinfo.be/info/monde/europe/297916/bug-de-l-an-2010-en-allemagne-plus-de-20-millions-de-cartes-bancaires-inutilisables) (in French). RTL. 2010-01-05. Retrieved 2016-10-16.
38. Raymond B. Howard. "The Case for Windowing: Techniques That Buy 60 Years". *Year/2000 Journal* (Mar/Apr 1998). "Windowing is a long-term fix that should keep legacy systems working fine until the software is redesigned and/or replatformed."
39. Green, Max. "CNN - Top 10 Y2K fixes for your PC - September 22, 1999" (http://www.cnn.com/TECH/computing/9909/22/top.y2k.idg). CNN.
40. "Millennium Bug Kit" (http://www.idg.net/go.cgi?id=162843).
41. Rudy Rupak
42. "The Year 2000 FAQ" (https://www-users.cs.umn.edu/~shekhar/5180/y2kfaq.txt). 1998-05-05. Retrieved 2020-03-01.
43. Ellen Friedman; Jerry Rosenberg. "Countdown to the Millennium: Issues to Consider in the Final Year" (http://davidjyoung.com/cmg98/images/8PDFILES/INT4108.PDF) (PDF).

44. Peter Kruskopfs. "The Date Dilemma" (https://web.archive.org/web/19961227162929/http://www.ibi.com/special/year2k/dilemma.html). *Information Builders*. Archived from the original (http://www.ibi.com/special/year2k/dilemma.html) on 1996-12-27. Retrieved 2020-03-15. "Bridge programs such as a date server are another option. These servers handle record format conversions from two to four digit years."
45. "Y2K bug rears its ugly head" (https://money.cnn.com/1999/01/12/technology/y2k_moneyline/). New York: CNN. 1999-01-12. Retrieved 2019-12-30.
46. Millennium bug hits retailers (http://news.bbc.co.uk/1/hi/business/582007.stm), from BBC News, 29 December 1999.
47. "Minor bug problems arise" (http://news.bbc.co.uk/1/hi/sci/tech/586620.stm). *BBC News*. 2000-01-01. Retrieved 2017-07-08.
48. "Japan nuclear power plants malfunction" (http://news.bbc.co.uk/2/hi/asia-pacific/585950.stm). *BBC News*. 1999-12-31.
49. Martyn Williams (2000-01-03). "Computer problems hit three nuclear plants in Japan" (https://web.archive.org/web/20041207161525/http://archives.cnn.com/2000/TECH/computing/01/03/japan.nukes.y2k.idg). *CNN*. IDG Communications. Archived from the original (http://archives.cnn.com/2000/TECH/computing/01/03/japan.nukes.y2k.idg) on 2004-12-07.
50. Wainwright, Martin (2001-09-13). "NHS faces huge damages bill after millennium bug error" (https://www.theguardian.com/uk/2001/sep/14/martinwainwright). *The Guardian*. UK. Retrieved 2011-09-25. "The health service is facing big compensation claims after admitting yesterday that failure to spot a millennium bug computer error led to incorrect Down's syndrome test results being sent to 154 pregnant women. ..."
51. Marsha Walton; Miles O'Brien (2000-01-01). "Preparation pays off; world reports only tiny Y2K glitches" (https://web.archive.org/web/20041207152504/http://archives.cnn.com/2000/TECH/computing/01/01/y2k.weekend.wrap/index.html). *CNN*. Archived from the original (http://archives.cnn.com/2000/TECH/computing/01/01/y2k.weekend.wrap/index.html) on 2004-12-07.
52. "BBC NEWS | Americas | US satellites safe after Y2K glitch" (http://news.bbc.co.uk/2/hi/americas/589836.stm). *news.bbc.co.uk*. Retrieved 2021-01-16.
53. "Leap Day Had Its Glitches" (https://www.wired.com/2000/03/leap-day-had-its-glitches). *Wired*. 2000-03-01. Retrieved 2020-02-25.
54. "HK Leap Year Free of Y2K Glitches" (https://www.wired.com/2000/02/hk-leap-year-free-of-y2k-glitches). *Wired*. 2000-02-29. Retrieved 2016-10-16.
55. "Leap Day Had Its Glitches" (https://www.wired.com/2000/03/leap-day-had-its-glitches). *Wired*. 2000-03-01. Retrieved 2016-10-16.
56. "The last bite of the bug" (http://news.bbc.co.uk/1/hi/sci/tech/1101917.stm). *BBC News*. 2001-01-05.
57. Rich Calder (2019-04-21). "New York's troubled wireless system has become a $900M money pit" (https://nypost.com/2019/04/21/new-yorks-troubled-wireless-system-has-become-a-900m-money-pit). *The New York Post*.
58. "NYC Wireless Network down due to Y2K-like software bug" (https://nypost.com/2019/04/10/nyc-wireless-network-down-due-to-y2k-like-software-bug). *The New York Post*.
59. Stokel-Walker, Chris. "A lazy fix 20 years ago means the Y2K bug is taking down computers now" (https://www.newscientist.com/article/2229238-a-lazy-fix-20-years-ago-means-the-y2k-bug-is-taking-down-computers-now/). *New Scientist*. Retrieved 2020-01-12.
60. Kohler, Iliana V.; Kaltchev, Jordan; Dimova, Mariana. "Integrated Information System for Demographic Statistics 'ESGRAON-TDS' in Bulgaria" (http://www.demographic-research.org/volumes/vol6/12/6-12.pdf) (PDF). 6 Article 12. Demographic Research: 325–354.
61. "The personal identity code: Frequently asked questions" (https://dvv.fi/en/personal-identity-code). Digital and Population Data Services Agency, Finland. Retrieved 2020-11-29.
62. "Uganda National Y2k Task Force End-June 1999 Public Position Statement" (http://parsifal.membrane.com/y2k/y2kugand.htm). 1999-06-30. Retrieved 2012-01-11.

63. "Y2K Center urges more information on Y2K readiness" (http://greenspun.com/bboard/q-and-a-fetch-msg.tcl?msg_id=001BVY). 1999-08-03. Retrieved 2012-01-11.
64. "Year 2000 Information and Readiness Disclosure Act" (https://corporate.findlaw.com/law-library/year-2000-information-and-readiness-disclosure-act.html). FindLaw. Retrieved 2019-05-14.
65. "Y2K bug: Definition, Hysteria, & Facts" (https://www.britannica.com/technology/Y2K-bug). *Encyclopædia Britannica*. 2019-05-10. Retrieved 2019-05-14.
66. DeBruce, Orlando; Jones, Jennifer (1999-02-23). "White House shifts Y2K focus to states" (http://www.cnn.com/TECH/computing/9902/23/shift.y2k.idg). CNN. Retrieved 2016-10-16.
67. Atlee, Tom. "The President's Council on Year 2000 Conversion" (https://www.co-intelligence.org/y2k_presidentscomm.html). The Co-Intelligence Institute. Retrieved 2019-05-14.
68. "FCC Y2K Communications Sector Report (March 1999) copy available at WUTC" (http://www.wutc.wa.gov/webdocs.nsf/b8da29aede8fdd67882571430005a9c1/c1cf57ff131085ca88256744007e1440/$FILE/Y2kcsr.pdf) (PDF). (1.66 MB)
69. "Statement by President on Y2K Information and Readiness" (https://clintonwhitehouse6.archives.gov/1998/10/1998-10-19-statement-by-president-on-y2k-information-and-readiness.html). *Clinton Presidential Materials Project*. National Archives and Records Administration. 1998-10-19.
70. "Home" (https://web.archive.org/web/20001205223500/http://www.y2k.gov/). *National Y2K Clearinghouse*. General Services Administration. Archived from the original (http://www.y2k.gov/) on 2000-12-05. Retrieved 2020-03-16.
71. Robert J. Butler and Anne E. Hoge, Wiley, Rein & Fielding (September 1999). "Federal Communications Commission Spearheads Oversight of the U.S. Communications Industries' Y2K Preparedness" (https://web.archive.org/web/20081009053904/http://www.opengroup.org/comm/the_message/magazine/mmv5n5/view.htm). *Messaging Magazine*. The Open Group. Archived from the original (http://www.opengroup.org/comm/the_message/magazine/mmv5n5/view.htm) on 2008-10-09. Retrieved 2016-10-16.
72. "Basic Internet Structures Expected to be Y2K Ready, Telecom News, NCS (1999 Issue 2)" (http://www.ncs.gov/library/telcom/tn_99-02.pdf) (PDF). (799 KB)
73. "U.S., Russia Shutter Joint Y2k Bug Center" (http://articles.chicagotribune.com/2000-01-16/news/0001160164_1_missile-peterson-air-force-base-y2k). *Chicago Tribune*. 2000-01-16. Retrieved 2017-01-28.
74. "Collection: International Y2K Cooperation Center records | University of Minnesota Archival Collections Guides" (https://archives.lib.umn.edu/repositories/3/resources/567). *archives.lib.umn.edu*.
75. Kirsner, Scott (1997-11-01). "Fly in the Legal Eagles". *CIO magazine*. p. 38.
76. "quetek.com" (http://www.quetek.com/dictionary/y2k-scare.html). quetek.com. Retrieved 2011-09-25.
77. Internet Year 2000 Campaign (http://www.cybertelecom.org/year2000/) archived at Cybertelecom.
78. Kunstler, Jim (1999). "My Y2K—A Personal Statement" (https://web.archive.org/web/20070927062527/http://kunstler.com/mags_y2k.html). Kunstler, Jim. Archived from the original (http://kunstler.com/mags_y2k.html) on 2007-09-27. Retrieved 2006-12-12.
79. "False Prophets, Real Profits - Americans United" (https://www.au.org/church-state/february-2000-church-state/featured/false-prophets-real-profits). Retrieved 2016-11-09.
80. Dutton, D., 31 December 2009 New York Times, "Its Always the End of the World as we Know it"
81. Coen, J., 1 March 1999, "Some Christians Fear End, It's just a day to others" Chicago Tribune
82. Hart, B., 12 February 1999 Deseret News, "Christian Y2K Alarmists Irresponsible" Scripps Howard News Service

83. Smith, B., 1999, I Spy with my Little Eye, MS Life Media, chapter 24 - Y2K Bug, http://www.barrysmith.org.nz/site/books/

84. "Col Stringer Ministries - Newsletter Vol.1 : No.4" (https://web.archive.org/web/20120320060233/http://www.colstringer.com/information/newsletters/vol-1-no-4.html). Archived from the original (http://www.colstringer.com/information/newsletters/vol-1-no-4.html) on 2012-03-20. Retrieved 2016-11-09.

85. Rivera, J., 17 February 1999, "Apocalypse Now – Y2K spurs fears", *The Baltimore Sun*

86. which was well underway by 1996: Tharp, Paul (1996-12-02). "Millennium Milllionairs: Counting past 2000; Mainframe mavens make their return". *The New York Post*. p. 27.

87. "low" #: US Senate Y2K committee's worldwide report for $200B;

88. Federal Reserve Bank of Minneapolis. "Consumer Price Index (estimate) 1800–" (https://www.minneapolisfed.org/about-us/monetary-policy/inflation-calculator/consumer-price-index-1800-). Retrieved 2020-01-01.

89. "Y2K: Overhyped and oversold?" (http://news.bbc.co.uk/2/hi/talking_point/586938.stm). 2000-01-06.

90. Mitchell, Robert L. (2009-12-28). "Y2K: The good, the bad and the crazy" (http://www.computerworld.com/s/article/9142555/Y2K_The_good_the_bad_and_the_crazy?taxonomyId=14&pageNumber=2). *Computerworld*.

91. David S. Joachim (2006-05-01). "A Mini-Y2K Looms, and Other Blips" (https://www.nytimes.com/2006/05/01/business/smallbusiness/01tech.html). *The New York Times*. "systems that automate the distribution of software fixes, called patches"

92. James Christie, (12 January 2015), Y2K – why I know it was a real problem, 'Claro Testing Blog' (https://clarotesting.wordpress.com/2015/01/12/y2k-why-i-know-it-was-a-real-problem/) (accessed 12 January 2015)

93. Y2K readiness helped New York after 9/11 (http://web.mit.edu/newsoffice/2002/terror-1120.html), article by Lois Slavin of MIT News, 20 November 2002.

94. "Finance & Development, March 2002 - September 11 and the U.S. Payment System" (http://www.imf.org/external/pubs/ft/fandd/2002/03/cumming.htm). *Finance and Development - F&D*.

95. Goldberg, Michael; Carr, Kathleen (2003-10-13). "The Next Time the Lights Go Out" (https://books.google.com/books?id=ig0AAAAAMBAJ&q=y2k+9%2F11+blackout+banking&pg=PP1). *CIO Magazine*.

96. Y2K readiness helped NYC on 9/11 (http://web.mit.edu/newsoffice/2002/terror.html), article by Rae Zimmerman of MIT News, 19 November 2002.

97. Dutton, Denis (2009-12-31), "It's Always the End of the World as We Know It" (https://www.nytimes.com/2010/01/01/opinion/01dutton.html), *The New York Times*.

98. Smith, R. Jeffrey (2000-01-04), "Italy Swatted the Y2K Bug" (https://www.washingtonpost.com/wp-srv/WPcap/2000-01/04/035r-010400-idx.html), *The Washington Post*.

99. White House: Schools lag in Y2K readiness: President's Council sounds alarm over K-12 districts' preparations so far (https://web.archive.org/web/20060506133440/http://eschoolnews.com/news/showstory.cfm?ArticleID=404), article by Jonathan Levine of *eSchool News*, 1 September 1999.

00. Hoover, Kent (2000-01-09), "Most small businesses win their Y2K gamble" (http://www.bizjournals.com/seattle/stories/2000/01/10/newscolumn5.html?page=all), *Puget Sound Business Journal*.

01. Lights out? Y2K appears safe (http://www.enquirer.com/editions/1999/02/14/fin_lights_out_y2k.html), article by Elizabeth Weise of *USA Today*, 14 February 1999.

02. John Quiggin, (2 September 1999), Y2K bug may never bite, 'Australian Financial Review' (https://web.archive.org/web/20080524084926/http://www.uq.edu.au/economics/johnquiggin/news/Millennium9908.html) (from The Internet Archive accessed 29 December 2009).

03. Eugene Clark; George Cho; Arthur Hoyel. "Y2K Litigation: More Bugs Hitting the Same Legal Fan" (http://www.austlii.edu.au/au/journals/JlLawInfoSci/1999/12.html).

04. T. August (2011). "Who Should Be Responsible for Software Security?". JSTOR 25835749 (https://www.jstor.org/stable/25835749).

External links

- Center for Y2K and Society Records (http://purl.umn.edu/53939), Charles Babbage Institute, University of Minnesota. Documents activities of Center for Y2K and Society (based in Washington DC) working with non-profit institutions and foundations to respond to possible societal impacts of the Y2K computer problem: helping the poor and vulnerable as well as protecting human health and the environment. Records donated by executive director, Norman L. Dean.
- International Y2K Cooperation Center Records, 1998–2000 (http://purl.umn.edu/41469), Charles Babbage Institute, University of Minnesota. Collection contains the materials of the International Y2K Cooperation Center. Includes country reports, news clippings, country questionnaires, country telephone directories, background materials, audio visual materials and papers of Bruce W. McConnell, director of IY2KCC.
- Preparing for an Apocalypse: Y2K (http://www.cbi.umn.edu/Y2K/index.html), Charles Babbage Institute, University of Minnesota. A web exhibit curated by Stephanie H. Crowe
- BBC: Y2K coverage (http://news.bbc.co.uk/1/hi/sci/tech/585013.stm)
- *In The Beginning There Was The Nerd* (https://www.bbc.co.uk/iplayer/episode/b00mz53r/Archive_on_4_In_the_Beginning_Was_the_Nerd/) – BBC Radio documentary about the history of computers and the millennium bug 10 years after using archival recordings.
- *The Surprising Legacy Of Y2K* (http://americanradioworks.publicradio.org/features/y2k/index.html) – Radio documentary by American Public Media, on the history and legacy of the millennium bug five years on.
- The Yawn of a New Millennium (http://www.benbest.com/computer/y2kfeb.html)
- CBC Digital Archives – The Eve of the Millennium (http://archives.cbc.ca/IDD-1-69-1998/life_society/y2k/)
- How the UK coped with the millennium bug (https://www.bbc.co.uk/news/magazine-30576670)
- Time running out for PCs at big companies (http://www.cnn.com/TECH/computing/9806/30/y2k.idg/index.html) *CNN*

Retrieved from "https://en.wikipedia.org/w/index.php?title=Year_2000_problem&oldid=1022043383"

This page was last edited on 8 May 2021, at 03:27 (UTC).